

Awareness im Enterprise 2.0

Empirische Untersuchung über Awareness im Arbeitsumfeld
einer Softwareentwicklungsfirma

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Medieninformatik

eingereicht von

Simon Diesenreiter

Matrikelnummer 9926534

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung
Betreuerin: Assoc. Prof. Dipl. Ingⁱⁿ Drⁱⁿ Hilda Tellioglu

Wien, 15.07.2013

(Unterschrift Verfasser/in)

(Unterschrift Betreuer/in)

Erklärung zur Verfassung der Arbeit

Simon Diesenreiter,
Wichtelgasse 20/28, 1160 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 15.07.2013

(Unterschrift Verfasser/in)

Danksagung

Ich möchte mich bei allen bedanken, die mich auf verschiedene Weise während meines Studiums unterstützt und begleitet haben, besonders bei meinen Eltern, Geschwistern, Freundinnen und Freunden, Studienkolleginnen und Kollegen. Vielen Dank!

Den Leuten von der VersicherungsGmbH möchte ich dafür danken, dass sie es mir ermöglicht haben, die empirische Untersuchung bei- und mit ihnen durchzuführen.

Ein Dank gilt auch Frau Professor Hilda Tellioglu für die stets motivierte Betreuung der Diplomarbeit und Frau Professor Ina Wagner dafür, dass sie mir qualitative Methoden nähergebracht hat.

Kurzfassung

Der Begriff *Awareness* in der CSCW-Forschung wird durch technische Neuerungen stetig vorangetrieben. Die Verwendung von Web 2.0-Technologien in Unternehmen – auch bekannt als *Enterprise 2.0* – stellt eine solche Neuerung dar. Das Ziel der vorliegenden Diplomarbeit ist es zu zeigen, wie Enterprise 2.0-Systeme gemeinsam mit klassischen Informationssystemen und Kommunikationsmechanismen im Arbeitsumfeld eingesetzt werden, um Awareness zwischen ArbeitskollegInnen herzustellen. Durch eine theoretische und empirische Analyse wird ein Modell von Awareness konstruiert, welches darstellt, welche Variablen, Mechanismen und Faktoren bei der Arbeit mit Awarenessinformationssystemen wirken und welche Beziehungen zwischen ihnen bestehen.

Für die empirische Untersuchung wurde ob der Komplexität des Themas ein qualitativer Forschungsansatz im Stile der Grounded Theory gewählt. Die ethischen Kriterien *Offenheit, theoriegeleitet, regelgeleitet* und *Schutz der untersuchten Personen* leiteten das Untersuchungsdesign. Die Untersuchung wurde in einer Softwareentwicklungsfirma in der Versicherungsbranche mit ca. 140 MitarbeiterInnen durchgeführt. Zur Datenerhebung wurden in der betreffenden Firma eine Intranetrecherche, teilnehmende Beobachtung und ExpertInneninterviews durchgeführt.

Die Analyse resultiert in einem Modell von Awareness, das aus den vier Kategorien *Awarenessbedürfnisse, Awarenessmechanismen, Toolverwendung* und *Faktoren* besteht und auf der Zusammenarbeit von KollegInnen und Teams in und mit einem Ökosystem von Anwendungen begründet ist. Das erarbeitete Modell baut in seinem Ursprung auf vorhandenen Modellen von Awareness auf, erweitert diese aber in einigen Punkten: Ereignis- und Kontextmodelle werden in der Kategorie *Awarenessbedürfnisse* durch die Inhalte von Awareness verfeinert und um den Begriff des Zustandes erweitert. Die Erfassung, Verarbeitung und Präsentation von Ereignissen und Zustand wird durch verschiedene *Awarenessmechanismen* ausdifferenziert. *Toolverwendung* nennt die Formalisierung von Awarenessinhalten als Grundvoraussetzung für die Umsetzung von Awareness mittels Tools. Bestehende Modelle von Awareness werden um organisatorische, tool- und informationsbasierte Faktoren ergänzt.

Abstract

The concept of *awareness* in the field of CSCW-research has always been driven by technological advances. The usage of web 2.0-technologies in organizations – also known as *Enterprise 2.0* – depicts such an innovation. The goal of this thesis is to show how enterprise 2.0-systems are used jointly with „classical“ information systems and communication mechanisms in a work environment to establish awareness between members of a team. By means of a theoretical and empirical analysis we construct a model of awareness that shows which variables, mechanics and factors are of effect and which relationships exist between those.

Due to the complexity of the topic a qualitative research approach was chosen for the empirical study, guided by grounded theory methodology as well as a set of ethical research guidelines. The study was carried out in a software development company of the insurance industry occupying around 140 people. Data acquisition was done by means of intranet research, participatory observation and expert interviews.

Analysis results in a model of awareness, comprising of four categories *awareness needs*, *awareness mechanisms*, *tool usage* and *factors* and is grounded in the cooperation of teammembers in an ecosystem of tools and applications. The developed model is based on existing models of awareness, but extends them in a few ways: *Awareness needs* extends and refines event- and context models by the notions of *contents of awareness* and *state*. Collection, processing and presentation of events and state is differentiated by several *awareness mechanisms*. *Tool usage* states the *formalization of contents of awareness* as a prerequisite for the implementation of awareness by means of tools. Existing models of awareness are complemented by organizational, tool and information based factors.

Inhaltsverzeichnis

1. Einleitung	1
1.1 Problemstellung.....	2
1.2 Fragestellungen	3
1.3 Abgrenzung.....	4
1.4 Aufbau der Arbeit	4
2. Awareness	5
2.1 Definitionen von Awareness.....	5
2.2 Ausprägungen von Awareness	5
2.3 Modelle von Awareness	8
2.4 Mechanismen für Awarenessverarbeitung	9
2.5 Inhalt von Awareness - Awarenessbedürfnisse	11
2.6 Faktoren für Awareness.....	14
2.7 Nutzen von Awareness	17
2.8 Enterprise 2.0 und Awareness.....	18
3. Methoden	21
3.1 Qualitative Methoden	21
3.2 Forschungsethik.....	21
3.3 Grounded Theory	22
3.4 Datenerhebung.....	25
4. Ergebnisse Kontextrecherche	31
4.1 Branchenkontext	31
4.2 Organisatorischer Kontext.....	32
4.3 Tools	33
4.3.1 Lotus Notes	33
4.3.2 TTS.....	35
4.3.3 REA.....	38
4.3.4 JIRA	39
4.3.5 Weitere Anwendungen	47
5. Ergebnisse Feldforschung	48
5.1 Entwicklung des Kategoriensystems.....	48
5.2 Das Kategoriensystem	49
5.3 Kategorie Awarenessbedürfnisse	50
5.3.1 Inhalt von Awareness.....	51
5.3.2 Kontext & Abhängigkeiten	60
5.3.3 Ereignisse & Zustand.....	62
5.3.4 Übersicht & Detail.....	63
5.4 Kategorie Awarenessmechanismen	63
5.4.1 Informationspushing – „Bescheid sagen“	64
5.4.2 Informationspulling – „Regelmässig nachschauen“	65
5.4.3 Subscription – „Erwarte, dass ich informiert werde“	66
5.4.4 Periphere Awareness	66
5.5 Kategorie Toolverwendung	67
5.5.1 Unterschiedliche Nutzung von JIRA	68
5.5.2 Formalisierung von Inhalten	73
5.6 Kategorie Faktoren	82
5.6.1 Organisatorische Faktoren.....	82
5.6.2 Tool- und Informationsfaktoren	83
5.7 Modell von Awareness	85
5.7.1 Verbindung zu bestehenden Modellen von Awareness	87
5.7.2 Anwendbarkeit des Modells	88

6. Zusammenfassung	89
6.1 Future Work.....	91
6.2 Reflexion.....	92
7. Bibliografie	93
Appendix A – Einverständniserklärung Firma	96
Appendix B – Projektinformation MitarbeiterInnen	99
Appendix C – Einverständniserklärung MitarbeiterInnen	101
Appendix D – Interviewleitfaden	102
Appendix E – Teilnehmende Beobachtung	104
Appendix F – Interviews	119
Interview 01_ENTW_DE_1.....	119
Interview 02_FK_TL.....	128
Interview 03_ENTW_DE_2.....	140
Interview 04_TST_TL.....	153
Interview 05_ENTW_DE_TL_STV_1	157
Interview 06_FK_2	166
Interview 08_PMM_1	171
Interview 09_ENTW_DE_3.....	180
Interview 10_RMM_1	189
Interview 11_PQM_1	199
Interview 12_ENTW_DE_TL_STV_2	205
Interview 13_FK_1	215
Interview 14_TST_3.....	222
Interview 16_ENTW_DE_AL.....	229
Interview 17_ENTW_DE_TL	234

1. Einleitung

Awareness wird in der Forschung über computerunterstützte kollaborative Arbeit („CSCW“) als „*das Gewährsein der Anwesenheit und des Verhaltens anderer Gruppenmitglieder*“ (Schwabe, Streitz & Unland 2001, p.184) definiert. Espinosa et al. (2007) bezeichnen Awareness als Kurzzeit- bzw. flüchtiges Wissen, im Gegensatz zu Langzeitwissen. Dourish und Bellotti (1992) definieren Awareness als „*an understanding of the activities of others, which provides a context for your own activity*“ (ibid, p. 1).

Verschiedene Ausprägungen von Awareness sind beispielsweise „Chronological Awareness“ (Espinosa et al. 2000), „Location Awareness“ (Priestersbach et al. 2001), „Informational-, Subscription based-, und Peripheral Awareness“ (Brush et al. 2002), „Social Awareness“ (Bardram & Hansen 2004), „Mutual Awareness“, „Taskwork Awareness“, „Workload Awareness“, „Teamwork Awareness“ (MacMillan et al. 2004), „Change Awareness“ (Tam & Greenberg 2006), „Task Awareness“ und „Team Awareness“ (Espinosa et al. 2007) oder „Anticipative Awareness“ (Prinz, Hinrichs und Kireyev 2010). Einige dieser Definitionen überschneiden sich oder beschreiben dasselbe Phänomen, die meisten bauen aber auf verschiedenen technischen Konzepten auf.

Technische Neuerungen sind es auch, die den Begriff von Awareness vorantreiben. Ein klassisches Beispiel für ein System mit Awarenessinformation stellen desktopbasierte Instant Messaging (IM) Programme dar, die Informationen über den Onlinestatus und häufig auch Ort und Zeitzone von Chatpartnern anzeigen. **Web 2.0-Technologien** stellen ein neues Beispiel für eine Technologie dar, die potentiell fähig ist, Awarenessinformationen zu generieren und anzuzeigen. Naaman, Boase und Lai (2010) bezeichnen Systeme wie Twitter¹ oder den Facebook²-Newsfeed demnach als „Social Awareness Streams“. Diese zeichnen sich laut diesen AutorInnen durch drei wesentliche Merkmale aus: 1.) die Kommunikation, bzw. Nachrichten sind öffentlich bis halböffentlich, 2.) Nachrichten sind relativ kurz, und 3.) die Plattformen sind „*a highly connected social space, where most of the information consumption is enabled and driven by articulated online contact networks*“ (ibid), d.h., die Kommunikation findet vorwiegend online statt und organisiert sich rund um diese Plattformen.

Die Verwendung von Web 2.0-Technologien in Unternehmen wird von McAfee (2006) als **Enterprise 2.0** bezeichnet. Moderne webbasierte **Issuetracking- und Projektmanagementsysteme** wie *JIRA*³ oder *TRAC*⁴, die in der Softwareentwicklung verwendet werden, erfüllen mit ihren Anwendungsmöglichkeiten und ihrer technologischen Basis die Definition von Enterprise 2.0 und stellen auch mehrere Arten von Awarenessinformationen- und Mechanismen zur Verfügung. „Collaborative Development Environments“ (CDEs) ist ein Begriff, den Treude und Storey (2010) verwenden, um die Integration und Verknüpfung von „Integrated Development Environments“ (IDEs) mit Issuetracking- und Projektmanagementsystemen zu bezeichnen. Beispiele dafür sind die Integration von *JIRA* in die *Eclipse*⁵-IDE mittels des Plugins *MyLyn*⁶ oder das IBM-Produkt *Rational Team Concert*⁷, welches die Eclipse-IDE und ein webbasiertes Issuetracking- und Projektmanagementsystem verknüpfen, und darüber hinaus auch schon IM-Funktionalitäten in die IDE integrieren.

Durch solche integrierten Ansätze wird die IDE/CDE zu einer immer mehr Funktionalitäten und Awarenessinformationen vereinigenden Anwendung. Treude und Storey (2010), die die Verwendung von Dashboards und Feeds in CDEs untersuchen, nennen die Gewinnung von Projektawareness und peripherer Awareness über Teamaktivitäten als die primären Vorteile

¹ www.twitter.com

² www.facebook.com

³ <http://www.atlassian.com/jira>

⁴ <http://trac.edgewall.org/>

⁵ <http://www.eclipse.org>

⁶ <http://www.eclipse.org/mylyn/>

⁷ <http://jazz.net/projects/rational-team-concert/>

der Verwendung solcher Systeme, stellen aber gleichzeitig fest, dass viele AnwenderInnen mit der Fülle an Informationen und Funktionalitäten in CDEs überfordert sind. Die AutorInnen fordern daher übersichtlichere und besser angepasste Awarenessinformationen in CDEs, bei gleichzeitiger weiterer Integration und Vereinheitlichung von IDEs, Issuetracking- und Projektmanagementsystemen.

1.1 Problemstellung

Treude und Storey (2010) liefern mit ihrer Studie einen wesentlichen Beitrag zu Awareness und Enterprise 2.0, die aber angesichts der Vielfältigkeit dieses Themas nicht vollständig sein kann, und weshalb die AutorInnen auch weitere Untersuchungen dazu anregen.

Ein Punkt der von Treude und Storey (2010) nicht betrachtet wird, ist die gemeinsame Verwendung unterschiedlicher Awarenessstools – also von „klassischen“ bis hin zu Enterprise 2.0-Awarenessinformationssystemen – innerhalb von bestehenden Arbeitssystemen- und Kontexten: *„One of the most important themes that came out of our work is the need to design studies that look beyond one individual tool and focus on the interplay of processes, artifacts and developers. We need to evaluate how awareness tools and features are used across different dimensions of collaboration“* (Treude & Storey 2010, p.373). Turner et al. (2010), die die Verwendung unterschiedlicher Kommunikationsmedien innerhalb eines Unternehmens untersuchen, zeigen, dass AnwenderInnen immer mehrere Kommunikationskanäle verwenden und dass durch das Hinzukommen neuer Kanäle weiterhin auch bisherige Kanäle verwendet werden. Wenn man Awareness, bzw. das Bereitstellen von Awarenessinformation, als eine Form von Kommunikation sieht, so ist auch bei der Verwendung von webbasierten Issuetrackingsystemen in einer Firma nicht anzunehmen, dass sie andere Kommunikations- und Awarenessmittel verdrängen, sondern sich in ein bestehendes Ökosystem von Anwendungen einfügen.

Über die Art und Weise, wie Enterprise 2.0 Systeme gemeinsam mit klassischen Systemen verwendet werden, und darüber, wie sich diese Integration auf die Awareness von ArbeitskollegInnen auswirkt, gibt es aber keine Studien, d.h., es fehlen vergleichende Daten über die gemeinsame Verwendung von „klassischen“ und Enterprise 2.0-Awarenessinformationssystemen. Turner et al. (2010) untersuchen zwar die gemeinsame Verwendung verschiedener „klassischer“ und webbasierter Kommunikationskanäle, gehen dabei aber nicht speziell auf die Themen Awareness oder Issuetrackingsysteme ein. Auch aufgrund der von Treude und Storey (2010) genannten Schwächen (Unübersichtlichkeit) von CDEs stellt sich die Frage, ob zur Gewinnung von Awareness über den Status und die Aufgaben von Kollegen CDEs das bevorzugte Werkzeug darstellen, oder ob klassische Kommunikationsmittel und Werkzeuge wie Face-to-Face-Gespräche, Telefon, Email, Chat und die Kontaktliste des Instant Messengers die primären Werkzeuge für die Gewinnung von Statusinformationen über Kollegen sind. Diese Frage wird durch die vorhandene Literatur nicht hinreichend beantwortet.

Weiters untersuchen Treude und Storey (2010) in ihrer Studie hauptsächlich SoftwareentwicklerInnen und ProjektmanagerInnen. In den Softwareentwicklungsprozess sind zumeist aber auch andere Rollen eingebunden, wie beispielsweise TesterInnen, SystemanalytikerInnen oder Releasemanagement, die ebenfalls häufig mit diesen – ursprünglich für EntwicklerInnen und ProjektmanagerInnen entwickelten – Issuetrackingsystemen arbeiten. Über die Art und Weise der Anwendung und Auswirkungen moderner Issuetrackingsysteme auf diese Rollen, vor allem in Bezug auf Awareness, gibt es ebenfalls keine Studien. *„As developers need to understand issues that span several teams and components, the line between requirements for dashboards and feeds becomes blurred and tool enhancement requests add up to a considerable overlap of both tools. Future work will have to be conducted to identify the ideal toolset for awareness in collaborative software development. Our findings suggest that such a toolset brings awareness of projects, teams and tasks together and no longer separates between awareness in the large and awareness in the small.“* (Treude & Storey 2010, p.374).

Ein Faktor, der bei der Untersuchung von Issuetrackingsystemen in Treude und Storey (2010) nicht erwähnt wird, ist räumliche Distanz zwischen KollegInnen. Distanz spielt laut Bardram und Hansen (2004) eine wichtige Rolle bei der Vermittlung von sozialer Awareness, da dadurch die Menge an vermittelbaren Kontextinformationen („rich social awareness“, (ibid, p. 194)) stark beeinflusst wird. Daher wäre zu untersuchen, wie sich webbasierte Issuetrackingsysteme auf die Awareness von räumlich verteilten KollegInnen auswirken.

Treude und Storey (2010) untersuchen die Verwendung einer konkreten CDE (IBM Jazz⁸). Da es aber eine ganze Reihe von ähnlichen Systemen gibt, sollten auch andere Systeme untersucht werden, um der beispielsweise von Diekmann (2000) angesprochenen Forderung, dass empirische Untersuchungen repliziert werden sollten, um „der wissenschaftstheoretischen Maxime, Hypothesen möglichst vielen, strengen Bewährungsproben zu unterziehen“ (ibid, p. 165), nachzukommen.

1.2 Fragestellungen

In dieser Diplomarbeit soll versucht werden, den in der Problemstellung identifizierten Fragen über die gemeinsame Verwendung von verschiedenen Awarenesssystemen für Teamarbeit nachzugehen. Ein besonderer Fokus soll auf der Betrachtung von webbasierten Issuetrackingsystemen liegen, welche gemeinsam mit anderen Kommunikationsmitteln in Softwareentwicklungsteams verwendet werden. Durch eine theoretische und empirische Untersuchung soll so ein Modell von Awareness im Enterprise 2.0 erarbeitet werden.

Zum einen soll die Frage geklärt werden, wie sich webbasierte Issuetrackingsysteme in ein bestehendes Ökosystem von Kommunikations- und Kollaborationsmitteln einfügen, und welche Auswirkungen sich dadurch für die Zusammenarbeit von KollegInnen in einer Softwareentwicklungsfirma ergeben. Insbesondere soll untersucht werden, ob sich zur Gewinnung von Awareness über den Status und die Aufgaben von KollegInnen in einem Unternehmen Issuetrackingsysteme und CDEs, oder „klassische“ Kommunikationsmittel- und Kanäle wie Face-to-Face-Gespräche, Telefon, Email, Chat oder die Kontaktliste des Instant Messengers besser eignen. Gibt es bestimmte Präferenzen in der Benutzung und Auswahl von Tools? Wird Kommunikation „gedoppelt“, d.h., wird bei Verwendung zum Beispiel einer Kommentarfunktion in einem Issuetrackingsystem darauf vertraut, dass die Kommunikationspartnerin diesen Kommentar liest, oder wird ein zusätzlicher Kommunikationskanal bemüht, um die Kommunikationspartnerin auf den Kommentar aufmerksam zu machen? Welche Awarenessbedürfnisse bestehen bei den an einem Projekt beteiligten Personen und Teams und wie werden diese befriedigt? Welche Awarenessbedürfnisse werden durch ein Issuetrackingsystem oder andere Tools abgedeckt, welche nicht? Diese Fragen sind insbesondere wichtig, als Treude und Storey (2010) auch die Frage stellen, wie ein „optimal toolset“ (ibid) – also eine geeignete Sammlung von Softwaretools – für die Gewinnung von Awareness aussieht.

Die gestellten Fragen sollen auf individueller, auf Teamebene und auf einer teamübergreifenden Ebene untersucht werden. Neben EntwicklerInnen und ProjektmanagerInnen sollen auch andere am Softwareentwicklungsprozess beteiligte Teams wie SystemanalytikerInnen, TesterInnen oder ReleasemanagerInnen betrachtet werden. Weiters soll untersucht werden, wie sich der Faktor räumliche Distanz auf die Gewinnung von Awareness durch webbasierte Issuetrackingsysteme auswirkt, bzw. ob solche Systeme helfen können den - zumeist negativ besetzten - Faktor Distanz zu überwinden.

Diese Fragen soll mittels einer empirischen Studie in einem Unternehmen, das solche Tools einsetzt, geklärt werden. Es soll ein Bild von den Arbeitsanforderungen und der Arbeitspraxis, den Arbeits- und Kommunikationsgewohnheiten und der Konfiguration der Arbeitsplätze in einem solchen Unternehmen gegeben werden. Es soll untersucht werden, ob Issuetrackingsysteme „solitär“, d.h. abgekapselt oder isoliert, verwendet werden oder ob Awarenessinformationen aus solchen Systemen Eingang in andere Kommunikationskanäle finden. Schließlich soll untersucht werden, wie „stark“ Issuetrackingsysteme im Gegensatz

⁸ <https://jazz.net/>

zu anderen Informations- und Kommunikationskanälen für die Gewinnung von Awarenessinformationen über KollegInnen und Aufgaben verwendet werden.

Auf Basis der empirischen Untersuchung sowie einer eingehenden Aufarbeitung der theoretischen Grundlagen zu dem Thema soll ein Modell von Awareness im Enterprise 2.0 entwickelt werden, welches darstellt, welche Variablen, Mechanismen und Faktoren darin wirken und welche Beziehungen zwischen diesen Variablen und Faktoren bestehen. Rogge (1995) stellt zur Konstruktion von Modellen fest: „*In jedem Fall wird ein Optimierungsproblem zu lösen sein, das darin besteht, herauszufinden, mit welcher Art und Menge von Variablen, deren Beziehungskonstellationen und kennzeichnenden Werte (Parameter) das Modell auszustatten ist, um die Realität (Phänomenbereich, Prozess) optimal zu repräsentieren.*“ (ibid, p. 52).

1.3 Abgrenzung

Die empirische Untersuchung soll sich auf „klassische“ Firmen, bzw. klassische Softwareproduktion innerhalb einer zentralen Firmenstruktur mit einem gewissen Maß an räumlicher Nähe und persönlichem (Face 2 Face) Kontakt beziehen, im Gegensatz zu (Open Source-) Softwareentwicklungsprojekten, welche ausschließlich über das Web organisiert sind und deren EntwicklerInnen keinen räumlichen Kontakt haben.

Weiters soll die empirische Untersuchung nicht darauf abzielen, das betrachtete Issuetrackingsystem zu evaluieren oder im Vergleich mit anderen ähnlichen Systemen zu bewerten, vielmehr soll dieses als ein Beispiel dienen, um grundsätzlich wirkende Mechanismen und Faktoren in einem Softwareentwicklungsteam zu erarbeiten.

1.4 Aufbau der Arbeit

In Kapitel 2 werden theoretische Grundbegriffe, die für das Thema wichtig sind erklärt. Zuerst wird der Begriff *Awareness* definiert, dazu werden verschiedene, in der Literatur erwähnte Ausprägungen davon präsentiert. Danach werden bestehende Modelle und Klassifikationen von Awareness beschrieben. Anschliessend wird untersucht, welche Mechanismen bei der Verarbeitung von Awareness zum Tragen kommen. Schließlich werden Faktoren herausgearbeitet, die beim Design von und bei der Arbeit mit Awarenesssystemen eine Rolle spielen. Danach wird der Nutzen von Awareness dargestellt. Zuletzt wird der Begriff *Enterprise 2.0* im Zusammenhang mit Awareness erklärt.

In Kapitel 3 wird die Methodik und die Umsetzung dargestellt. Zuerst wird erklärt, warum qualitative Methoden als Untersuchungsmethode verwendet werden. Ethische Grundlagen für die Untersuchung werden ebenfalls dargestellt. Die Methodik der Grounded Theory wird überblicksmäßig mit den für die in dieser Arbeit wichtigen Elementen dargestellt. Danach werden die für die Datenerhebung verwendeten Methoden und deren Umsetzung im Feld beschrieben, ebenso die Aufbereitung der Daten.

In Kapitel 4 und 5 werden schließlich die Ergebnisse der Untersuchung dargestellt. Kapitel 4 enthält die Ergebnisse der Intranetrecherche: Zuerst werden der Branchen- und Firmenkontext erklärt, anschliessend werden die in der Firma verwendeten Tools anhand ihres Zweckes und ihrer potentiellen Möglichkeiten für Awareness analysiert. In Kapitel 5 werden die in den Interviews und der teilnehmenden Beobachtung erhobenen Daten mittels der Grounded Theory Methodik analysiert und dargestellt. Das Ergebnis besteht aus einem Modell von Awareness, das aus vier Kategorien „Awarenessbedürfnisse“, „Awarenessmechanismen“, „Toolverwendung“ und „Faktoren“ besteht.

In Kapitel 6 werden die Ergebnisse der Diplomarbeit nochmals zusammengefasst. Ein persönliches Resümee über den Forschungsvorgang wird gezogen, und Vorschläge für zukünftige Forschungsmöglichkeiten werden diskutiert.

2. Awareness

Im folgenden Kapitel wird der Begriff „Awareness“ an sich erklärt, dazu werden verschiedene in der Forschungsliteratur vorkommende Ausprägungen von Awareness beispielhaft gegenübergestellt. Anschließend werden Erklärungsmodelle für Awareness und Awarenesssysteme vorgestellt, die es erlauben, verschiedene Ausprägungen von Awareness einzuordnen, oder die verschiedene Mechanismen und Anforderungen von Awareness und Awarenesssystemen andeuten. Solche Mechanismen werden anschließend in einem eigenen Abschnitt ausführlicher behandelt. Danach werden in zwei separaten Abschnitten die eigentlichen Inhalte von Awarenesssystemen- und Ausprägungen, sowie bei der Arbeit mit diesen zum Tragen kommende Faktoren, ausgearbeitet und analysiert. Der Nutzen von Awareness wird nochmals in einem eigenen Abschnitt erläutert. Zuletzt wird der Begriff Enterprise 2.0 und dessen Definition erklärt, und wie sich damit Awarenessmechanismen theoretisch umsetzen lassen.

2.1 Definitionen von Awareness

Awareness ist ein sehr breiter Begriff, der von verschiedensten Fachrichtungen verwendet wird. In der Forschung über computerunterstützte kollaborative Arbeit („CSCW“) wird Awareness als „*das Gewährsein der Anwesenheit und des Verhaltens anderer Gruppenmitglieder*“ (Schwabe, Streit & Unland 2001, p.184) definiert. Espinosa et al. (2007) definieren Awareness als Kurzzeit- bzw. flüchtiges Wissen, im Gegensatz zu Langzeitwissen. Dourish und Bellotti (1992) definieren Awareness als „*an understanding of the activities of others, which provides a context for your own activity*“ (ibid, p. 1). In Gegensatz zu solchen personenzentrierten Definitionen von Awareness kann auch die Behaftung von technischen Systemen mit Kontextinformation / Kontext-Awareness gestellt werden, wie sie beispielsweise von Veiel, Haake und Lukosch (2009) beschrieben wird.

Im Kontext dieser Diplomarbeit soll der Begriff Awareness dazu verwendet werden, um menschliche Aufmerksamkeit und Wissen in Bezug auf die sie (die Menschen) umgebende Entitäten im Arbeitsumfeld – Kollegen, Aufgaben, Projekte, technische Systeme – zu bezeichnen. „Aufmerksamkeit“ bedeutet dabei, dass Ereignisse und Änderungen an den Entitäten wahrgenommen werden, „Wissen“ bezieht sich auf den (momentanen) Zustand von Entitäten insofern, als dieser wichtig für den momentanen Arbeitskontext einer Person sein kann.

2.2 Ausprägungen von Awareness

Verschiedene Ausprägungen von Awareness werden meist durch neue technische Konzepte geformt. Auch die Beobachtung von Gruppenprozessen führte zur Definition verschiedener Formen von Awareness, beispielsweise in (MacMillan et al. 2004). Einige dieser Definitionen überschneiden sich oder beschreiben dasselbe Phänomen, die meisten bauen aber auf verschiedenen technischen Konzepten auf, und zielen teilweise auf verschiedene spezielle Problemstellungen ab. Einige dieser Ausprägungen sollen im Folgenden überblicksmässig dargestellt werden.

Context Awareness und Situation Awareness: Schmidt et al. (1999) definieren Context Awareness als „Wissen um den Zustand von AnwenderInnen und Geräten“. Um einen Kontext zu beschreiben, verwenden sie die drei Kategorien *environment* (physische und soziale Umgebung von AnwenderInnen oder Geräten), *self* (der Zustand der/des Anwenderin/Anwenders oder des Gerätes selbst) und *activity* (welche Aktivität führt der/die AnwenderIn gerade aus). Ein ähnliches Konzept stellt die von Antunes et al. (2010) erwähnte Situation Awareness dar, welche laut den AutorInnen eine Effizienzsteigerung bei der Erledigung von Aufgaben bewirken kann. Zur Beurteilung einer Situation erwähnen die AutorInnen drei Stufen: „*perception of elements in the current situation, comprehension of current situation and projection of future status.*“ (ibid, p. 8).

Location Awareness in (Spriestersbach et al. 2001) bezeichnet die Behaftung von mobilen Applikationen mit ortsbezogenen Daten. Die AutorInnen sehen Location Awareness als ein Subset von *Environmental Awareness*, welche wiederum ein Subset von *Context Awareness* darstellt. Weiters unterscheiden sie zwischen dem physischen Ort („physical location“) und dem semantischen Ort („semantic location“). Ein physischer Ort sind absolute Ortsdaten, wie beispielsweise eine Adresse oder GPS-Koordinaten, semantische Orte sind relative Positionen „in einem grösseren Kontext“ (ibid, p. 56), beispielsweise „im Konferenzraum“ oder „im Bus“. Eine andere Unterteilung von ortsbezogenen Awarenessarten nehmen Antunes et al. (2010) vor: Sie unterscheiden Awareness über „geographical space“, der sich auf dieselben Attribute wie die „physical location“ bei (Spriestersbach et al. 2001) bezieht und damit konkrete Ortsangaben, Distanzen und Orientierung beinhaltet, „physical space“, mit dem die AutorInnen Attribute in Bezug auf Mobilität beschreiben, und schließlich „virtual space“, welcher die Verortung von Daten in virtuellen Strukturen, z.B. Ordnernstrukturen, beschreibt.

Eine dritte Klassifizierung von Verortungen in Awarenesssystemen wird ebenfalls in (Antunes et al. 2010) festgehalten: „*same place, different place and any-place*“ (ibid, p.5), bzw. „*co-located, virtual-co-located and remote places*“ (ibid, p.6). Diese Unterscheidungen werden verwendet, um die Möglichkeiten eines direkten persönlichen, bzw. zumindest virtuellen Kontaktes zu beschreiben. Bei „same place“ und „co-located“ stehen die handelnden Personen in (sehr) nahem räumlichem Kontakt, „virtually co-located“ bedeutet Kontakt oder eine Kontaktmöglichkeit mittels eines (elektronischen) Mediums. Bei „different place“, „any place“ und „remote places“ ist ein direkter physischer Kontakt nicht ohne grösseren Aufwand möglich.

Change Awareness und **Chronological Awareness**: Diese beiden Awarenessarten behandeln das Wahrnehmen von Änderungen über die Zeit. Change Awareness wird von Tam und Greenberg (2006) als „*die Möglichkeit asynchrone Änderungen in kollaborativen Dokumenten oder grafischen Workspaces durch Dritte zu verfolgen*“ (ibid, p. 583) definiert. Die Möglichkeit zur Verfolgung soll dabei ohne mündliche oder schriftliche Informationen durch andere möglich sein. Ein ganz ähnliches Konzept stellt die von Espinosa et al. (2000) definierte Chronological Awareness dar, welche sich auf Dokumente in einer verteilten Umgebung bezieht, für die ersichtlich ist, „*was verändert wurde, wann und von wem*“ (ibid, p. 393). In (Antunes et al. 2010) wird Zeit gemeinsam mit einem Ort verschränkt als „time/place“-Beziehung.

Informational Awareness, Subscription-based Awareness, und Peripheral Awareness sind laut Brush et al. (2002) die drei wichtigsten Arten von Awareness, die von Dokumentenverwaltungssystemen (DMS) unterstützt werden. *Informational Awareness* bedeutet, dass AnwenderInnen über Änderungen in Dokumenten erfahren. Informationen über Änderungen können von DMS automatisch generiert werden, oder sie können von der verändernden Person explizit angegeben werden. *Subscription-based Awareness* erlaubt AnwenderInnen sich bei Dokumenten zu registrieren, um in der Folge bei Änderungen automatisch benachrichtigt zu werden. Änderungen können sofort pro Dokument, oder auch gesammelt über einen längeren Zeitraum für ein oder mehrere Dokumente verschickt werden, was erhöhtes Email-Aufkommen verhindern kann. Es besteht auch die Möglichkeit, sich nur über bestimmte Arten von Änderungen informieren zu lassen. *Peripheral Awareness* beschreibt Awarenessinformation über Aktivitäten anderer BenutzerInnen die mittels einer geeigneten Visualisierung in Echtzeit und peripher sichtbar gemacht werden.

Social Awareness im Arbeitsumfeld wird von Bardram und Hansen (2004) definiert als das Wissen um den Arbeitskontext von KollegInnen. Als Beispiel für computervermittelte Social Awareness geben die AutorInnen die „Buddy-“, oder Statusliste von Instant Messenger-Programmen (IM) an, bemerken allerdings, dass die Vermittlung von Awarenessinformation mittels IM sich oft auf die diskursiven Möglichkeiten des Mediums (d.h. einer Kommunikation mittels Chatnachrichten) verlässt, was in der Folge zu Unterbrechungen der Arbeit von KollegInnen führt. Laut den AutorInnen entsprechen solche Unterbrechungen nicht der ursprünglich von der CSCW-Forschung intendierten Form von sozialer Awareness. Um einen Arbeitskontext zwischen KollegInnen zu vermitteln, gibt es laut den AutorInnen zwei Konzepte: „Displaying“ und „Monitoring“: Monitoring bedeutet das Beobachten des Verhaltens von KollegInnen, Displaying bedeutet

„implicit or explicit signals a given actor uses to show specific aspects of his or her current situation, which could be useful or relevant for the other actors in the context“ (ibid, p. 193). Eine grosse Rolle bei der Vermittlung von sozialer Awareness spielt laut den AutorInnen der Faktor Distanz, da dadurch die Menge an vermittelbaren Kontextinformationen („rich social awareness“, (ibid, p. 194)) stark beeinflusst wird. Antunes et al. (2010) definieren das Wissen um (Gruppen-)Zugehörigkeit (Membership) und Aufmerksamkeit (Attention) als Awareness über „Social Space“.

Mutual Awareness, Taskwork Awareness, Workload Awareness, Teamwork Awareness: MacMillan et al. (2004) untersuchen *Mutual Awareness* – „das Ausmaß in dem Teammitglieder über das Verhalten anderer Teammitglieder Bescheid wissen“ – im Kontext von Zusammenarbeit in einer Gruppe von Armeeangehörigen. Laut den AutorInnen setzt sich Mutual Awareness aus drei Punkten zusammen: *Taskwork Awareness* (welche Aufgabe führen andere Teammitglieder zu einem bestimmten Zeitpunkt aus), *Workload Awareness* (wie stark sind andere Teammitglieder ausgelastet) und *Teamwork Awareness* (wie gut funktioniert die Zusammenarbeit im Team).

Task Awareness und Team Awareness: Espinosa et al. (2007) untersuchen Koordination in der Softwareentwicklung, mit besonderem Fokus auf den Unterschieden zwischen EntwicklerInnen am selben, bzw. an verschiedenen Orten. Die AutorInnen vertreten die Meinung, dass Koordination durch verschiedene Arten von Wissen ermöglicht wird. Sie unterscheiden dabei einerseits zwischen Langzeitwissen und Kurzzeit- bzw. flüchtigem Wissen, welches sie auch „Awareness“ nennen. Andererseits unterscheiden sie noch zwischen Wissen um die Aufgabe („Task“) und Wissen um das Team, bzw. um Prozesse. Diesen Unterscheidungen folgend, gibt es also vier Kategorien von Wissen bzw. Awareness: „Task Knowledge“, „Team Knowledge“, „Task Awareness“ und „Team Awareness“. Weiters identifizieren sie noch drei Aspekte von Teamarbeit, die koordiniert werden müssen: technische, zeitliche und prozessorientierte. Ihre Untersuchung zeigt, dass diese drei Aspekte abhängig von der Rolle der Teammitglieder unterschiedlich wichtig sind: Für Manager sind zeitliche und prozessorientierte Koordination wichtig, für Entwickler hauptsächlich technische Koordination. Außerdem stellen sie fest, dass für Teams an verschiedenen Orten Team Knowledge und Team Awareness noch wichtiger sind als Task Knowledge und Task Awareness, bzw. dass es dabei am häufigsten zu Problemen kommt. Bei Teams am selben Ort ist die größte Herausforderung Task Knowledge, Teamwissen ist eher ein kleineres Problem. Task Awareness ist laut den AutorInnen eher ein geringeres Problem, da dieses Wissen großteils bereits durch moderne Softwareentwicklungsumgebungen und Versionsverwaltungssysteme zugänglich gemacht wird.

Anticipative Awareness: Prinz, Hinrichs und Kireyev (2010) beschreiben ein Konzept, das sie „Expectations“ („Erwartungen“) oder auch „teilnehmende Awareness“ nennen. Dabei kann man Dokumente in einem CSCW-System (in dem Projekt wurde BSCW verwendet) mit Objekten verknüpfen, welche überprüfen, ob in einem bestimmten Zeitraum (z.B. ab Erstellung eines Dokumentes bis zu einem spezifizierten Zeitpunkt) bestimmte Aktionen (z.B. Lesen, Editieren) durch bestimmte User durchgeführt wurden. Am Ende des spezifizierten Zeitraums bekommt der Ersteller der „Expectation“ eine Nachricht, ob die Erwartung eingetroffen ist oder nicht. In der Evaluierung mittels Tests durch häufige BenutzerInnen solcher Systeme und einer anwendungsnahen Implementierung zeigte sich, dass dieses Konzept alternativ zur Erstellung von sog. Tasks (also beispielsweise das Erstellen und Zuweisen einer bestimmten Aufgabe (z.B. „lesen sie dieses Dokument“) an eine bestimmte Person) verwendet werden könnte.

Social Awareness Stream: Naaman, Boase und Lai (2010) bezeichnen Systeme wie Twitter oder den Facebook-Newsfeed auch als „Social Awareness Streams“. Diese zeichnen sich durch drei wesentliche Merkmale aus: 1.) Die Kommunikation, bzw. Nachrichten sind öffentlich bis halböffentlich, 2.) Nachrichten sind relativ kurz, und 3.) die Plattformen sind „a highly connected social space, where most of the information consumption is enabled and driven by articulated online contact networks“, d.h., die Kommunikation findet vorwiegend online statt und organisiert sich rund um diese Plattformen und die darin bestehenden (Personen-)Netzwerke.

2.3 Modelle von Awareness

Als grundlegende Gemeinsamkeit von Awarenesssystemen nennen Gross und Koch (2007), dass diese die Erfassung, Verarbeitung und Präsentation von Awarenessinformationen ermöglichen. Im **Modell für Awareness-Informationsverarbeitung** (siehe Abb. 1) von Gross und Koch (2007) handelt es sich bei Awarenessinformationen um Ereignisse (beispielsweise Benutzeraktionen), welche mittels Sensoren erfasst, mittels eines Client-Server-Modelles verarbeitet und verfügbar gemacht, und schließlich von *Indikatoren* angezeigt werden.

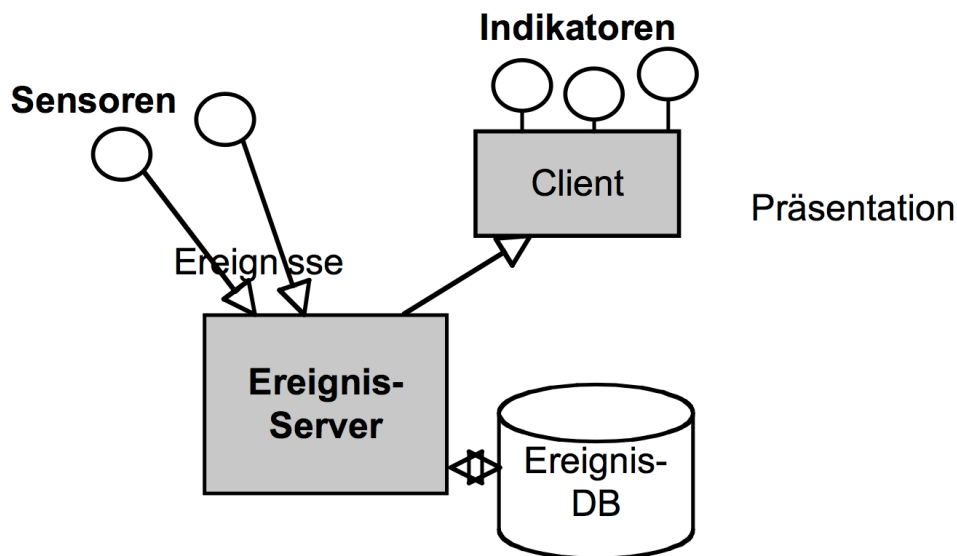


Abb. 1 Modell für Awareness-Informationsverarbeitung (Gross & Koch 2007, p.60).

Die **Awareness-Pipeline** (siehe Abb. 2) von Gross und Koch (2007) stellt den Fluss von der Ereignissentstehung zur Ereignispräsentation dar. Zwischen Entstehung und Präsentation können drei Arten von „Filtern“ angelegt werden, die einigen wesentlichen Anforderungen für Informationssysteme Rechnung tragen (nach (Gross & Koch 2007)):

- Die Ereignis-ErzeugerInnen können Filter zum Schutz ihrer Privatsphäre anlegen und somit bestimmen, welche Informationen von ihnen preisgegeben werden.
- Eine Organisation kann Regularien beispielsweise zum Datenschutz oder zu Vertraulichkeitsstufen anlegen.
- Ereignis-KonsumentInnen können festlegen, für welche Ereignisse sie sich interessieren.

Des Weiteren halten Gross und Koch (2007) fest, dass der Informationsfluss in der Awarenesspipeline *reziprok* sein kann, d.h., dass Ereignis-ErzeugerInnen gleichzeitig auch Ereignis-KonsumentInnen sein können, der Informationsfluss also in beiden Richtungen stattfinden kann.

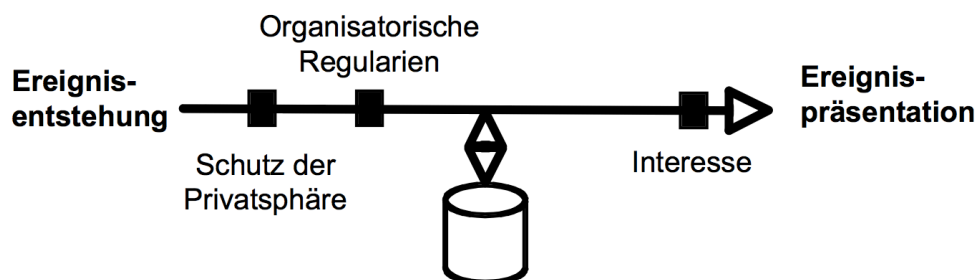


Abb. 2 Awareness-Pipeline (Gross & Koch 2007, p.61).

Gross und Koch (2007)) bieten drei verschiedene konzeptionelle Modelle an, um verschiedene Ausprägungen von Awareness beschreiben zu können: räumliche-, ereignisbasierte- und Kontextmodelle. Diese drei Modelle sollen im Folgenden kurz beschrieben werden.

Räumliche Modelle bilden räumliche Gegebenheiten zwischen verschiedenen Objekten in einem System ab. Nach Gross und Koch (2007) kann es sich bei diesen Objekten um menschliche Akteure oder Artefakte handeln. Awareness bezieht sich dann auf das Wahrnehmen der (räumlichen) Präsenz zwischen Objekten innerhalb räumlicher Gegebenheiten. Laut Gross und Koch (2007) kann diese Wahrnehmung realisiert werden, indem Objekte eine „gedankliche Aura“, sowie einen Fokus besitzen: *„Dabei sind Objekte gedanklich von einer Aura umgeben, d.h. einem Bereich in dem sie wahrgenommen werden. Objekte haben auch einen Fokus, d.h. einen Bereich den die wahrnehmen können. Überlappende Auren und Foci bestimmen dann die Awareness zwischen Objekten.“* (ibid, p. 62). Laut Gross und Koch (2007) gibt es verschiedene Systeme, die weitergehende räumliche Modelle umsetzen, die aber laut den AutorInnen grossen Aufwand im Betrieb dieser Systeme bedeuten.

Ereignisbasierte Modelle: Hier bezieht sich Awareness auf die Wahrnehmung von Ereignissen in Systemen, Gruppen oder Artefakten. Erste Systeme, die dieses Modell umsetzten, verwendeten laut Gross und Koch (2007) einfaches Logging aller User- oder Systemaktionen, welche in der Folge präsentiert werden konnten. In aufwendigeren Systemen werden laut Gross und Koch (2007) nicht nur einzelne Aktionen geloggt, sondern es werden Situationen als Objekte und Verbindungen zwischen diesen Objekten modelliert. Spezifizierbare Ereignisse innerhalb dieser Situationen können dann als Awarenessinformationen an interessierte BenutzerInnen weitergegeben werden (ibid). Gross und Koch (2007) halten für diese aufwendigere Form von ereignisbasierten Modellen fest, dass diese aufwendig in der Modellierung sind, und auch nur schwer wieder verwendet werden können, da sie für jede Anwendung individuell angepasst werden müssen.

Kontextmodelle: In Kontextmodellen wird versucht, den Kontext von Ereignissen oder Objekten in die Awarenessvermittlung miteinzubeziehen. *„Kontextmodelle führen den Begriff des Kontexts als die verwobenen Bedingungen ein, in denen etwas existiert oder sich ereignet. Sie unterscheiden typischerweise den Entstehungskontext, in dem sich etwas ereignet und den Präsentationskontext, in dem sich der interessierte Benutzer aktuell befindet.“* (Gross & Koch 2007, p.63). Als grossen Vorteil von Kontextmodellen sehen Gross und Koch (2007), dass diese semantische Informationen ausnützen können, den Nachteil sehen sie in einem erhöhten Aufwand für die BenutzerInnen. So muss beispielsweise beim Durchführen von Handlungen angegeben werden, warum diese Handlung durchgeführt wird: *„Kontextoren beschreiben Aktionen innerhalb eines spezifischen Kontexts. Die Benutzer klassifizieren ihre Aktionen und Artefakte manuell, indem sie ihnen Kontextoren zuweisen, d. h. wenn ein Benutzer eine Handlung vollzieht wird sie oder er automatisch vom System nach einem Kontextor gefragt (Rittenbruch, 1999).“* (Gross & Koch 2007, p.63).

2.4 Mechanismen für Awarenessverarbeitung

In der Literatur und in Forschungsarbeiten zu Awareness kommen verschiedene technische, organisatorische und kommunikative Mechanismen zum Einsatz. Diese Mechanismen legen fest, wie mit den Informationen in einem Awarenesssystem gearbeitet werden kann, bzw. wie Informationen verbreitet werden. Auch diese Mechanismen kommen in der Praxis kaum isoliert vor, sondern werden meist in Kombination verwendet. Im Folgenden soll versucht werden, einen Überblick über verschiedene Mechanismen zu geben.

Aktiv/Passiv, Implizit/Explizit: Bei der Betrachtung von Awarenesssystemen und Mechanismen ist zu unterscheiden zwischen der Erzeugung, der Weitergabe und der Präsentation von Awarenessinformationen, was von Gross und Koch (2007) in der bereits beschriebenen Awareness-Pipeline hervorgehoben wird. In diesem Zusammenhang spielen die vier Begriffe aktiv, passiv, implizit und explizit eine wichtige Rolle, da sie die Art, wie

sowohl menschliche BenutzerInnen als auch technische Systeme (Awareness-) Informationen generieren, weitergeben und verarbeiten, grundlegend beeinflussen.

Beim Erzeugen von Awarenessinformation kann dies entweder aktiv durch eineN BenutzerIn geschehen, indem dieseR selbst ihren/seinen Status in ein technisches System eingibt. Wenn die Informationserzeugung ohne Zutun von menschlichen BenutzerInnen beispielsweise durch einen Sensor geschieht, soll in diesem Zusammenhang von einer passiven Erzeugung gesprochen werden. Gleiches gilt für den Austausch von Informationen, hier muss man allerdings weiter differenzieren zwischen InformationserzeugerInnen und EmpfängerInnen. So können BenutzerInnen Informationen aktiv und explizit an andere BenutzerInnen oder auch technische Systeme weiterleiten, bzw. kann Information aktiv und explizit von anderen BenutzerInnen und Systemen abgefragt werden. Bei der passiven Weitergabe von Informationen übernimmt wiederum ein technisches System diese Verbreitung, ohne dass das explizite Zutun von BenutzerInnen nötig wäre. Bei der Verarbeitung und Interpretation von Awarenessinformationen schließlich kann es sich um explizite und implizite, aktiv oder passiv erzeugte Informationen handeln, wobei explizite Informationen beispielsweise einer genauen Statusangabe entsprechen, implizite Informationen sich aus dem Kontext ergeben, beispielsweise aus der Historie eines Artefaktes.

Informationspulling und Informationspushing: Die Mechanismen Informationspulling und Informationspushing werden beispielsweise von Gross und Koch (2007) und von Convertino et al. (2008) genannt. Informationspulling beschreibt das Konzept, dass bei zwei oder mehreren Personen oder Objekten dasjenige, das etwas wissen will, aktiv diese Informationen direkt bei den anderen Objekten, oder bei einer zwischengeschalteten Instanz abrufen. Es kommt also nur nach expliziter Anforderung zu einem Informationsaustausch. Beim Informationspushing wird ein Objekt, welches Informationen besitzt, von sich aus aktiv, und gibt diese Informationen entweder an alle oder an ausgewählte Objekte. Ein Informationsaustausch findet statt, wenn der/die InformationsbesitzerIn dies für sinnvoll erachtet. Ob oder wie die EmpfängerInnen diese Information verwerten, bleibt ihnen überlassen. Convertino et al. (2008) zeigen, dass bei der Erarbeitung einer gemeinsamen Wissensbasis in einer Gruppe diese beiden Mechanismen zum Tragen kommen, dass Informationspushing dabei aber der effizientere Weg ist.

Checking: Checking in der Kommunikation bedeutet Rückfragen, Verifizieren und Vergewissern. Checking kann somit auch als eine beschränkte Form von Informationspulling gesehen werden. Checking wird beispielsweise nötig, wenn der Kontext von vorhandenen Informationen nicht klar ist, oder wenn die vorhandenen Informationen unvollständig, bzw. für die InformationsempfängerInnen nicht ausreichend sind. Checking kann auch gesehen werden als eine „komprimierte“ Form von Informationspulling, das sich auf ein vorangegangenes Kommunikationsereignis, bzw. auf bestehende Informationen bezieht. Convertino et al. (2008) und Convertino et al. (2009) vermuten, dass für weniger komplexe Gruppenaufgaben weniger Checking notwendig ist als für komplexere Gruppenaufgaben.

Polling: Polling ist ebenfalls eine Form des Informationspullings, allerdings erfolgt eine Abfrage nach der Definition von King und Lyons (2011) nicht direkt bei der anderen Person, sondern über eine Informationsplattform, an der BenutzerInnen aktiv Statusinformationen von anderen AnwenderInnen nachsehen bzw. abfragen müssen.

Displaying und Monitoring: Bardram und Hansen (2004) definieren die Begriffe Displaying und Monitoring als Mechanismen für die Übermittlung von Informationen über den Arbeitskontext. Monitoring bedeutet das Beobachten des Verhaltens von KollegInnen, Displaying bedeutet „*the implicit or explicit signals a given actor uses to show specific aspects of his or her current situation, which could be useful or relevant for the other actors in the context*“ (ibid, p. 193).

Subscription: Subskriptions- oder abonnementbasierte Awarenessmechanismen erlauben es den InformationskonsumentInnen sich an einem Awarenessserver oder Dokument für bestimmte Ereignisse oder eine bestimmte Art von Informationen zu registrieren, um in der Folge bei Änderungen automatisch benachrichtigt zu werden (Brush et al. 2002).

Änderungen können sofort pro Dokument, oder auch gesammelt über einen längeren Zeitraum für ein oder mehrere Dokumente verschickt werden, was erhöhtes Email-Aufkommen verhindern kann. Es besteht auch die Möglichkeit, sich nur über bestimmte Arten von Änderungen informieren zu lassen (ibid), in diesem Fall entspräche dieser Mechanismus den in der Awareness-Pipeline vorkommenden *Interessensfiltern* für KonsumentInnen von Awarenessinformation.

Notification: *Notifications* oder Benachrichtigungen sind eine Möglichkeit, über Änderungen in einem System informiert zu werden. Brush et al. (2002) nennen als Benachrichtigungsformen Email und periphäre Informationsanzeigen.

Periphäre Informationsanzeige: Periphäre Informationsanzeigen gehören zur Gruppe von Benachrichtigungssystemen. Solche Anzeigen sind periphär sichtbar - d.h. zum Beispiel am Rande des Bildschirms oder in einer Symbolleiste – wodurch sie wenig Platz einnehmen und solange aus dem Blickfeld verschwinden, bis sich die angezeigte Information ändert. Informationen werden in Echtzeit oder synchron übertragen. Die Menge der anzeigbaren Informationen ist eher gering, daher ist laut Brush et al. (2002) beim Design solcher Anzeigen auf eine geeignete Visualisierung zu achten.

Filter: Filter stellen eine Möglichkeit dar, die in der Awareness-Pipeline gestellten Forderungen nach Privatsphäre, bzw. Selbstbestimmtheit bei der Freigabe von Informationen, organisatorischen Einschränkungen sowie der Möglichkeit für BenutzerInnen, die Menge von angezeigten Informationen zu steuern, umzusetzen. Filter können manuell oder auch automatisch kontextbasiert erstellt werden. Eine Art von Filterung liegt auch jedem technischen System durch seinen Funktionsumfang zu Grunde, indem dieser Funktionsumfang festlegt welche Informationen verfügbar sind. Gross und Koch (2007) nennen kontextbasierte Filter als eine Möglichkeit, den „Aufwand beim Konsumieren von Awareness-Information“ (ibid, p.70) zu senken.

Sensoren: Gross und Koch (2007) nennen Sensoren als Mechanismen für das Registrieren oder Erfassen von Awarenessinformationen. Somit „tracken“ sie gewissermaßen bestimmte, an einem System registrierte, oder im Kontext eines Systems stehende Entitäten. Gross und Koch (2007) nennen Sensoren als Teile von Awareness-Informationssystemen, spezifizieren aber nicht näher, ob es sich dabei um Software oder Hardware handelt, sondern verstehen den Begriff als abstraktes Konzept.

2.5 Inhalt von Awareness - Awarenessbedürfnisse

Bei Betrachtung der verschiedenen angeführten Modelle und Mechanismen von Awareness wird ersichtlich, dass diese abstrakte Konzepte sind, die konkrete inhaltliche Aspekte außen vor lassen. Auch die verschiedenen vorgestellten Ausprägungen von Awareness unterscheiden sich häufig nur in ihrer Benennung, beschreiben aber inhaltlich dieselbe Art von Awareness. Deshalb soll im Folgenden versucht werden, die angeführten Modelle und Ausprägungen von Awareness auf ihre Inhalte hin näher zu untersuchen. In der Literatur wird in diesem Zusammenhang auch von **Awarenessbedürfnissen** („Awareness needs“) gesprochen, damit ist jenes Wissen gemeint, das jemand über etwas anderes benötigt.

Konkrete Entitäten: Wie man anhand der verschiedenen Ausprägungen und Modelle von Awareness sehen kann, bezieht sich Awareness stets auf *etwas*. Dieses *etwas* können konkrete Entitäten sein, über deren Zustand, Zustandsänderungen oder Aktivitäten man Bescheid wissen möchte. Beispiele für solche konkreten Entitäten im Kontext der CSCW-Forschung sind physische oder digitale Artefakte und Artefaktgruppen (Gross & Koch 2007), Dokumente ((Espinosa et al. 2000), (Tam & Greenberg 2006), (Brush et al. 2002)), einzelne Personen oder Gruppen und Teams ((Schmidt et al. 1999), (MacMillan et al. 2004)), Geräte (Schmidt et al. 1999), Aufgaben (Espinosa et al. 2007), Treude & Storey 2010) oder reale oder virtuelle Arbeitsplätze (Tam & Greenberg 2006).

Komplexe Sachverhalte: Neben diesen konkreten, mehr oder weniger greifbaren und sichtbaren Entitäten, über die man Awareness erlangen will, gibt es auch komplexere Dinge oder Sachverhalte, über die man Awareness erlangen möchte wie Projekte (Treude &

Storey 2010), Teamwork (MacMillan et al. 2004) oder Arbeitskontext (Bardram & Hansen (2004). Diese Begriffe zeichnen sich dadurch aus, dass sie sich meist nicht mittels einzelner Entitäten oder Aktivitäten beschreiben oder operationalisieren lassen, sondern dass sie sich aus einer Vielzahl formaler und informeller Aspekte zusammensetzen. Treude und Storey (2010) verwenden die beiden Begriffe **High-Level** und **Low-Level Awareness**, um eine sich solchermaßen zusammensetzende Awareness zu beschreiben: High-Level Awareness meint dabei Awareness über den gesamten Projektstatus in einem Softwareprojekt, welche sich wiederum aus Awareness über einzelne, kleinere Teilaufgaben, Änderungen im Sourcecode oder erfolgreiche und gescheiterte Tests zusammensetzt, welche sie mit „Low-Level Awareness“ bezeichnen: *„Rather than source code alone, the status of the project arises from an aggregation of data on open and closed development tasks, successful and failed builds, delivered and pending changes, and successful and failed tests as well as evolutionary information.“* (Treude & Storey 2010, p.365) und weiter: *„There is still a lack of understanding of how to achieve highlevel awareness of project management issues with low-level awareness of more fine grained activities such as source code changes and development task creation.“* (ibid).

Kontext: Hilfreich bei der Betrachtung sowohl von einfachen als auch von komplexeren Awarenessbegriffen ist der schon in den Kontextmodellen besprochene **Kontext**, in dem sowohl die interessierenden als auch die interessierten Entitäten stehen. Gross und Koch (2007) sprechen dabei von „**Entstehungskontext**“ und „**Präsentationskontext**“. Beim Entstehungskontext einer Entität kann es sich also beispielsweise um den Ort handeln, an dem sich Dokumente oder Personen befinden (Spriestersbach et al. 2001) oder auch um die Gründe für Aktivitäten oder Handlungen (siehe z.B. (Tam & Greenberg 2006)). Das Wissen um diesen Kontext kann beispielsweise dazu beitragen, eine konkrete (Awareness-) Information nachvollziehbar zu machen. Der Präsentationskontext wiederum legt fest, welche Informationen in einer bestimmten Situation gewünscht sind, bzw. von Nutzen sind. So zeigen Espinosa et al. (2007), dass in einem Team abhängig von der Rolle der Mitglieder bestimmte Aspekte von Koordination wichtiger sind als andere.

Zustand und Ereignisse: Schließlich ergibt sich die Frage, was man über einzelne Entitäten, Sachverhalte oder deren Kontext wissen möchte. Gutwin, Greenberg und Roseman (1996) formulieren eine Liste von Fragen, die man über einen Arbeitskontext wissen möchte (siehe Abb. 3).

Element	Relevant Questions
Identity	Who is participating in the activity?
Location	Where are they?
Activity Level	Are they active in the workspace? How fast are they working?
Actions	What are they doing? What are their current activities and tasks?
Intentions	What are they going to do? Where are they going to be?
Changes	What changes are they making? Where are changes being made?
Objects	What objects are they using?
Extents	What can they see?
Abilities	What can they do?
Sphere of Influence	Where can they have effects?
Expectations	What do they need me to do next?

Abb. 3 Elemente von Workspace Awareness (Gutwin, Greenberg & Roseman 1996, p.7)

Bei den Fragen, die dabei an die Entitäten gerichtet werden, handelt es sich um Fragen zum **Zustand** der Entitäten, bzw. deren Kontext, sowie zu **Ereignissen** (hier vor allem Änderungen) innerhalb eines Kontextes. Fragen, die den Zustand betreffen, richten sich vor allem an die Zusammensetzung, bzw. das Gesamtbild eines Kontextes, beispielsweise welche Personen sind beteiligt, was sind ihre Fähigkeiten und Intentionen, welche Artefakte oder Aktivitäten gibt es. Mit Ereignissen sind hauptsächlich Änderungen am Zustand einer Entität oder eines Kontextes gemeint. Ereignisse bilden damit auch die Basis für das bereits erwähnte Modell für Awareness-Informationsverarbeitung sowie die Awareness-Pipeline (beides in (Gross & Koch 2007)), da Ereignisse an sich technisch leicht zu erfassen sind, da sie – wenn man auf den zuvor eingeführten Begriff von High-Level und Low-Level Awareness zurückgeht – im Prinzip die „niedrigste“ Maßeinheit darstellen.

Fragen an Entitäten: Tam und Greenberg (2006) definieren weiters eine Liste von sechs Fragen über Änderungen bezogen auf Dokumente und Arbeitsplätze: *Wer* hat eine Änderung gemacht, *was* wurde geändert, *wo* fand die Änderung statt, *wann* wurde die Änderung gemacht, *wie* wurden Dinge geändert und *warum* wurde eine Änderung gemacht.

Diese Fragen in Bezug auf Änderungen an einer Entität oder eines Kontextes besitzen laut den AutorInnen zwei Eigenschaften: Zum einen gibt es eine **hierarchische Ordnung** in der Fragestellung, begonnen mit der höchsten Frage „Gibt es irgendwelche Änderungen?“, bis hin zu detaillierteren Fragen, wie „Warum gibt es eine Änderung“. Die zweite Eigenschaft ist, dass solche Fragen aus einer bestimmten **Perspektive** heraus gestellt werden können, d.h. Personen interessieren sich möglicherweise an bestimmten Entitäten innerhalb eines Kontextes: beispielsweise personenzentriert („Welche Personen befinden sich in einem Kontext?“), artefaktzentriert („Welche Artefakte existieren in einem Kontext?“) oder arbeitsplatzzentriert (z.B. „Welche Orte gibt es in einem bestimmten Kontext?“).

Schließlich ist zu beachten, dass es bei all diesen Fragen zum Zustand und zu Ereignissen auch eine **zeitliche Komponente** gibt. So können sich Fragen auf den momentanen Zustand beziehen (z.B. „Wer nimmt teil?“), auf die Vergangenheit, bzw. bereits geschehene Ereignisse („Was wurde geändert?“), aber auch auf zukünftige Vorhaben oder Intentionen („Was werden sie machen?“, „Was soll ich als nächstes machen?“).

2.6 Faktoren für Awareness

In der bestehenden Literatur über Awareness werden verschiedene Faktoren genannt, die bei der Entstehung von Awareness, bzw. bei der Arbeit mit Awarenesssystemen zu berücksichtigen sind und die auch sehr stark verschiedene Awarenessbedürfnisse und somit den Inhalt von Awareness beeinflussen. Diese sollen nachfolgend beschrieben und diskutiert werden.

Informationsmenge: Die Menge oder der Umfang von Informationen spielt vor allem bei der Präsentation von Awarenessinformationen eine wichtige Rolle. Zum einen kann es technische Beschränkungen in der Menge der anzeigbaren Informationen geben – beispielsweise durch Beschränkungen des vorhandenen Bildschirmplatzes – andererseits muss es für BenutzerInnen auch möglich sein, die angezeigten Informationen zu verarbeiten. Tam und Greenberg (2006) bemerken, dass die praktische Umsetzung von Change Awareness-Funktionalitäten in einem Userinterface nicht trivial ist, da beispielsweise die alleinige Anzeige aller Änderungen in einem System die BenutzerInnen überfordern könnte. Laut Lim, Brdiczka und Bellotti (2010) kann dieser „cognitive load“ bei den Rezipienten von Awarenessinformationen durch eine Abstraktion von Informationen umgangen werden. Technische Systeme, die Informationen semantisch verstehen können, können hier hilfreich sein. Holmes und Walker (2010) entwickeln ein System, welches vorhandene Awarenessinformationen in einer Softwareentwicklungsumgebung – in Form von Commitnachrichten in einem Softwarerepository – automatisch nach ihrer Relevanz für einzelne EntwicklerInnen filtert, und weiters nach ihren möglichen Auswirkungen auf die Funktionsfähigkeit von Codesegmenten einstuft. Der Herausforderung in Punkto technischer Beschränkungen, zum Beispiel durch beschränkten Platz am Bildschirm, kann durch konzeptionelle Mechanismen, wie beispielsweise Abstraktion und geeignetes Informationsdesign, begegnet werden.

Zeitpunkt und Häufigkeit: Der Zeitpunkt spielt sowohl bei der Präsentation von Awarenessinformationen durch Benachrichtigungssysteme als auch beim Sammeln von Information durch Pullmechanismen eine Rolle. In beiden Fällen können dadurch nämlich Unterbrechungen von BenutzerInnen hervorgerufen werden, bzw. nötig machen, was von mehreren AutorInnen ((Gross & Koch 2007), (Lim, Brdiczka & Bellotti 2010), (Tang & Birnholtz 2010)) als Problem erkannt wird. Auch Brush et al. (2002) nennen die Häufigkeit von Benachrichtigungen als einen wichtigen Faktor beim Design von Informationssystemen. Su und Mark (2008) bemerken, dass der Wechsel zwischen verschiedenen Kommunikationsereignissen und Aufgaben Stress verursachen, und dazu führen kann, dass weniger Zeit für die eigentliche Arbeit („solitary work“ – in diesem Zusammenhang als Gegensatz zu „Kommunikationsarbeit“) bleibt.

Gedächtnis: Dix et al. (2004) erwähnen das menschliche Gedächtnis als einen grundlegenden Faktor für den Fachbereich der Mensch-Computer-Interaktion, wozu auch Awarenessinformationssysteme zählen, da die unterschiedlichen Eigenschaften von Lang- und Kurzzeitgedächtnis die menschlichen Wahrnehmungs- und Verarbeitungsmöglichkeiten beeinflussen. Lim, Brdiczka und Bellotti (2010) sprechen den Faktor menschliches Gedächtnis an, indem sie das Problem von zu grosser „cognitive load“ bei der Verarbeitung von Awarenessinformationen behandeln: „...*this corroborates that simplifying awareness information can reduce cognitive load on users and increase their rate of assimilating that information.*“ (ibid, p. 269). Dieses Problem steht auch in direktem Zusammenhang mit dem oben genannten Faktor *Informationsmenge*.

Distanz: Laut Bardram und Hansen (2004) spielt Distanz eine wesentliche Rolle bei der Vermittlung von sozialer Awareness, da dadurch die Menge an vermittelbaren Kontextinformationen („*rich social awareness*“, (ibid, p. 194)) stark beeinflusst wird. Auch Schmidt und Rodden (1996) heben die Auswirkungen von Distanz auf die Bandbreite von Kommunikation im Vergleich zu direkter Face-to-Face-Kommunikation hervor, betonen aber, dass geeignete technische Systeme diese Auswirkungen kompensieren können: „*Distance in space is only an issue in so far as the techniques of communication available for interaction across that space offers restricted bandwidth and delayed feedback compared with the full capacity of face-to-face interaction. By providing techniques of communication*

in the form of file sharing, shared view, email, computer conferencing, and video conferencing that increase the bandwidth and reduce the turnaround time of communication, CSCW systems can augment the capacity of cooperating ensembles of articulating their distributed activities in spite of distance.“ (Schmidt & Rodden 1996, p.159). Auch Brush et al. (2009) bestätigen, dass es besonders bei entfernten KollegInnen schwierig ist, über deren Status auf dem Laufenden zu bleiben. Brush et al. (2009) führen zudem eine interessante Unterscheidung zwischen verteilten („*remote*“) und nicht verteilten („*collocated*“) KollegInnen ein: „Nicht verteilt“ definieren sie als *"people who you work with and who you would visit in person (rather than by phone, email, or instant message) if you had an important question that was best resolved by immediately talking with your coworker."* (ibid, p. 2092). Verteilte KollegInnen definieren sie als *"coworkers that you work with on a project or need to interact with frequently, but who do not qualify as collocated coworkers. For these coworkers, you would typically phone, email or instant message them, rather than go to their desk if you have an important question for them. For example, they may be coworkers who are located on another floor, in another building, in another city, or in another time zone."* (ibid, p. 2092). Hierbei ist bemerkenswert, dass also auch KollegInnen die sich nur in einem anderen Stockwerk befinden, bereits als „entfernt“ bezeichnet werden. In der Folge machen die AutorInnen auch die Beobachtung, dass mehr Informationen besonders über solche KollegInnen gewünscht werden, die ohnehin in der Nähe sind: *„A few steps away from your primary work location“* (ibid, p. 2094).

Privatsphäre: Beim Design von Awarenessinformationssystemen muss beachtet werden, dass BenutzerInnen ein Recht auf ihre informationelle Selbstbestimmung haben, bzw. dass Systeme, welche diese Forderung ignorieren, möglicherweise von den BenutzerInnen nicht akzeptiert werden. King und Lyons (2011) fanden in einer Studie über ein Awarenesssystem, welches automatisch bestimmte Informationen über BenutzerInnen sammelt und verbreitet, heraus, dass sich viele AnwenderInnen zwar mehr Awarenessinformationen von KollegInnen wünschen, selbst aber häufiger Bedenken haben, Informationen über sich selbst preiszugeben. Auch Tellioglu (2010) meint, dass es nicht zu einer Asymmetrie kommen dürfe: *„Though, less awareness information would be better to avoid participant asymmetry. Coordination 2.0 must consider this problem and offer user-configurable awareness feature in instance messaging tools.“* (ibid, p.5). Lim, Brdiczka und Bellotti (2010) wollen das Problem von mangelnder Privatsphäre mittels Abstraktion von vorhandenen Informationen durch semantische Analysen lösen.

Transparenz von Kommunikation: Für die Gestaltung von Systemen mit automatischen Benachrichtigungsmechanismen - wie sie in vielen Awareness-Systemen verwendet werden - nennen Brush et al. (2002) als einen von mehreren wichtigen Faktoren Transparenz, in dem Sinne, als ersichtlich sein sollte, wer benachrichtigt wird, um doppeltes Verschicken von Nachrichten zu vermeiden: *„If users cannot see who is subscribed for notifications they may send e-mail about important changes and duplicate automatic notifications.“* (ibid, p.4).

Transparenz von Informationen: Transparenz im Sinne von Offenlegung von Informationen aller Art betreffend einer kollaborativen Arbeit und an dieser Arbeit beteiligter Entitäten (KollegInnen, Artefakte, Orte, Aufgaben, etc.) ist eigentlich das zentrale Thema von Awareness, da ohne eine solche Offenlegung keine Informationen ausgetauscht werden können. Souza et al. (2007, p.154) beispielsweise nennen Transparenz und Offenlegung von Informationen über den Status und die Aktivitäten von ArbeitskollegInnen gemeinsam mit geeigneten, leicht zugänglichen, offen kommunizierten Visualisierungen als Grundlage für (verbesserte) Awareness. Somit steht Transparenz aber auch in direktem Zusammenhang - und unter ungünstigen Umständen sogar im Gegensatz - mit dem Faktor Privatsphäre. Auch der Faktor „Transparenz von Kommunikation“ könnte als ein Teilaspekt von „Transparenz von Information“ gesehen werden, da das Wissen um Kommunikationsvorgänge ja auch eine Art von Information darstellt.

Aufwand / Nutzen - Technisch: Das Verhältnis von Aufwand und Nutzen bei der Bedienung von technischen Systemen spielt auch für Awarenesssysteme eine Rolle. *„Bei der Bereitstellung von Awareness-Information muss wie bei allen anderen Groupware-Diensten auf ein möglichst gutes Aufwand/Nutzen-Verhältnis geachtet werden. D.h. der Aufwand für das Bereitstellen der Information muss (für den Benutzer) möglichst niedrig sein und der Aufwand für das Wahrnehmen der Information für den Konsumenten*

möglichst gering (bei möglichst hohem Nutzen für beide).“ (Gross & Koch 2007, p.70). Auch Larsson et al. (2010) betonen, dass indem der (technologische) Aufwand für Wissens-Teilung verringert wird, das Ausmaß von Wissens-Teilung steigt.

Bei der Steuerung von Aufwand und Nutzen spielen die Begriffe *aktiv* und *passiv*, bzw. *explizit* und *implizit* eine große Rolle, da sie sowohl den menschlichen Aufwand, als auch den technischen Aufwand beeinflussen: Ein aktives / explizites Bedienen von technischen Systemen durch menschliche BenutzerInnen bedeutet inhärent einen höheren Aufwand für diese BenutzerInnen, kann aber auch mit einem höheren Nutzen verbunden sein, da zum Beispiel gezielter nach Information gesucht werden kann. Bei einer passiven Benutzung von Awarenessinformationssystemen übernimmt ein technisches System die Erzeugung der Informationen und macht diese für andere BenutzerInnen sichtbar. Die Erzeugung erfolgt entweder, indem es Aktionen von BenutzerInnen protokolliert, oder indem inhärent über die BenutzerInnen verfügbare Daten verwendet werden. Eine solche passive Erzeugung von Informationen verlangt aber zumeist auch ein aufwendigeres technisches System. Lim, Brdiczka und Bellotti (2010) zum Beispiel entwickeln ein Tool, welches den Inhalt jener Dokumente semantisch analysiert, die einE bestimmteR BenutzerIn bearbeitet. Daraus leiten sie die Aktivität der Benutzerin / des Benutzers in einer abstrahierten Form ab, und geben diese Aktivität in der Folge als Awarenessinformation auch für andere BenutzerInnen weiter. Holmes und Walker (2010) entwickeln ein System, welches vorhandene Awarenessinformationen in einer Softwareentwicklungsumgebung – in Form von Commitnachrichten in einem Softwarerepository – automatisch nach ihrer Relevanz für einzelne EntwicklerInnen filtert, und weiters nach ihren möglichen Auswirkungen auf die Funktionsfähigkeit von Codesegmenten einstuft.

Nutzen - Informatorisch: Espinosa et al. (2000) zeigen, dass der Nutzen von bestimmten Awarenessinformationen stark von der zu erfüllenden Aufgabe abhängt und den Erfolg sowohl positiv wie auch negativ beeinflussen kann. Die AutorInnen untersuchten ein Awarenessstool für verteiltes, asynchrones Arbeiten, welches einer Arbeitsgruppe zwar einerseits half mittels eines Tools schneller zu einer Lösung zu finden, andererseits war aber die Qualität der Lösung schlechter, da laut den AutorInnen die zu Verfügung stehenden Informationen weniger gründlich, da von weniger ProbandInnen, überprüft wurden. Auch verringerte sich das gemeinsame Wissen („Common Ground“) der ProbandInnen.

Appropriation: Dourish (2003) untersucht CSCW-Systeme aus einer technischen – im Gegensatz zur sozialen – Perspektive, und beleuchtet die Rolle von *Appropriation* (hier: das Ändern, Anpassen und Rekonfigurieren von Technologie) für die erfolgreiche Einführung von Technologien in Unternehmen. Er betrachtet (kooperative) Arbeit als eine Abfolge von Handlungen, die, bedingt durch äußere Umstände, immer wieder verändert, angepasst, rekonfiguriert oder improvisiert werden. Daraus schließt er, dass Technologie und die Art, wie sie verwendet wird, ebenfalls dieser ständigen Änderung unterworfen ist. Seiner Meinung nach muss eine Technologie oder Software „anpassbar“ sein, um erfolgreich zu sein, da dies den Bedürfnissen der User und der Arbeitspraxis entspricht. Im Weiteren zitiert er MacLean et al. (1990), die bemerken, dass dieses Anpassen innerhalb einer Firma oder Organisation aber auch organisatorisch (i.e. es ist erlaubt), technisch (i.e. es ist technisch möglich) und sozial (i.e. es wird von Kollegen nicht missbilligt) akzeptiert und gefördert werden muss.

Durchdringung: Espinosa et al. (2000) bemerken, dass es bei der Verwendung von Kollaborations- oder Awarenessstools wichtig ist, dass möglichst alle Mitglieder einer Gruppe diese Tools verwenden müssen um einen positiven Effekt zu erzielen.

Organisatorische Regularien: Gross und Koch (2007) erachten es als wichtig, dass auch organisatorische Vorgaben oder Regularien in Awarenesssystemen Niederschlag finden können. Diese Vorgaben betreffen laut den AutorInnen die Erfassung und die Weitergabe von Informationen, also wer darf welche Informationen an wen weitergeben. Ganz im Gegensatz dazu steht allerdings die Aussage von Larsson et al. (2010), dass, indem keine fixen Rollen, Privilegien, Prozessschritte oder Taxonomien (technologisch) vorgegeben werden, sich Wissen weiter verbreiten und über einen längeren Zeitraum erhalten bleiben kann. Da auch Awarenessinformationen eine Form von Wissen sind („*Kurzzeitwissen*“),

muss davon ausgegangen werden, dass solche äußeren Rahmenbedingungen bei der Verwendung von Awarenessinformationen also auch ein Faktor sein können.

Rollen: Espinosa et al. (2007) zeigen, dass in einer Gruppenarbeit abhängig von der Rolle von Teammitgliedern das Wissen über verschiedene Aspekte von Arbeit (technische, räumliche und prozessorientierte) unterschiedlich wichtig sind. King und Lyons (2011) zeigen weiters, dass mit steigender Erfahrung und „Seniority Level“ der AnwenderInnen der Wunsch nach mehr Awareness steigt. Spriestersbach et al. (2001) stellen fest, dass Location Awareness für Anwendungen die Usability verbessern kann, bemerken aber auch, dass Orte oder Orts-Awareness je nach Anwendungsfall oder Rolle verschieden implementiert und interpretiert werden muss. Auch Turner et al. (2010) halten fest, dass sich die Wahl der Kommunikationsmittel beispielsweise in Abhängigkeit der Empfänger und deren Rolle im Team entscheidet.

2.7 Nutzen von Awareness

Verschiedene Arbeiten der Forschungsliteratur zeigen die positiven Effekte von Awareness auf Kommunikation, Koordination, soziale Beziehungen und letztendlich den Erfolg von Gruppen und Teams in ihrer Zusammenarbeit.

Gross und Koch (2007) nennen Awareness als einen wesentlichen Faktor für den Erfolg von Gruppen, da „*Gruppenmitglieder zur effektiven und effizienten Kommunikation Informationen über einander, über gemeinsame Artefakte und über den Gruppenprozess*“ (Gross & Koch 2007, p.60) benötigen. Diese Kommunikation ist im Weiteren die Basis für Koordination. Laut Espinosa et al. (2007) wird Koordination durch verschiedene Arten von Wissen, - wozu nach deren Definition auch Awareness zählt – ermöglicht. Auch Tellioglu (2010) sieht Awareness als ein Mittel, um wechselseitige Abhängigkeiten zwischen verschiedenen Entitäten zu koordinieren. Antunes et al. (2010) stellen eine Effizienzsteigerung bei der Erledigung von Aufgaben in Gruppen durch verbesserte Situationsawareness fest. Nach Cataldo et al. (2006) werden Projekte und Arbeitsaufgaben meist auf mehrere Leute oder Teams und in mehrere, voneinander abhängige Unteraufgaben aufgeteilt. Diese beteiligten Aufgaben, Personen und Abhängigkeiten gilt es zu koordinieren. Laut den AutorInnen entwickeln sich die dazu nötigen Kommunikations- und Informationskanäle um Aufgaben herum, allerdings gibt es sich schnell ändernde Arbeitsanforderungen und Abhängigkeiten, auf welche Standardprozesse nicht angewendet werden können. Diese Änderungen gilt es laut den AutorInnen mittels geeigneter Awarenessmittel zu erkennen, um die Kommunikationswege entsprechend anpassen zu können.

Mark (2002) meint, dass in einer Gruppensituation durch Beobachten der Handlungen anderer und im Vergleich mit dem eigenen Handeln ein Lerneffekt ausgelöst werden kann, der die Zusammenarbeit einer Gruppe positiv beeinflusst. Bessere Awareness kann den BenutzerInnen eines Informationssystems die Gruppensituation, die gegenseitige Abhängigkeit und die Konsequenzen des individuellen Handelns für die Gruppe besser bewusst machen, und so die Etablierung und Einhaltung von Konventionen unterstützen. Tang und Birnholtz (2010) beobachten, dass Awarenessinformation in einem Instant Messaging System einen positiven Effekt auf die sozialen Beziehungen zwischen Gruppenmitgliedern haben kann, und erklären diesen Effekt damit, dass das (Kommunikations-)Verhalten von KollegInnen besser nachvollziehbar wird. Die Awarenessinformation in diesem Experiment besteht dabei aus einem Screenshot des Bildschirms von ChatpartnerInnen, sowie einer Angabe über die „Wichtigkeit“ der momentanen Aufgabe der PartnerInnen. Das Experiment zeigt, dass sich durch solch eine Awarenessinformation das Timing für Unterbrechungen von ChatpartnerInnen verbessert, und im Folgenden auch die Leistung der solcherart unterbrochenen PartnerInnen steigt.

Warta (2011) kann in einer Studie über die Verwendung von Wikis in Unternehmen die positiven Auswirkungen von Awarenessinformationen sogar quantitativ nachweisen: „*In erfolgreich im Unternehmen etablierten Wikis werden Awareness-Funktionen stärker genutzt als in weniger oder noch nicht erfolgreich etablierten Unternehmenswikis.*“ (ibid, p.208).

2.8 Enterprise 2.0 und Awareness

McAfee (2006) bezeichnet mit dem Begriff *Enterprise 2.0* die Verwendung von Web 2.0 Technologien in Unternehmen: „*These new digital platforms for generating, sharing and refining information are already popular on the Internet, where they're collectively labeled 'Web 2.0' technologies. I use the term 'Enterprise 2.0' to focus only on those platforms that companies can buy or build in order to make visible the practices and outputs of their knowledge workers.*“ (McAfee 2006, p.23). In seiner Diskussion des Begriffes grenzt er diesen vor allem auch gegen herkömmliche Intranetstrukturen, wie sie in Unternehmen häufig verwendet werden, ab.

Auch Convertino et al. (2010) beschäftigen sich mit dem zunehmenden Gebrauch von Web 2.0 Tools in Unternehmen. Tools und die Praktiken, wie z.B. das Generieren, Teilen und Organisieren von Content mittels Wikis, bezeichnen sie als „Collective Intelligence“. Als „intelligent“ bezeichnen sie diese Prozesse, da sie es den Usern erlauben ihre „soziale, informationelle und physische Umgebung zu adaptieren und kontrollieren“ (ibid). Sie heben hervor, dass Web 2.0 Tools meist zuerst für private Zwecke verwendet werden, bevor sie in Unternehmen oder Organisationen Einzug finden. Die Grundvoraussetzungen aber, warum, wie und von wem diese Tools in privaten gegenüber unternehmerischen oder organisationellen Kontext eingeführt und verwendet werden, unterscheiden sich laut den AutorInnen. Private Organisationen sind eher „Bottom-up“ organisiert, Innovationen werden „von unten“ eingeführt, und es setzt sich diejenige Technologie durch, die bei den BenutzerInnen am besten Anklang findet. Unternehmerische Organisationen dagegen sind „Top-down“ organisiert, Technologien werden von oben vorgegeben und die NutzerInnen können sich nicht aussuchen, was und womit sie arbeiten.

Die sechs wichtigsten Funktionalitäten, aus denen sich Enterprise 2.0 zusammensetzt, fasst McAfee (2006) unter dem Akronym *SLATES* zusammen. Hinchcliffe (2007) erweitert McAfee's Begriff von Enterprise 2.0 um vier soziale Aspekte, die der Arbeit mit Web 2.0-Technologien innewohnen. Diese fasst er mit den *SLATES*-Funktionalitäten zu dem Akronym *FLATNESSES* zusammen. Die Begriffe sollen im Folgenden beschrieben werden. Zudem soll dargelegt werden, wie die verschiedenen Enterprise 2.0-Funktionalitäten die besprochenen Mechanismen und Faktoren von Awareness theoretisch umsetzen und beeinflussen können. Wie diese Umsetzung in der Praxis aussieht, soll in den anschließenden Kapiteln empirisch untersucht werden.

Search: Nach McAfee (2006) ist das Finden von Informationen ein zentrales Element bei der Arbeit mit Informationstechnologien. Laut ihm werden Such-Funktionalitäten wie Schlagwort-Suche vermehrt verwendet, da sie gegenüber einer traditionellen Navigation über strukturierte Dokumente oder Verzeichnisse in einem Intranet schnellere Zugriffe erlauben.

Eine Suche stellt prinzipiell eine Form von aktivem Informationspulling durch einen BenutzerIn dar. Gleichzeitig ist sie durch die Spezifikation der Suchanfrage auch eine Art von Filter, der ad-hoc von Awarenesskonsumenten erzeugt wird. Die durch eine Suche gelieferte Informationsmenge kann durch Verfeinern einer Suche eingeschränkt werden, zudem kann ein Suchergebnis einer allgemeineren Suchanfrage, welches mehr als ein Ergebnis liefert, auch einen Überblick über vorhandene Informationen vermitteln, und macht diese somit quantifizierbar. Zeitpunkt und Häufigkeit von Suchen sind vom KonsumentInnen selbst bestimmbar. Geeignete Suchen in offenen, nicht durch Privilegien oder Barrieren beschränkten Systemen machen auch vermeintlich versteckte Informationen sichtbar. Schließlich gilt auch, dass je einfacher sich eine Suchfunktion bedienen lässt, desto geringer ist der Aufwand und umso höher der Nutzen für die KonsumentIn.

Links: Links sind die Basis für effektive Suchalgorithmen. Hier hat das Internet den Vorteil gegenüber einem Intranet, dass mehr User damit arbeiten und Links erstellen. Durch eine automatische Analyse von Links und deren Veränderung über die Zeit können nach McAfee (2006) interessante oder aktuelle Themen identifiziert werden. Links machen Themen und Verknüpfungen sichtbar und schaffen so eine Form von Kontextawareness.

Durch das Auslagern von Kontextinformation in Querverweise können die Informationsmenge und der informatorische Nutzen von Inhalten sowohl positiv als auch negativ beeinflusst werden.

Authoring: Damit ist die Möglichkeit des (kollaborativen) Verfassens und Bearbeitens von Texten im Internet gemeint. Als Beispiele gibt McAfee (2006) Wikis und Blogs an. Als Vorteile von Authoring in Firmen nennt er bessere Aktualisierungen von Informationen und bessere Verlinkungen. Zum Authoring muss festgestellt werden, dass es nicht nur das Verfassen von Informationen meint, sondern, dass im Web 2.0, respektive Enterprise 2.0, das Schreiben von Text immer mit einem sofortigen Publizieren einhergeht. Weiters stellen die beiden Funktionalitäten *Tags* und *Links* – sofern sie von BenutzerInnen und nicht automatisch von einem Informationssystem oder *Extensions* generiert werden - auch eine Form von Authoring dar. Beim Authoring kann man zwei Arten des Generierens von Awarenessinformationen unterscheiden: Aktives Generieren von Awarenessinformationen durch eine BenutzerIn, und passives Generieren von Awarenessinformationen durch eine Authoring-Plattform selbst. Beim aktiven Generieren durch eine BenutzerIn muss diese explizit Informationen über einen Zustand oder einen Kontext angeben. Beim passiven Generieren von Awarenessinformationen durch eine Authoring-Plattform werden Metadaten über die AutorIn, Zeitpunkt der Veröffentlichung oder sonstige Kontextinformationen automatisch protokolliert und als Awarenessinformation zugänglich gemacht.

Tags: Tags sind Schlagwörter, die zur Kategorisierung von vorhandener Information, und damit in der Folge auch zur Verbesserung der Suche dienen. Web 2.0-Technologien bieten die Möglichkeit, alle BenutzerInnen in die Verschlagwortung einzubinden. Schlagwörter oder Kategorien können entweder vorgegeben sein, oder können sich aus einer freien Verschlagwortung durch BenutzerInnen heraus entwickeln. Solche Kategorisierungen aus freien Schlagworten werden auch *Folksonomies* genannt (McAfee 2006). Einen möglichen Nutzen einer freien Verschlagwortung sieht McAfee (2006) in einer verbesserten Sichtbarmachung von Mustern und Prozessen in der Wissensarbeit. Das aktive Erstellen von Tags bedeutet gleichzeitig ein explizites Erzeugen von Kontextinformation (und somit Kontextawareness) durch Angabe des Tags. Zudem bedeutet das Einordnen von Content und Informationen mittels Tags in eine vorhandene Tag-Struktur wiederum ein implizites Erzeugen von Kontextinformation.

Extensions: Mit *Extensions* meint McAfee (2006) Systeme, welche BenutzerInneninteressen „mitdenken“, indem sie NutzerInnenverhalten analysieren und beispielsweise geeignete Vorschläge zu den BenutzerInnen interessierenden Informationen unterbreiten. *Extensions* im Sinne von künstlicher Intelligenz und semantischen Analysen sind somit ein mächtiges Konzept, welches auf alle für Awareness wichtigen Faktoren Einfluss haben kann.

Signals: Signals sind Benachrichtigungen, die eineN BenutzerIn über neue Informationen unterrichten und entsprechen somit direkt den *Notifications* aus den Awarenessmechanismen. Als traditionelle Benachrichtigungsinstrumente nennt McAfee (2006) Emailnachrichten, bei diesen ortet er aber das Problem, dass zu viele Nachrichten erzeugt werden und folglich die Posteingänge wiederum unübersichtlich werden. Als Instrumente von Web 2.0 nennt McAfee (2006) RSS-Feeds und Feed-Aggregatoren, welche neue Informationen aus dem Web übersichtlich und kompakt aufbereiten. Mit dieser Technologie erspart man sich laut McAfee (2006) das regelmässige Überprüfen von Websites auf Aktualisierungen.

Freeform: Freeform meint die möglichst freie Verwendung von Web 2.0-Technologie in Unternehmen, ohne Vorgaben, welche Tools für welche Aufgaben genutzt werden sollen. Die BenutzerInnen sollen selbst entscheiden welche Tools sie verwenden, da sie selbst am besten einschätzen können, welche Medien für ihre jeweiligen Aufgaben am besten geeignet sind. Diese Eigenschaft von Enterprise 2.0 steht potentiell im Gegensatz zur Forderung von Gross und Koch (2007) nach der Möglichkeit von organisatorischen Vorgaben.

Social: Damit meint Hinchcliffe (2007), dass man sich in Organisationen die Stärken von Web 2.0 in Bezug auf soziale Aspekte von Gruppen zunutze machen sollte, wie zum Beispiel Transparenz und die Unterstützung von nichthierarchischen Organisationsformen.

Emergence: Emergence meint das Entstehen von neuen (gewinnbringenden) Ideen, Informationen und Netzwerken, die aus der freien und transparenten Zusammenarbeit von beliebigen NutzerInnen im Web-, bzw. Enterprise 2.0 entstehen können. Grundlage dafür sind die beiden zuvor genannten Begriffe *Freeform* und *Social*. *Emergence* ist stark verwandt mit Awareness, da doch beide Konzepte ähnliches bedeuten, nämlich eine verbesserte, gewinnbringendere Zusammenarbeit ausgelöst durch bessere Kommunikation und Informationsteilung.

Network-oriented: Hinchcliffe (2007) meint damit, dass nicht nur Technologien und Tools selbst vernetzt sein müssen, sondern auch der Inhalt von Informationsplattformen Netzwerk-orientiert sein muss: „*These aspects of Enterprise 2.0 must apply not only to applications that are fundamentally delivered over a network but that their content be fully Web-oriented, addressable, and reusable*“ (Hinchcliffe 2007).

3. Methoden

Im folgenden Abschnitt werden die für die empirische Untersuchung verwendeten Datenerhebungs- und Analysemethoden erklärt. Zuerst wird dargelegt, warum ein qualitativer Ansatz als Forschungsmethode ausgewählt wurde, welche forschungsethischen Grundlagen für die Durchführung maßgebend waren und wie diese Grundlagen umgesetzt wurden. Danach wird der für die Datenerhebung- und Analyse verwendete Forschungsstil der Grounded Theory und dessen Umsetzung beschrieben. Anschliessend wird die konkrete Datenerhebung im Feld beschrieben: die Kontaktaufnahme zum Untersuchungsfeld, die drei eingesetzten Erhebungsmethoden „teilnehmende Beobachtung“, „Intranetrecherche“ und „Interviews“ und deren Umsetzung im konkreten Forschungsprojekt. Zuletzt wird die Transkription und die Aufbereitung der erhobenen Daten beschrieben.

3.1 Qualitative Methoden

Bei der vorliegenden Diplomarbeit wird für die empirische Datenerhebung- und Analyse bewusst ein ethnographischer Forschungsansatz mit qualitativen und nicht quantitativen Methoden verwendet, aus dem Grund, als sich Awareness nur schwer in Zahlen fassen lässt. Gläser und Laudel (2009, p.79) meinen, dass es Begriffe gibt, die *„so komplex sein können, dass sie sich nicht auf eindimensionale Variablen reduzieren lassen. Soziale Sachverhalte können ‚unauflösbar komplex‘, d.h. nur in mehreren Dimensionen gleichzeitig beschreibbar sein. (...) Sie können in den verschiedenen Dimensionen unterschiedlich skaliert sein. Das schließt ein, dass sie in allen Dimensionen nominalskaliert sein können, die Merkmalsausprägungen also ausschließlich verbal beschreibbar sind.“* (ibid, p.79). Awareness und seine Ausprägungen können wohl auch als solchermaßen komplexe Begriffe gesehen werden, die verschiedene Merkmale besitzen, und deren Wahrnehmung sich besser verbal, als durch Zahlen oder durch in Zahlen erfassbare Zusammenhänge beschreiben lassen.

Allerdings hat Warta (2011) in einer vergleichenden Untersuchung von Wikis in Unternehmen eine quantitative Methode für die Messung von Awareness entwickelt, indem bestimmte Funktionalitäten oder Kennzahlen eines Wiki-Systems, wie beispielsweise die Anzahl abonniertes Artikel pro BenutzerIn, als Maßzahlen für Awareness deklariert wurden. Für das vorliegende Untersuchungsfeld wäre eine ähnliche Metrik denkbar, allerdings müsste man auch hier verschiedene Aspekte berücksichtigen, beispielsweise die Tatsache, dass im hier untersuchten System abhängig von der Systemkonfiguration automatisch Abonnements angelegt werden, was zu falschen Schlüssen führen könnte.

3.2 Forschungsethik

Bevor die für die Diplomarbeit verwendeten Methoden näher beschrieben werden, soll noch kurz auf einige forschungsethische Grundlagen eingegangen werden, die das Untersuchungsdesign beeinflusst haben. Gläser und Laudel (2009) nennen verschiedene Kriterien, denen eine (empirische) Forschungsarbeit und Forschungsmethoden genügen müssen:

Offenheit: Eine Forschungsarbeit und die verwendeten Methoden müssen offen sein für neue oder unerwartete Erkenntnisse. Dieses Prinzip wird in der vorliegenden Arbeit einerseits durch die Methode der teilnehmenden Beobachtung erfüllt, dessen Erkenntnisinteresse sehr offen und eher explorativ angelegt war. Auch die Interviews wurden bewusst nicht zu stark strukturiert, um Raum für eigene Themen der InterviewpartnerInnen zu lassen.

Theoriegeleitet: Empirische Untersuchungen sollen auf theoretischem Vorwissen aufbauen. Diesem Prinzip wurde Rechnung getragen, indem vor Beginn der empirischen

Untersuchung eine umfangreiche Literaturrecherche- und Aufarbeitung durchgeführt, und der Problemstellung und Zielsetzung zugrundegelegt wurde.

Regelgeleitet: Forschungsarbeiten, Methoden und Ergebnisse müssen reproduzierbar, nachvollziehbar und überprüfbar sein. Im Besonderen nennen Gläser und Laudel (2009) die Vollständigkeit von Ergebnissen, die Nachvollziehbarkeit des Zustandekommens von Ergebnissen, und die Offenlegung von Interessen und Abhängigkeiten. Diese Kriterien sollen in der vorliegenden Arbeit durch eine genaue Beschreibung der empirischen Methoden, sowie der Offenlegung der Beobachtungsprotokolle und Interviewtranskripte erfüllt werden.

Schutz der untersuchten Personen: Dies ist ein weiteres wichtiges Kriterium, das Gläser und Laudel (2009) beim Durchführen von Untersuchungen nennen. Die untersuchten Personen dürfen durch das Untersuchungsergebnis oder Veröffentlichungen nicht geschädigt werden, sollen über Zweck und gesammelte Daten der Untersuchung aufgeklärt werden, und dürfen jedenfalls nur nach freiwilliger, informierter Einwilligung untersucht werden. Um diesem Kriterium Rechnung zu tragen, wurden zum einen Daten jeweils nur nach freiwilliger, informierter Einwilligung erhoben, zum anderen wurden vor Beginn der empirischen Datenerhebungen in der untersuchten Firma Zusagen zum Schutz der untersuchten Personen und zur Wahrung des Betriebsgeheimnisses der Firma gemacht. Zur Umsetzung dieser Zusagen wurden nach Beendigung der Datenerhebung Namen von InterviewpartnerInnen oder unternehmenskritische Daten in Transkripten, Beobachtungsprotokollen und Bildmaterial anonymisiert. Da eine solche Anonymisierung für angefertigte Bildschirmvideos technisch zu aufwendig war, wurden davon Screenshots, welche ebenfalls anonymisiert wurden, angefertigt, und die Videos anschließend gelöscht.

3.3 *Grounded Theory*

Die Datenerhebung und die Analyse dieser Daten wurde angelehnt an den Forschungsstil der Grounded Theory (GT). Der Begriff „Grounded Theory“ bezeichnet dabei sowohl eine Methode als auch das Ergebnis dieser Methode welches, eine aus den Daten heraus entwickelte, und in den Daten verankerte Theorie sein soll (Kuckartz 2007). Breuer (2010) fasst die Grounded Theory Methodik folgendermaßen zusammen: *„Eine Forschungslogik, bei der es um das Erfinden und und Ausarbeiten gegenstandsangemessener Begriffe, von Modellierungen und Theorien auf der Basis empirischer Erfahrungen, im Austausch zwischen Daten(-erhebung) und Theorie(-entwicklung) geht.“* (ibid, p.9)

Ein grundlegendes Werkzeug der GT und auch anderen Analysemethoden wie der Qualitativen Inhaltsanalyse ist das **Codieren** von erhobenen Daten. Nach Kuckartz (2007) bezeichnet Codieren das Zuordnen von Textteilen zu „Codes“ oder „Kategorien“. Die zugeordneten Textteile selbst können einzelne Zeichen, Wörter, Sätze, Absätze, Textteile oder ganze Dokumente bei der Arbeit mit Texten, oder Bilder, Filme, Ausschnitte oder Sequenzen bei der Arbeit mit Medien sein (Kuckartz 2007). Die Begriffe „Kategorie“ und „Code“ werden in der sozialwissenschaftlichen Fachliteratur häufig wechselseitig verwendet, und verschiedene Methoden oder AutorInnen verwenden unterschiedliche Begriffe dafür, zum Beispiel „Label“, „Merkmal“, „Variable“, „Verschlagwortung“, „Klassifikation von Textmerkmalen“, „Konzepte“ (Kuckartz 2007) oder „Operatoren“ (Gläser & Laudel 2009).

Ein weiteres grundlegendes Werkzeug der GT sind sogenannte **Memos**. Das sind Notizen oder Skizzen, die während des gesamten Prozesses der Datenerhebung und der Analyse gemacht werden und die helfen sollen, eine Theorie zu entwickeln. Memos können entweder Textstellen oder Codes selbst zugeordnet sein, oder als theoretische Codes ohne Bezug zum Text verfasst werden (Kuckartz 2007).

Bei der Grounded Theory ist das Codieren gleichbedeutend mit der Analyse (Kuckartz 2007, p.72-82). Ziel ist es, in einer zyklischen Vorgehensweise interessante (interessierende) Phänomene im Datenmaterial zu finden, diese zu kategorisieren, und die so gefundenen **Kategorien** schrittweise zu sogenannten Schlüsselkategorien zu verfeinern, zu abstrahieren oder zu subsumieren (ibid). Die GT entspricht einem induktiven

Forschungsansatz und betont sehr stark den Aspekt der Offenheit, von daher sollten vor Beginn des Codierens keine Kategorien vorgegeben, und auch keine Dimensionalisierungen vorweggenommen werden.

Das so entwickelte Kategoriensystem dient als Basis für die zu entwickelnde Theorie. In der Literatur über GT werden drei Codiermethoden genannt, die gleichzeitig drei (aufeinanderfolgende) Analyseschritte beschreiben: **offenes Codieren**, **axiales Codieren** und zuletzt **selektives Codieren** (Kuckartz 2007).

Beim ersten Analyseschritt werden mittels offenem Codieren die Texte aufgearbeitet, indiziert und exploriert. Codes können entweder auf theoretischem Vorwissen oder basierend auf der Fragestellung vergeben werden (*konzeptuelle Codes*), oder sich direkt aus dem Text und Aussagen von Interviewpartnern in Form von sogenannten In-Vivo-Codes ergeben. Neben Codes können auch sogenannte Konzepte identifiziert werden, das sind Ereignisse, die als Beispiele für Phänomene dienen (Kuckartz 2007).

Anschließend werden die gefundenen Codes und Konzepte zu sogenannten Kategorien zusammengefasst: „Kategorie ist für Strauss ein unabhängiges begriffliches Element einer Theorie, eine Klassifikation von Konzepten. Diese Klassifikation wird erstellt, wenn Konzepte miteinander verglichen werden und sich offenbar auf ein ähnliches Phänomen beziehen. So werden die Konzepte unter einem Konzept höherer Ordnung zusammengruppiert – ein abstrakteres Konzept, genannt Kategorie“ (Kuckartz 2007, p.74). Zu den gefundenen Kategorien werden ausserdem Merkmale und Dimensionen dieser Merkmale festgehalten.

Im nächsten Schritt erfolgt das axiale Codieren, bei dem die einzelnen Kategorien genauer untersucht und zueinander in Beziehung gesetzt werden sollen. Ein Hilfsmittel dafür stellt dabei das sogenannte *Codierparadigma nach Strauss* (siehe Abb. 4) dar. Dieses sieht vor, dass für interessante Phänomene oder Kategorien deren Kontext, ursprüngliche- und intervenierende Bedingungen, Handlungsstrategien und Konsequenzen davon herausgearbeitet werden (Kuckartz 2007).

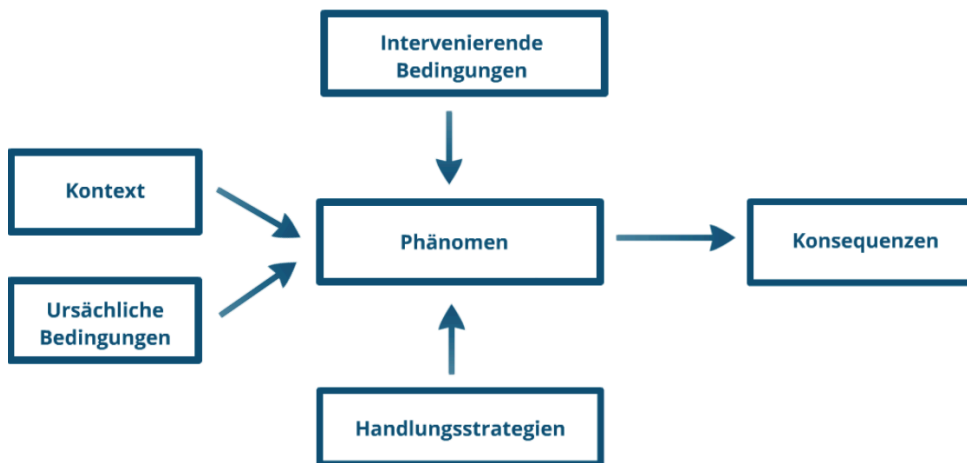


Abb. 4 Codierparadigma nach Strauss (Quelle: selbst angefertigt nach Vorlage von (Kuckartz 2007))

Im abschließenden selektiven Codieren geht es schließlich um das Ausformulieren und Darstellen eines Modelles oder einer Theorie, indem eine sogenannte *Schlüsselkategorie* benannt wird, um die herum die zuvor gefundenen Kategorien angeordnet und abermals zu einander in Beziehung gesetzt werden (Breuer 2010). „*Selektives Kodieren ist gewissermassen axiales Kodieren auf einem Niveau höherer Abstraktion, allgemeinerer Theoretisierungsstufe. Es soll eine Schlüssel- bzw. Kernkategorie, ein konzeptionelles Zentrum der entwickelten Theorie ausgewählt, bzw. festgelegt werden. Alle anderen Theoriebestandteile (Kategorien) werden sodann um dieses Zentralkonzept herum angeordnet und mit diesem verknüpft (...). Zudem ergibt sich aus der Wahl der Kernkategorie eine Entscheidung des Forschenden über die sogenannte „Story Line“, den roten Faden bzw. den Bogen der nun zu erzählenden Geschichte, der Ergebnisdarstellung*

des Forschungsprojekts, die Fokussierungsperspektive der Themenbearbeitung bzw. der gegenstandsbezogenen Theorie“ (Breuer 2010, p.92). Breuer (2010) stellt den gesamten Prozess der Theorieentwicklung von den Daten hin zur in diesen Daten verankerten Theorie mittels der verschiedenen Codiermethoden schematisch in einer Übersicht dar (siehe Abb. 5).

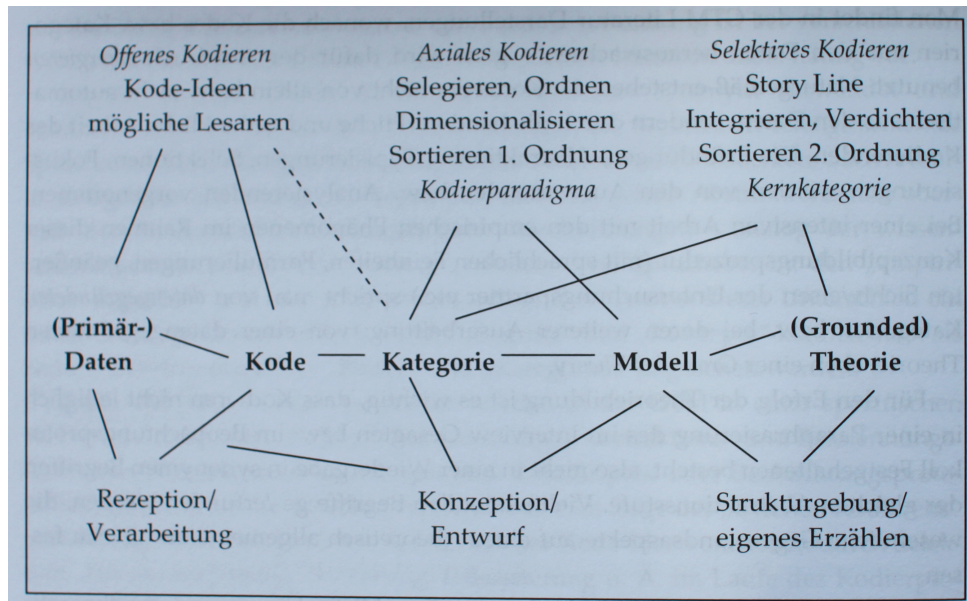


Abb. 5 Schematische Abfolge von Codiermethoden (Breuer 2010, p.76)

Ein weiteres Merkmal der Grounded Theory ist die zyklische Arbeitsweise. Diese wird nicht nur auf die verschiedenen Codierschritte angewendet, was bedeutet, dass zwischen diesen vor- und zurückgesprungen werden kann oder sogar muss (Breuer 2010), sondern auf den gesamten Forschungsprozess inklusive der Aufarbeitung von theoretischer Fachliteratur und Basiswissen und der Datenerhebung. In der Praxis bedeutet dies, dass nach jeder Datenerhebung, d.h. beispielsweise nach jedem Interview, diese Daten codiert und analysiert werden, um mit den daraus gewonnenen Erkenntnissen das nächste Interview durchzuführen. Diese Arbeitsweise wird auch „Methode des permanenten Vergleiches“ genannt (Kuckartz 2007). Der Begriff „theoretisches Sampling“ bedeutet, dass beispielsweise Interviewpartnerinnen danach ausgewählt werden, ob sie mögliche neue Erkenntnisse für die (bisher entwickelte) Theorie bringen können (ibid).

Schließlich gibt es in der Grounded Theory Methodik noch den Begriff der „theoretischen Sättigung“, der eine Art Kriterium für die Abgeschlossenheit eines Forschungsprozesses, bzw. einer Theorieentwicklung ist: „Die Kategorien und ihre Modellkonfiguration müssen durch oftmalige Konfrontation mit Fällen und Daten hinsichtlich ihrer Strukturen und Bedingungsbeziehungen den auftretenden empirischen Variationen standhalten und theoretisch detailliert, kohärent und dicht ausgearbeitet sein.“ (Breuer 2010, p.110). Breuer (2010) hält dazu aber auch fest, dass es sich bei einer Theorieentwicklung schon aus forschungslogischer Sicht nie um eine endgültige Theorie handeln kann, und dass zudem auch pragmatische Faktoren eine Rolle für den Abschluss einer Theorieentwicklung spielen können.

In der Fachliteratur über qualitative Methoden und die Grounded Theory Methodik wird zudem stets betont, dass es kein einheitliches Prozedere für die Durchführung und Darstellung eines Forschungsprojektes nach solchen Methoden gibt. Das Anpassen von Methoden ist nach Kuckartz (2007) und anderen Lehrbüchern ein gangbarer Weg: „Ebenso betont er, dass es wichtig sei, die Methode immer an die je konkrete Fragestellung anzupassen. Entscheidend ist weniger ein ganz bestimmtes Prozedere als vielmehr das Ziel, nämlich Theorie.“ (Kuckartz 2007, p.77). Auch Mayring (2000) befürwortet eine Kombination von Qualitativer Inhaltsanalyse mit anderen qualitativen Ansätzen: „Allerdings lassen sich die beschriebenen Verfahrensweisen auch gut mit anderen qualitativen Ansätzen kombinieren. Dabei sollte immer die Fragestellung der Studie und

die Charakteristik des zu untersuchenden Materials das Primat haben, nicht der Methode der Analyse.“ (Mayring 2000, p.9).

3.4 Datenerhebung

Als Untersuchungsobjekt für die vorliegende Diplomarbeit wurde eine Firma aus der Versicherungsbranche gewählt, für die der Autor dieser Diplomarbeit davor selbst mehr als zwei Jahre als Softwareentwickler gearbeitet hatte. Diese Verbindung ist für eine empirische Untersuchung einerseits von Vorteil, kann aber auch Nachteile haben.

Von Vorteil war es, dass der Autor leichter Zugang zur untersuchten Firma, zu Firmenstrukturen und zu MitarbeiterInnen hatte, und überdies großes Entgegenkommen von Seiten der Firma erfuhr, wodurch beispielsweise die Interviews während der Arbeitszeit durchgeführt werden konnten. Ein weiterer Vorteil war, dass der Autor bereits mit vielen Prozessen, Personen und dem in der Firma verwendeten Vokabular (fachlich, technisch und auf Prozessebene) vertraut war, und dadurch die Eingewöhnungsphase für den empirischen Teil relativ kurz war. Der wichtigste Grund für die Auswahl dieser Firma war aber das Firmengefüge, welches mit seiner Zusammensetzung aus verschiedenen Teams an unterschiedlichen Orten und mit unterschiedlichen Teilprojektaufgaben ein großes Ausmaß an Kommunikation und Koordination verlangt. Für diese Anforderung wird eine weite Palette von klassischen- und Enterprise 2.0 Technologien eingesetzt, und somit stellt das ausgewählte Unternehmen für eine empirische Untersuchung zu dem behandelten Thema ein sehr ergiebiges Untersuchungsobjekt dar.

Ein Nachteil der früheren Tätigkeit des Autors bei er untersuchten Firma ist ein möglicher Verlust von Objektivität, sowie eine mögliche Voreingenommenheit gegenüber verschiedenen Handlungsabläufen im Untersuchungsfeld, was dem Prinzip der Offenheit in der qualitativen Forschung zuwiderlaufen könnte: *„Das Prinzip der Offenheit verlangt von den Forschern, beobachtete Tatbestände nicht vorschnell unter bekannte Kategorien zu subsumieren. Dabei ist unwichtig, ob diese Kategorien aus der theoretischen Analyse entstanden sind, oder ob es sich um unsere Vor-Urteile handelt, die wir aus dem Alltagswissen über den Gegenstand bilden.“* (Gläser & Laudel 2009, p.30). Girtler (1992, p.34) jedoch sieht in einer starken Integration in eine untersuchte Gruppe, bzw. eines guten Vorverständnisses einen Vorteil, wenn nicht sogar eine Notwendigkeit, um *„die Bedeutung des Handelns zu erfahren“* (ibid, p34), und *„um entsprechende ‚wirkliche‘ Ergebnisse zu erhalten“* (ibid, p35). Auch Gläser und Laudel (2009) sehen es für *„unbestritten, dass es eine von theoretischem und alltagsweltlichem Vorverständnis freie empirische Untersuchung gar nicht geben kann“* ((Meinfeld 2005) in (Gläser & Laudel 2009, p.31)). Durch eine halbjährige Pause zwischen dem Ende der beruflichen Tätigkeit des Autors für die Firma und dem Beginn der empirischen Untersuchung wurde versucht, die Sichtweise des Autors zu distanzieren. Tatsächlich gab es in dieser Zeit in der untersuchten Firma auch viele Änderungen in den Prozessen, den verwendeten Tools, sowie in der personellen Zusammensetzung.

3.4.1 Sampling und Kontaktaufnahme

Die Kontaktaufnahme war durch die frühere Tätigkeit des Autors in der untersuchten Firma relativ unkompliziert, und erfolgte zuerst mit den Chefs des österreichischen Büros. Nach deren prinzipiellem OK zu einer Beobachtung erfolgte eine erste schriftliche Anfrage an die zu untersuchende Firma in Norddeutschland, diese enthielt eine Beschreibung des Forschungsvorhabens inklusive der angedachten Untersuchungsmethoden und der geplanten Veröffentlichungen. Nachdem die Firma auch ihr Einverständnis signalisierte, wurde kurz vor Beginn der eigentlichen Beobachtung noch eine formelle Einverständniserklärung von der Firma eingeholt (siehe Appendix A – Einverständniserklärung Firma).

Des Weiteren wurde auch ein Beitrag für den internen Newsletter der Firma verfasst, in diesem wurden ebenfalls kurz das Thema und die geplanten Datenerhebungen (Interviews

und teilnehmende Beobachtung) erklärt. Zu Beginn der teilnehmenden Beobachtung im österreichischen Büro wurde das Vorhaben den dortigen MitarbeiterInnen in einer Besprechung offiziell vorgestellt und ein mündliches Einverständnis eingeholt.

Cirka zwei Wochen vor Beginn der Interviews wurde dann der Kontakt zu den einzelnen AbteilungsleiterInnen gesucht um Termin- und Durchführungsdetails zu besprechen, sowie um das Einverständnis der AbteilungsleiterInnen für die Interviewdurchführung mit ihren MitarbeiterInnen einzuholen. Dieser Schritt war auch notwendig, da die Interviews in der Arbeitszeit stattfanden und deshalb firmenintern auch abgerechnet werden mussten. Die einzelnen InterviewpartnerInnen selbst wurden dann jeweils vor Ort kontaktiert. Die InterviewpartnerInnen wurden mündlich und schriftlich über das Projekt aufgeklärt, und pro Interview wurde eine schriftliche Einverständniserklärung eingeholt (siehe Appendix B – Projektinformation MitarbeiterInnen und Appendix C – Einverständniserklärung MitarbeiterInnen).

3.4.2 Teilnehmende Beobachtung

Ziel der teilnehmenden Beobachtung war es, konkrete Arbeitssituationen zu dokumentieren, und zu untersuchen, in welcher Weise dabei Awarenessinformation zur Anwendung kommt, bzw. erzeugt wird. Dazu sollte die Kommunikation zwischen Kolleginnen beobachtet werden um zu sehen, welche Awarenessinformationen kommuniziert werden, woher die Awarenessinformationen stammen, oder durch welche Mechanismen Awarenessinformationen verbreitet werden. Ein besonderer Fokus der Beobachtung lag auf dem untersuchten webbasierten Issuetrackingsystem, nämlich in wieweit dieses in die alltägliche Arbeit und Kommunikation einfließt und wie es sich auswirkt, also ob es als Auslöser für Kommunikationsereignisse wirkt, ob es als „Kommunikationsartefakt“ dient mittels dem über bestimmte Themen gesprochen werden kann, und auch wie „klassische“ Tools im Vergleich dazu verwendet werden. Des Weiteren sollte die teilnehmende Beobachtung auch als erster Kontakt und als Akklimatisationsphase mit dem Untersuchungsfeld dienen.

Die Beobachtung im österreichischen Büro selbst erfolgte über einen Zeitraum von ca. sechs Wochen in denen 15 einzelne Tage mit insgesamt 153 Kommunikationsereignissen protokolliert wurden. In dieser Zeit hatte der Autor einen Sitzplatz an einem von vier Arbeitsplätzen des zentralen Büroraumes mit Blick auf die drei MitarbeiterInnen dieses Zimmers, und mit Blick auf ein, daran offen anschliessendes, zweites Zimmer, in dem drei weitere EntwicklerInnen ihre Arbeitsplätze hatten. Für die Protokollierung wurde ein Tabellenkalkulationsprogramm verwendet mit Spalten für Datum, Wochentag, Uhrzeit, gegebenenfalls einer freien Kontextbeschreibung (z.B. „*Situation: Donnerstag Vormittag, „Eincheckdeadline“ war Mittwoch Nachmittag (gestern)*“), einer Beschreibung des Kommunikationsereignisses in wörtlicher, paraphrasierender oder beschreibender Form, sowie zwei Spalten für Codierung und Kommentare, bzw. Memos. Bei der Teilnahme an Besprechungen wurden handschriftliche Notizen angefertigt und später in die Tabelle übertragen.

Als ein Problem bei diesem Teil der empirischen Erhebung stellte sich heraus, dass bei beobachteten Telefonaten eine „Hälfte“ der Kommunikation nicht registriert werden konnte, und sich Informationen über GesprächspartnerInnen und über das Thema oft erst aus dem Zusammenhang oder teilweise auch auf Nachfrage erörtern ließen. Bei diesem Punkt war allerdings das Vorwissen des Autors über die Personen und Prozesse in der untersuchten Firma von Vorteil, als dadurch eben der Zusammenhang leichter hergestellt werden konnte. Ein weiteres Problem war auch, dass das Kommunikationsaufkommen stark schwankte, und es einerseits Phasen ohne interessante Ereignisse gab, aber auch Phasen mit sehr vielen interessanten Ereignissen, die aber aufgrund der Gleichzeitigkeit des Auftretens nur schwer protokolliert werden konnten.

3.4.3 Intranet-Recherche

Ziel der Intranet-Recherche war es, die in der Firma verwendeten (Software-)Tools und Kommunikationsmittel zu dokumentieren, und deren Awarenessfunktionalitäten festzuhalten. Da die zu Verfügung stehenden Kommunikations- und Arbeitsmittel direkten Einfluss auf die Kommunikations- und Arbeitsweise der MitarbeiterInnen haben, wird dieser Schritt vom Autor als wichtig für das Verständnis der Kommunikations- und Arbeitsabläufe angesehen.

Ein weiterer Punkt, der über die Intranet-Recherche in Erfahrung gebracht werden sollte, waren Daten über die Organisationsstruktur der Firma, da auch diese direkten Einfluss auf die Kommunikationswege der MitarbeiterInnen hat. Dies sollte mittels dezidierter verfügbarer Informationen im Intranet wie Organigrammen geschehen, aber auch implizit durch eine Analyse der verwendeten Tools und der damit arbeitenden Personen und Teams.

Die Intranetrecherche wurde während der teilnehmenden Beobachtung durchgeführt. Für die Dokumentation stand dem Autor ein PC der untersuchten Firma mit Anbindung an deren Firmennetzwerk und Accounts für die wichtigsten Tools zur Verfügung. Die Dokumentation geschah mittels Screenshots und Notizen. Um besser zwischen organisatorischen Merkmalen, die spezifisch für die untersuchte Firma sind, und solchen, die für die gesamte Versicherungsbranche gelten, unterscheiden zu können, wurde zusätzlich zur Intranetrecherche auch geeignete Literatur zur Versicherungswirtschaft studiert.

3.4.4 Interviews

Zweck der Interviews war es, die eingangs in der Problemstellung gestellten Fragen zu beleuchten: wie wird ein Enterprise 2.0-Issue Trackingsystem konfiguriert und verwendet, wie wirkt es sich auf die Teamarbeit aus, welche Unterschiede bestehen zu „klassischen“ Awareness Tools, welche Unterschiede ergeben sich abhängig von Rollen und Teams, und wie wirkt sich der Faktor Distanz auf die Awareness von Kolleginnen aus.

Die Interviews wurden als halbstrukturierte Experteninterviews im Sinne der Definition von Gläser und Laudel (2009) angelegt: *„Schließlich gibt es eine Art besonderen Wissens, über das eigentlich jeder von uns verfügt. Es ist das Wissen über die sozialen Kontexte, in denen man arbeitet, über die eigenen Arbeitsprozesse (...). Nur die unmittelbar Beteiligten haben dieses Wissen, und jeder von ihnen hat aufgrund seiner individuellen Position und seiner persönlichen Beobachtungen eine besondere Perspektive auf den jeweiligen Sachverhalt.“* (Gläser & Laudel 2009, p.11). Die InterviewpartnerInnen wurden also als ExpertInnen für ihr jeweiliges Team, für ihre jeweilige Rolle und für die nötigen Arbeitsabläufe und Softwareanwendungen in dieser Rolle betrachtet. Dabei wurde versucht, eher kurze Fragen zu stellen, und die InterviewpartnerInnen im Sinne eines offenen Forschungsansatzes möglichst frei sprechen zu lassen.

Bei der Auswahl der InterviewpartnerInnen wurde versucht, Leute aus verschiedenen Abteilungen, die gemeinsam an demselben Projekt arbeiten und dafür miteinander kommunizieren müssen, zu finden. Ausgangspunkt dafür waren die EntwicklerInnen in Österreich, die an verschiedenen Projekten im Kompositbereich arbeiten, und bei denen bereits die teilnehmende Beobachtung durchgeführt wurde. Für die Interviews wurde dafür also versucht, Leute aus den ebenfalls für den Kompositbereich zuständigen Teams in Norddeutschland auszuwählen. Zu diesen Teams zählten die Abteilungen Produktmanagement am Anfang der Produktionskette, Fachkonzeption, welche für die Businessanalysen zuständig sind und logistisch zwischen Produktmanagement und Entwicklung stehen, die Entwicklung selbst, und schließlich die Testabteilung am Ende der Produktionskette. Begleitend dazu wurde auch versucht, Personen aus dem Releasemanagement, welche mit dem Deployment des produzierten Sourcecodes beauftragt sind, und die ebenfalls häufigen Kontakt zu den EntwicklerInnen haben, zu interviewen. Des Weiteren wurde auch darauf geachtet, in den einzelnen Abteilungen auch Personen mit verschiedenen Rollen und aus verschiedenen Hierarchieebenen zu befragen.

Da die Interviewdurchführung verschiedenen zeitlichen, örtlichen und terminlichen Einschränkungen unterworfen war, konnten die an den Projekten beteiligten Personen nicht vollständig abgedeckt werden. Andererseits ergab sich aber die Möglichkeit, mit Personen außerhalb dieser Projekte zu sprechen, und deren Sicht auf die untersuchte Software zu erfahren, was zu einem differenzierteren Bild vom Gebrauch der Software beitrug.

Für die Interviewdurchführung vor Ort in Norddeutschland wurde abermals eine schriftliche Einverständniserklärung eingeholt, zudem wurde für die einzelnen Interviews eine Projektinformation und eine Einverständniserklärung im Sinne eines „*informed consent*“ vorbereitet (siehe Appendix B – Projektinformation MitarbeiterInnen und Appendix C – Einverständniserklärung MitarbeiterInnen). Die Interviewtermine selbst wurden vor Ort mit den jeweiligen Personen vereinbart, hierbei war das firmeninterne (Lotus Notes-basierte) Kalendertool hilfreich. Die Interviews wurden – um den laufenden Betrieb nicht zu stören – jeweils in einem Konferenzraum abgehalten, dieser wurde zu Beginn jeweils für eine Stunde reserviert, später immer für 75 Minuten um eine Nachbereitung der Interviews zu ermöglichen und dabei Unterbrechungen durch andere Konferenzraum-BenutzerInnen zu vermeiden. Im Konferenzraum wurde jeweils vor Interviewbeginn ein Interviewsetup aufgebaut (siehe Abb. 6), dieses enthielt vorbereitet eine Projektinformation und Einverständniserklärung für die InterviewpartnerInnen, einen zweiseitigen Interviewleitfaden für den Interviewer - dieser wurde deshalb auf zwei Seiten komprimiert, um in der Interviewsituation nicht die Übersicht zu verlieren – ein digitales Tonaufzeichnungsgerät mit voreingestelltem Aufnahmepegel, sowie einem Laptop mit Internetanschluss für den Login in den jeweiligen JIRA-Account der Interviewpartner. Für die Aufzeichnung dieses Teils wurde die Screenrecordingsoftware „CamStudio“⁹ verwendet.



Abb. 6 Interviewsetup: Projektinformation und Einverständniserklärung links, Interviewleitfaden rechts, Tonaufnahmegerät und Laptop für JIRA-Login mitte, Waffeln als Dankeschön.

In der Vorbereitung auf die Interviews wurde ein Interviewleitfaden konstruiert (siehe Appendix D – Interviewleitfaden). Dieser, und somit die Interviewstruktur, enthielt vier wesentliche Bestandteile. Zuerst wurden allgemeine Fragen zum Team und zum Aufgabenbereich der InterviewpartnerInnen gestellt, mit welchen Personen und warum kommuniziert wird, sowie mit welchen Tools im Arbeitsalltag gearbeitet wird, ohne aber

⁹ <http://camstudio.org/>

noch speziell auf JIRA einzugehen. Hier wurde vor allem auch versucht, Unterschiede in der Kommunikation mit den eigenen Teams und mit anderen Teams zu erfragen. Im zweiten Teil wurde versucht, spezifischer bestimmte Awareness Themen zu ergründen, beispielsweise, ob man über die Arbeit von KollegInnen Bescheid weiß, wie man darüber erfährt, oder ob das für die eigene Arbeit eine Rolle spielt. Als Drittes wurden spezifischere Fragen zu bestimmten Tools gestellt und wie diese im Vergleich zu JIRA wahrgenommen werden. Im letzten Teil schließlich loggte man sich gemeinsam mit den InterviewpartnerInnen in deren JIRA-Account ein. Anschließend sollten sie zeigen, wie sie selbst JIRA konfiguriert hatten, welche Funktionen, Filter und Widgets verwendet werden, wie das Tool für die Arbeit verwendet wird und wie sie sich damit zurechtfinden. Dabei hatte es sich bewährt, den InterviewpartnerInnen die Maus zu überlassen und sie frei reden zu lassen. Zwischenfragen wurden gestellt wenn dies notwendig oder interessant war. Der Interviewleitfaden wurde abhängig von den InterviewpartnerInnen teilweise etwas angepasst. So wurden beispielsweise für Personen aus der Fachkonzeption keine Fragen zu MyLyn gestellt, da dort dieses Tool nicht genutzt wird, und bei manchen Personen – vor allem in späteren Interviews – wurden zusätzliche Fragen notiert, die aus vorhergegangenen Interviews auftauchten.

In Summe wurden schließlich in einem Zeitraum von eineinhalb Wochen 17 Interviews mit jeweils einer Dauer von 25 bis 75 Minuten durchgeführt. Von diesen 17 Interviews wurde bei zweien allerdings kein Einverständnis für eine Tonaufzeichnung gegeben, weshalb diese im Sinne einer Nachvollziehbarkeit nicht verwertet werden konnten.

3.4.5 Transkription

Die Interviews sollten wegen auf Grund der Nachvollziehbarkeit der Ergebnisse, und als Grundlage für die folgende Analyse vollständig transkribiert werden. Vor der Transkription der Interviews, bzw. während der Transkription des ersten Interviews wurde ein Transkriptionsleitfaden erstellt, der eine einheitliche Vorgehensweise sicherstellen sollte. Dabei wurde den Empfehlungen von Gläser und Laudel (2009, p.193-194) gefolgt, die folgendes System verwenden (siehe Abb. 7):

- Es wird in Standardorthographie verschriftet und keine literarische Umschrift verwendet (z.B. „hast du“ statt „haste“, Kowal/O’Connell 2007: 441);
- Nichtverbale Äußerungen (z.B. Lachen, Räuspern, Husten, Stottern) werden nur dann transkribiert, wenn sie einer Aussage eine andere Bedeutung geben,
- Besonderheiten der Antwort mit „Ja“ oder „Nein“, (z.B. zögernd, gedehnt, lachend) werden vermerkt,
- Unterbrechungen im Gespräch werden vermerkt,
- Unverständliche Passagen werden gekennzeichnet.

Abb. 7 Transkriptionsregeln (Gläser & Laudel 2009, p.194)

Diesem System folgend wurden für die vorliegende Arbeit folgende **Transkriptionsregeln** aufgestellt:

1. Verkürzende umgangssprachliche Äußerungen wie beispielsweise „haste“ oder „wir ham“ wurden ohne weitere Kennzeichnung durch Standardorthografie wie „hast du“ oder „wir haben“ ersetzt, da es die Lesbarkeit wesentlich verbessert, und in diesem Fall inhaltlich keine Verfremdung, und auch keine eigene Interpretation des Autors bedeutet. Auch Kuckartz (2007, p.39) meint, dass für transkriptbasierte Analysen der Genauigkeitsgrad von Transkriptionen abhängig vom Zweck des Interviews variieren darf, und dass verschiedene Transkriptionssysteme zulässig sind.
2. Besonderheiten von Antworten wie Lachen oder Dehnungen wurden in Klammern vermerkt (z.B. „(lacht)“, oder „(gedehnt)“), da sie für die Interpretation von Aussagen relevant sein können.
3. Indirekte Sprache wurde in Anführungszeichen gesetzt. Z.B.: „Und da beantrage ich formal eine Abnahme vom Produktmanagement und sage "hier, Thema xy, Fachkonzept ist fertig, Entwickler haben es abgenommen, Kunde hat es

abgenommen, bitte nehmt mir das auch ab, und bestätigt mir, dass ich mit dem Auftrag, den Auftrag erledigt habe".“

4. Umgangssprachlich kommt es sehr häufig vor, dass Sätze unterbrochen und nicht oder anders weitergeführt werden. Um dabei die Lesbarkeit zu gewährleisten, wurden solche Unterbrechungen mit drei Punkten ... gekennzeichnet. Z.B.: "*Hier müsste jetzt bei Glas... das zum Beispiel, so das ist in der neuen Version.*"
5. Pausen oder Unterbrechungen im Gespräch wurden mit drei Punkten in Klammern gekennzeichnet: (...)
6. Unverständliche Passagen wurden mit „(unv.)“ gekennzeichnet, daneben wurde die vermutete Aussage notiert.
7. Anonymisierte Namen von Personen oder Produkten wurden zur Kennzeichnung **fettgedruckt**.

Die Transkription selbst wurde mit dem frei verfügbaren Tool „f4“¹⁰ vorgenommen. Die Transkription der Interviews dauerte im Schnitt je nach SprecherIn und Länge des Interviews inklusive Pausen jeweils zwischen einem halben und eineinhalb Tagen, in Summe ca. vier Wochen. Während der Transkription wurden die Transkripte mit ersten Memos und Codes versehen.

¹⁰ www.transkription.de

4. Ergebnisse Kontextrecherche

Im folgenden Kapitel wird der wirtschaftliche, organisatorische und technische Kontext des Untersuchungsfeldes dargestellt. Zuerst wird kurz die untersuchte Firma und die Branche vorgestellt, gefolgt von einer kurzen Beschreibung des in der untersuchten Projektgruppe verwendeten Softwareentwicklungsprozesses und der damit einhergehenden organisatorischen Aufteilung. Anschließend werden die wichtigsten in der Firma verwendeten Softwaretools und Technologien beschrieben. Bei den beschriebenen technischen Systemen handelt es sich um jene Tools, die zur alltäglichen Zusammenarbeit zwischen den in der Untersuchung beobachteten und interviewten Personen verwendet werden.

4.1 Branchenkontext

Da bei dem Thema Awareness und Teamarbeit soziale und organisatorische Faktoren eine wichtige Rolle spielen, sollen an dieser Stelle der Firmen- und Projektkontext einführend beschrieben werden. Da es sich bei dem untersuchten Projekt außerdem um ein Projekt im Versicherungswesen handelt, sollen einige grundlegende, und im Laufe der Arbeit und der Interviews öfter vorkommende Begriffe aus dem Versicherungsbereich kurz erklärt werden.

Bei der untersuchten Firma handelt es sich um eine Softwareentwicklungsfirma mit Sitz in Norddeutschland, sowie zwei weiteren Entwicklungs- und Vertriebsstandorten in Süddeutschland und Österreich. Zum Zeitpunkt der Untersuchung beschäftigte die Firma am Standort in Norddeutschland ca. 140 MitarbeiterInnen, am Standort in Österreich waren acht externe MitarbeiterInnen beschäftigt.

Die Firma erstellt Softwarelösungen für die Versicherungsbranche und bietet als zentrales Produkt eine Softwaresuite für verschiedene Versicherungssparten an. Eine typische Versicherungsanwendung ist modular aufgebaut und besteht im Wesentlichen aus jenen Komponenten, die auch in Abb. 8 dargestellt sind. Die für das untersuchte Projekt relevanten Komponenten sind dabei die Produktmaschine und das Bestandssystem für die Sparte Komposit. Schnittstellen bestehen zu weiteren Komponenten wie Schaden und Leistung, Partnersystem, Dokumentenmanagement und Druck, Inkasso und Exkasso. Da die Versicherungsbranche ein hochgradig spezialisierter Bereich ist, der branchenintern ein sehr spezielles Vokabular verwendet, schlägt sich dieses Vokabular auch in der alltäglichen Softwareentwicklungsarbeit für diesen Bereich nieder. Zur Klärung von Begriffen sei an dieser Stelle auf Aschenbrenner Michael et al. (2010) verwiesen, einzig der Begriff „Kompositversicherung“ soll kurz definiert werden, da er im Lauf der Untersuchung öfter vorkommt: „*Unter Kompositsystemen werden Verwaltungen für Sach-, Unfall-, Haftpflicht- und Kfz-Versicherungen subsumiert.*“ (Aschenbrenner Michael et al. 2010, p.43).

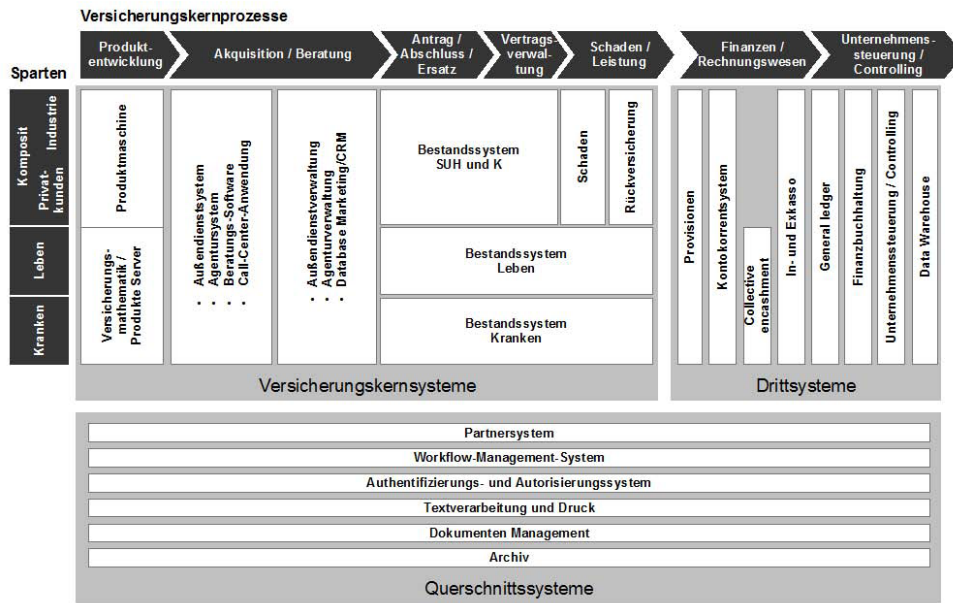


Abb. 8 Komponenten einer Versicherungsanwendung (Aschenbrenner Michael et al. 2010, p.42)

4.2 Organisatorischer Kontext

Das Vorgehensmodell für die Softwareentwicklung im untersuchten Projekt folgte einer Wasserfallstruktur mit drei Projektmeilensteinen pro Jahr. Wasserfallstruktur bedeutet in diesem Fall, dass jeder Meilenstein mit fest definierten Arbeitspaketen im Voraus geplant wird. Einzelne Arbeitspakete werden von verschiedenen, organisatorisch getrennten Abteilungen abgewickelt, wobei jede Abteilung intern wiederum in Teams für die einzelnen Versicherungssparten aufgeteilt ist. Zusätzlich zur Auftrennung in Sparten gibt es pro Abteilung auch Teams für die verschiedenen Komponenten. In Abb. 9 ist diese Aufteilung schematisch dargestellt.

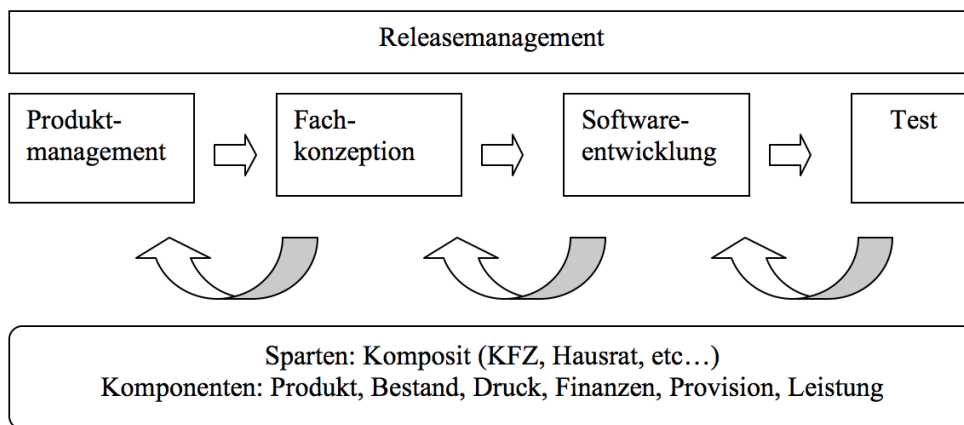


Abb. 9 Schematische Darstellung des Entwicklungsprozesses für den Bereich Komposit mit den beteiligten Abteilungen und der Prozessabfolge (Pfeile).

Eine Ausnahme von dieser Spartenauftrennung stellt die Abteilung Releasemanagement dar, diese kümmert sich sparten- und komponentenübergreifend um das Deployment des produzierten Codes.

In der Abteilung Produktmanagement werden gemeinsam mit den Kunden die Anforderungen für den nächsten Meilenstein definiert, anschließend werden diese Anforderungen in der Abteilung Fachkonzeption genauer spezifiziert und für die

Umsetzung in der Softwareentwicklungsabteilung ausgearbeitet. Die hierbei ausgearbeiteten Dokumente – in der untersuchten Firma werden diese „Fachkonzepte“ genannt - enthalten getrennte Beschreibungen jeweils über die Funktionalitäten des Produktkerns und der Bestandsverwaltung, dazu werden noch Testlisten und Testfallbeschreibung für die Testabteilung ausgearbeitet. In der Abteilung Softwareentwicklung werden die ausgearbeiteten Fachkonzepte schließlich in Sourcecode umgesetzt. Auch hierbei gibt es innerhalb einer Sparte meist eine Spezialisierung in ProduktentwicklerInnen und BestandsentwicklerInnen. Der produzierte Code wird von den EntwicklerInnen in das vom Releasemanagement betreute Buildsystem eingepflegt, und wird in der Folge für die Testabteilung verfügbar gemacht. Diese testet den produzierten Code und die entwickelten Funktionalitäten ebenfalls anhand der Fachkonzepte und meldet Fehler zurück an die Entwicklungsabteilung, bzw. die Fachkonzeption.

Die EntwicklerInnen im untersuchten Komposit-Projekt waren räumlich auf die Standorte in Österreich und Norddeutschland aufgeteilt, wobei in Österreich ausschließlich Produkt- und Bestandssystem entwickelt wurden. In Deutschland, wo sich auch die Teamleitung und HauptentwicklerInnen befanden, wurde ebenfalls am Produkt- und Bestandssystem gearbeitet, dazu wurden dort aber auch die benötigten Umsysteme, wie beispielsweise Leistung oder Druck, entwickelt. Zwei Personen auf österreichischer Seite waren neben der Programmierung auch mit koordinativen Belangen zwischen den beiden Standorten betraut, und fungierten als Schnittstelle zur Teamleitung in Norddeutschland.

4.3 Tools

Im Folgenden werden die in der untersuchten Firma verwendeten Tools, bzw. diejenigen Tools, die für das untersuchte Projekt und für die tägliche Arbeit der am Projekt beteiligten Personen wichtig sind, beschrieben. Dabei soll besonders auch auf Awarenessfeatures der Tools eingegangen werden. Die beiden wichtigsten hierbei erwähnten Tools sind *Lotus Notes*¹¹, welches einen Email-Client, Instant Messenger und weitere intranetbasierte Datenbankanwendungen enthält, sowie das Taskmanagementsystem *JIRA*, welches mit seiner technischen Basis und seinen Funktionalitäten der Definition von Enterprise 2.0 entspricht.

Eine Beschreibung dieser Systeme ist wichtig, da sie ein wichtiger Teil der täglichen Arbeit in der untersuchten Firma sind, und dort auch eine Art von Kommunikationsmittel oder Kommunikationsmechanismus darstellen, bzw. bestimmte Prozesse manifestieren. Informationen zwischen Mitarbeitern und Teams werden mittels dieser Systeme ausgetauscht. Außerdem soll somit auch ein Vergleich zwischen verschiedenen Awareness-Systemen – konkret zwischen einem webbasierten Issuetrackingsystem und „klassischen“ Kommunikations- und Awareness-Systemen - ermöglicht werden, bzw. soll es auch beitragen, die von Treude und Storey (2010) aufgeworfene Frage nach dem „optimal Toolset“ zu beantworten.

4.3.1 Lotus Notes

Bei Lotus Notes handelt es sich um eine verteilte, datenbankbasierte Groupware auf Client-Server-Basis, deren Basisfunktionalitäten unter anderem einen Email-Client, (geteilte) Kalender, Instant Messaging und einen Internetbrowser umfassen. Außerdem ist es möglich, in Lotus Notes eigene Anwendungen zu programmieren, die im Prinzip auf verteilten Datenbanken mit Eingabe-/Anzeigemasken, Visualisierungen und BenutzerInnenrechten beruhen. Im Prinzip ähneln solche Anwendungen herkömmlichen Internet-Services, mit dem Unterschied, dass sie eben ihre eigene, proprietäre Infrastruktur benötigen, welche aus den Lotus Notes Clients und Lotus Notes Servern („Domino-Server“) besteht. Zum Zeitpunkt der Untersuchung wurde Lotus Notes in der Version 8.5 verwendet.

¹¹www.ibm.com/software/at/lotus/

Lotus Notes dient als Basisinfrastruktur für viele Anwendungen der untersuchten Firma, wie beispielsweise ein internes Issuetrackingsystem („TTS“), eine Anwendung für die Verwaltung von Sourcecodenachlieferungen („REA“), eine Anwendung für Datenbank- und Systemdokumentationen, oder ein internes Zeiterfassungssystem. Diese Anwendungen werden über den Lotus Notes Client auf die gleiche Weise wie die Standardanwendungen gestartet. Abb. 10 zeigt einen Screenshot von Lotus Notes mit der geöffneten Emailanwendung. Oben in der Tab-Leiste sieht man auch, dass die Anwendung „TTS“ geöffnet ist. Auf der rechten Seite ist der integrierte Instant Messenger „Sametime“ geöffnet, darunter Plugins für Feed-Reader, Kalenderübersicht und Aktivitäten.

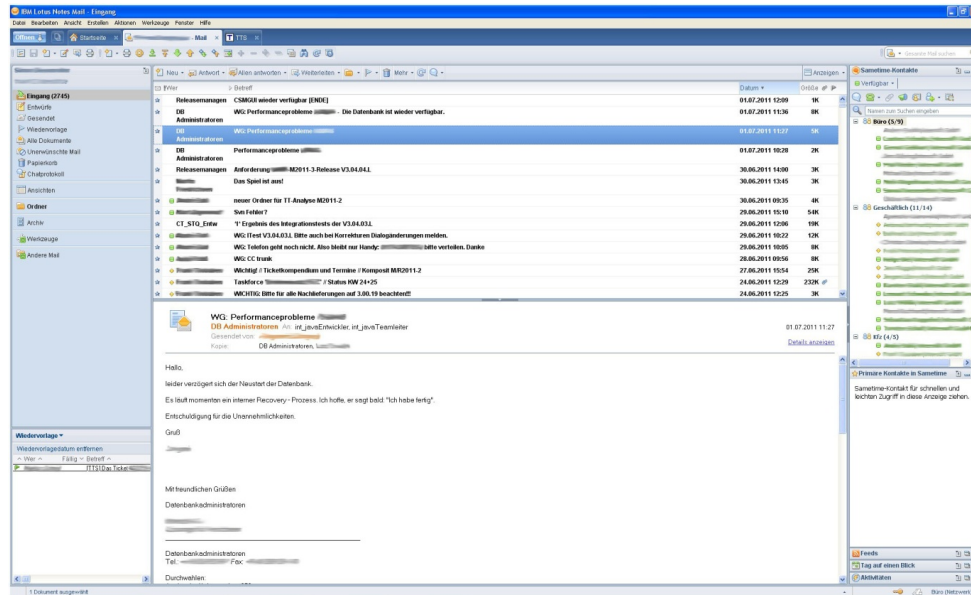


Abb. 10 Lotus Notes Client mit Ordnerstruktur links, Tableiste mit den Anwendungen Email und „TTS“, Nachrichtenübersicht Mitte oben, ausgewähltes Email Mitte unten, Sametime Instant Messenger rechts oben, Plugins für Feed-Reader, Kalenderübersicht und Aktivitäten rechts unterhalb von Sametime.

4.3.1.1 Lotus Notes Email

Der Emailclient von Lotus Notes (siehe auch Abb. 10) ähnelt sehr stark herkömmlichen Emailclients wie beispielsweise Mozilla Thunderbird¹². Die Ansicht bietet eine Ordnerstruktur für eingegangene, entworfene, gesendetete, gelöschte, etc. Emails. Ein weiteres Fenster zeigt eine Übersicht der Emails in einem ausgewählten Ordner. Diese Übersicht enthält pro Email den Absender, einen Betreff, Datum, Größe, Anhänge und Kennzeichnungen für besonders dringende oder wiedervorzulegende Emails. Neben den normalen Email-Adressen für die einzelnen MitarbeiterInnen gibt es auch diverse Emailverteiler für verschiedene Gruppen, dies sind zum Beispiel einzelne Teams, Abteilungen, die Gruppe der Java-Programmierer, oder andere mehr. Diese Emailverteiler spielen auch für Sametime eine Rolle. Zudem werden zusammengehörende Nachrichten (Nachrichten die als Antwort auf andere Nachrichten erzeugt wurden) automatisch zu einer Art „Konversation“ zusammengefasst. Bei solchen Nachrichten, die Teil einer Konversation sind, wird ein Pfeil vor der Betreffzeile angezeigt, wenn man diesen anklickt, werden die damit assoziierten Nachrichten aufgeklappt.

Der Emailclient enthält ein besonderes Awarenessfeature, und zwar wird – falls der/die BenutzerIn ebenfalls in Sametime eingeloggt ist – der Onlinestatus der AbsenderInnen angezeigt. Die Anzeige ist farblich codiert, im Tooltip sieht man auch den Ort. Weiters wird beim Eintreffen von neuen Nachrichten ein Briefsymbol im „Infobereich“ des Betriebssystems rechts unten angezeigt. Diese Form der Benachrichtigung stellt eine Art von peripherer Awareness nach Brush et al. (2002) dar.

¹²www.mozilla.org/de/thunderbird

4.3.1.2 Lotus Notes Sametime

Bei Sametime handelt es sich um den Instant Messenger von Lotus Notes. Dieser ist ein integrierter Bestandteil von Lotus Notes, und öffnet sich defaultmäßig beim Starten von Lotus Notes auf der rechten Seite (siehe auch Abb. 10). Die Benutzernamen werden mit dem firmenweiten Adressverzeichnis synchronisiert und sind identisch mit den jeweiligen Emailadressen der BenutzerInnen. Kontakte können aus dem Verzeichnisserver der Firma in die Kontaktliste des Messengers übernommen, und wahlweise in Gruppen organisiert werden. Es ist auch möglich ganze Emailverteiler als Gruppe zu importieren. Das hat den Vorteil, dass man bei personellen Änderungen (Zu- und Abgang von KollegInnen, Wechsel zwischen Teams) selbst keine Änderungen oder Anpassungen in der Kontaktliste vornehmen muss. Kontakte, die offline sind, werden bei dieser Art der Gruppenorganisation allerdings gar nicht angezeigt, ansonsten werden sie ausgegraut angezeigt.

Die Kontaktliste des Messengers bietet eine Reihe von Awarenessinformationen. Für jeden Kontakt ist dessen Onlinestatus sichtbar und farblich markiert (Grün: der Kontakt ist in Sametime angemeldet (online) und aktiv. Gelb: der Kontakt ist in Sametime angemeldet, aber nicht aktiv, d.h. es erfolgten keine Eingaben am Computer über einen bestimmten Zeitraum). Gleichzeitig ist auch der eigene Onlinestatus für andere, in deren Kontaktliste man sich selbst befindet, sichtbar. Wenn man einen Kontakt markiert oder eine Konversation startet, werden außerdem die Telefondurchwahl des Kontaktes, die lokale Zeit, der Ort und eine Statusnachricht („Ich bin verfügbar“, „Ich bin abwesend“, „In einer Besprechung“, „Bitte nicht stören“, etc.) angezeigt. Die Statusnachricht kann individuell angepasst werden. Weiters kann auch ein Bild des Kontaktes angezeigt werden. Durch die Integration von Sametime in Lotus Notes gemeinsam mit der Möglichkeit, eigene Anwendungen für Lotus Notes zu programmieren bleibt Sametime auch in den eigenen Anwendungen stets periphär sichtbar.

4.3.1.3 Weitere Lotus Notes Plugins

In der Randleiste von Lotus Notes gibt es neben der Sametimekontaktliste noch vier weitere Anzeigen (siehe auch Abb. 10): eine Liste mit favorisierten Kontakten, eine Anzeige für RSS-Feeds, eine Übersicht für verschiedene Kalendereinträge („Tag auf einen Blick“), sowie „Aktivitäten“. Bei „Aktivitäten“ handelt es sich um ein Tool, um online Aufgaben oder Todos, und dazugehörige Personen, Termine, Dokumente, Chatausschnitte oder Emails zu organisieren.

In den Interviews und bei der teilnehmenden Beobachtung zeigte sich aber, dass diese Features kaum genutzt werden. Nur eine Person gab an, die Kalenderübersicht in der Randleiste „selten“ geöffnet zu haben (siehe Interview 10_RMM_1: 137). Allerdings war eine quantitative Erfassung der Nutzung dieser Features nicht der Fokus der Untersuchung, weshalb diese Beobachtungen nicht repräsentativ sind.

4.3.2 TTS

TTS nennt sich das interne Ticketsystem der untersuchten Firma. Dabei handelt es sich um eine Eigenentwicklung auf Basis von Lotus Notes. TTS wird verwendet, um Fehler in der entwickelten Software in Form von sogenannten „Tickets“ zu dokumentieren. Beim Starten der Anwendung gelangt man auf eine Übersichtsseite, welche eine Liste mit allen existierenden Tickets enthält. Diese Liste kann durch verschiedene vordefinierte Filter („mir zugewiesen“, „von mit eröffnet“, „Tickets für Kunde XY“, etc., siehe Abb. 11) eingegrenzt werden. Die Übersichtsliste beinhaltet den Bearbeitungsstatus der Tickets („offen“, „geschlossen“, „wiedereröffnet“, „wartet auf Kundenrückmeldung“, „testbereit“), durch wen es erstellt und bearbeitet wurde, welches Team dafür verantwortlich ist, welches Programmmodul betroffen ist, eine Priorität (5 Stufen 1, 3, 5, 7, 9, mit 9 als der höchsten Priorität), eine interne und (falls vorhanden) eine externe Ticket-ID, eine Kurzbeschreibung des Fehlers, und einige andere Angaben mehr. Die Ansicht lässt sich im Layout etwas konfigurieren: Einzelne Spalten lassen sich per Drag&Drop umsortieren und in ihrer Breite

verändern. Außerdem können die Datensätze spaltenweise sortiert werden. Die Möglichkeit zur Sortierung ist zum Zeitpunkt der Untersuchung allerdings nicht für alle Filterungen gegeben.

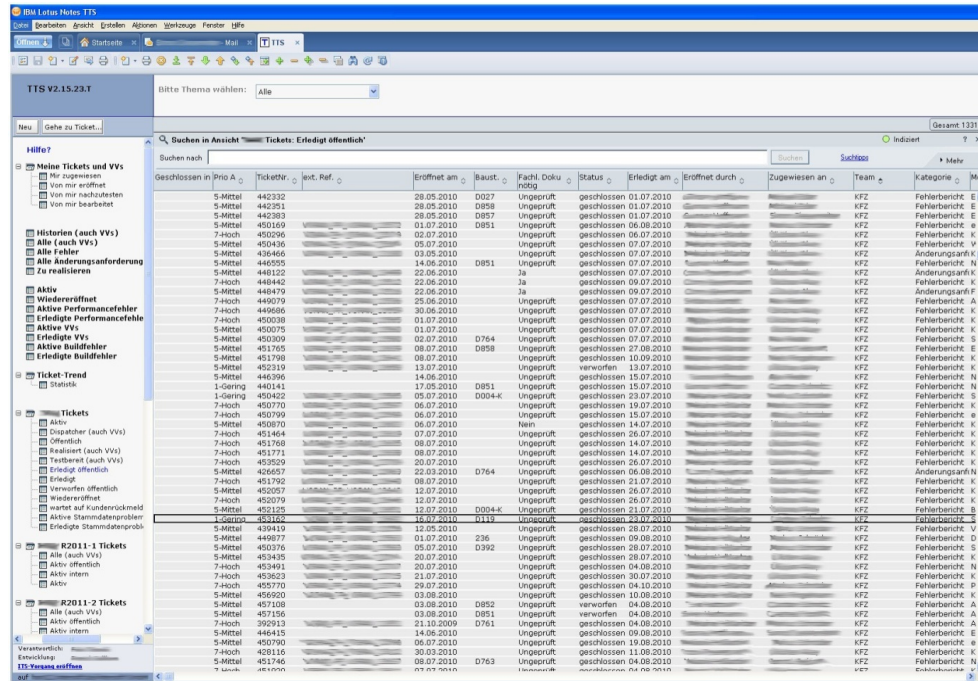


Abb. 11 TTS-Übersichtsseite: im linken Frame stehen verschiedene Filtermöglichkeiten zur Verfügung, rechts die (gefilterten) Tickets im Listenformat mit sortierbaren Spalten.

Für einzelne Tickets kann eine Detail- und Bearbeitungsansicht geöffnet werden. In dieser sind neben den in der Übersicht angezeigten Informationen weitere detailliertere Informationen ersichtlich, bzw. können diese hier eingetragen oder geändert werden (siehe Abb. 12). Durch Lotus Notes als Basis für diese Anwendung ist es auch möglich, direkt von der Detailansicht eines Tickets aus eine Email mit einem Link zu dem entsprechenden Ticket zu erzeugen („Verknüpfungsnachricht“). Diese erhält automatisch den Bearbeiter des Tickets als Adressat, die ID des Tickets als Betreff, sowie die Kurzbeschreibung, die externe ID und einen Link zu dem Ticket als Textkörper. Auch REA-Anträge (siehe nächster Punkt) können von TTS aus erzeugt werden.

Ticket: 453162 - Allgemein

Status :	geschlossen seit 23.07.2010	Eröffnet am :	16.07.2010
Meldung von :	██████████ Tel: ██████████	Wiedereröffnet am :	
Kunde* :	L9334	Erledigt am :	23.07.2010
Kategorie :	Fehlerbericht	TPR Ticket :	
Performancefehler :	<input type="checkbox"/>	Parallel-Ticket :	
Stamdatenfehler :	<input type="checkbox"/>	Dringlichkeit :	1-Gering
Klassifizierung :	Produktion	Reihenfolge :	
Eröffner :	██████████	Freigabestatus :	öffentlich
Testart :		Reproduzierbar* :	Ja
Kurztext* :	SUCHS D119+D392 V2, Schreibfehler beim AS-Grund + der Uebersetzung im Faehnchen	Geschätzter Aufwand (PT) :	
Nachttester :	██████████	Gebuchter Aufwand (PT) :	0,16
Zugewiesen an* :	██████████	PAZ-Aktivität :	
Team :	KFZ	Nachttest seitens ██████████ erledigt :	
Interner Status :	Ungeprüft	Status Analyse :	
Fachliche Doku nötig :		Status Analyse seit :	
Info zu fachl. Doku :		Abgestimmte Lieferversion :	
Referenz :	██████████		
Behebung bis :	2010-3		
Eröffnet in KW :	2010 28		
Erledigt in KW :	2010 29		

Softwarebezug

Produkt :	██████████	Plattform :	WindowsXP-Pro
Aufgetreten in Version :	V2.86.18.L	Datenbank :	Oracle
Umgebung / Schema* :	Test - TSTTEST01	Sprache :	deutsch
Zielversion* :	2010-3 (V2.88)	Lieferdatum :	
Realisiert in Version :	V2.88.07.L	Sparte* :	KFZ
Geschlossen in Version :	V2.88.07.L	Baustein :	D119 Vorgangs-Liste
GeVo :	keiner	Ausfertigungsgrund :	
Job :	keiner		

Testfallbezug

Interne Testfall-ID : ██████████
 Externe Testfall-ID : KO_FU_2501TG002F1,Step38
 Kommentar : 57175731; D119+D392

Ausführliche Beschreibung

Bitte berichtigen Sie im D119 den Ausfertigungsgrund und dessen Uebersetzung im Faehnchen. Es soll heissen: Storno zum Ablaufdatum nach Antragsmonierung und nicht Storno zu Ablaufdatum..... Das Gleiche gilt fuer den Baustein D392. Auch hier heisst es faelschlicherweise Storno zu..... Die Screenshots sind beigefuegt.

Hardcopies, Bearbeitungsvermerke und Kundenkontakt

Typ	Kurzbeschreibung/Titel	Erstellt	Zuletzt bearbeitet
Hardcopy	Nachttest erfolgreich (siehe HC)	23.07.2010 15:07:31	23.07.2010 15:07:53
Bearbeitungsvermerk	Kurz- und Bezeichnung im Ausfertigungsgrund korrigiere	16.07.2010 13:33:01	16.07.2010 13:33:36
Hardcopy		16.07.2010 12:41:52	16.07.2010 12:42:05

Zusätzliche Benachrichtigung wenn Ticket testbereit

mich als Interessent hinzufügen

Interessent: _____ Mailtext: _____

Bearbeitet durch

Abb. 12 TTS-Detailansicht: Der Bildausschnitt hier wurde zur vollständigen Übersicht aus zwei Bildschirmen zusammengefügt.

TTS verwendet Email als Benachrichtigungs-/Awarenessmechanismus bei Neuerstellung oder Änderung von Tickets. Dabei wird an die betroffene Person (jene, die ein Ticket zur Bearbeitung oder zum Nachttest zugewiesen bekommt) eine Email mit der Ticketnummer und einer kurzen Beschreibung des Ereignisses (z.B. „TT 123456 wurde Ihnen zugewiesen“ oder „TT 123456 ist zum Nachttesten bereit“) versandt.

Weitere Awarenessinformationen sind in TTS insofern gegeben, als die Übersicht und alle Tickets von allen in Lotus Notes eingeloggten Personen eingesehen werden können. Somit ist es möglich, auch die Tickets von anderen Teams oder Personen anzusehen, womit potentiell die Möglichkeit gegeben ist, sich über den Zustand eines Projektes, bzw. dessen Problemstellen oder sich über die Ticketanzahl von einzelnen Kolleginnen oder Teams zu informieren.

4.3.3 REA

REA ist ebenfalls eine Eigenentwicklung auf Lotus Notes Basis. Fehlertickets die im Produktivsystem eines Kunden auftreten und auch dort gelöst werden müssen, müssen formal mit dem Kunden über sogenannte „Nachlieferungsanträge“ abgestimmt werden. Dieser Prozess wird mit Hilfe von REA abgewickelt. Entweder von REA, oder auch direkt von TTS aus, kann ein neuer Nachlieferungsantrag erstellt werden. Dieser enthält im Wesentlichen eine oder mehrere Ticketnummern, die betroffenen Softwarekomponenten, sowie die Softwareversion(en), die nachgebessert werden soll(en). Überdies hat jeder Antrag einen Status: „eröffnet“, „genehmigt“, „abgelehnt“, „zurückgezogen“, „buildbereit“ oder „durchgeführt“. EröffnerIn eines Antrages ist im Normalfall der- oder diejenige, die das zugrunde liegende Fehlerticket bearbeitet. Für die REA-Anträge gibt es analog zu TTS eine Übersichts- und eine Detailseite. Die Übersichtsseite stellt auch hier auf der linken Seite verschiedene Filterungen über die REA-Datenbank zur Verfügung („Eröffnete“, „Genehmigte“, „Buildbereite Komponenten“, etc.), und rechts davon eine Übersicht der durch die Filterung ausgewählten Anträge mit Namen und Team der EröffnerInnen, Status des Antrages und anderen Daten. Der Status eines Antrages ist auch das primäre Filterkriterium für die Übersichtsfilter.

Awareness über Änderungen von einzelnen REA-Anträgen ist einerseits durch die Übersicht gegeben, die ein „Monitoring“ erlaubt: die Übersicht, und auch alle Detailinformationen eines Antrages ist, wie bei TTS, für alle eingeloggten BenutzerInnen ersichtlich. Über die Filterungen kann man sehen, in welchem „Filtertopf“ sich Anträge befinden. Diese Vorgehensweise erfordert allerdings ein wiederholtes Wechseln in die Übersicht, bzw. wiederholte Aktualisierungen der Übersicht. Ein weiteres Awarenessinstrument sind automatische Emailbenachrichtigungen, die bei Änderungen des Status an die betroffenen Personen verschickt werden.

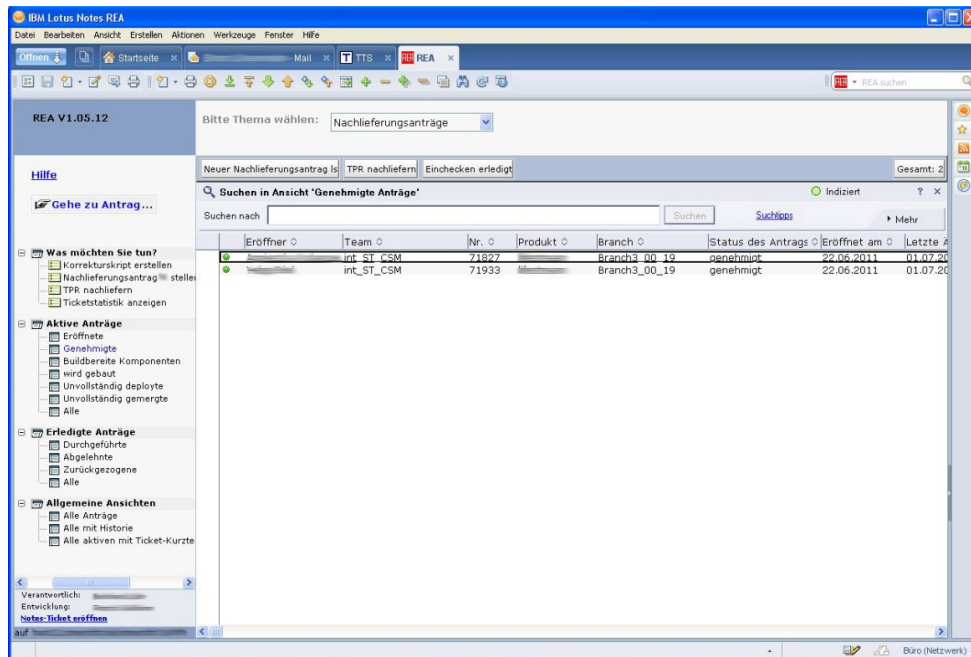


Abb. 13 REA Übersichtsseite: Im linken Frame stehen verschiedene Filtermöglichkeiten zur Verfügung, rechts die (gefilterten) REA-Anträge im Listenformat mit sortierbaren Spalten.

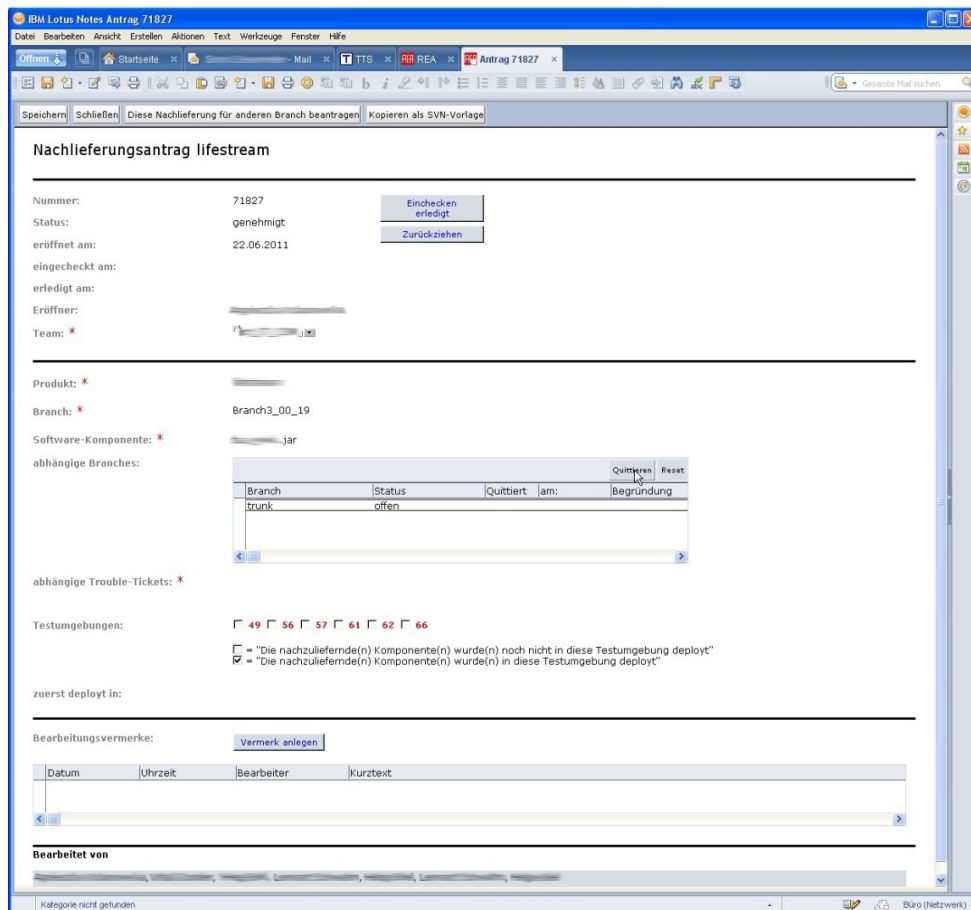


Abb. 14 REA Detailansicht: Neben spezifischen Informationen zum Antrag selbst im oberen Teil, bietet der untere Abschnitt die Möglichkeit Kommentare zu hinterlassen. Ganz unten ist außerdem ersichtlich, welche Personen den Antrag bereits bearbeitet haben.

4.3.4 JIRA

JIRA ist ein webbasiertes Issuetrackingsystem, welches in erster Linie zur Verwendung durch Softwareentwicklungsteams gedacht ist, und wird in der vorliegenden Arbeit als Repräsentant für Enterprise 2.0-Issuetrackingsysteme betrachtet. Aus diesem Grund wird es an dieser Stelle etwas ausführlicher beschrieben. Da der Fokus der Untersuchung aber auf einer personenzentrierten Betrachtung liegt, kann und soll hier keine vollständige Funktionsbeschreibung gegeben werden, sondern es wird nur versucht, die für den Kontext der Arbeit wichtigen Aspekte abzudecken. Diese sind die Klassifikation von JIRA als Enterprise 2.0-System, sowie eine Beschreibung der grundlegenden Strukturen und Funktionalitäten von JIRA, beziehungsweise von Issuetrackingsystemen allgemein, die für das weitere Verständnis der Untersuchung notwendig sind. Wo nötig, wird an späterer Stelle detaillierter auf weitere Funktionalitäten von JIRA eingegangen.

4.3.4.1 JIRA als Issuetrackingsystem

JIRA dient vorrangig zur Organisation von Projekten und Aufgaben, diese sind strukturell in drei hierarchische Ebenen gegliedert: Projekte (Projects), Aufgaben (Tasks) und Unteraufgaben (Subtasks). Aufgaben und Unteraufgaben werden auch als „Vorgänge“ oder „Issues“ bezeichnet. Auf der ersten Ebene stehen Projekte, darunter stehen Aufgaben, und darunter Unteraufgaben. Auf jeder Ebene können beliebig viele Elemente angelegt werden. Jedes Element der drei Ebenen muss mit einem Titel benannt werden, Aufgaben und Unteraufgaben erhalten zudem eine generische ID, die dem Titel vorangestellt angezeigt wird. Die Abbildungen Abb. 15, Abb. 16 und Abb. 17 zeigen diese drei Ebenen auf ihrer jeweiligen Übersichtsseite.

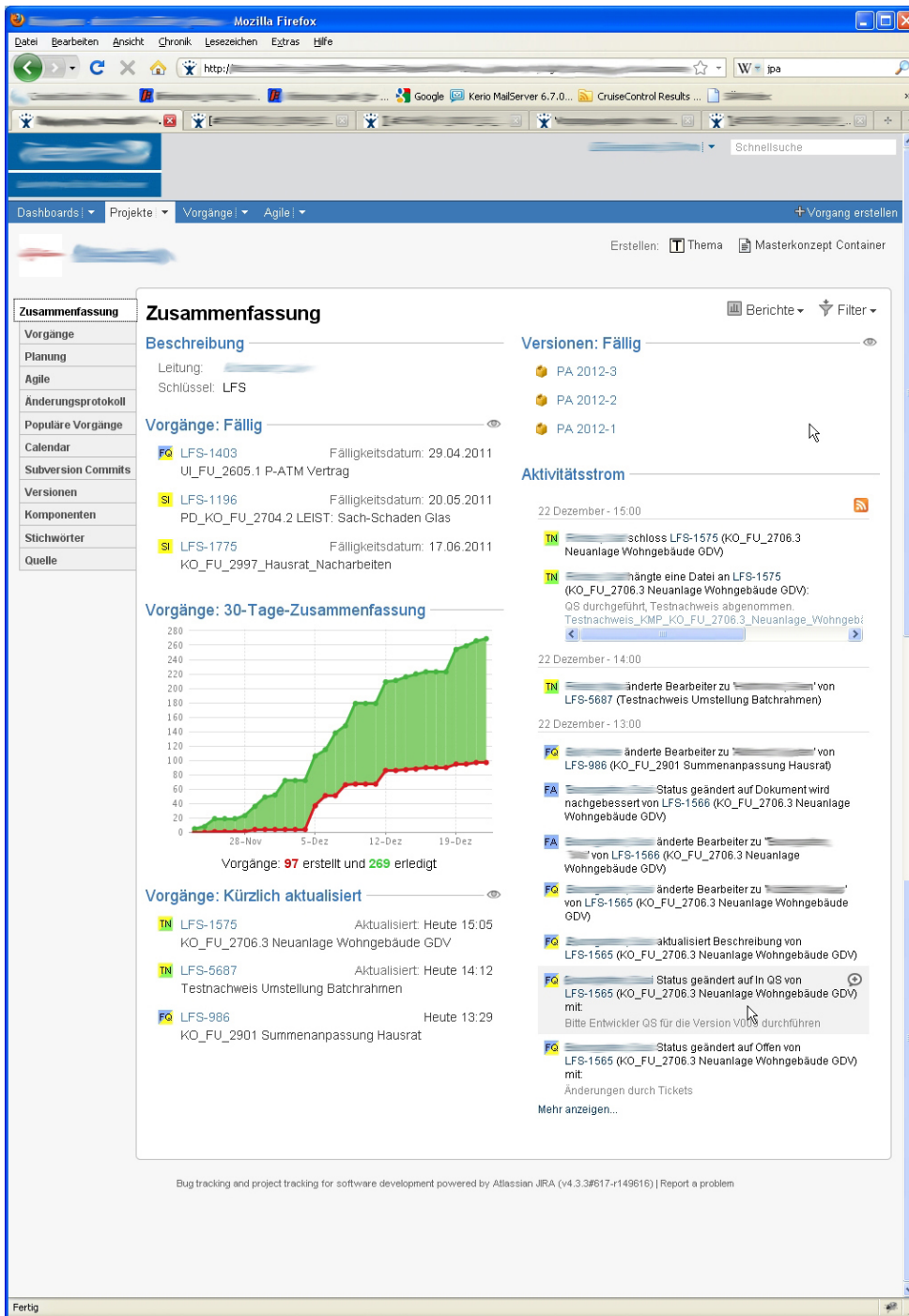


Abb. 15 Übersichtsseite für ein JIRA-Projekt: Links sind verschiedene weitere Ansichten für das Projekt (Vorgänge, Agile, Komponenten, Versionen, etc.); Zentral sind Diagramme, kürzlich aktualisierte Vorgänge, u.a. zu sehen.

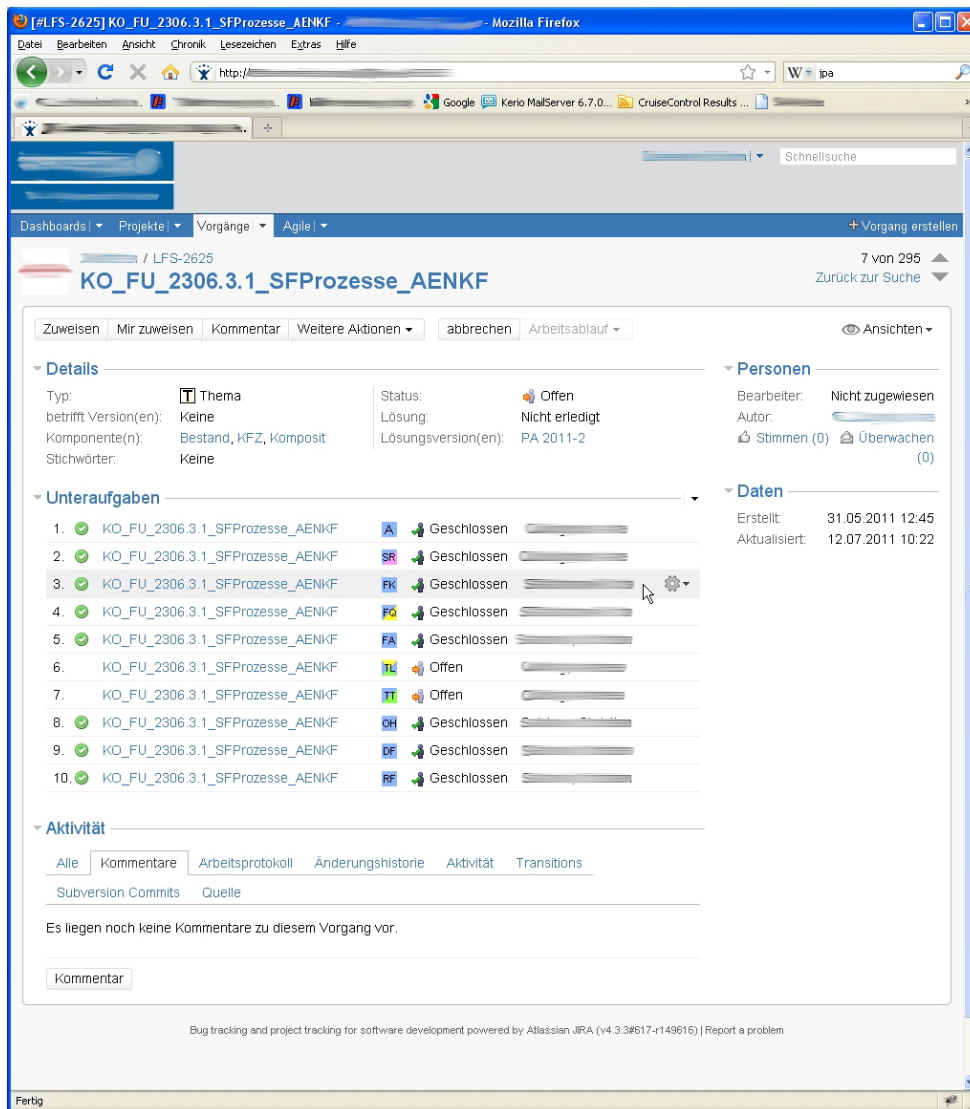


Abb. 16 Übersichtsseite für eine JIRA-Hauptaufgabe: Bietet einen Überblick über die Aufgabe mit den dazugehörigen Attributgruppierungen Details, Personen, Daten, Unteraufgaben und Aktivität.

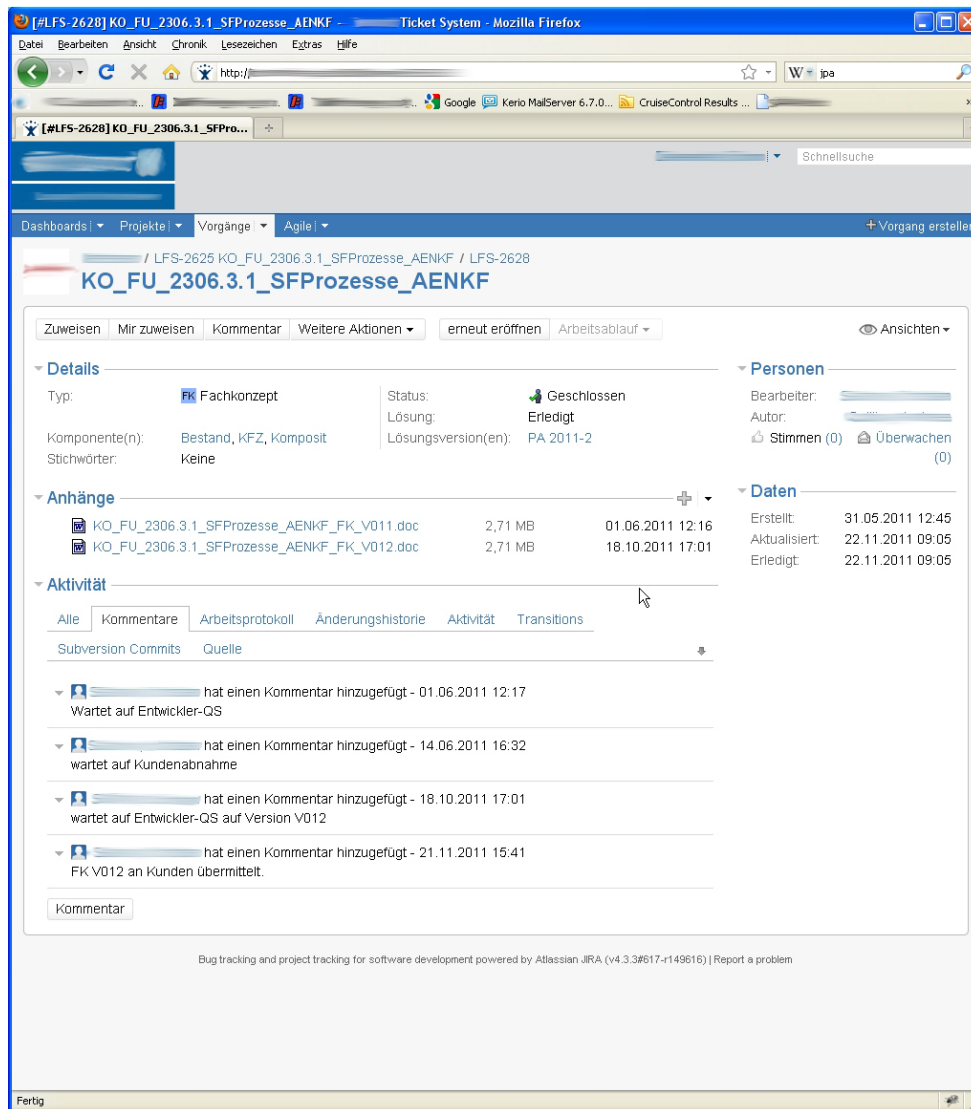


Abb. 17 Übersichtsseite für eine JIRA-Unteraufgabe: Zeigt Details, Anhänge, Zeitschätzungen, Bearbeiter, Bearbeitungsstatus, u.a.m.

Zusätzlich zu der Ebenen-Struktur von JIRA mit Projekten, Aufgaben und Unteraufgaben gibt es weitere Attribute um eine Strukturierung zu ermöglichen, welche thematisch zu „Details“, „Personen“ und „Daten“, „Anhänge“ und „Aktivitäten“ gruppiert sind (siehe Abb. 17). Auf der Ebene der Aufgaben gibt es dazu noch die Gruppe „Unteraufgaben“.

In der Gruppe „Details“ kann *Typ* als Merkmal frei spezifiziert werden und enthält in der untersuchten Firma beispielsweise die Ausprägungen „Thema“ oder „Fachkonzept“. *Versionen* sind ein wichtiges Element in der Softwareentwicklung und beziehen sich auf einen bestimmten Entwicklungsstand. In JIRA können für jedes Projekt beliebige Versionen oder Versionsnummern angelegt werden um in der Folge den dazugehörigen Aufgaben eine oder auch mehrere Versionen zuzuordnen. *Komponenten* beziehen sich ebenfalls auf die Softwareentwicklung und bezeichnen einzelne Objekte oder Pakete innerhalb eines Softwareproduktes. *Stichwörter* sind freie Schlagwörter, die für Projekte oder Aufgaben vergeben werden können um ebenso wie *Versionen* oder *Komponenten* effizientere Suchen zu ermöglichen und zusätzliche semantische Informationen vergeben zu können. *Status* und *Lösung* beschreiben den momentanen Zustand einer Aufgabe, typische Ausprägungen sind hier „offen“ oder „in Arbeit“ für *Status*, und „erledigt“ oder „nicht erledigt“ für *Lösung*.

Die Gruppe „Personen“ enthält Angaben über AutorIn und momentane BearbeiterIn eines Tasks, dazu gibt es hier noch die Möglichkeit über Tasks abzustimmen oder sich mittels „Überwachen“ über Änderungen am Task benachrichtigen zu lassen. Die Informationen,

wer wofür gestimmt hat und wer den Task überwacht, sind für alle SystembenutzerInnen ersichtlich.

Die Gruppe „Daten“ enthält zeit- und datumsrelevante Angaben zu einzelnen Aufgaben, wie Erstellungs- und Aktualisierungsdatum, oder eine Angabe, wann eine Aufgabe erledigt wurde. Neben solchen Angaben ist es in JIRA auch möglich Arbeitszeiten für einzelne Aufgaben mit geschätzter, verbleibender und protokollierter Dauer zu erfassen, diese Möglichkeit wurde aber zum Zeitpunkt der Untersuchung in der Firma nicht genutzt. Arbeitszeiten wurden in einer separaten Lotus Notes Anwendung erfasst.

Die Gruppe „Anhänge“ beinhaltet Dokumente, welche an einen Task hochgeladen wurden. Als zusätzliche Angaben sind die Dokumentgröße, sowie das Datum des Hochladens sichtbar. Obwohl es sich bei der Darstellungsform um eine Liste handelt, ist diese nicht sortierbar.

Die Gruppe „Aktivitäten“ schließlich beinhaltet alle mit einem Task einhergehenden Aktivitäten und Änderungen, und als wichtiges Kommunikationsinstrument die Möglichkeit, *Kommentare* zu Aufgaben und Unteraufgaben abzugeben. Kommentare können jederzeit von allen BenutzerInnen hinzugefügt werden, bei manchen Änderungen an einem Task ist das Verfassen eines Kommentares auch Pflicht. Das *Arbeitsprotokoll* für einen Task zeigt bei entsprechender Konfiguration die erfassten Arbeitszeiten an. Die *Änderungshistorie* loggt sämtliche Veränderungen mit Zeitpunkt, Benutzernamen, dem optionalen Kommentar der Änderung sowie den alten und neuen Wert eines geänderten Attributs. JIRA bietet weiters Integrationsmöglichkeiten mit anderen Softwareentwicklungstools wie Coderepositories oder Buildsystemen. Bei einer entsprechenden Konfiguration der Systemlandschaft sind Repositorycommits im Reiter *Aktivität* betrachtbar, ebenso wie die Ergebnisse von Buildvorgängen.

In der Gruppe „Unteraufgaben“ werden alle einer Hauptaufgabe zugeordneten Unteraufgaben gelistet. Die Listenelemente enthalten den Titel der Unteraufgabe, den Aufgabentyp, Aufgabenstatus (offen, geschlossen, etc.) sowie den oder die BearbeiterIn der Aufgabe. Die Liste ist nicht sortierbar, allerdings lassen sich Unteraufgaben durch ein beim Mouse-Over erscheinendes Menü bearbeiten, ohne dass die Unteraufgabe selbst geöffnet werden muss.

Neben diesen Detailattributen, die Informationen über den Typ und den Zustand eines JIRA-Tasks enthalten, gibt es auch die Möglichkeit, je nach Konfiguration eine freie, mit „Rich Text“ oder wiki-mässig formatierbare, textuelle Beschreibung zu einem Task hinzuzufügen.

Die wichtigste Funktionalität von Issuetrackingsystemen ist die Möglichkeit Aufgaben einer Person - oder auch einer Gruppe – zuzuweisen, und eine Aufgabe mit einem Status zu versehen. In JIRA gibt es standardmäßig verschiedene Statusse wie „offen“, „geschlossen“ oder „in Arbeit“, diese Statusse können aber beliebig konfiguriert, benannt oder erweitert werden. Die Zuordnung von Aufgaben zu Personen oder Gruppen erfolgt auf mehrere Arten direkt und indirekt: Zum einen wird bei einer Aufgabe immer der oder die ErstellerIn der Aufgabe referenziert („Autor“). Weiters kann eine Aufgabe einer Person oder Gruppe zugewiesen werden („Bearbeiter“), diese sollte sich im Normalfall dann um die Bearbeitung der Aufgabe kümmern. Aufgaben in JIRA können aber auch von dritten Personen bearbeitet werden, auch diese werden in jedem Fall referenziert und sind im Folgenden aus den Kommentaren oder der Änderungshistorie einer Aufgabe ersichtlich.

Generell gilt, dass jede Änderung einer Aufgabe mit einer Person verknüpft ist. Personen oder Gruppen müssen sich einmalig in JIRA registrieren um mit dem System arbeiten zu können. Für jede registrierte Person in JIRA wird automatisch eine Profildatei angelegt, diese enthält Informationen und Kontaktdaten zu der jeweiligen Person, Daten zu Gruppenzugehörigkeiten, oder Informationen über Aktivitäten dieser Person in JIRA. Diese Informationsseiten können von den BenutzerInnen erweitert werden, und beispielsweise um Profilfotos ergänzt werden. Die Verknüpfung von Aufgaben mit Personen stellt ein weiteres Instrument dar, um JIRA zu strukturieren, und „durchsuchbar“ zu machen.

Ein wichtiges Element zur Übersicht über JIRA sind sogenannte *Dashboards* (Dashboard = „Schwarzes Brett“). Bei einem Dashboard handelt es sich um eine Übersichtsseite, deren Inhalt registrierte BenutzerInnen selbst mittels sogenannter *Widgets* gestalten können. Widgets wiederum sind kleine Tabellen, Diagramme, Kalenderansichten oder ähnliches, die auf dem Dashboard frei platziert werden können und die es erlauben, bestimmte Inhalte zu visualisieren. Der Inhalt der Widgets schließlich wird mittels sogenannter *Filter* bestimmt. Filter sind eine Möglichkeit, die in den Widgets anzuzeigenden Tasks und Informationen nach bestimmten Kriterien wie Personen, Status oder Terminen einzugrenzen. Technisch wird dies mit einer SQL-ähnlichen Abfragesprache realisiert. Die BenutzerInnen haben dabei die Möglichkeit solche Abfragen direkt in der Abfragesyntax zu spezifizieren, mittels eines Editors bestimmte Filterkriterien zu erstellen, oder vorgefertigte Filter zu verwenden, wobei einige vorgefertigte Filter standardmässig in JIRA integriert sind (z.B. „Mir zugewiesene Aufgaben“). Es können auch eigene Filter erstellt, und mit anderen registrierten BenutzerInnen geteilt werden. In Abb. 18, Abb. 19 und Abb. 20 sind drei Beispiele für Kalender, eindimensionale und zweidimensionale Filterwidgets abgebildet.



Abb. 18 Kalenderwidget: Anzeige aller offenen Unteraufgaben von eingeloggtten BenutzerInnen mit Fälligkeitsdatum im aktuellen Monat

Filterergebnisse: KMP-Benutzer / zugewiesene Unteraufgaben					
S	Schlüssel	Zusammenfassung	Priorität	Fällig	Status
RT	LFS-1134	LFS-1119 / KO_FU_2909 Nacharbeiten LEIST-Kfz		29.07.2011	Offen
TK	LFS-1133	LFS-1119 / KO_FU_2909 Nacharbeiten LEIST-Kfz		29.07.2011	Offen
IMP	LFS-1131	LFS-1119 / KO_FU_2909 Nacharbeiten LEIST-Kfz		29.07.2011	In Arbeit
FQ	LFS-1122	LFS-1119 / KO_FU_2909 Nacharbeiten LEIST-Kfz			Dokument wird nachgebessert

Vorgänge 1 bis 4 der 4 entsprechenden Vorgänge werden angezeigt.

Abb. 19 Eindimensionales Filterwidget: Anzeige aller zugewiesenen Unteraufgaben von eingeloggtten BenutzerInnen

Zweidimensionale Filterstatistik: LFS / KMP-Bearbeiter / alle Unteraufgaben / offen				
Vorgangstyp	Status			S:
	Offen	In Arbeit	In QS	
DT Doku-Pool technisch	1	0	0	1
FQ Fachkonzept QS	0	0	1	1
IMP Implementierung	20	0	0	20
IMP Implementierung mit interner QS	2	0	0	2
RT Release Notes technisch	2	0	0	2
SI Schätzung Imp	0	1	0	1
TK Technisches Konzept	2	0	0	2
TS Testdurchführung STQ	1	0	0	1
TL Testliste	1	0	0	1
TN Testnachweise	2	0	0	2
Summe, Vorgänge:	31	1	1	33

Zeigt 10 von 10 Statistiken an.
Filter: LFS / KMP-Bearbeiter / alle Unteraufgaben / offen

Abb. 20 Zweidimensionales Filter-Widget: Anzeige aller Aufgaben von eingeloggtten BenutzerInnen nach Vorgangstyp und Status

4.3.4.2 JIRA als Workflowmanagementsystem

JIRA erlaubt die Abbildung von Geschäfts- oder Arbeitsprozessen mittels sogenannter „Workflows“. Grundlage dafür ist die Möglichkeit Taskparameter wie Tasktyp oder Taskstatus anzupassen und zu konfigurieren. Ein Workflow sorgt dann dafür, dass abhängig vom Tasktyp beispielsweise der Taskstatus nur gewechselt werden darf, wenn bestimmte Kriterien innerhalb eines Tasks erfüllt sind, und es werden auch die möglichen Zustände eingeschränkt, in die gewechselt werden kann. Ein Beispiel für einen solchen Workflow würde beispielsweise ein JIRA-Task darstellen, dessen mögliche Statusausprägungen- und Wechsel der Abfolge „Offen“, „In Arbeit“, „Realisiert“ und „Geschlossen“ unterliegen, wobei beispielsweise vom Status „Geschlossen“ nur wieder in

den Status „Geöffnet“ gewechselt werden kann. Mittels eines Buttons kann der für einen Tasktyp vorgesehene Workflow jederzeit betrachtet werden. Die Erstellung und Konfiguration von Workflows und Tasktypen kann durch verschiedene Benutzerrechte beschränkt werden.

4.3.4.3 JIRA als Enterprise 2.0-System

Wie schon eingangs erwähnt, erfüllt JIRA mit seinen Funktionalitäten die Kriterien für Enterprise 2.0 nach der Definition von McAfee (2006): Es bietet eine umfangreiche *Suchfunktionalität*, *Links* bilden die Basis für die Navigation innerhalb von JIRA und dienen auch als Verbindung nach außen. *Authoring* wird einerseits durch verschiedene Kommentarfunktionen erfüllt, andererseits ist es registrierten Usern möglich eigene Projekte und Issues zu erzeugen und auch selbst erstellte Filter können für andere BenutzerInnen verfügbar gemacht werden. *Tagging* wird strukturell durch einerseits vorkonfigurierbare Attribute wie beispielsweise Versionen oder Komponenten unterstützt, andererseits durch frei angebbare Tags in dafür vorgesehenen Feldern. *Extensions* werden durch verschiedene Widgets und Visualisierungen realisiert, die einem User potentiell für sie interessierende Aktivitäten oder Statistiken präsentieren. *Signals* werden beispielsweise durch Email-Benachrichtigungen im Fall von Änderungen von Aufgaben realisiert.

Auch die Forderungen von Hinchcliffe (2007) an Enterprise 2.0-Systeme werden von JIRA erfüllt: Sobald das System auf einem Server aufgesetzt wurde, kann es prinzipiell über jeden Browser verwendet werden. Es gibt keine Vorgaben für Rollen und es ist den BenutzerInnen freigestellt, welche Art von Aufgaben sie damit verwalten oder mit welcher Art von Inhalt das System befüllt wird, wodurch es den Aspekt *Freeform* erfüllt. Auch der Aspekt *Network-Oriented* wird erfüllt, da sich der Inhalt von JIRA webbasiert durchsuchen, verlinken oder mit anderen BenutzerInnen teilen lässt. Standardisierte Schnittstellen erlauben auch die Verbindung von und zu externen Systemen und Inhalten. Auch die Aspekte *Social* und *Emergence* werden von JIRA erfüllt, da es eine transparente, frei konfigurierbare Struktur bietet, die keine fix vordefinierten Abläufe vorschreibt, sondern den BenutzerInnen in erster Linie eine Plattform für die Organisation und Kommunikation von Aufgaben bietet, und es aber schließlich auch an den BenutzerInnen liegt das System für sie gewinnbringend zu verwenden. Weitere Funktionalitäten wie die Möglichkeit Abstimmungen durchzuführen tragen potentiell zu den von Hinchcliffe (2007) angesprochenen sozialen Aspekten von Enterprise 2.0 bei.

4.3.4.4 Awarenessmechanismen in JIRA

Awarenessinformation in JIRA wird sowohl über Push- als auch über Pullmechanismen verbreitet. Das Informationspushing erfolgt ereignisbasiert bei Änderungen von Aufgaben und Projekten, die Benachrichtigung erfolgt über Email oder auch über RSS-Feeds. Die generierten Emailnachrichten enthalten neben Informationen über den Vorgang selbst an erster Stelle Informationen über Veränderungen am Vorgang: Wer, wann und was wurde verändert (siehe Abb. 21). Die Registrierung („Subscription“) für Benachrichtigungen erfolgt teils automatisch, indem die jeweiligen AutorInnen und BearbeiterInnen von Aufgaben standardmäßig vom System für Benachrichtigungen registriert werden, aber auch Dritte können sich mittels eines „Überwachen“-Buttons für Änderungsbenachrichtigungen anmelden, bzw. die entsprechenden Feeds für eine Aufgabe abonnieren.

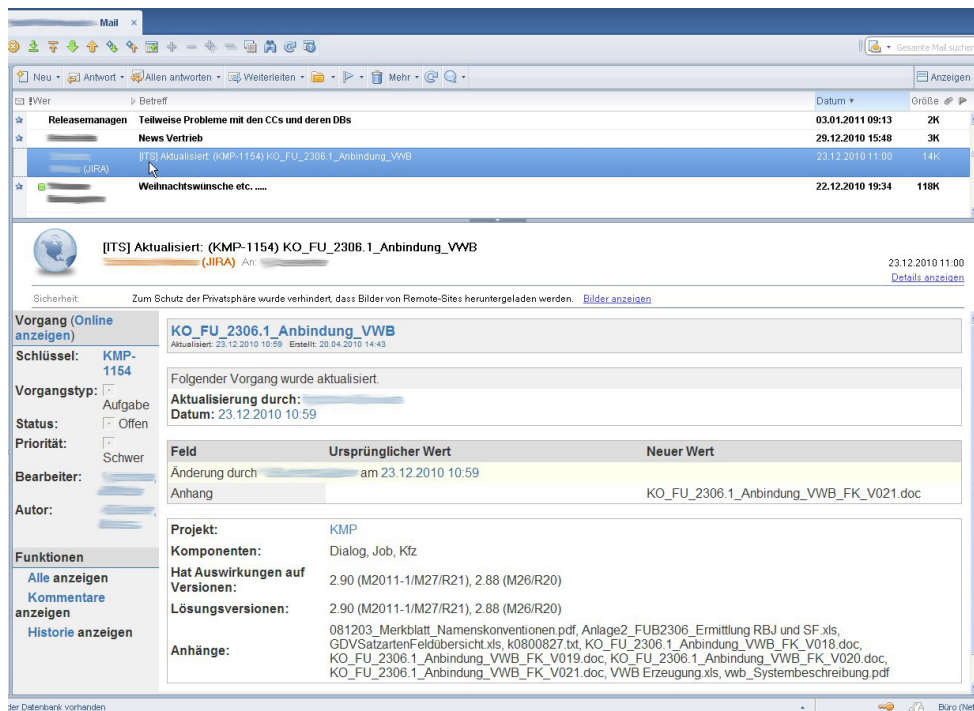


Abb. 21 Von JIRA generierte Benachrichtigungsemail: Informationen über Änderungen (wer, wann und was), dazu Informationen über den Vorgang selbst.

Neben diesen pushbasierten Awarenessinformationen über Änderungen bietet JIRA eine Reihe von Informationen, die BenutzerInnen im Pullverfahren selbst auslesen müssen. Bezogen auf einzelne Tasks gibt es dazu in erster Linie die Informationen zum aktuellen Status eines Tasks, die durch die Detailansicht an sich ausgelesen werden können. Zudem gibt es zu jedem Task eine Änderungshistorie, welche den Verursacher, den alten und den neuen Wert jeder Änderung darstellt. Bezogen auf Projekte bieten die verschiedenen Filter und Widgets, die auf den Dashboards und den Projektübersichtsseiten konfiguriert werden können, die Möglichkeit, Status und Veränderungen in Projekten und Aufgaben zu verfolgen.

4.3.5 Weitere Anwendungen

Neben den genannten Anwendungen Lotus Notes und JIRA werden eine Reihe weiterer Anwendungen für die Produkt- und Softwareentwicklung in der untersuchten Firma verwendet. Dazu zählen herkömmliche Büroanwendungen wie Textverarbeitungs- und Tabellenkalkulationsprogramme, Programme zur Erstellung von Projektabläufen, und spezielle für die Softwareentwicklung eingesetzte client- und serverseitige Anwendungen wie Coderepositories (SVN), Tools für Continuous Integration (CruiseControl), Entwicklungsumgebungen (Eclipse) oder verschiedene Plugins dafür (z.B. MyLyn). Neben diesen Anwendungen stehen innerhalb der Firma auch eine Reihe von Netzlaufwerken zur Verfügung, die zum Austausch von Daten verwendet werden.

5. Ergebnisse Feldforschung

Im folgenden Abschnitt werden die Ergebnisse der teilnehmenden Beobachtung und der Interviews in der Form eines Kategoriensystems, welches mittels der Grounded Theory Methodik entwickelt wurde, dargestellt. Zuerst wird das Kategoriensystem und dessen Entwicklungsprozess vorgestellt. Danach werden die vier entwickelten Hauptkategorien mit ihren Ausprägungen und Merkmalen dargestellt. Zur besseren Illustration der Beschreibungen und der Analysen werden Ausschnitte aus den Interviews eingefügt. Zuletzt werden die einzelnen Kategorien und ihre Zusammenhänge als Modell von Awareness dargestellt.

5.1 Entwicklung des Kategoriensystems

Die Analyse der empirischen Daten erfolgte wechselseitig mit der Analyse der zu Verfügung stehenden Literatur und Basistheorien. In einem ersten Schritt der Aufarbeitung der Beobachtungsprotokolle lag der Fokus noch darauf, jene auch in den Kapiteln 2.2 und 2.4 aufgelisteten Ausprägungen und Mechanismen von Awareness zu identifizieren und zu klassifizieren. Auch zu Beginn des offenen Codierens der ersten Interviews wurde versucht, die Ausprägungen und Mechanismen des Theorieteils auf die beobachteten Phänomene und Aussagen als Grobkategorien anzuwenden. Eine Zuordnung dieser (theoretischen) Codes zu den Interviewtexten stellte sich aber rasch als problematisch heraus, da sich einerseits die in der Literatur identifizierten Ausprägungen und Mechanismen viel zu stark überschneiden, und andererseits die getätigten Aussagen viel zu vielschichtig waren, um sie einer oder mehreren der Grobkategorien sinnvoll zuordnen zu können. Auch andere, vorab erdachte Codes und Grobkategorien, die hauptsächlich aus dem Interviewleitfaden stammten, bzw. dem Versuch die anfänglichen Leitfragen nach der Verwendung mehrerer Tools zu beantworten geschuldet waren, erwiesen sich besser als Indizierung der Texte denn als brauchbare Kategorien zur Strukturierung der Texte.

Im weiteren Verlauf der verschiedenen Codierschritte entwickelten sich jedoch eine Reihe von konzeptuellen- und In-Vivo-Codes aus den Interviewtexten heraus, wie beispielsweise „Risiken“, „Unterschiedliche Verwendung von JIRA“, „Bescheid sagen“ oder „Transparenz“, welche sich als praktikabler und auch sinnvoller für die Kategorisierung der zugrundeliegenden Interviews und Beobachtungsprotokolle erwiesen. Auf Basis solcher Erkenntnisse, nicht nur aus dem offenen Codieren sondern auch aus späteren Analysestadien wie dem axialen- und selektiven Codieren, wurde auch die Literatur regelmässig neu betrachtet und erneut analysiert, was schließlich in der theoretischen Ausarbeitung verschiedener Inhalte und Faktoren von Awareness wie sie in den Kapiteln 2.5 und 2.6 angegeben sind resultierte. Umgekehrt resultierte diese erneute Beschäftigung mit der Literatur wiederum in der Weiterentwicklung des Code- und Kategoriensystems. Nicht nur das Wechselspiel zwischen Literatur- und Textanalyse folgte einem zyklischen Ablauf, auch die einzelnen Analysephasen von offenem-, axialen- und selektivem Codieren wurden zyklisch, mit Vor- und Rücksprüngen durchgeführt. Vergebene Codes wurden ausdifferenziert, umsortiert, gestrichen oder um andere Codes oder Kategorien subsummiert.

Ein weiterer wichtiger Aspekt, der die Entwicklung des Kategoriensystems anleitete, war die Beobachtung verschiedener Phänomene, wie beispielsweise die Beobachtung, dass bestimmte technische Awarenessmechanismen aus verschiedenen Gründen im alltäglichen Arbeitskontext nicht wie gedacht funktionieren, oder dass handelnde Personen eigene Kommunikationsmechanismen entwickeln, da die zu Verfügung stehenden technischen Möglichkeiten ihre Awarenessbedürfnisse nicht befriedigen. Eine 1:1 Überführung dieser Phänomene in eigene Kategorien war häufig nicht möglich, eine Analyse mittels des bereits beschriebenen *Codierparadigmas nach Strauss* nach ihren Merkmalen, Ursachen und Auswirkungen stellte sich jedoch als sinnvoll für die Weiterentwicklung des Kategoriensystems heraus.

5.2 Das Kategoriensystem

Schließlich entstanden mittels der beschriebenen Instrumente der Grounded Theory Methodik vier Hauptkategorien, die gemeinsam eine Schlüsselkategorie „Awareness im Enterprise 2.0“ konstituieren. Diese vier Hauptkategorien sind „Awarenessbedürfnisse“, „Toolverwendung“, „Awarenessmechanismen“ und „Faktoren“. Die vier Kategorien spiegeln sich durch die zuvor beschriebene Arbeitsweise teilweise auch im Theorieteil dieser Arbeit wider. Im Sinne des Codierparadigma nach Strauss (siehe Abschnitt 3.3) stellt „Awarenessbedürfnisse“ die zentrale Kategorie dar. „Awarenessmechanismen“ kann als die Handlungsstrategien für die Umsetzung von Awareness betrachtet werden. „Toolverwendung“ kann im Codierparadigma als Konsequenz von Awarenessbedürfnissen betrachtet werden. „Faktoren“ schließlich fasst Kontext, ursächliche und intervenierende Bedingungen der drei vorangehenden Kategorien zusammen. Eine schematische Übersicht über das Kategoriensystem ist in Abb. 22 dargestellt.

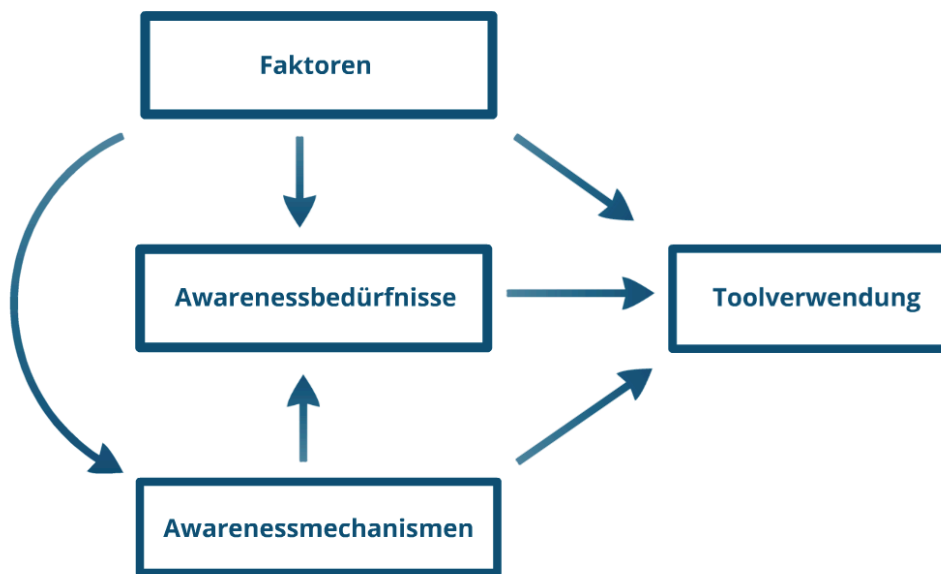


Abb. 22 Hauptkategorien von Awareness im Enterprise 2.0 im Codierparadigma nach Strauss

Die **Kategorie Awarenessbedürfnisse** ergibt sich aus einer Betrachtung der Texte nach dem Inhalt des Awareness-Fokus der interviewten Personen, also den auch im theoretischen Teil angesprochenen „Entitäten“ welche im Lauf der Analyse zu einer gemeinsamen Überkategorie zusammengefasst wurden. Der Begriff „Awarenessbedürfnisse“ selbst ist der einschlägigen Literatur entliehen, spiegelt aber am besten die von den Interviewten getätigten Aussagen wider. Die einzelnen Subkategorien sind Awareness über Aufgaben, Projekte, Termine, Artefakte, Tools und Infrastruktur, Personen und Gruppen, Prozesse, Risiken und Probleme.

Die **Kategorie Awarenessmechanismen** beschreibt verschiedene Mechanismen, bzw. Kommunikationsweisen- und Konzepte, die von den handelnden Personen verwendet werden um Awareness herzustellen und zu erlangen. Die verschiedenen Kommunikationsmuster können den auch im Theorieteil identifizierten Awarenessmechanismen Push und Pull, Subscription und Notification, sowie periphere Awareness zugeordnet werden. In der Untersuchung zeigt sich, dass meist eine Mischung aus toolbasierter und direkter Kommunikation verwendet wird.

Die **Kategorie Toolverwendung** umfasst Phänomene die die Verwendung der verschiedenen, in der untersuchten Firma in Verwendung stehenden Tools betreffen. Ein zentrales Phänomen ist die Beobachtung, dass das untersuchte Enterprise 2.0-Tool „JIRA“ auf viele verschiedene Arten verwendet und konfiguriert wird, sowohl um Prozesse auf Team- oder Projektebene zu unterstützen, als auch um individuelle Bedürfnisse zu befriedigen. Die Abbildung von Awarenessinhalten in den Tools ist die

Grundvoraussetzung für toolvermittelte Awarenessmechanismen. Die Abbildung von Aufgaben und abhängigen Inhalten in der untersuchten Firma in JIRA und anderen Tools ermöglicht Awareness über eigene und fremde Themen, stellt sich aber als komplexe Herausforderung dar, die Wissen über die Konfigurationsmöglichkeiten der Tools selbst und die im System abgebildeten Prozesse und Konventionen benötigt.

Die **Kategorie Faktoren** schließlich beschreibt Einflussfaktoren, die sich auf einzelne oder auch mehrere der zuvor genannten Kategorien auswirken. Faktoren wie „Rollen“ oder „Teamgrösse“ entstehen aus dem organisatorischen Kontext heraus, andere Faktoren ergeben sich aus den Tools und den darin abzubildenden und zu kommunizierenden Awarenessinformationen.

Im Folgenden werden die einzelnen Kategorien mit ihren Ausprägungen und Merkmalen beschrieben, abschliessend wird das Zusammenspiel und Beziehungsgefüge der einzelnen Kategorien als Modell von Awareness im Enterprise 2.0 zusammengefasst.

5.3 Kategorie Awarenessbedürfnisse

Die Kategorie Awarenessbedürfnisse umfasst mit ihren Subkategorien jene Begriffe, auf welche die in dem untersuchten Projekt handelnden Personen ihre Aufmerksamkeit richten und worüber sie Awareness erlangen wollen und müssen. Die in den Interviews und der teilnehmenden Beobachtung identifizierten Entitäten und Begriffe sind Awareness über Aufgaben, Projekte, Termine, Artefakte, technische Systeme, Personen und Gruppen, Prozesse und schließlich Risiken und Probleme (siehe Abb. 23).



Abb. 23 Kategorie Awarenessbedürfnisse mit Ausprägungen

Diese *Inhalte von Awareness* sind begründet durch das Arbeitsumfeld und sind untereinander stark gekoppelt. Neben Ereignissen wie Änderungen die an diesen Ausprägungen passieren und für die handelnden Personen von Interesse sind, wird der Zustand der Entitäten auch ohne das Vorhandensein von konkreten Ereignissen abgefragt. Nicht nur Übersichtswissen ist gefragt, auch Hintergründe und Abhängigkeiten sind für die handelnden Personen von Relevanz.

5.3.1 Inhalt von Awareness

Im folgenden Abschnitt werden jene Inhalte dargestellt, auf welche die in dem untersuchten Projekt handelnden Personen ihre Aufmerksamkeit richten und worüber sie Awareness erlangen wollen und müssen.

5.3.1.1 Aufgaben

Typische im Feld beobachtete Aufgaben sind die Erstellung eines Fachkonzeptes oder von Teilen eines Fachkonzeptes, die Umsetzung einzelner Anforderungen eines Fachkonzeptes in Sourcecode, das Beheben von Fehlern in der bestehenden Software, oder das Testen von einzelnen Funktionalitäten der Software. Diese Aufgaben verfügen meist über eine beliebige **Granularität**, d.h. sie können selbst wiederum in weitere Unteraufgaben aufgeteilt werden. Beispielsweise umfasst die Aufgabe „Technische Implementierung einer fachlichen Anforderung“ das Erstellen von Einträgen auf Datenbankebene, die Erstellung von Zugriffsmethoden auf Codeebene und vieles mehr.

Awareness über Aufgaben betrifft im wesentlichen vier Aspekte: Zuerst stellt sich die Frage, ob es **Änderungen** an einer Aufgabe gibt, ob neue Anforderungen hinzugekommen sind, ob es neue Fachkonzeptversionen gibt die umzusetzen sind, oder auch, ob Aufgaben als Ganzes verworfen wurden. Die zweite Frage betrifft die **Ursachen** und den Kontext von Änderungen: Warum wurde eine Aufgabe geändert, warum gibt es eine neue Version eines Fachkonzeptes. Die dritte Fragestellung betrifft **personelle Aspekte** von Änderungen, also beispielsweise wer eine Aufgabe oder ein Ticket erstellt hat, von wem wurde ein Ticket getestet, wer hat an einer Aufgabe gearbeitet. Ein vierter Aspekt der erfragt wird, betrifft den Zustand oder **Status** von Aufgaben: „Wie weit ist eine Umsetzung vorangeschritten“, „ist eine Aufgabe abgeschlossen“ oder „welche Unteraufgaben sind noch nicht abgeschlossen“.

Das Awarenessbedürfnis über Aufgaben ist abhängig von der **Rolle** die eine handelnde Person einnimmt: Für EntwicklerInnen, FachkonzepterInnen oder TesterInnen bedeuten Änderungen an Aufgaben, dass sie selbst aktiv werden müssen und wenn nötig diese Änderungen umsetzen müssen. Hintergrundinformationen benötigen sie, um den Arbeitsaufwand und etwaige Schwierigkeiten bei der Umsetzung abschätzen und beurteilen zu können. Arbeitsaufwände und Schwierigkeiten werden in der Regel auch den jeweiligen TeamleiterInnen mitgeteilt. Über die personellen Aspekte müssen sie informiert sein, um sich gegebenenfalls mit den betreffenden Personen über Details oder Unklarheiten der auszuführenden Inhalte austauschen zu können. Desweiteren ist das Wissen, von wem eine Änderung oder Anmerkung zu einer Aufgabe kommt, hilfreich bei der Einschätzung der Dringlichkeit einer Aufgabe. Über den Status von Aufgaben müssen sie schließlich Bescheid wissen, da beispielsweise bestimmte Arten von Aufgaben wie Fehlertickets nicht von den umsetzenden Personen selbst für abgeschlossen erklärt werden können, sondern erst von anderen Personen geprüft oder getestet werden.

Die Fragen die TeamleiterInnen über Aufgaben stellen, sind relativ ähnlich zu denen ihrer MitarbeiterInnen. Ein grosser Unterschied besteht allerdings im Fokus der Aufgaben: Während „normale“ MitarbeiterInnen hauptsächlich ihre eigenen Aufgaben betrachten, haben TeamleiterInnen vor allem die Aufgaben ihrer MitarbeiterInnen im Blick. Bei Änderungen von Aufgaben müssen sie gegebenenfalls dafür sorgen, dass diese auch umgesetzt werden, und dazu beispielsweise die Aufgaben einer neuen oder anderen Person zuweisen. Zudem müssen sie auch den Arbeitsaufwand und eventuelle Risiken einschätzen, was häufig gemeinsam mit den umsetzenden MitarbeiterInnen geschieht. Falls diese Risiken sehr groß sind, liegt es in ihrem Aufgabenbereich diese Änderungen mit anderen involvierten Teams zu besprechen, und aktuelle oder zukünftige Maßnahmen zu überlegen. Dieses Interesse wird beispielsweise aus Interviewausschnitt 1 ersichtlich. Zudem müssen sie auch ohne konkrete Änderungen an Aufgaben über deren Fortschritt und Status informiert sein, um die Einhaltung von Planungszielen einschätzen zu können.

B: Ich bevorzuge grundsätzlich den direkten Austausch mit den Kollegen. Natürlich kriegen wir die Emails über das JIRA-Tool. Wenn irgendetwas neu zugewiesen worden ist, oder verändert worden ist. Oder bestimmte Sachen geändert worden ist, oder eine neue Version

eines Dokumentes erstellt worden ist, dann werden wir automatisch vom Tool informiert. Das ist schon sehr nützlich. Aber nichts desto trotz: der Weg zu dem direkten Kollegen wird bevorzugt, wenn irgenetwas anders sein sollte, oder eine neue, nachträglich dokumentierte Anforderung im Tool erfasst worden ist, dann ist das für mich immer so ein Zeichen ein Gespräch zu suchen. Das reicht mir nicht im Tool zu lesen "Ok, Version drei ist erstellt worden", mir ist es wichtig zu wissen die Hintergründe, können wir das im Prozess abfangen, ist das notwendig gewesen, kann man das nicht anders priorisieren dass das vielleicht der Entwicklungsprozess dadurch nicht gestört wird, und, und, und. Weil das Problem das wir grundsätzlich haben ist, dass die Liefertermine fest definiert sind. Und die können wir nicht verschieben. Von daher: alles was mittendrin im Prozess anders sein soll ist immer ein Riesenproblem welches wir abfangen müssen. Und deshalb immer der direkte Kontakt.

Interviewausschnitt 1 [04_TST_TL: 17]: Erkennen von Risiken und Anpassung von Prozessen

5.3.1.2 Projekte

Ein Projekt kann definiert werden als eine übergeordnete Aufgabe oder eine organisatorische Einheit, die verschiedene Aufgaben und Personen zusammenfasst, mit dem Ziel diese Aufgabe umzusetzen. In der untersuchten Firma wurden anstatt des Begriffs „Projekt“ häufig synonym die Begriffe „Meilenstein“ oder „Produktionsabschnitt“ verwendet, was aus dem in Abschnitt 4.2 dargestellten Vorgehensmodell resultiert. Diese Begriffe beschreiben dasselbe formale Konstrukt wie ein Projekt, tragen aber den zusätzlichen Aspekt der zeitlichen oder ressourcenbedingten Beschränkung von Aufgaben und Projekten bereits im Namen.

Analog zu Awareness über Aufgaben wird auch für Projekte Awareness über **Änderungen** am Projekt selbst, oder an Aufgaben eines Projektes benötigt, beispielsweise ob es es neue Aufgaben im Projekt gibt, oder ob es bei einzelnen Aufgaben des Projektes zu Verzögerungen kommt. Damit einhergehend stellt sich die Frage, ob es dadurch **Auswirkungen** auf andere Aufgaben und Aspekte des Projektes gibt. Die für Aufgaben gestellten Fragen nach Ursachen und personellen Aspekten von Änderungen sind auch für Projekte relevant. Ebenso die Frage nach dem **Status** eines Projektes – auch unabhängig von konkreten Änderungen – stellt sich, beispielsweise wieviele der Aufgaben eines Projektes abgeschlossen, und wieviele noch offen sind.

Awareness über Projekte ist für TeamleiterInnen tendentiell wichtiger als für Nicht-TeamleiterInnen, da sie für die Erstellung und Zuteilung von Aufgaben zu ihren MitarbeiterInnen zuständig sind und diese Zuteilung gegebenenfalls auch anpassen müssen. Zusätzlich sind sie für die Kommunikation über den Projektstatus mit der Abteilungsleitung verantwortlich. Interviewausschnitt 2 illustriert dieses „Beischeidwissen“ eines Teamleiters über den Projektstatus mittels JIRA.

B: (klickt) Und das ist mein Hauptfilter, das sind noch die einzelnen dazu. Hier sieht man das im einzelnen auch noch heruntergebrochen, mit den einzelnen Workflow-Schritten in denen sich das gerade befindet momentan. Wenn man sieht, meinerwegen hier bei den Implementierungs-Aufgaben sind so-und-so viele offen, die sind momentan gerade 'in Arbeit'... Das kann man sich anschauen, da kann man auch Statistiken drüber ziehen. Und was-weiss-ich nicht alles. Man sieht einen Gesamt-Meilenstein - "wie sieht das aus?", "wieviel habe ich schon erledigt?", "wieviel ist offen?". Das ist so meine... Hauptinformationsquelle. Um wirklich an Informationen heranzukommen, jetzt zu sagen "wie weit sind wir denn", oder jetzt "wo hakt es".

Interviewausschnitt 2 [17_ENTW_DE_TL:56] Übersicht über Meilenstein in JIRA

5.3.1.3 Termine

Aufgaben und Projekte sind zudem meist einer zeitlichen Beschränkung unterworfen, d.h. es gibt verschiedene Termine zu denen ganze oder Teile von Aufgaben fertiggestellt sein sollen, oder eine Aufgabe oder Projekt muss innerhalb einer gewissen **Dauer** erledigt werden. Dabei sind Termine für ein Projekt oder einen Meilenstein zumeist fest von aussen vorgegeben, d.h. sie werden innerhalb der Firma oder mit einem Kunden im Vorfeld

vereinbart und anschließend wird die Projekt- oder Meilensteinplanung auf diese Termine hin abgestimmt. In Interviewausschnitt 3 wird dieses Vorgehen für die Planung eines Produktionsabschnittes erklärt.

B: Machbarkeit: also der Kunde reicht eine Liste an Themen normalerweise ein die er bearbeitet haben möchte. Der Produktionsabschnitt ist fest terminiert, das heisst die Dauer des Produktionsabschnittes ist fest. Ich habe in diesem Produktionsabschnitt in meinem Team eine bestimmte Kapazität zur Verfügung die ich mir vorher ermittle, weiss wer welche Themen bearbeiten kann in meinem Team. Die Themen werden dann in der sogenannten Woche Null des Produktionsabschnittes den Fachkonzeptern zur Schätzung vorgelegt, die lesen sich dann ein in das entsprechende Thema, kriegen eine Funktionsumfangsbeschreibung (Anm.: im Folgenden auch „FUB“ genannt) vom Kunden meistens, versuchen darauf eine Fachkonzeptschätzung zu erstellen - wie lange brauche ich für das Fachkonzept, wie lange brauche ich für die Testliste dazu. Es gibt von meinem Produktmanagement, da kriege ich dann noch eine Liste wie die Priorisierung der Themen aussieht, und gemäss der Priorisierung und meiner Kapazitäten und den Schätzaufwänden die ich von meinen Leuten bekommen habe, kucke ich halt dann was in diesem Produktionsabschnitt von uns gemacht werden kann. Also wieviel passt rein, und daraus ergibt sich dann die Machbarkeit. Das heisst die Themen die wir auch zeitlich bearbeiten können, wo wir davon ausgehen, kriegen die Machbarkeit "ja". Wenn Themen dabei sind die vielleicht ein bisschen zu gross sind, oder wo die Anforderungen noch so schwammig sind dass man überhaupt nicht weiss wo es da hinlaufen könnte, kann es passieren, dass die erst mal eine Machbarkeit "offen" kriegen. Das heisst da muss man sich noch mal drüber unterhalten. Und Themen die jetzt einfach nicht mehr reinpassen in den Produktionsabschnitt kriegen gleich von vornherein eine Machbarkeit "nein". Das liefere ich meinem Auftraggeber zurück, das ist das Produktmanagement, und das Produktmanagement übernimmt über die Machbarkeit dann die Kommunikation mit dem Kunden.

Interviewausschnitt 3 [02_FK_TL:6] Produktionsplanung für Fachkonzeption

Innerhalb eines Meilensteines werden die Liefertermine für die einzelnen Abteilungen festgelegt, deren Arbeitsergebnisse ja nach dem in Abschnitt 4.2 beschriebenen Vorgehensmodell die Voraussetzungen für die jeweils folgenden Abteilungen darstellen. Neben solchen, im Voraus vereinbarten Terminen, gibt es weitere zeitliche Einteilungen die den Arbeitsrhythmus bestimmen, wie beispielsweise zweiwöchentliche Iterationsplanungen oder regelmässige, zumeist wöchentliche oder zweiwöchentliche Code-Abspaltungen. Neben den fixen, übergeordneten Terminen und den regelmässigen Terminen gibt es noch eine dritte Art von Terminen, nämlich jene von (Unter-)Aufgaben. Deren Planung orientiert sich an den übergeordneten Terminen. Zumeist werden jeder Person eine oder mehrere (Unter-)Aufgaben zur Umsetzung innerhalb eines Produktionszykluses zugewiesen.

In Bezug auf Awareness sind Verschiebungen der fixen, übergeordneten Termine von Interesse, da diese Auswirkungen auf untergeordnete Termine haben. Regelmässige Termine, wie solche für Codeabspaltungen, werden meist nicht zu weit in die Zukunft geplant, sondern das tatsächliche Datum und Uhrzeit dieser Ereignisse wird häufig relativ kurzfristig, meist wenige Tage vor Eintreten, bekannt gegeben. Übergeordnete und regelmässige Termine werden ausserdem meist „top-down“ vorgegeben, d.h. vom Management an die MitarbeiterInnen. Wichtig sind auch Änderungen von Aufwandsschätzungen, also die Dauer für bestimmte Aufgaben. Hier liegt die Abschätzung von Arbeitsaufwänden oder die Kommunikation von Verzögerungen bei einzelnen Aufgaben häufig bei den umsetzenden MitarbeiterInnen selbst, d.h. die Kommunikation erfolgt „bottom-up“.

5.3.1.4 Artefakte

In der untersuchten Firma spielen zwei Arten von Artefakten eine Rolle: Produktionsartefakte und administrative Artefakte. Zur Koordination von Aufgaben werden **administrative Artefakte** in Form von JIRA-Tasks, Fehlertickets in TTS und Projektpläne in Dokumentenform erzeugt und verwendet. Diese Art von Artefakten hat keinen direkten Nutzen für Dritte und besitzt nach Beendigung einer Aufgabe nur noch dokumentarischen Charakter. Während der Arbeit an einem Projekt jedoch unterliegen diese Artefakte

ständigen Anpassungen und Änderungen von innen wie von aussen, d.h. entweder müssen sie selbst angepasst werden, oder eine Anpassung an ihnen löst eine Anpassung bei anderen Artefakten, bzw. weitere Handlungen durch Personen aus. In diesem Sinne stellen Awarenessfunktionalitäten solcher Artefakte und der Umgebung in der sie verwaltet werden einen wichtigen Aspekt für Koordination dar.

Neben rein administrativen Artefakten, gibt es Artefakte, die das Ergebnis einer Aufgabe, bzw. des Produktionsprozesses darstellen. Sie sollen im Folgenden als **Produktionsartefakte** bezeichnet werden. Im untersuchten Feld sind das beispielsweise Fachkonzept-Dokumente, Testlisten oder Sourcecode, die als solches an einen Kunden geliefert werden. Diese Artefakte unterliegen einem Entstehungsprozess, in dessen Dauer sie ständig erweitert und korrigiert werden. Fachkonzepte stellen dabei die Spezifikation und somit die fachliche Basis für den zu programmierenden Sourcecode, sowie für die Testspezifikation der Testabteilung dar. Somit müssen Änderungen im Fachkonzept im Sourcecode und den Testlisten nachgezogen werden. Die EntwicklerInnen und TesterInnen müssen also über Fachkonzeptänderungen Bescheid wissen, genauso müssen FachkonzepterInnen über etwaige technische Beschränkungen durch die ProgrammiererInnen informiert werden. Eine weitere Quelle von (zu kommunizierenden) Anpassungen von Fachkonzept und Sourcecode stellen Fehlertickets durch die Testabteilung oder den Kunden, oder Änderungswünsche durch den Kunden dar.

5.3.1.5 Technische Infrastruktur und Tools

Die Umsetzung von Aufgaben und die Erstellung von Artefakten geschieht mit Hilfe einer technischen Infrastruktur aus Netzwerkverbindungen, PCs und Servern, Netzlaufwerken, lokalen- und onlinebasierten Tools und Software wie Coderepositories und Tools für Continuous Integration wie Cruise Control, die beispielsweise von den EntwicklerInnen verwendet werden. Das Funktionieren dieser Tools an sich ist eine Grundvoraussetzung für die tägliche Arbeit der MitarbeiterInnen. Vor allem Dinge wie die Geschwindigkeit der Netzwerkverbindung, welche sich wiederum auf die Performance von onlinebasierten Tools wie Lotus Notes oder dem Coderepository auswirkt, scheint in der Beobachtung von Bedeutung zu sein. In der Untersuchung zeigt sich, dass dieser „**technische**“ **Status** der Infrastruktur regelmässig, bzw. beim Auftreten von Problemen, zwischen den EntwicklerInnen kommuniziert wird.

Neben der Erreichbarkeit von Tools an sich ist auch der „**inhaltliche Status**“ von Coderepositories und Cruise Control, vor allem für die Entwicklungsabteilung und für das Releasemanagement, von Interesse. Diese beiden Tools werden genutzt, um die verschiedenen Sourcecode-Teile der einzelnen EntwicklerInnen zu integrieren, und geben Rückschluss darüber, ob Sourcecodeteile einzelner EntwicklerInnen für sich compilier- und lauffähig sind und dazu verschiedene Testroutinen bestehen. Darüber hinaus werden diese Tools abteilungsübergreifend genutzt, und geben somit Aufschluss darüber, ob sich Sourcecodeteile oder Funktionalitäten verschiedener Personen und Abteilungen gegenseitig (negativ) beeinflussen. Die teilnehmende Beobachtung und die Interviews zeigten, dass der inhaltliche Status dieser Tools von den damit arbeitenden Personen ständig überwacht wird, und auftauchende Probleme in der Folge häufig kommuniziert werden, sofern sie das eigene Team betreffen. Neben dem inhaltlichen Status dieser eher produktionsorientierten Tools wird auch der inhaltliche Status von **administrativen Tools** wie JIRA und TTS regelmässig betrachtet und kommuniziert.

5.3.1.6 Personen

Awareness über Personen betrifft im wesentlichen drei Aspekte: Der erste Aspekt betrifft die **Verfügbarkeit** oder Erreichbarkeit von Personen am Arbeitsplatz. So geben viele Interviewte an, dass sie vor Tätigung eines Telefonates erst in der Kontaktliste des Instant Messengers nachsehen, ob die gesuchte Person auch am Arbeitsplatz anwesend ist. Bei der teilnehmenden Beobachtung, bei der die untersuchten Personen räumlich sehr nah zueinander waren, wurde im Fall des nicht-Auffindens einer Person häufig auch die Frage gestellt (an die anderen anwesenden KollegInnen), wo eine Person denn sei, oder wann sie wieder zurück sei. Diese tiefergehenden Fragen dienen in erster Linie der zeitlichen Abschätzung, wann eine Person wieder zu Verfügung stünde.

Der zweite Aspekt der beobachtet werden konnte, betrifft die (momentane) **Tätigkeit** einer Person, also beispielsweise woran jemand gerade arbeitet, oder wie einzelne MitarbeiterInnen ausgelastet sind, d.h. wieviele Aufgaben und Tickets sie abzuarbeiten haben. Diese Themen werden regelmässig in Teammeetings besprochen, und die TeamleiterInnen behalten die Menge an zugeteilten Aufgaben für ihre Teammitglieder mittels Statistiken und Übersichtsdarstellungen im Auge. Auch wenn solche Auslastungen primär für die TeamleiterInnen von Interesse sind und viele Befragte in den Interviews angeben, sich allgemein nicht für die Auslastung ihrer KollegInnen zu interessieren, so geben manche doch auch an, sich zumindest für die Auslastung jener KollegInnen zu interessieren, mit denen man in einem Zimmer sitzt.

Der dritte Aspekt von Awareness über Personen beschäftigt sich mit der **Person als Kontext** von anderen Aufgaben und Artefakten, zum Beispiel wer ein bestimmtes Ticket erstellt hatte oder von wem eine spezifische Anforderung kam. Dieser Aspekt deckt sich mit der Fragestellung über personelle Aspekte von Aufgaben, bzw. Änderungen in Abschnitt 5.3.1.1 Diese Fragen werden häufig in der Diskussion über Aufgaben gestellt und helfen im Folgenden beispielsweise bei der Einschätzung von Prioritäten und Dringlichkeiten dieser Aufgaben. So zeigte sich, dass das Eintreffen von Fehlermeldungen von Seiten der Kunden meist eine grössere Aufmerksamkeit der EntwicklerInnen und TeamleiterInnen nach sich zieht. Auch bei Veränderungen im Sourcecode wird die Frage gestellt, durch welche Person eine Modifikation erfolgt war.

In der Untersuchung konnte beobachtet werden, dass sich das Awarenessbedürfnis über Aufgaben und andere Personen abhängig von der **Rolle** der handelnden Personen unterscheidet. Für TeamleiterInnen ist es durch ihren Aufgabenbereich obligatorisch, sich für die Aufgaben ihrer Teammitglieder zu interessieren und den Fortschritt ihrer Aufgaben im Auge zu behalten. Die TeamleiterInnen in der untersuchten Firma verwenden dabei eine Reihe von Mechanismen, um über die Aufgaben und das Vorankommen ihrer MitarbeiterInnen informiert zu sein: Meetings, persönliche Gespräche und Telefonate im Bereich der direkten Kommunikation, Statistiken, Übersichts- und Detailseiten im Umgang mit Fehlertickets und dem Ticketsystem, und diverse Filter im Umgang mit Aufgaben und JIRA. Übersichten, Statistiken und Filter werden dabei für ganze Teams und für einzelne Personen eines Teams angelegt.

Im Gegensatz zu TeamleiterInnen haben EntwicklerInnen und FachkonzepterInnen ohne direkte Verantwortung für andere Personen auch hauptsächlich ihre eigenen Aufgaben im Blick. Für Aufgaben, die anderen Personen zugeordnet sind, interessieren sie sich meist nur „*wenn es mich auch betrifft*“. Zwar geben sie teilweise an, dass sie ab und zu Aufgaben anderer aus Interesse oder aus Neugier betrachten, jedoch würden sie das nur machen, wenn sie auch **Zeit** dafür hätten, d.h. wenn das Stressniveau niedrig ist und somit Zeit lässt, um sich Tasks und Tickets anderer Personen in JIRA und TTS anzusehen.

Die meisten der interviewten EntwicklerInnen aus dem Kompositsteam benutzen das JIRA-Dashboard welches der Teamleiter für die Gruppe als Standarddashboard konfiguriert hatte. Dieses verwendet vor allem Filter, die ihre eigene Person betreffen, d.h. sie sehen jene Aufgaben, die momentan ihnen selbst zur Bearbeitung zugewiesen sind. Aufgaben anderer Personen und Teams sind nicht unmittelbar sichtbar. Im Gegensatz dazu zeigt sich aber bei der Verwendung von TTS, dass sie in dieser Anwendung sehr wohl die Tickets des eigenen Teams, und zum Teil auch von anderen Teams regelmässig betrachten. Dieses Verhalten ist beispielsweise aus Interviewausschnitt 4 und Interviewausschnitt 5 ersichtlich:

B: (...) Also bei dem Ticketsystem finde ich es gut, dass man alle Tickets auf einen Schlag hat. Man kann, man hat alle auf einen Schlag, man kann nach den Namen sortieren, ich kann sie nach dem Datum sortieren, ich kann da so ein bisschen rumsuchen, und ich sehe auch, wenn ich ein bisschen blättere, wieviele Tickets bei **ENTW_DE_1** liegen, oder bei **ENTW_DE_5** liegen, oder wieviele nachgetestet sind. Also das sehe ich alles, ohne dass ich da gross suchen kann. Ich gehe da rein und habe das gleich.

Interviewausschnitt 4 [03_ENTW_DE_2:79] Übersichtlichkeit von TTS, Ansehen der Tickets von TeamkollegInnen

I: Dann eine Frage noch zum Ticketsystem: Wie nützlich findest du das, um Übersicht zu behalten über das Team oder über den Projektablauf?

B: Ziemlich unübersichtlich. Also ich wäre schon froh, wenn wir das in JIRA hätten. Weil es wesentlich leichter zu verwalten ist in JIRA. Man kann über JIRA mehr.. also (unv. jeder) Statistiken ausdrucken, die Filter entsprechend setzen und so. Und das ist mit dem Notes-System was wir da haben ist das halt ziemlich schwierig.

I: Und welche Tickets siehst du dir an im Ticketsystem? Nur deine eigenen, oder..

B: Nein, eigentlich kucke ich mir alle an.

I: Alle? Also von allen Kompositentwicklern?

B: So gut wie, ja.

I: Fachkonzepter auch? Weil da gehen ja auch ab und zu dann Tickets rüber.

B: Die.. da.. Fachkonzept da schaue ich so ein-, zweimal die Woche rüber, ob da überhaupt irgendetwas Neues aufgetreten ist, weil wir kriegen ja nicht unbedingt immer alles gleich mit. Also normalerweise gehen die Tickets ja im Prinzip erstmal an die Entwicklerteams und die geben das an die Fachkonzepter weiter. Aber ab und an bei den Fachkonzeptern schaue ich dann mal ein-, zweimal die Woche rüber ob da noch irgendwelche Tickets herumliegen die demnächst auf uns zukommen könnten.

Interviewausschnitt 5 [12_ENTW_DE_TL_STV_2: 67-74] Ansehen von Tickets anderer Teams

Aus den beiden Interviewausschnitten (Interviewausschnitt 4 und Interviewausschnitt 5) wird aber auch ersichtlich, dass es unterschiedliche Ansichten über die Bedienbarkeit und den Nutzen von TTS und JIRA gibt. Diese betreffen die **Komplexität** und **Usability** der Tools. Während die erste Person (Interviewausschnitt 4) angibt, in TTS im Vergleich zu JIRA schneller und leichter eine Übersicht zu bekommen, bemängelt die zweite Person (Interviewausschnitt 5) die fehlenden weitergehenden Auswertungs- und Verwaltungsmöglichkeiten in TTS.

Während TTS schon jahrelang in der untersuchten Firma verwendet wurde und keine Konfiguration nötig ist um alle erfassten Tickets aller Personen und Teams zu sehen, war JIRA zum Zeitpunkt der Untersuchung erst seit kurzem eingeführt worden. Für JIRA müssen zur Übersicht über alle Teams und Personen eigene Filter angelegt werden. Dazu muss man alle relevanten Task-Typen angeben, welche dazu auch ständig erweitert werden (in TTS gibt es keine unterschiedlichen Task-Typen), und man muss die betreffenden Personen der Teams kennen und angeben, da in JIRA nur einzelne Personen und keine Teams zugewiesen werden (in TTS werden Personen und Teams angegeben).

JIRA erscheint somit wesentlich komplexer als TTS. Dadurch sind einerseits spezifischere Abfragen möglich, andererseits benötigt man aber auch mehr Wissen über das System und seine Konfigurationsmöglichkeiten und braucht somit auch eine längere **Einarbeitungszeit**. In den Interviews zeigte sich, dass sich vor allem die TeamleiterInnen weitaus mehr mit den Konfigurationsmöglichkeiten von JIRA auseinandergesetzt hatten und sicherer im Umgang damit wirken als normale EntwicklerInnen, da sie einerseits schon bei der Konfiguration des Gesamtsystems eingebunden waren, und auch mehr Zeit darauf verwenden, Übersichten und Filter für sich und ihre MitarbeiterInnen zu erstellen.

5.3.1.7 Gruppen

Awareness über Gruppen stellt eine Verallgemeinerung von Awareness über Personen dar und kann somit also analog zu Awareness über (Einzel-)Personen betrachtet werden: Sie betrifft das Gewahrsein über Verfügbarkeit und Auslastung von Gruppen, oder auch das Interesse daran, von welcher Gruppe beispielsweise Änderungen oder Erweiterungen von Artefakten oder Aufgaben kommen.

Generell zeigt sich, dass die Frage nach dem „Team“ in den Interviews häufig einer genaueren Spezifikation bedurfte, da die Befragten sich häufig mehreren Gruppen zugehörig fühlen: Jener Gruppe der sie abteilungstechnisch zugeordnet sind, der Gruppe der sie innerhalb einer Abteilung zugeordnet sind (z.B. DialogentwicklerInnen im Gegensatz zu ProduktentwicklerInnen), mit jenen Personen mit denen sie abteilungsübergreifend an einem Thema arbeiten, oder mit jenen Personen mit denen sie gemeinsam in einem Büro sitzen.

So wie sich einzelne Personen mehreren Gruppen zugehörig fühlen, gibt es umgekehrt das Phänomen, dass manche Fachbereiche von Aussenstehenden eher als abgeschlossene, homogene Gruppe wahrgenommen werden und dass nicht über einzelne Personen aus dem Fachbereich gesprochen wird, sondern immer über den Fachbereich als solches. So zeigt sich in der teilnehmenden Beobachtung, dass die EntwicklerInnen in Österreich meist über das Releasemanagement als Gruppe sprechen und seltener über einzelne Personen dieses Teams.

5.3.1.8 Prozesse

Prozesse regeln und organisieren die Zusammenarbeit zwischen Personen und Gruppen, bzw. die Abfolge von Arbeitsschritten. Prozesse können unterteilt werden nach ihrem Formalisierungsgrad, d.h. in wie weit sie in Tools oder Artefakten abgebildet sind und wie „langlebig“ sie sind.

Formalisierte Prozesse sind Prozesse, bei denen die Zusammenarbeit mittels eines Artefaktes geregelt ist und toolgestützt durchgeführt wird. Dazu zählt beispielsweise die Beantragung von Nachlieferungsaufträgen von Softwareteilen in die Produktionsumgebung mittels des Tools REA. Anträge werden von EntwicklerInnen oder anderen Personen in REA durch Anlegen eines neuen Datenbankeintrages gestellt. Diese werden in der Folge vom Releasemanagement begutachtet und geprüft und entweder stattgegeben, nicht stattgegeben oder mit Anmerkungen an die Entwickler zurückgegeben. Wenn sie stattgegeben werden, können die Entwickler in der Folge mit der ID des REA-Antrages Sourcecode in die betreffenden Coderepositories einspielen, und dies anschliessend im REA-Antrag vermerken. Daraufhin wird der REA-Antrag vom Releasemanagement weiter bearbeitet.

Informelle Prozesse sollen an dieser Stelle definiert werden als Koordinationsereignisse, bei denen sich die beteiligten Personen kurzfristig formieren und mündlich oder per Email eine Vorgehensweise zur Bewältigung einer Aufgabe vereinbaren. Als Beispiel sei hier die dokumentierte Situation in Interviewausschnitt 6 genannt, bei der ein unvorhergesehenes, aber dringendes Problem auftritt, sich die EntwicklerInnen auf eine Vorgehensweise zur Lösung einigen, und diese Vorgehensweise wiederum an die TeamleiterInnen und andere Beteiligte kommunizieren.

5	Situation: Donnerstag Vormittag, „Eincheckdeadline“ war Mittwoch Nachmittag (gestern).
5A	ENTW_OE_TL [geht zu ENTW_OE_1 und ENTW_OE_4 (reden ohnehin gerade miteinander)-> mündlich]: „ ENTW_DE_TL_STV_2 hat gerade angerufen, der ist gerade beim Kunden in Süddeutschland , die haben einen Fehler bei [Funktionalität] gefunden, das funktioniert nicht mehr so wie früher. Sie werden zwei TTs eröffnen und euch zuweisen. Die TT-Nummern sind xxxx und xxxx.. Dann haben die beiden jetzt mal Priorität“
5B	ENTW_OE_1 und ENTW_OE_4 machen sich an die Arbeit (an ENTW_OE_4s Rechner), TT-Beschreibung ist in ihren Augen schlecht, und unklar woher der Fehler kommt. TT-Beschreibung (Screenshots) werden ausgedruckt, TT-Nummern o.Ä. werden noch über Chat von ENTW_OE_4s Rechner an ENTW_OE_1s Rechner geschickt (für Verwendung eines bestimmten Tools das nur auf ENTW_OE_1s Rechner ist?). Anschliessend gemeinsam Arbeiten an ENTW_OE_4 -Rechner. Nach einer Weile versuchen sie ENTW_DE_TL_STV_2 telefonisch zu erreichen, erwischen ihn nicht. Machen weiter mit Fehleranalyse. Nach einer Weile Anruf von ENTW_OE_4 an ENTW_OE_3 (dieser ist gerade in Norddeutschland wegen fachlicher Frage. ENTW_OE_4 erklärt ENTW_OE_3 anhand eines

	Beispiels (dem TT?) die Ausgangslage und fragt wie das verarbeitet werden muss. Nach einer Weile ist Telefonat beendet, ENTW_OE_4s Fazit (zu ENTW_OE_1 , der daneben sitzt): „dieser Fall ist im Produktkern nicht vorgesehen“. ENTW_OE_4 geht rüber zu ENTW_OE_TL und sagt auch dem sein Fazit. Danach kommt er wieder zurück zu ENTW_OE_1 . Sie überlegen und diskutieren weiter welche Werte rauskommen sollten.
5C	ENTW_DE_TL_STV_2 ruft an, ENTW_OE_4 geht ran. ENTW_OE_4 : "Punkt eins: das hat nie funktioniert, Punkt zwei: wird von Produktkern nicht unterstützen, aber wir haben jetzt eine Idee wie man es machen könnte. Aber wir müssen hier das FK anpassen, weil es gibt kein eigenes Kapitel dafür. Es steht zwar was drin, aber das ist total verstreut, sollte geordnet werden, und auch an den Kunden kommuniziert werden." [ENTW_DE_TL_STV_2 spricht...]. ENTW_OE_4 erklärt nochmal den Fehler und wie das zustande kommt. [...diskutieren weiter...]. ENTW_OE_4 -Frage "wo ist ENTW_DE_TL ?", ENTW_OE_4 -Frage "wie sollen wir das einchecken", ENTW_OE_4 -Frage "kannst du ankündigen lassen dass FK geändert werden muss? sonst kann ich das auch machen" "OK, danke"
5D	ENTW_OE_4 und ENTW_OE_1 machen weiter. ENTW_OE_1 (nach einer Weile): "und wie sollen wir jetzt vorgehen?" ENTW_OE_4 : "du machst das..., ich mach das..., das so und so...". ENTW_OE_4 : "ok, ich schick dir das dann", ENTW_OE_1 geht zurück zu seinem Platz.
5E	ENTW_OE_4 verlässt auch seinen platz, trifft ENTW_OE_TL am Gang: ENTW_OE_TL "und? was war?" ENTW_OE_4 erklärt kurz vorgehensweise (FK muss geändert werden). ENTW_OE_TL "und was ENTW_DE_TL_STV_2 dazu?" ENTW_OE_4 : "für den ist das ok" (+ weiteres erklären der Vorgangsweise).

Interviewausschnitt 6 [Teilnehmende Beobachtung: 5-5E] Erstellung und Kommunikation eines informellen Prozesses

Zwischen diesen beiden Formalisierungsgraden gibt es beliebige weitere Ausprägungen. Beispielsweise stellt die Erstellung einer Machbarkeitsliste, wie sie in dem Gesprächsausschnitt in Interviewausschnitt 3 beschrieben ist, einen Prozess dar, der zwar nicht in einem Artefakt formalisiert ist, aber durch die regelmässige Praxis im Arbeitsalltag festgelegt ist.

Awareness über Prozesse betrifft zwei Aspekte: Die „**ablauftechnische**“ und die „**inhaltliche**“ Seite eines Prozesses. Analog zum „technischen“ und inhaltlichen Status bei technischer Infrastruktur (siehe Abschnitt 5.3.1.5 beschreibt die „ablauftechnische“ Seite, wie ein Prozess aussieht, d.h. aus welchen Arbeitsschritten er besteht, welche Personen oder Artefakte beteiligt sind. Awareness über die „ablauftechnische“ Seite von Prozessen wird benötigt, da Prozesse immer wieder (temporären) Änderungen, Anpassungen und Abweichungen unterliegen. Diese müssen unter den Beteiligten und teilweise, zwecks Nachvollziehbarkeit, auch nach Aussen (also beispielsweise an die TeamleiterInnen) kommuniziert werden. Im untersuchten Projekt betreffen solche Anpassungen beispielsweise die Zuständigkeit von Personen für bestimmte Aufgaben, das Abweichen vom üblichen Vorgehen für das Einchecken von Code oder auch das Abkürzen von Teilschritten eines Prozesses. Die Gründe für solche Abweichungen liegen zumeist an personellen, technischen oder terminlichen Gegebenheiten, die Anpassungen erforderlich machen. In der Untersuchung zeigt sich, dass Änderungen an den Prozessen vorwiegend mündlich oder per Email kommuniziert werden, nicht aber mittels der Tools, in denen sie abgebildet sind.

Die „inhaltliche“ Seite von Prozessen beschreibt dagegen, wie weit ein Prozess fortgeschritten ist, bzw. an welcher Stelle eines Prozesses man sich befindet. Vor allem bei formalisierten Prozessen wird Awareness über den inhaltlichen Zustand, wie z.B. Zustandsänderungen von Artefakten, benötigt, um eine effiziente Abfolge der einzelnen Arbeitsschritte zu gewährleisten. Diese Art von Awareness wird in JIRA und TTS durch verschiedene Awarenessmechanismen der Tools, wie Notification durch Emails und Monitoring von Filtern und Übersichtsdarstellungen, bewerkstelligt.

5.3.1.9 Risiken und Probleme

Probleme und unvorhergesehene Ereignisse fachlicher, technischer, personeller oder sonstiger Art stellen ein Risiko für die Einhaltung von Terminen und Plänen dar. Das Erkennen von Risiken wird vor allem in Interviews mit TeamleiterInnen häufig als wichtiges Awarenessbedürfnis erwähnt, was beispielsweise aus Interviewausschnitt 7 ersichtlich wird. Der Interviewausschnitt zeigt auch, dass Risiken, sobald sie als solches erkannt werden, in schriftlicher Form erfasst, und kommuniziert werden. Die angesprochene Risikoliste in Tabellenform wird dabei als Austauschmedium zwischen dem Abteilungsleiter und den TeamleiterInnen der einzelnen Abteilungen verwendet.

B: Wir haben natürlich Rahmenbedingungen in den Terminen drin - das ist ganz klar - das sind zum Beispiel "Feature Complete Version" oder "Testoffensive" oder wann die erste Auslieferung ist. An diesen Terminen.. die sind natürlich fix. Und wenn du dann so und so bis da fertig sein musst, hast du halt nicht so viel Spielraum da jetzt die Termine hin- und her zu schieben. Das muss man da natürlich abstimmen. Da komme ich dann wieder zu dieser Produktionsplanung von **ENTW_DE_AL**. Bei der hängt hinten auch immer so eine Risikoliste mit drin. Wenn man jetzt sagt "oh, ich kann jetzt diesen Termin für dieses Thema nicht einhalten" dann muss man da ein Risiko anmelden. Dann sagt man "Risiko: Ticket-Eingang Produktion verhindert Fertigstellung von Thema XY. Die Wahrscheinlichkeit dass das Risiko eintritt ist so-und-so hoch, wenn das passiert, dann verschiebt sich der Termin um zwei Wochen" meinetwegen. Das meldet man dann an, und wenn der Fall dann eingetreten ist, dann ist man wenigstens darauf vorbereitet dass sowas passiert.

Interviewausschnitt 7 [05_ENTW_DE_TL_STV_1:125] Fixe Termine und Risikoliste

Eine häufige Quelle von Verzögerungen und Problemen sind Unklarheiten in fachlichen Spezifikationen, welche erst während der Umsetzungsphase in der Implementierung offensichtlich werden, und auch eingehende Fehlermeldungen können sich als unterschiedlich zeitaufwendig erweisen – ein Problem, das zum Beispiel aus Interviewausschnitt 8 ersichtlich wird. Die Schwierigkeit dabei ist, dass sich häufig nicht vorab sagen lässt, ob eine auftauchende Unklarheit oder ein neues Fehlerticket eine Unklarheit bleibt und sich schnell lösen lässt, oder ob es sich eben zu einem Problem, bzw. Risiko auswächst, indem es mehr Ressourcen benötigt als geplant oder vorgesehen und sich somit auch auf andere Aufgaben und Personen auswirkt.

Von vornherein kann man sagen "dafür brauche ich ungefähr so-und-so lange", das kannst du auch in JIRA eintragen als Aufwandsschätzung, aber ob du das nachher einhältst, das ist ziemlich schwierig bei diesen Problemen. Bei Tickets. Ich glaube das kennst du aber selber auch, aus dem **ABC**-Bereich. Wenn du sagst "ja, das könnte eigentlich nur eine Kleinigkeit sein", nachher stellt sich aber heraus, das ist doch so gross. Das lässt sich halt sehr schwierig einschätzen. Gerade so Tickets oder so lassen sich immer sehr schwierig einschätzen. Also die meisten zumindest. Bei einigen sagst du "ok, das ist eine Kleinigkeit", aber eigentlich kann sich ja auch herauskristalisieren da brauchst du nachher zwei Wochen für.

Interviewausschnitt 8 [12_ENTW_DE_STV_2: 164] Abschätzung des Aufwandes für Tickets

Das breite Spektrum von möglichen Problemen – diese können in jeglicher Form auftreten, von Krankheiten einzelner Personen, über fachliche Unklarheiten bis hin zu technischen Problemen – erfordert somit auch die Verwendung und Überwachung eines breiten Spektrums an Tools, Medien und Kommunikationsmechanismen. Die richtige Einschätzung des Ausmaßes von Problemen und das Erkennen von Risiken kann zwar durch verschiedene Tools, Artefakte und Mechanismen unterstützt werden, erfordert aber auch die Expertise und Erfahrung der beteiligten Personen, was die Aussagen in Interviewausschnitt 9 und Interviewausschnitt 10 belegen:

I: Und wie genau musst du jetzt zum Beispiel Bescheid wissen über - also jetzt nicht nur über **Österreich**, sondern auch über die Entwickler hier - über.. eben deren Probleme oder woran sie arbeiten?

B: Also sobald diese Probleme ein grösseres Ausmaß annehmen. Also sobald die Probleme

dazu führen, dass sie nicht mehr weiterarbeiten können, dass sie nur noch mit doppeltem Zeitaufwand oder soetwas.. Also natürlich ist nicht jedes kleine Problem, soetwas wie "meine Umgebung baut gerade nicht" oder sonst irgendwas, das wäre ein bisschen banal. Aber da ist ein bisschen Augenmerk oder so auch gefragt. Das kann man dann immer wieder auch (unv. offen kommunizieren), sagen "ja dann lös das Problem einfach, nerv mit damit nicht" - in Anführungsstrichen - sondern, "da mach das einfach, frag nicht dauernd rum und lös es". Aber sobald die halt nicht mehr weiterkommen.. Das halte ich immer so, sobald ein 'Showstopper' in der Gegend herumsteht, muss man Bescheid wissen. Alles andere kriegt man auch in regelmässigen Meetings raus. Dass man so sagt dann.. "das kann auch zwei Tage später passieren", dass man sagt "hier, der Gang hat ein bisschen länger gebraucht, weil wir haben einen neuen Mitarbeiter", oder "der Rechner ist ausgefallen" oder ich weiss nicht was. Das ist aber der... man kann da halt kein festes Maß nehmen. Man kann nicht sagen "immer wenn 'Schlimmigkeit sieben' eintritt oder so, das geht nicht. Das muss man irgendwie mit Augenmaß machen.

Interviewausschnitt 9 [17_ENTW_DE_TL: 24-25] Augenmaß, Showstopper, "Schlimmigkeit 7"

I: Wie oft schaust du ins Ticketsystem?

B: Täglich.

I: Das schon?

B: Ja, das ja. Einfach auch, weil es eine Rolle spielt für alle möglichen Auslieferungen. Und ich erstelle ja auch wöchentlich Berichte zu diesen Tickets, um zu kucken wie liegen die Teams im Rennen, muss man da spezielle Aufmerksamkeit darauf lenken, und spezielle Aktionen planen, dass da nicht etwas aus dem Ruder läuft. Und ich kucke wirklich fast täglich rein glaube ich.

B: Und vor allen Dingen auch, um die Produktion.. den Produktivbetrieb, den Codefreezebetrieb da im Auge zu behalten.

I: Und wie merkt man das dann, ob es aus dem Ruder läuft?

B: Wenn Ticket.. wenn Zahlen offener Tickets kontinuierlich steigen über mehrer Wochen. Wenn es heftige Ausschläge gibt. Also.. das sind so Trenddiagramme, die auch pro Woche ausweisen was dazugekommen ist, wurde gelöst irgendwie.. Und wenn da einmalig etwas anderes ist, das hat noch gar nichts zu sagen. Aber wenn es über ein Monat hinweg oder so eine Tendenz aufweist die dann... Spätestens dann muss man sich mal informieren woran das liegt - das kann auch gute Gründe haben die jetzt keine speziellen Aktionen erfordern, oder Probleme von denen man glaubt dass sie sich in den nächsten Wochen ohnehin erledigen. Oder man muss dann tatsächlich mal speziell etwas planen irgendwie.

Interviewausschnitt 10 [16_ENTW_DE_AL:51-57] TTS, "Aus dem Ruder laufen"

5.3.2 Kontext & Abhängigkeiten

Im vorigen Abschnitt wurden eine Reihe von Inhalten von Awareness – Projekte, Aufgaben, Termine, Tools, Personen und Gruppen, Artefakte, Risiken und Probleme – identifiziert. Diese entspringen einem gemeinsamen Kontext – dem Arbeitsauftrag der Firma – und weisen auch untereinander starke Abhängigkeiten auf, stellen also auch zueinander einen Kontext dar.

Der **Arbeitsauftrag** des Unternehmens und dessen Umsetzung in einer kooperativen Weise stellt den Ursprung und den gemeinsamen Kontext aller zuvor identifizierten Inhalte von Awareness dar: Von den Zielen, Aufgaben und Projekten, über die handelnden Personen und die Zusammensetzung der Gruppen bis zu den eingesetzten Artefakten, Tools und Prozessen sind alle diese Entitäten auf die Erfüllung dieses Arbeitsauftrags ausgerichtet.

Die verschiedenen Inhalte von Awareness können zwar teilweise einzeln und unabhängig voneinander – also als eigenständige Entitäten – betrachtet werden, doch die bisherige

Betrachtung der Inhalte zeigt schon, dass es gewisse **Abhängigkeiten** zwischen den einzelnen Entitäten – Aufgaben, Projekte, Termine, Artefakte, etc. – gibt, die sie zu komplexeren Sachverhalten zusammenfügen. Diese Abhängigkeiten können zeitlicher, logischer, personeller und auch fachlicher Natur sein: Manche Aufgaben und Projekte bauen aufeinander auf, womit eine Aufgabe eine Voraussetzung für eine andere Aufgabe darstellt oder manche Aufgaben können nur von Personen mit einem bestimmten Fachwissen durchgeführt werden. Fachliche Abhängigkeiten werden beispielsweise im beschriebenen Entstehungsprozess von Produktionsartefakten ersichtlich. Abhängigkeiten zwischen den genannten Entitäten existieren zudem nicht nur in eine Richtung, sondern sind durchaus **wechselseitig** vorhanden: Zwar sind einzelne Aufgaben einem Projekt untergeordnet, Projekttermine wiederum richten sich nach übergeordneten Lieferterminen aus, und Verschiebungen dieser Termine bewirken Verschiebungen bei den Aufgaben. Jedoch können Probleme bei einzelnen Aufgaben umgekehrt dazu führen, dass übergeordnete Termine nicht eingehalten werden können und gegebenenfalls verschoben werden müssen.

Als Beispiel für die Abhängigkeit der Inhalte voneinander und der **Komplexität** dieser Verknüpfungen sei die Umsetzung einer beliebigen Kundenanforderung im Rahmen des Produktionsprozesses genannt: Dazu werden eine Reihe von JIRA-Tasks erstellt, die jeweils einer Person einer Abteilung zur Bearbeitung zugeteilt werden. Stellt sich beispielsweise während der Implementierungsphase heraus, dass ein Teil eines Fachkonzeptes nicht ausreichend genau oder fehlerhaft spezifiziert ist, oder dass es technische Unzulänglichkeiten gibt, die eine vorgesehene Umsetzung behindern, wird das zwischen Entwicklung und Fachkonzeption zumeist mündlich oder schriftlich kommuniziert, in manchen Fällen werden auch Fehlertickets erfasst. Um den Status der Umsetzung kundzutun wird möglicherweise der JIRA-Task der Entwicklung auf „warten“ gesetzt. Gegebenenfalls muss in der Folge das Fachkonzept und möglicherweise auch die Testliste geändert werden, neue Dokumentenversionen werden wiederum über die entsprechenden Tasks in JIRA ausgetauscht, teilweise wird auch schon vorab mündlich oder per Email über die geänderten Inhalte informiert. Mit jeder Anpassung müssen die entsprechenden Personen verständigt werden, je nach Grösse und Auswirkungen einer Änderung müssen auch das Produktmanagement, die Team- und AbteilungsleiterInnen informiert sein, besonders, wenn durch Änderungen im Voraus festgesetzte Termine nicht eingehalten werden können. Schließlich müssen die Anpassungen in der Spezifikation wiederum in der Entwicklungs- und der Testabteilung – zeitlich aufeinander abgestimmt – umgesetzt werden. Eine Übersicht dieser Kommunikationswege und Abhängigkeiten wird in Abb. 24 grafisch dargestellt:

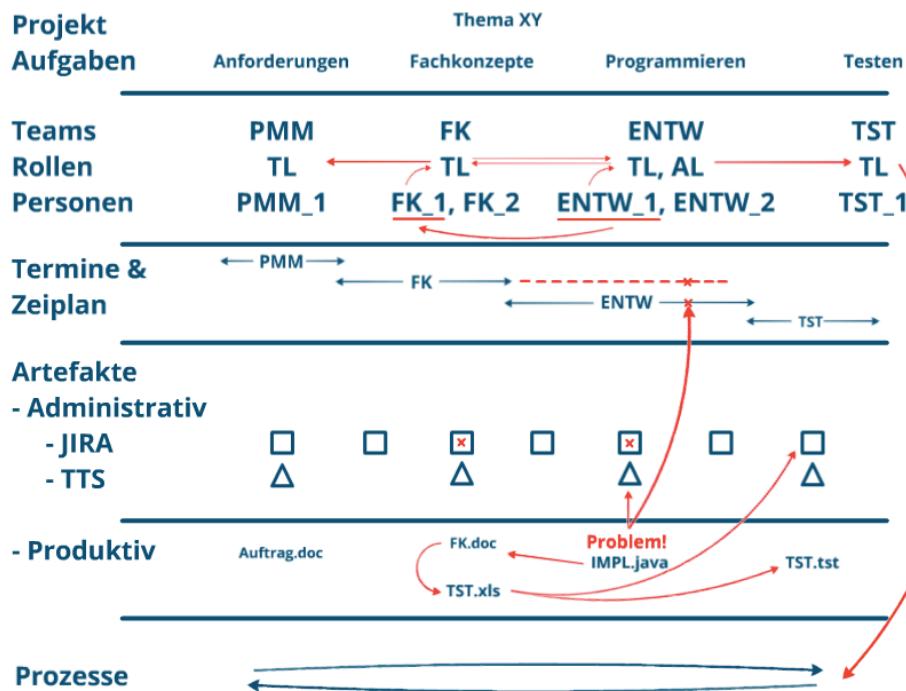


Abb. 24 Kommunikationswege und Abhängigkeiten zwischen Inhalten von Awareness

Das in Abb. 24 dargestellte und skizzierte Beispiel stellt einen Extremfall von Abhängigkeiten dar, der nicht für jedes auftretende Problem und jede Entität gegeben sein muss. Die Abhängigkeiten zwischen den einzelnen Inhalten verfügen vielmehr über eine gewisse Stärke oder Reichweite d.h. ein Ereignis an einer Entität kann lokal begrenzt bleiben und beispielsweise zwischen zwei betroffenen Personen, ohne Einbindung Dritter, geklärt werden. Es kann aber auch weitere Kreise ziehen und Auswirkungen auf eine Reihe anderer Entitäten haben.

5.3.3 Ereignisse & Zustand

Awareness über die beschriebenen Inhalte wird also benötigt, um **Ereignisse** wahrzunehmen die die Entitäten selbst betreffen, oder ihren Kontext (also jene Entitäten mit denen sie verknüpft sind). Solche Ereignisse die in den Interviews und der Beobachtung festgestellt wurden sind zum Beispiel das Hinzukommen neuer Aufgaben, der Abschluss bestehender Aufgaben, der Wechsel von Zuständigkeiten von Personen für Aufgaben, Änderungen an Terminen, oder auch das Eintreten von Terminen, Änderungen an Artefakten, kurzzeitige Anpassungen von Prozessen und Probleme mit der technischen Infrastruktur.

In der Untersuchung zeigt sich zudem, dass nicht nur Ereignisse an konkreten Entitäten ein Interesse an ihnen und ihrem Kontext auslösen, sondern dass der momentane **Zustand** einer Aufgabe oder eines Projektes häufig auch ohne Vorliegen von offensichtlichen, konkreten Ereignissen abgefragt wird. So kann beispielsweise das Nicht-Eintreten eines Ereignisses – wie die Fertigstellung einer Aufgabe – dazu führen, dass der Status der Aufgabe durch eine Person abgefragt wird. Dieses Vorgehen zeigt sich beispielsweise bei der Bearbeitung von dringenden Fehlertickets die zwar in TTS erfasst sind, die TeamleiterInnen aber häufig nicht auf Änderungen des Status der Fehlertickets warten, sondern von sich aus den Fortschritt der Fehlerbehebung bei den EntwicklerInnen erfragen. Das Abfragen von Zuständen kann entweder anlassbezogen erfolgen – das ist beispielsweise der Fall bei einer dringenden Ticketbearbeitung – oder es erfolgt regelmässig als Teil der Routine von TeamleiterInnen oder im Rahmen von regelmässigen Arbeitsabläufen und Prozessen wie täglichen Scrummeetings oder wöchentlichen Teammeetings.

5.3.4 Übersicht & Detail

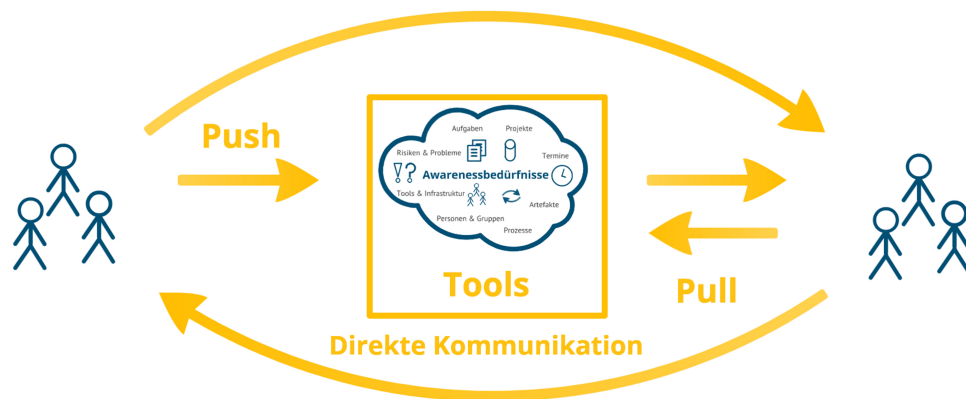
Während Ereignisse und Zustand von einzelnen Entitäten oder ganzen Projekten oftmals nur **überblicksmässig** erfasst und behandelt werden, zeigt sich in der Betrachtung der einzelnen Inhalte, dass das Eintreten von Ereignissen, oder das Vorhandensein bestimmter Zustände häufig auch detailliertere Fragen nach dem **Hintergrund** dieser Ereignisse und Zustände auslösen. Solche Hintergrundinformationen betreffen abhängige Inhalte und werden beispielsweise erfragt, um mögliche Probleme oder Verbesserungspotentiale erkennen zu können (wie in Interviewausschnitt 1 erkennbar), oder bei Zuweisung neuer Aufgaben um beispielsweise Ansprechpersonen zu finden, wie in 5.3.1.1 beschrieben.

Das Bedürfnis nach Hintergrundinformationen ist nicht immer gleich groß. Während **geplante** Zustandsänderungen von Aufgaben (beispielsweise die Zuweisung eines QS-Tasks von der Fachkonzeption an die Entwicklung zu Beginn des Produktionszyklus) im Normalfall bei dem / der betroffenen EntwicklerIn wohl nicht die Frage nach dem „Warum“ aufwerfen, lösen **ungeplante** Änderungen eher die Frage nach den Hintergründen aus. Ebenso wichtig ist der **Zeitpunkt** von Ereignissen: Während fachliche Änderungen am Beginn eines Produktionszyklus als normal erachtet werden, stellen sie mit Näherkommen eines Liefertermines ein Risiko dar.

5.4 Kategorie Awarenessmechanismen

In der Untersuchung lassen sich eine Reihe von Mechanismen beobachten, die eingesetzt werden, um Awareness über die im vorigen Abschnitt identifizierten Inhalte zu erlangen und die dabei gestellten Fragen zu Zustand und Ereignissen, Kontext und Abhängigkeiten, Übersicht und Details der Inhalte zu beantworten. Interviewaussagen wie „Bescheid sagen“, „Erwarte, dass ich informiert werde“ oder „Regelmässig nachschauen“ können den im Theorieteil dargestellten Awarenessmechanismen Pushing und Pulling oder Subscription zugeordnet werden.

Diese Mechanismen, die immer einen Informationsaustausch oder eine Informationsabfrage bedeuten und somit eine Form von Kommunikation darstellen, lassen sich in regelmäßige und ereignisbasierte Kommunikationsereignisse unterteilen. Der Austausch findet entweder in Form direkter Kommunikation, oder indirekt über den Weg eines Tools oder Artefaktes statt. In der Untersuchung zeigte sich, dass häufig ein Mix, bzw. eine Abfolge von direkter und indirekter Kommunikation verwendet wird. Weiters kann die Kommunikation zwischen einzelnen Personen oder auch mehreren Personen oder Gruppen erfolgen und zudem unterschiedlich detailliert sein. Zudem muss unterschieden werden zwischen InformationserzeugerInnen/HalterInnen und InformationsempfängerInnen/Abfragenden. Abb. 25 stellt den Austausch von (Awareness-) Informationen basierend auf Tools, Artefakten und direkter Kommunikation schematisch dar. Im Folgenden werden die Mechanismen Push, Pull, Subscription und periphere Awareness einzeln besprochen.



Awarenessmechanismen

Abb. 25 Austausch von Awarenessinformationen basierend auf Tools, Artefakten und direkter Kommunikation.

5.4.1 Informationspushing – „Bescheid sagen“

Informationspushing umfasst mündliche Kommunikationsweisen wie „Bescheid sagen“, aber auch beispielsweise das Setzen eines Status in einem Tool wie JIRA oder TTS von „In Arbeit“ auf „Realisiert“. Auch das Erfassen anderer Informationen in einem Tool oder Artefakt stellt als Weitergabe von Information eine Art von Pushing dar.

Im Gegensatz zur (mündlichen) direkten Mitteilung von einer Person oder Gruppe an jemand anders, benötigt das Erfassen von Information in einem System weitere Komponenten um Awareness bei der anderen Person auszulösen: Entweder einen **Notification**-Mechanismus wie das automatische Verschicken von Email-Nachrichten über Änderungen, oder die Zielperson muss sich ohne ein bestimmtes Ereignis, beispielsweise durch regelmässiges Kontrollieren (**Monitoring**) von Tools, einen Überblick in dem Tool verschaffen. Im letzteren Fall ist es notwendig, dass Informationssysteme ein Erkennen und Nachverfolgen von Änderungen ermöglichen, indem sie beispielsweise geänderte Daten als solche markieren.

Informationspushing kann **ereignisbasiert** oder auch ohne Vorliegen eines konkreten Ereignisses erfolgen. Diese zweite Form stellt beispielsweise die tägliche Erfassung von Arbeitszeiten, gemeinsam mit einer Angabe über die ausgeübte Tätigkeit dar. Bei dieser Form des Pushings handelt es sich um einen **regelmässigen** Vorgang.

Ereignisbasiertes Informationspushing markiert meist das Überschreiten eines bestimmten Prozessschrittes, den Abschluss einer Aufgabe, die Neuerfassung von Aufgaben, oder auch das Auftreten von Problemen. In diesen Fällen, wo durch das Eintreten eines Ereignisses eine andere Person gefordert wird, sind Awarenessmechanismen besonders wichtig. In der Untersuchung zeigt sich, dass in diesen Fällen die Kommunikation häufig **gedoppelt** wird, also dass nicht bloss ein Eintrag in den Tools erfasst und darauf vertraut wird, dass ein Awarenessmechanismus ausgelöst und von der Zielperson wahrgenommen wird, sondern dass zusätzlich ein mündliches Bescheidgeben erfolgt. Dieses Vorgehen wird besonders bei zeitlich **dringenden** Angelegenheiten praktiziert, wie beispielsweise dem Einlangen neuer Fehlertickets, wie es in Interviewausschnitt 11 dargestellt wird:

5	Situation: Donnerstag Vormittag, „Eincheckdeadline“ war Mittwoch Nachmittag (gestern).
5A	ENTW_OE_TL [geht zu ENTW_OE_1 und ENTW_OE_4 (reden ohnehin gerade miteinander)]: „ ENTW_DE_TL_STV_2 hat gerade angerufen, der ist gerade beim Kunden in Süddeutschland , die haben einen Fehler bei Funktionalität gefunden, das funktioniert nicht mehr so wie früher. Sie werden zwei TTs eröffnen und euch zuweisen. Die TT-Nummern sind xxxx und xxxx.. Dann haben die beiden jetzt mal Priorität“

Interviewausschnitt 11 [Teilnehmende Beobachtung: 5-5a] Dopplung von Kommunikation bei dringenden Fehlertickets

In dem hier angeführten Beispiel werden also Fehlertickets in TTS erfasst, und obwohl TTS zum einen über eine automatische Emailbenachrichtigung bei Zuweisung von Tickets zu Personen verfügt, und zum anderen alle EntwicklerInnen angeben, regelmässig in TTS nachzusehen ob es neue Tickets gäbe, gibt es in diesem Fall zusätzlich eine direkte, mündliche Anweisung dazu. Das geschieht deshalb, weil der Ersteller des Tickets sicherstellen möchte, dass das Ticket auch tatsächlich sofort behandelt wird. Das Benachrichtigungsemail von TTS stellt keine **Garantie** dar, dass das Ticket auch tatsächlich registriert wird, zumal viele der untersuchten Personen auch angeben, automatisch generierte Emails der Tools mittels **Emailfilter** auszusortieren, was wiederum praktiziert wird, da zu viele automatisch generierte Emails von den verschiedenen Tools einlangen. Ebenso ist durch das mehr oder weniger regelmässige Nachsehen im Tool durch die Empfänger nicht sichergestellt, wann oder ob sie ein Fehlerticket registrieren.

Ein weiterer Vorteil eines direkten, (mündlichen) Informationspushings ist zudem, dass der Umfang, bzw. Detailgrad der ausgetauschten Information sofort, d.h. innerhalb des gleichen Kommunikationsereignisses, je nach Bedarf angepasst werden kann. Erfolgt eine Benachrichtigung durch Notification (über einen toolgestützten Mechanismus), welche einen Bedarf nach mehr Hintergrundinformation nach sich zieht, muss diese Information wiederum in einem separaten Pullvorgang (bei einer Person oder in einem Tool) eingeholt werden.

5.4.2 Informationspulling – „Regelmässig nachschauen“

Informationspulling umfasst mündliche Kommunikationsweisen wie „Nachfragen“ oder das Einholen von Information in einem Tool oder Artefakt durch „Nachschauen“. In Abschnitt 2.4 wird dieses tool- oder artefaktbasierte Einholen von Information auf „Polling“ genannt. Beide Arten von Informationspulling können ereignisbasiert oder auch ohne Vorliegen eines konkreten Ereignisses stattfinden. Ereignisbasiertes Einholen von Informationen wird ausgelöst durch Notification-Mechanismen von Tools, beispielsweise automatisch verschickte Emails von JIRA oder TTS bei Statusänderungen von Ticktes oder Tasks. Auch mündlich übertragene Informationen, beispielsweise im Zuge eines Gespräches oder Meetings, können ein Einholen von weiteren, möglicherweise tiefergehenden Informationen auslösen.

Auftretende Veränderungen in Filter-Ansichten von JIRA oder Übersichtsseiten in TTS können ebenfalls Ereignisse darstellen, welche ein Einholen von zusätzlichen Informationen nach sich ziehen. Dies bedingt allerdings, dass die betrachtende Person die Filter-, bzw. Übersichtsseiten, regelmäßig betrachtet, um Unterschiede im Zustand erkennen zu können. Dieses **Monitoring**, welches auf der Veränderungen von Übersichtsdarstellungen basiert, erweist sich in der Untersuchung als ein häufig genutzter Mechanismus, um über den Zustand oder Arbeitsumfang von eigenen oder Teamaufgaben am Laufenden zu bleiben. Interviewausschnitt 12 verdeutlicht dieses Verhalten:

B: Genau. Ja, das ist hier so die Hauptübersicht mit der ich arbeite. Was ist offen, was in Arbeit und was in QS. Beziehungsweise wie viele. Und mit den Zahlen merke ich mir einfach ungefähr "ok, wo stehst du", ist was Neues dazugekommen. Also wenn ich abends reinkomme, dann weiss ich "Hä?" 25 Implementierungstasks hatte ich doch heute Morgen nicht, da ist was dazugekommen. So ungefähr. Und das war es dann auch fast schon. Das ist so die Übersicht für mich, und zum Scrollen... da schaue ich schon nicht mehr.

Interviewausschnitt 12 [01_ENTW_DE_1: 88] Monitoring von Filtern in JIRA

5.4.3 Subscription – „Erwarte, dass ich informiert werde“

Subscription, also ein Mechanismus, bei dem man sich als InteressentIn an einer Entität einträgt, um über Änderungen informiert zu werden, kann nicht nur an Artefakten und Tools stattfinden, sondern auch an Personen. Von den untersuchten Tools wird Subscription meist sogar implizit unterstützt, d.h. man muss sich nicht explizit als InteressentIn eintragen, da man als AutorIn oder BearbeiterIn eines JIRA-Tasks oder TTS-Tickets in der Regel automatisch über Änderungen in Form von Emailnachrichten informiert wird. Bei JIRA besteht die Möglichkeit, sich mittels eines „Überwachen“-Buttons über Änderungen informieren zu lassen, allerdings sagt keine der interviewten Personen aus, diese Funktion zu benutzen. Bei TTS besteht die Möglichkeit einer expliziten Subscription nicht, nur die als AutorIn, BearbeiterIn und TesterIn eingetragenen Personen erhalten Email-Nachrichten. Wie schon im vorigen Abschnitt beschrieben, geben viele der untersuchten Personen an, automatisch generierte Emails, aufgrund deren meist hoher Anzahl, mittels der Filtermöglichkeiten des Emailclients automatisiert aus dem Hauptpostfach zu löschen, bzw. in separate Ordner zu sortieren.

Mit Personen funktioniert Subscription auf explizite und auf implizite Weise: Explizit, indem man eine Person bittet, beispielsweise über die Fertigstellung eines Artefaktes informiert zu werden, implizit, indem über die Rollen, die die einzelnen Personen einnehmen, auch gewisse Informationsketten inhärent vorgegeben sind. Interviewausschnitt 13 soll diesen Gedanken verdeutlichen: Der Produktmanager *erwartet* vom Teamleiter der Fachkonzeption über Verzögerungen im Projekt benachrichtigt zu werden.

Aber derzeit ist es eben so, dass dafür verantwortlich ist der Teamleiter der gesagt hat am Anfang "ich mache das". Und der ist auch dafür zuständig dass die Planung im Prinzip bei ihm so funktioniert. Er gibt ja am Anfang eine Machbarkeit ab, und muss auch entsprechende Risiken melden und so weiter wenn das nicht klappt. Oder wir müssen dann entsprechend sagen "ok, wir sind in Zeitverzug, der Aufwand hier erhöht sich, dafür kippt hinten das andere Thema runter" - das muss dann natürlich schon mit uns abstimmen. Aber es ist nicht so, dass es der einzelne Konzepter machen muss. Das läuft immer eigentlich über die Teamleiter.

I: Ok. Und hast du zum Beispiel den Teamleiter in deiner Sametime- Kontaktliste? **FK_TL?**

B: Den habe ich da mit drin, ja, klar. Aber ich persönlich bevorzuge da so mehr das persönliche Gespräch. Dass ich also schon entweder nachfrage "wie ist der Stand der Dinge, seid ihr noch d'accord?", oder aber wenn er merkt dass es nicht hinhaut, dass ich schon erwarte dass er mich unverzüglich benachrichtigt. Weil wir dem Kunden gegenüber natürlich auch in der Pflicht sind und sagen müssen "pass auf, es gibt da eine Verschiebung". Oder wir kommen hinten raus, oder wie auch immer.

Interviewausschnitt 13 [08_PMM_1: 34-36] "Subscription" an Personen durch Aufgabenverteilung

5.4.4 Periphäre Awareness

Periphäre Awarenessmechanismen in Tools sind eng gekoppelt mit Notification- und Subscriptionmechanismen. Zwar zeigt sich in der Untersuchung, dass viele der automatisch verschickten Emailnachrichten aussortiert werden, diese landen aber nicht im meist

geöffneten Hauptpostfach des Email-Clients, sondern in dafür konfigurierten Nebenordnern. Die Emails werden großteils zwar auch nicht gelesen, aber die fettmarkierte Zahl neben dem Ordnersymbol des Email-Clients, welche auf ungelesene Nachrichten hinweist, macht Aktivitäten in dem verschickenden System ersichtlich. Auch das Erkennen von Änderungen in JIRA-Filtern kann eine Form von periphärer Awareness darstellen.

Neben dieser toolvermittelten Form von periphärer Awareness, die hauptsächlich eigene Aufgaben betrifft, gibt es im direkten Kontakt von KollegInnen auch das Phänomen des periphären Zuhörens: Dabei unterhalten sich zwei KollegInnen über ein Problem und eine dritte, danebensitzende Person, die periphär zuhört, aber an diesem Problem eigentlich nicht direkt beteiligt ist, schaltet sich ein, und gibt einen Lösungsvorschlag, oder macht die beiden Ersten darauf aufmerksam, dass sie noch etwas in ihren Überlegungen vergessen hätten. Diese Situation wird in der teilnehmenden Beobachtung (siehe Interviewausschnitt 14) festgehalten:

41	ENTW_OE_1 zu ENTW_OE_4 : "Du, wir müssen noch REAs für 3.16 und 3.018 stellen." ENTW_OE_4 :"hab ich schon gemacht" ENTW_OE_1 "super" ENTW_OE_3 schaltet sich dazwischen:"aber für xxx müsst ihr auch noch stellen!" ENTW_OE_4 und ENTW_OE_1 :"achja, stimmt!". Danach machen sich ENTW_OE_1 und ENTW_OE_4 (jeder an seinem Recher) an die Arbeit Codes in die verschiedenen Branches zu mergen, fragen dazu wechselseitig ob bestimmte Code-Fragmente schon in bestimmte Branches gemerged wurden, schicken sich REA-Nummern via Chat zu, etc. ENTW_OE_TL kommt auch kurz mal dazu, fragt?/gibt Anweisung? zu einer bestimmten Branch-version und geht gleich wieder. Zeitrahmen: REAs müssen bis 12:00 beantragt/nachgeliefert sein --> nur 20 min Zeit!
42	ENTW_OE_4 und ENTW_OE_1 diskutieren über FK-Anforderung. ENTW_OE_3 schaltet sich ein, da er die Diskussion periphär mithört, und korrigiert sie in einer Aussage.
43	ENTW_OE_4 und ENTW_OE_1 diskutieren über FK-Anforderung. Sie telefonieren mit FK_2 , und besprechen wie es genau umgesetzt gehört. Eventuell handelt es sich dabei um eine Änderungsanforderung. ENTW_OE_3 sitzt gegenüber an seinem Platz und diskutiert ebenfalls mit, bzw. schaltet sich ab und zu ein. ENTW_OE_3 schickt einen Auszug aus seinem Konzept an FK_2 mittels Chat, und gibt ENTW_OE_4 und ENTW_OE_1 Bescheid dass er ihm das geschickt hat. Sie kommen zu dem Schluss dass eine Änderung im Code erforderlich ist, ENTW_OE_4 sagt FK_2 dass sie noch "ein, zwei" Sachen umbauen müssen, und dass er ihm am nächsten Tag eine Liste mit den Änderungen schicken werde, damit dieser das ins FK einbauen kann, bzw. davon Kenntniss hat. Sie machen auch einen Screenshot den sie für die Liste mit Änderungen mitschicken.

Interviewausschnitt 14 [Teilnehmende Beobachtung: 41-43] Periphäre Awareness

Diese Art von periphärem Zuhören, die auch eine Art von periphärer Awareness darstellt, erweist sich als durchaus wertvoll für die Problemlösung, funktioniert aber scheinbar nur bei Vorhandensein von direktem Kontakt und räumlicher Nähe. Zumindest konnte diese Art von periphärer Awareness, die ein Aktiv-werden und zu-einer-Lösung-beitragen bei Dritten bewirkt, in der Arbeit mit den untersuchten Informationssystemen nicht beobachtet, bzw. in den Interviews nicht erfragt werden. Zum einen ist hierfür vielleicht die Methode der teilnehmenden Beobachtung nicht geeignet, um eine solche toolvermittelte, periphäre Awareness zu erfassen, zum anderen wurde auch in den Interviews nicht gezielt danach gefragt.

5.5 *Kategorie Toolverwendung*

In den vorigen Abschnitten wurde gezeigt, welche Awarenessbedürfnisse in der täglichen Zusammenarbeit von EntwicklerInnen, FachkonzepterInnen, TesterInnen, TeamleiterInnen und anderen, an der Entwicklung beteiligten Personen bestehen (5.3) und welche Awarenessmechanismen zur Anwendung kommen (5.4). An dieser Stelle soll nun zuerst gezeigt werden, wie JIRA und andere Tools für die individuelle, teaminterne und teamübergreifende Arbeitspraxis genutzt werden. Danach wird gezeigt, wie

Awarenessbedürfnisse und Awarenessmechanismen in JIRA und anderen zu Verfügung stehenden Tools umgesetzt werden und welche Herausforderungen bei der Umsetzung von Awareness mittels solcher Informationssysteme zu beachten sind.

5.5.1 Unterschiedliche Nutzung von JIRA

Wie in Abschnitt 4.3 dargestellt, wird für die Erledigung der täglichen Arbeitsaufgaben eine Reihe von Tools verwendet, diese reichen von allgemeinen Kommunikationstools wie Email und Sametime über klassische Officeanwendungen bis hin zu spezialisierten Datenbankanwendungen für die Abwicklung von Nachlieferungsanträgen in REA. Die vorhandenen Tools bestehen aus „klassischen“, intranetbasierten Anwendungen auf Basis von Lotus Notes hin zum Enterprise 2.0-Tool JIRA. Die grundlegenden Awarenessmechanismen von JIRA, Lotus Notes und darauf basierender Plugins wurden in Abschnitt 4.3 dargestellt. In diesem Abschnitt soll nun dargestellt werden, dass die verschiedenen Teams und Abteilungen der Firma sehr unterschiedliche Arbeitsmethoden haben und dementsprechend das untersuchte Enterprise 2.0-Tool JIRA auf sehr unterschiedliche Art verwenden.

In der Untersuchung wurden Personen aus drei verschiedenen organisierten Teams, bzw. Projekten befragt. Es zeigt sich, dass die Teams sehr unterschiedliche Arbeits- und Awarenessbedürfnisse haben, und dass sie die in der Firma vorhandenen Tools dementsprechend sehr unterschiedlich verwenden und konfigurieren. Demnach wird JIRA nicht nur für das Erstellen, Verteilen und Controlling von einzelnen Aufgaben („Taskmanagement“) verwendet, sondern es wird auch für die Abbildung und Durchführung von Prozessen („Workflowmanagement“) und die Verwaltung und den Austausch von Dokumenten („Dokumentenmanagement“) verwendet. Taskmanagement ist der eigentliche Zweck, für den JIRA ausgelegt ist. Workflowmanagement erlaubt durch Adaptierung und Anpassung von Aufgabenattributen die Implementierung von Firmenprozessen. Dokumentenmanagement wird durch verschiedene ETP 2.0 Features wie Links und Suche ermöglicht. Allerdings zeigt sich auch, dass mit steigender Komplexität und Umfang der abgebildeten Inhalte, wie es bei einer teamübergreifenden Nutzung der Fall ist, die Nutzung und Administration des Tools schwieriger wird.

5.5.1.1 Die untersuchten Teams

Die meisten der befragten Personen sind Mitglieder jener Abteilungen, die am hauptsächlich untersuchten Kompositprojekt arbeiten, bei dem mehr als zwanzig Personen mitwirken. Die Teamstruktur und der Entwicklungsprozess des Kompositprojektes folgt jener in Abschnitt 4.2 beschriebenen Aufteilung, d.h. die Beauftragung, Konzeption, Entwicklung und Test von Themen sind auf mehrere, fachlich und räumlich getrennte Abteilung aufgeteilt und werden teamübergreifend abgewickelt. JIRA dient sowohl zur teaminternen Aufgabenplanung- und Übersicht, wie auch zur teamübergreifenden Bearbeitung von Aufgaben. Das Tool wird ausserdem zum Austausch von aufgabenbezogenen Fachkonzepten und QS-Dokumenten verwendet. In JIRA sind mehrere Projekte angelegt, die Projekte sind so konfiguriert, dass nur die TeamleiterInnen und deren Stellvertreter neue Tasks anlegen können.

Das zweite untersuchte Team ist Releasemanagement. Dieses Team besteht aus weniger als zehn Personen, die räumlich auf wenige angrenzende Büros aufgeteilt sind. Die MitarbeiterInnen kümmern sich hauptsächlich um den Betrieb, die Wartung und Weiterentwicklung der hauseigenen Datenbank-, Server- und Entwicklungsinfrastruktur, sowie um die Entwicklung von firmenintern genutzten Tools wie zum Beispiel Lotus Notes Plugins. Entwicklungsaufträge kommen dabei in erster Linie aus dem eigenen Team, bzw. werden diese firmenintern beauftragt. Konzeption, Entwicklung und Test von Neu- und Weiterentwicklungen werden größtenteils teamintern abgewickelt. Gleichsam wird auch JIRA hauptsächlich teamintern verwendet, es sind mehrere Projekte, jeweils eines für jedes zu entwickelnde oder betreuende Tool, angelegt. Die Mitarbeiter können selbst neue Tasks in JIRA anlegen.

Das dritte Team, mit dem Interviews geführt wurden, kümmert sich um die Entwicklung der provisions- und partnerspezifischen Komponenten der Versicherungssoftware und umfasst ca. 15 Personen. Im Gegensatz zum Kompositsteam wird im Provisionsprojekt nicht nach einem Wasserfallmodell gearbeitet, sondern nach einem agilen Softwareentwicklungsmodell mit zweiwöchigen Sprintzyklen (siehe Interviewausschnitt 15). FachkonzepterInnen und EntwicklerInnen die an denselben Themen arbeiten, sitzen teilweise in denselben Zimmern. Im Provisionsprojekt wird nicht mit JIRA gearbeitet, stattdessen werden Emails und Netzlaufwerke für den Austausch von Testnachweisen und Spezifikationen verwendet.

5.5.1.2 Provision – Agiles Vorgehensmodell

Im Vergleich zwischen dem Kompositsteam und dem Provisionsteam zeigt sich, dass das angewendete Vorgehensmodell erhebliche Auswirkungen auf die Awarenessbedürfnisse der MitarbeiterInnen hat: Die Person des Provisionsteams in Interviewausschnitt 15 gibt an, sehr genau wissen zu müssen, woran die ihr vorgelagerte Person gerade arbeitet, aus dem Grund, als deren Arbeitsfragment, das selbst noch nicht abgeschlossen ist, die Grundlage für das eigene, „on-the-fly“ entwickelte Arbeitsfragment darstellt:

I: Weisst du zum Beispiel Bescheid was die Entwickler, woran die gerade dran sind? Oder spielt das für dich eine Rolle?

B: Ja. Ja, in der Regel weiss ich das schon. Spielt auch eine ganz bedeutende Rolle, weil in dem Projekt "**Provisionsprojekt**" das wir jetzt gerade machen - also das Provisionsprojekt für die Außendienstabrechnung - da ist ganz wichtig dass ich es weiß welche Features als nächstes fertiggestellt werden, weil dazu gibt es keine separaten Testlisten, sondern die Tests die definiere ich aufgrund von Testnachweisen die erstellt worden sind. Oder von erkannten Problemen aus den Konzepten - sofern welche vorhanden sind - und da ist es ganz wichtig dass ich das ganz eng dran bin an der Entwicklung, und genau weiss welche Features wann fertig sind, damit ich auch mein Zeitfenster zum Testen habe.

B: Wir haben alle zwei Wochen ein Sprint-Review. Da stellen die Entwickler ihre Fortschritte - das was sie fertig entwickelt haben - stellen die vor, und das ist auch der maßgebliche Punkt, wo.. Ja.. In einem Sprint sind halt gewisse Themen zugeordnet an den Entwicklern, die bis dahin zu schaffen sind. Da wird das einmal vorgestellt, und für mich heisst das, wenn es dann voll-umfänglich OK ist, aus Entwicklersicht, dass ich dann erst mit meinen Tests anfangen kann. Und das findet alle zwei Wochen..

I: .. in einem Besprechungsraum..

B: Ganz genau, in einem Besprechungsraum, wo dann alle Entwickler die dem Projekt zugeordnet sind dabei sind. Auch Produktmanagement, der Projektleiter ist mit dabei, und natürlich auch die Konzeption - oder der Fachkonzepter der da mitgewirkt hat.

Interviewausschnitt 15 [14_TST_3: 23-33] Agiles Vorgehen im Provisionsteam

Im Gegensatz dazu ist im Wasserfallmodell vorgesehen, dass Arbeitsergebnisse (Produktionsartefakte) bei der Übergabe von einem Team an das nächste (also beispielsweise bei der Übergabe des Fachkonzeptes von der Fachkonzeption an die Entwicklung oder bei der Freigabe von Sourcecode von der Entwicklung an die Testabteilung) bereits größtenteils abgeschlossen sind. Aus diesem Sachverhalt ergibt sich ein scheinbar geringeres Awarenessbedürfnis über die Tätigkeiten anderer Personen, wie aus Interviewausschnitt 16 ersichtlich wird:

I: Spielt das für dich eine Rolle, woran deine Kollegen im Projekt gerade arbeiten. Also wenn du jetzt - also nicht nur Kollegen wie **ENW_DE_2**, die mit dir im Büro sitzt - sondern jetzt auch Fachkonzepter oder Tester oder so, ist das für dich wichtig, dass du weisst woran die gerade arbeiten? Also wenn du zum Beispiel ein Fachkonzept bekommst, zum Beispiel von **FK_1**, ist das für dich von Bedeutung ob sie gerade das eine macht oder das andere?

B: Eigentlich nicht, weil ich von dem nicht direkt abhängig bin. Also wenn ich anfangen zu arbeiten, dann ist deren Arbeit eigentlich schon passiert, grösstenteils, und dann gibt es

bloss die Fälle dass sich irgendwann herausstellt, technisch ist etwas nicht so umsetzbar, oder nicht so einfach, oder unverständlich. Und dann ist es meistens so, dass man dieses Problem direkt zu dem hingibt, und einfach wartet und Druck macht, bis die Lösung kommt. Das sind immer so Kleinigkeiten. Genau. Das funktioniert dann meistens auch über diese Ticketsystem. Also wenn es ein Problem gibt, dann gibt es dann zu 90 Prozent in Form eines Tickets. Also ausser es sind wirklich Unklarheiten. Aber meistens ist es irgendwie so etwas wie "der Kunde will dann irgendetwas anders als es beschrieben ist". Und das sind alles so Klärfälle die gehen wirklich zur Konzeption, und das ist sehr oft in Form von einem Ticket. Und genauso ist es zum Testteam rüber, da ist es genauso. Wenn ich ein Ticket realisiere, dann ist es das einzige was ich über deren Arbeit wissen will, der Punkt wo ich sage "ich habe es gefixt, es ist testbereit, es gibt Zeitdruck, bitte mach doch mein Ticket zu. Bitte teste das Ticket nach." Das sind so die einzigen beiden Punkte, wo ich interessiert an der Arbeit anderer bin. Auf welchem Punkt die stehen.

Interviewausschnitt 16 [01_ENTW_DE_1:40-41] Awareness über KollegInnen

Interviewausschnitt 16 macht aber auch deutlich, dass im Kompositteam, im Fall von Anpassungen, diese häufig nicht mehr in JIRA abgebildet werden, sondern in TTS in Form von Tickets, was bedeutet, dass somit auch die für das Thema relevanten Informationen auf zwei verschiedene Tools (respektive Informationsplattformen) aufgeteilt werden. Wenn die entsprechenden Artefakte in den beiden Tools nicht verlinkt, oder auf andere Weise nachvollziehbar gemacht werden, geht somit aber auch der Überblick über den Zusammenhang zwischen Aufgabe und Problem – zumindest für Aussenstehende – verloren. Dasselbe Problem wird durch die doppelte Erfassung von Aufgaben in JIRA einerseits (durch das EntwicklerInnenteam) und einer tabellarischen Übersicht andererseits (durch die Abteilungsleitung) erzeugt (siehe Interviewausschnitt 17).

B: (...) Da komme ich dann wieder zu dieser Produktionsplanung von **ENTW_DE_AL**. Bei der hängt hinten auch immer so eine Risikoliste mit drin. Wenn man jetzt sagt "oh, ich kann jetzt diesen Termin für dieses Thema nicht einhalten" dann muss man da ein Risiko anmelden. Dann sagt man "Risiko: Ticket-Eingang Produktion verhindert Fertigstellung von Thema XY. Die Wahrscheinlichkeit dass das Risiko eintritt ist so-und-so hoch, wenn das passiert, dann verschiebt sich der Termin um zwei Wochen" meinetwegen. Das meldet man dann an, und wenn der Fall dann eingetreten ist, dann ist man wenigstens darauf vorbereitet dass sowas passiert. Dann wird der Termin diesen (unv. Führer) in den Excel-Listen sage ich mal entsprechend angepasst. Der Termin der JIRA-Task wird nicht immer angepasst. Es kann auch sein, dass die einfach mal da liegen. Und dann hast du eben diese Diskrepanz zwischen Excel-Liste, JIRA, .. ja, "was stimmt jetzt?"/... Solange es um Termine geht ist es ok, aber sobald du irgendwelche Statusabweichungen dann dazwischen hast, das ist dann schon ein bisschen komisch. Da wird es schwierig dann das synchron zu halten, wenn du jetzt im JIRA sagst "ich habe die Aufgabe schon abgeschlossen", in der Excel-Liste ziehst du den Status dann nach, sagt "ah, ja, hier da, fertig", irgendwann fällt dem Entwickler ein "ach nein, ich habe was vergessen", macht sie wieder auf, dann ist die da wieder offen, und der Status, dann hat das wieder keiner mitgekriegt... Da gibt es schon diverse Organisationsschwierigkeiten manchmal. Das ist schon echt grausam teilweise.

Interviewausschnitt 17 [05_ENTW_DE_TL_STV_1:125] Doppelte Pflege von Terminen

5.5.1.3 Releasemanagement - Teaminterne Nutzung

Im Vergleich der JIRA-Nutzung von Releasemanagement und Kompositteam zeigt sich der Einfluss von Grösse und Struktur eines Teams auf die Nutzungsmöglichkeiten- und Notwendigkeiten von Systemen wie JIRA. Das Releasemanagementteam ist im Gegensatz zum Kompositteam kleiner, Aufgaben müssen zwischen weniger Personen abgestimmt werden und die Personen sind auch räumlich näher. Die Softwareentwicklung ist nicht nach einem Meilensteinzyklus gegliedert. Die Teammitglieder können Tasks selbst anlegen, diese dienen zudem weniger der Koordination zwischen verschiedenen Personen, als mehr der Aufgabenverwaltung für Einzelne. Durch alle diese Faktoren ist scheinbar weniger (toolvermittelte) Awareness notwendig und die angebotenen Awarenessmechanismen bei Änderungen der Tasks stellen eher einen zusätzlichen Informationskanal dar. Diese Aspekte werden im Interview mit einer Person aus dem Releasemanagement in Interviewausschnitt 18 evident:

B: Und das mache ich auch manchmal, das habe ich zum Beispiel gemacht.. (sucht) mal kucken.. es gibt ja.. bei 'SÄM' müsste das sein... genau. 'SÄM-518', das war so eine Geschichte von den DB-Adminstratoren die sie haben wollten für SÄM. Und da habe ich, da ich das nicht gleich bearbeitet habe, habe ich - weil viele da gesagt haben 'ok, das ist ein etwas komplizierteres Thema, da müssen wir uns ein bisschen Gedanken darüber machen' - habe ich dafür ein Ticket aufgemacht. Das aber auch gleich mir zugewiesen ist, gehe ich mal davon aus, genau.

B: Und das schöne daran ist natürlich auch, wenn ich 'Meldung durch' eintrage, weiss ich, dass **RMM_DB_2** dann eine Email kriegt wenn ich es schliesse. Oder wenn ich etwas kommentiere. So dass sie dann Bescheid weiss. Wobei ihr das eh immer dann nochmal sagen würde. Da sie ja praktisch nur einen Raum weiter ist, ist das ja auch immer ziemlich unkompliziert. Und so mache ich das natürlich bei.. auch wenn **RMM_TL** irgendetwas haben möchte, dann..

I: Ok... Diesen Aktivitätsstrom da jetzt noch mal zum Beispiel. Also du siehst da schon immer rein, und... ich weiss nicht...

B: Ich kucke mir das schon an was passiert, das ist auch immer ganz interessant. Sind natürlich in diesem Fall selten Sachen von denen ich NICHT weiss. Aber bei der CSM-Gui, da macht **RMM_4** manchmal einfach so Sachen, und da kucke ich, und dann sehe ich das. Das ist dann natürlich ganz interessant. Es gibt natürlich auch Sachen die er da nicht einträgt, und davon weiss ich dann natürlich nicht. Also.. es ist.. man weiss nicht.. Es ist kein vollständiges Abbild der Änderungen die es gibt. Darüber muss man sich dann eben auch bewusst sein. Aber das ist eh so mehr... Na sagen wir mal es wäre nicht so weiter dramatisch, wenn ich das nicht sehen würde, und nicht wissen würde. Das ist mehr so 'kucken aus Interesse'.

Interviewausschnitt 18 [10_RMM_1: 69-74] JIRA als zusätzlicher Kommunikationskanal in kleinen Teams.

5.5.1.4 Komposit – Teamübergreifende Nutzung

Im Gegensatz zum Releasemanagement dient JIRA den für den Kompositbereich zuständigen Abteilungen nicht nur zur individuellen Aufgabenverwaltung, bzw. zur Verwaltung von Aufgaben eines einzigen Teams, sondern zur abteilungsübergreifenden Zusammenarbeit. Diese Zusammenarbeit besteht großteils aus dem Austausch und der Überprüfung von Produktionsartefakten zwischen den Abteilungen. Durch die Überprüfung wird ein **Qualitätssicherungsprozess** in der Zusammenarbeit implementiert, der vorsieht, dass der Arbeitsoutput der Fachkonzeption, also Schätzraster, Fachkonzepte und Testlisten, vor Beginn der weiteren Umsetzung von den anderen Abteilungen auf Korrektheit und Vollständigkeit geprüft wird.

Zur Abbildung dieser teamübergreifenden Zusammenarbeit in JIRA wurde ein eigenes Projekt angelegt. Dieses enthält jeweils eine Aufgabe für jedes vom Kunden beauftragte Thema. Der Aufgabentitel stammt wiederum aus einem Verwaltungstool des Produktmanagements und ist identisch mit dem später dazu erstellten Fachkonzept. Die Aufgabentitel bestehen aus einer generischen Nummer (z.B. „2703.3“), einer Unterscheidung in Produkt und Bestand (z.B. „PD_KO_FU“ für Produkt, „KO_FU“ für Bestand), dem Spartenamen sowie einem Titel, der einer groben Beschreibung des Inhalts entspricht (z.B. „TÄ/nTÄ“ oder „Umsetzung VWB“). Dazu werden standardisierte Unteraufgaben angelegt, die denselben Namen wie die Hauptaufgabe tragen. Durch eigens konfigurierte JIRA-Aufgabentypen kann nach dem Inhalt der Aufgabe und den beteiligten Teams differenziert werden: Eine farbliche Hinterlegung des Aufgabentyps legt die Zuordnung zu einem oder mehreren Teams fest (hier: Rosa für Produktmanagement, blau für Fachkonzeption, gelb für Entwicklung, grün für Testabteilung), der Aufgabentyp legt den Inhalt der Aufgabe fest (hier z.B.: „FP – Fachkonzept PRM Abnahme“). In Abb. 26 und Abb. 27 ist diese Strukturierung ersichtlich. Man sieht auch, dass die Fachkonzeption (blauer Farbcode) an jedem Task beteiligt ist.

Wenn die Fachkonzeption ein Fachkonzept abgeschlossen hat, wird dieses in den entsprechenden, bereits angelegten und der Fachkonzeption selbst zugeordneten, JIRA-

Task hochgeladen. Ausserdem wird ein eigenes, sogenanntes „QS-Dokument“ in den entsprechenden QS-Task („Qualitätssicherungs-Task“), welcher Fachkonzeption und Entwicklung, bzw. Testabteilung zugeordnet ist, hochgeladen. Dabei handelt es sich um ein Formular, welches eine Tabelle mit Fragen und Antworten für die verschiedenen Fachkonzeptversionen bereithält. Bei der erstmaligen Erstellung von QS-Tasks werden diese den jeweiligen TeamleiterInnen der Test- und Entwicklungsabteilung zugewiesen, welche die Tasks wiederum ihren MitarbeiterInnen zuweisen. Haben diese die QS abgeschlossen, wird der Taskstatus wiederum auf „erledigt“ oder „nachzubessern“ gesetzt, und der/ dem FachkonzepterIn zugewiesen.

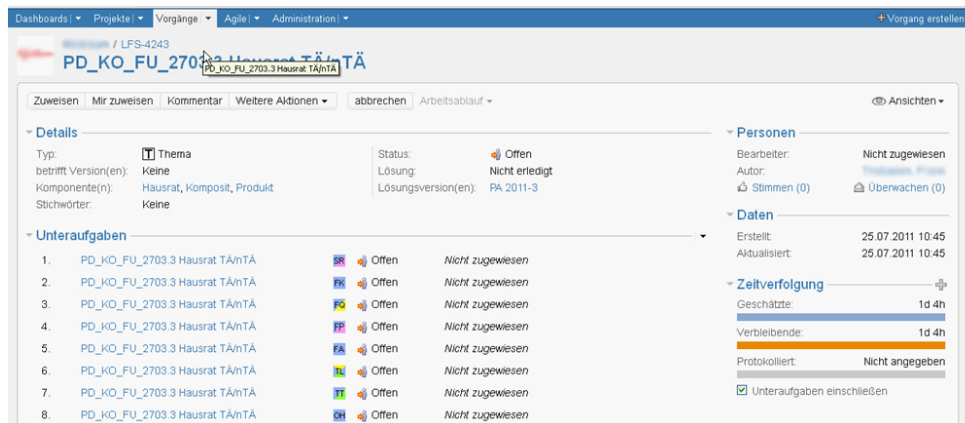


Abb. 26 Aufgabe in JIRA mit standardisierten Unteraufgaben

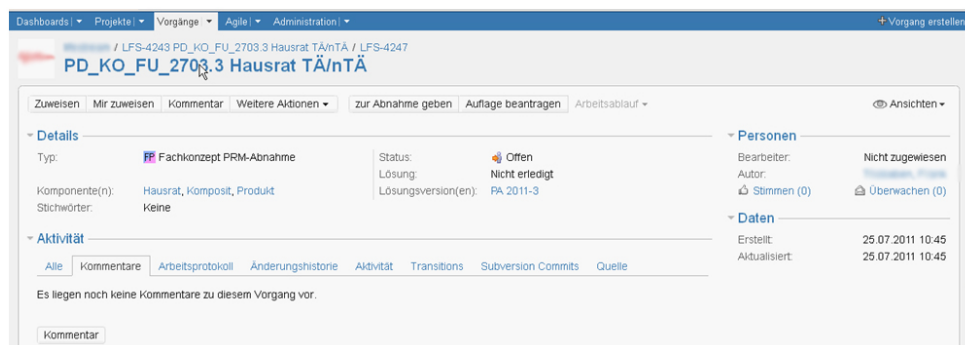


Abb. 27 Standardisierte Unteraufgabe in JIRA

Durch die hier dargestellte Nutzung von JIRA, nicht nur zur Prüfung und zum Austausch, sondern, im Endeffekt, auch zur Ablage von Fachkonzepten und anderen Dokumenten, wird JIRA gleichsam zum **Dokumentenmanagementsystem** (DMS). Für diese Verwendung ist JIRA per se nicht ausgelegt, unterstützt diesen Bedarf aber durch verschiedene ETP 2.0-Funktionalitäten wie Links oder Suche. Was im Vergleich zu tatsächlichen DMS fehlt, sind Volltextsuchen in angehängten Dokumenten oder beispielsweise eine automatische Versionierung von hochgeladenen Dokumenten.

Die Tatsache, dass Fachkonzepte und QS-Dokumente über JIRA, statt wie früher mittels Email, ausgetauscht werden, bringt laut den Befragten einen grossen Vorteil in Bezug auf **Transparenz**, da dadurch zum Beispiel im Krankheitsfall von einzelnen Personen deren damit verbundene Aufgaben nicht „*liegenbleiben*“: In JIRA sind neue Fachkonzepte, die in den entsprechenden JIRA-Tasks hochgeladen und von der Fachkonzeption für andere Abteilungen zugewiesen werden, prinzipiell für alle sichtbar und auch zugänglich, im Gegensatz zu Email, wo nur die EmpfängereInnen Zugriff auf diese Fachkonzepte haben. Dieser Aspekt wird in Interviewausschnitt 19 ersichtlich, in welchem einer der Teamleiter erklärt, wie er mittels JIRA-Filter Überblick über neue, zuzuweisende Aufgaben erhält.

B: Das ist im Prinzip hier das allgemeine **LFS**-Projekt wo wir uns ja nur noch mit beschäftigen. Was ich jetzt nochmal öfters dann immer kontrolliere sind halt für die beiden wichtigen Leute in dem Team noch. **ENW_DE_TL**, was da bei ihm noch aktuell immer rumliegt, beziehungsweise bei **ENW_DE_TL_STV_1**, weil da können ja im Prinzip auch die

ganzen QSen, Fachkonzepte die gehen ja im Prinzip erst mal an die Teamleitung, beziehungsweise stellvertretenden Teamleiter. Das schaue ich mir dann eben an was bei einem dann..

B: Hier ist eigentlich der Haupt.. Augenmerkmal. Hier kommen sie im Prinzip alle an. Hier schaue ich dann mal nach was hier jetzt aktuell noch so rumliegt. Ob hier irgendwelche Sachen sind um die sich irgendwelche Entwickler noch kümmern müssten, wie hier zum Beispiel Testlisten QSen. (...) Also hier interessiert mich im Prinzip alles was noch zu QSen reinkommt. Also Testlisten-QS, Fachkonzept-QSen, da schaue ich dann rüber was herumliegt.

I: Und das sind Sachen die dann eben an andere Entwickler weitergeleitet werden müssen.

B: Genau, die leite ich im Prinzip hier nur weiter.

Interviewausschnitt 19 [12_ENTW_DE_STV_2: 88-91] Übersicht über zuzuweisende Tasks in JIRA

5.5.2 Formalisierung von Inhalten

Um Awarenessbedürfnisse und Awarenessmechanismen durch Tools umsetzen zu können, ist eine **Formalisierung von Awarenessinhalten** notwendig, d.h. die Inhalte müssen in digitalen Artefakten abgebildet werden. Abb. 28 zeigt diesen Schritt in einer schematischen Übersicht. Administrative Tools und Artefakte wie JIRA-Tasks, TTS-Tickets oder auch Projektpläne in anderer Form ermöglichen eine solche Formalisierung von Aufgaben, Projekten und ihrem Kontext – also jenen in Abschnitt 5.3.1 dargestellten Inhalten von Awarenessbedürfnissen. Es lassen sich Kontext und Abhängigkeiten, Zustand und Ereignisse, Übersicht und Details erfassen und auslesen.

Durch die Formalisierung können weiters die verschiedenen Awarenessmechanismen umgesetzt werden: Informationspushing wird durch das Eintragen von Informationen in die Tools möglich. Das Wahrnehmen von Ereignissen wird durch Subscription und Emailnotification sowie durch Monitoring von Filteransichten umgesetzt. Filter und Widgets, sowie die Suchfunktion stellen das hauptsächliche Mittel für Informationspulling in JIRA dar.

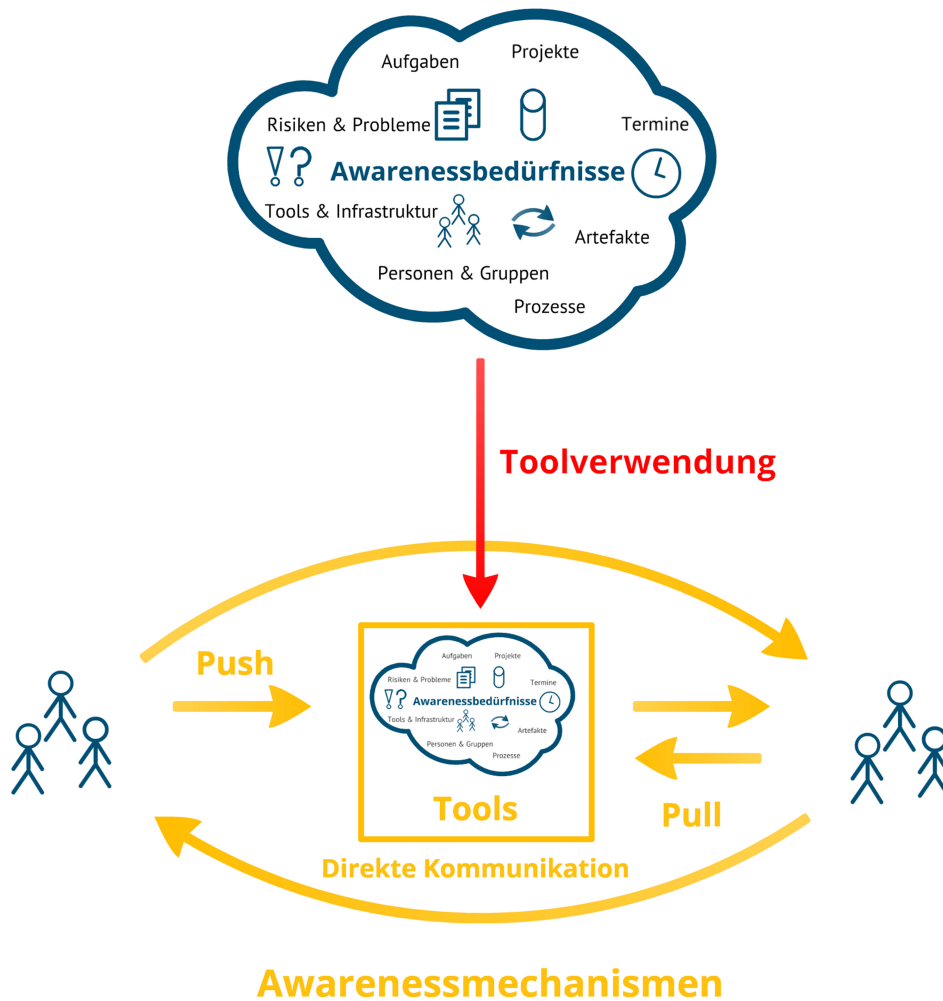


Abb. 28 Formalisierung von Awarenessbedürfnissen durch Toolverwendung

Die Untersuchung zeigt, dass die Umsetzung von Awareness in Tools nicht ganz einfach ist, da Informationen nicht immer vollständig oder aktuell erfasst werden, die Erstellung und Pflege von Filtern relativ komplex ist und dadurch die ersichtlichen Informationen falsch oder irreführend sein können, bzw. falsch interpretiert werden.

5.5.2.1 Zustand, Kontext, Abhängigkeiten, Detail

In JIRA ist es potentiell möglich, alle der in Abschnitt 5.3.1 identifizierten Inhalte von Awareness auf explizite oder implizite Weise in einem Task abzubilden. Dazu stehen als Datenstruktur die in Abschnitt 4.3.4 beschriebenen Attribute und Attributgruppen, Textfelder, Kommentare, Attachments und die Änderungshistorie zu Verfügung. Dadurch lassen sich Zustand, Fortschritt, Abhängigkeiten und Hintergrundinformationen darstellen.

Der Inhalt und Zustand von Aufgaben lässt sich mittels Titel, Aufgabentyp, expliziten Statusattributen wie Lösungsversionen und freien, textlichen Beschreibungen des Aufgabeninhaltes in den Beschreibungsfeldern, in Kommentaren oder auch durch angehängte oder verlinkte Dokumente (d.h. ein im Implementierungstask verlinktes Fachkonzept legt fest, was implementiert werden soll) definieren. Auch die Zuordnung zu einem Projekt kann Aufschluss über den Inhalt und das Ziel einer Aufgabe geben. Zeitliche und personelle Abhängigkeiten und Rahmenbedingungen können ebenfalls durch die entsprechenden Attribute festgelegt werden. In Bezug auf Artefakte können andere JIRA-Tasks verlinkt werden, ebenso können Produktionsartefakte angehängt werden. Informationen über technische Systeme wie SVN und Cruise Control und deren Inhalt und Status sind bei entsprechender Konfiguration möglich. Der Task-Typ einer Aufgabe lässt eine Zuordnung zu einem Prozess, bzw. zu einem Prozessschritt zu. Schließlich können

Informationen zu Problemen und Risiken explizit als freie Textbeschreibungen und Kommentare und implizit beispielsweise aus zeitlichen Angaben (Verzögerungen) oder der Taskhistorie erfasst und ausgelesen werden.

Zur Darstellung des Fortschrittes werden in administrativen Artefakten Attribute vorgesehen, die **explizit** darüber Bescheid geben: in JIRA ist das der Status von Tasks, beispielsweise „offen“ oder „erledigt“, in TTS „in Arbeit“, „realisiert“ oder „geschlossen“ (siehe auch Abschnitt 4.3.4 zu Attributen in JIRA), die in der Firma gesehenen Projektpläne tragen farbliche Markierungen und prozentuelle Angaben über den Fertigstellungsgrad von Teilaufgaben. Bei JIRA und TTS kann die Angabe des Status überdies darstellen, in welchem Schritt eines Prozesses sich die Aufgabe befindet, nämlich dann, wenn die Aufgabe teamübergreifend verwendet wird. Im Falle von Tickets bedeutet beispielsweise das Einstellen des Status „realisiert“, dass der / die EntwicklerIn den Fehler im Sourcecode gelöst hat, und das Ticket somit an das Releasemanagement wandert, welches sich um die Kompilierung und Auslieferung auf eine Testumgebung kümmert. Ist dieser Schritt erfolgt, wird das Ticket auf „testbereit“ gestellt, womit es nun in der Verantwortung des eingetragenen Nachtesters liegt.

Neben expliziten Attributen lässt sich der Zustand und Hintergründe von Aufgaben und Abhängigkeiten auch über freie Beschreibungen erschliessen: In TTS wird diese Möglichkeit durchwegs für die Beschreibung des Problemes, verwendet, stellt also eine Spezifikation der Aufgabe dar. In JIRA in der untersuchten Firma zeigte sich, dass dieses Feld sehr unterschiedlich verwendet wird, häufig ist darin keine weitere Information zu finden, teilweise werden in Entwicklungstasks Einzelschritte, die zur Umsetzung nötig sind, von den Entwicklern erfasst und zum Teil auch mit Markierungen über die Fertigstellung versehen. Solche Arten von „Todo-Listen“ sind in der hauptsächlich untersuchten Gruppe aber eher selten, EntwicklerInnen geben an, sich Notizen über „private“ Todos eher auf handgeschriebene Zettel zu machen, siehe Interviewausschnitt 20:

I: Also du machst dir nicht einen eigenen Task dafür auf.. keine Ahnung, "das und das gehört noch gemacht"..

B: Ja, also wichtig ist einfach so dass man so einen kleine Liste hat in der man sieht "ach ja, 'das muss man noch machen', 'das muss man noch machen', 'das muss man noch machen'", und dann weiss man einfach noch vom Vortag, oder vielleicht ein paar Tage früher dass das noch ein Problem war. Und dann setzt man sich da wieder ran. Häufig ist es auch so, dass man, oder dass ich mir Abends kleine Notizen auf Papier und Zettel, und dass ich dann Morgens weiss "ach ja, da war ich dran", und dann kann ich da weitermachen.

Interviewausschnitt 20 [09_ENTW_DE_3:126-127] Entwickler über Todo-Listen

In einem anderen Fall aus einem anderen Team lässt sich aber durchaus der Nutzen der Aufgabenerfassung in digitaler- statt in Papierform erkennen: Durch die Erfassung eines „Todos“ in digitaler und webbasierter Form kann das Artefakt als Kommunikationsmedium genutzt werden, und stellt dazu gleich noch den Kontext für die Kommunikation (also das Kommunikationsthema) dar, siehe Interviewausschnitt 21 und Interviewausschnitt 22:

B: ... aber in diesem Fall ist wahrscheinlich **RMM_TL** zu **RMM_4** gegangen und hat gesagt "wir hätten das gerne noch, dass es eine Suchfunktion geben würde". Und dann hat er, um sich das nicht auf einen gelben Zettel irgendwo schreiben zu müssen, das in JIRA eingetragen. Und man kann dann ja manchmal Kommentare noch dazu schreiben, und dann ist ja auch ganz praktisch, dass das dann jeder sehen kann. Kommt hier selten, aber wie man sieht gelegentlich vor. In diesem Fall war es ein Kommentar der sich darauf bezieht wann er das Ticket, beziehungsweise dass das Ticket nicht umgesetzt wird bis die die Serverstruktur umbauen. Und solche Sachen.

Interviewausschnitt 21 [10_RMM_1:70]: Todos in JIRA & Kommentare als Kommunikationsform

B: Da kann ich 'in Arbeit nehmen' und 'warten', und muss dann das auch noch kommentieren. Das macht alles Sinn und Zweck, weil jedesmal wenn ich, oder wer auch

immer dann ein Stück weiter weg sitzt, da rein kuckt und das Ding steht auf 'warten', dann weiss ich warum. Ich kann da reinkucken, und dann steht halt drin 'weil Entwickler krank' oder ich weiss nicht was.

Interviewausschnitt 22 [17_ENTW_DE_TL:77] Kommentar bringt Kontextinformation

5.5.2.2 Ereignisse – Subscription und Notification

Awareness über Ereignisse wird in JIRA durch Subscription und Notification umgesetzt. In den vorangegangenen Abschnitten 5.4.1, 5.4.3 und 5.4.4 wurden die beiden Mechanismen bereits diskutiert, an dieser Stelle sollen die Erkenntnisse nochmals zusammengefasst werden.

Subscription funktioniert in JIRA implizit: alle als AutorIn oder BearbeiterIn in einem Task eingetragene Personen werden automatisch in die InteressentInnen-Liste eingetragen, und über die verschiedenen Notification-Mechanismen informiert. Explizite Subscription wird über einen „Überwachen“-Button eingetragen, allerdings gab in der Untersuchung niemand an, diese Funktion zu nutzen. Ein Problem bei JIRA liegt darin, dass immer nur eine Person als BearbeiterIn eingetragen sein kann. Wenn eine Aufgabe beispielsweise von der Fachkonzeption an die Entwicklung wechselt, geht der Task somit auch aus dem Awarenessfokus der Fachkonzeption verloren. Im Gegensatz dazu sind in TTS mehrere Personen und ein Team erfassbar, Tickets bleiben somit auch bei mehreren Personen im Fokus, wie die Interviewantworten zum Thema TTS (siehe Abschnitt 5.3.1.6) zeigen.

Als Notification-Mechanismen stehen Email und RSS-Feeds zu Verfügung, wobei keine der befragten Personen angab, RSS zu benutzen. Email bedingt, dass auch der Emailclient geöffnet ist und regelmässig überprüft wird, diese Bedingung wird aber – wie die Befragung zeigt – erfüllt. Allerdings gaben die meisten der befragten Personen an, automatisch generierte Emails (diese werden nicht nur von JIRA verschickt, sondern auch von TTS, REA oder CruiseControl) im Emailclient automatisiert zu löschen, bzw. in separate Ordner zu sortieren. Der Grund dafür liegt in der zu hohen Anzahl an verschickten Emails. Durch diese Vorgehensweise wird der Emailclient zu einer Art peripherem Awarenessmechanismus: Aussortierte, (ungelesene) Emails werden durch eine (fettgedruckte) Zahl neben den entsprechenden Ordnern angezeigt. In der Befragung zeigte sich, dass die Emails zwar meist nicht im Detail gelesen würden, neu eingelangte Emails werden aber dennoch registriert und zum Teil zumindest anhand ihres Betreffs beurteilt. In der Folge wird dann in die Weboberfläche von JIRA gewechselt, um detailliertere Informationen einzuholen. Dieses Verhalten wird in Interviewausschnitt 23 ersichtlich:

I: Und ins JIRA, wie oft siehst du da rein?

B: Hm, eigentlich fast nur wenn ich Emails bekomme. Also immer wenn irgendwelche Aufgaben zugewiesen werden oder geändert werden kriegst du (...). Teilweise, also meistens ist es so, du kriegst eine ganze Reihe von Tasks, "Testnachweis", "Konzept" und so weiter, irgendwie eine ganze Menge, und anstelle die alle durchzulesen (Anm.: die Emails) ist dann meiner Meinung nach nicht so sinnvoll. Ich lösche die alle, und denke dann, "ok, es ist irgendetwas in JIRA passiert, ich kuck mal", und dann habe ich das eigentlich im Kopf was zu wann fertig sein muss. Dann mache ich es meistens so, dass ich immer wenn ein Email kommt ich da rein gehe, und ja, Abends dann meistens auch noch, nur um zu kucken, um den nächsten Tag so zu planen. Was liegt denn eigentlich an, was ist denn noch offen bei mir, habe ich vielleicht irgendwelche Tasks schon erledigt aber noch nicht geschlossen? Ja, einmal am Abend kann man sagen so.

I: Ok. Diese Emails die du angesprochen hattest - ist das quasi ein aggregiertes Email, wo dann von allen Tasks etwas drinnenstehen, oder bekommst du etwas von jedem Task?

B: Nein, ich bekomme zehn Emails "irgendetwas wurde getan", und das macht halt auch überhaupt keinen Sinn in der Form. Kann man sich auch wegfiltern unter Umständen. Ich hab halt einen Filter eingebaut, wie gesagt so "schupp" (Anm.: Wischgeste) - weg damit.

Interviewausschnitt 23 [01_ENTW_DE_1: 28-29] Emailnotification und JIRA

Awareness über Aufgaben wird also aus einer Kombination aus Emailnotification und dem Monitoring von Filterdarstellungen im Taskmanagementtool erreicht. Einen weiteren Mechanismus stellt mündliche, direkte Kommunikation dar. Wie bereits in Abschnitt 5.4.1 gezeigt wurde, wird besonders in zeitkritischen Situationen – zusätzlich zum Zuweisen von Tickets oder Aufgaben in den jeweiligen Tools – mündlich Bescheid über Zuweisungen (oder andere Änderungen) gegeben. Dies geschieht, weil Emailnotification und Monitoring von Filtern keine Garantie darstellen, dass Änderungen von der anderen Person auch tatsächlich registriert werden. Zudem werden im Zuge der mündlichen Kommunikation häufig auch Hintergrundinformationen für die Gründe der Zuweisung oder andere Detailinformationen ausgetauscht.

5.5.2.3 Übersicht – Filter und Suche

Dashboards und die darauf abgelegten Filter und Widgets stellen eine wichtige Quelle für Übersicht und Awareness über Aufgaben und Projekte in JIRA dar. Aus der Verknüpfung von fachlichen, personellen und zeitlichen Angaben innerhalb eines Artefaktes, gemeinsam mit (expliziten) Attributen über den Zustand von Aufgaben lassen sich der Zustand und Fortschritt von Aufgaben und Projekten feststellen. Dies wird durch die Aussage eines Teamleiters in Interviewausschnitt 24 illustriert:

B: Da kann man halt jederzeit sehen, wer, was, wo dran ist. Der hat hier noch viele Implementierungs-Aufgaben für den Meilenstein liegen, das sind... diese drei vor allen Dingen und noch ein paar andere. Man kann sehen, wieviele Aufgaben die Leute insgesamt haben. Man kann sehen.. also... (...) Und man kann halt auch sehen, welche Aufgaben dran hängen. Was den Vorteil hat, wenn man sagt "ok, ich will jetzt noch wissen was für Fachkonzept-QSen sind denn noch offen" das sind dann die Aufgaben hier. Man kann hier unten auf die Zahl drücken (klickt) und sich ankucken "ok, diese Leute haben diese Fachkonzept-QSen noch liegen, die sollten sich gefälligst mal drum kümmern". Und im Zweifelsfall muss man da mal hinterher laufen.

B: (klickt) Und das ist mein Hauptfilter, das sind noch die einzelnen dazu. Hier sieht man das im einzelnen auch noch heruntergebrochen, mit den einzelnen Workflow-Schritten in denen sich das gerade befindet momentan. Wenn man sieht, meinerwegen hier bei den Implementierungs-Aufgaben sind so-und-so viele offen, die sind momentan gerade 'in Arbeit'... Das kann man sich anschauen, da kann man auch Statistiken drüber ziehen. Und was-weiss-ich nicht alles. Man sieht einen Gesamt-Meilenstein - "wie sieht das aus?", "wieviel habe ich schon erledigt?", "wieviel ist offen?". Das ist so meine... Hauptinformationsquelle. Um wirklich an Informationen heranzukommen, jetzt zu sagen "wie weit sind wir denn", oder jetzt "wo hakt es".

Interviewausschnitt 24 [17_ENTW_DE_TL:55-56] Verknüpfung von Aufgaben, Personen und Terminen in JIRA

Filter und ihre Filterkriterien werden mittels einer Maske oder mittels einer speziellen Abfragesprache spezifiziert (siehe dazu ebenfalls Abschnitt 4.3.4). Diese Filter können ausserdem von einer Person erstellt (und gepflegt) werden, und anschliessend anderen Personen im System zu Verfügung gestellt werden. In der Untersuchung zeigte sich, dass von der Möglichkeit, Filter und Dashboards zu teilen, auch viel Gebrauch gemacht wird. Beinahe alle befragten KompositentwicklerInnen geben an, dass von der Teamleitung konfigurierte Dashboard zu verwenden, viele haben aber zusätzlich auch eigene Dashboards und Filter angelegt, um die sie betreffenden Aufgaben im Blick zu haben.

Bei der Verwendung von Filtern ist zu beachten, dass für Filterdarstellungen eben jene expliziten Attribute der einzelnen JIRA-Tasks notwendig sind. Informationen die als Freitext, Kommentar oder implizit in der Task-Historie vorhanden sind, sind durch solche Filter nur schwer zu erfassen.

5.5.2.4 Probleme

Filter sind also ein wichtiges Mittel, um Übersicht über eigene Aufgaben zu haben und den Fortschritt des Teams zu verfolgen. Ausserdem dienen Filter im untersuchten Projekt auch als Einstiegspunkt für die Suche nach Fachkonzepten. Dabei treten aber auch Probleme auf.

Das Auslesen von Informationen aus Tools und Artefakten bedingt eine gewisse Kenntnis der Tools, bzw. deren Konfiguration und der darin abgebildeten Prozesse. Dies wird beispielsweise aus den Aussagen in Interviewausschnitt 25 und Interviewausschnitt 26 ersichtlich, und bedeutet, dass es für Aussenstehende nicht ohne weiteres möglich ist, sich aus den Tools einen Überblick über den Zustand von Aufgaben oder Projekten zu machen, bzw. dass es zu Fehlinterpretationen kommen kann.

Eine Gefahr die natürlich besteht ist, wenn derjenige vorher nicht bei den Prozessen der Erstellung - also wie ich das in JIRA abbilde - mit dabei gewesen ist, kann es passieren, dass er Sachen fehlinterpretiert. Wenn er das Handbuch noch nicht gelesen hat wie das jetzt gerade zu verstehen ist, was bedeutet das wenn ein Task "wartet". Weil wir uns natürlich für uns bestimmte Regeln natürlich überlegt haben, und gesagt haben, keine Ahnung, er wird auf warten gestellt wenn es länger als drei Tage dauert.

Interviewausschnitt 25 [02_FK_TL: 141] Kenntnis von Tools und Konfiguration

B: Das ist, auch solche Sachen wie dieses 'Neunzig-Prozent-Phänomen': ich melde immer 90 Prozent fertig, habe aber noch 80 Prozent der zeitlichen Aufwände offen.. oder irgendwie so. Also das Thema ist zu 90 Prozent erledigt, aber die letzten 10 Prozent kosten mich 90 Prozent der Zeit. Auch das kann ich damit fast nicht berücksichtigen.

Interviewausschnitt 26 [08_PMM_1: 93] "90 Prozent-Phänomen"

Generell zeigt sich, dass die Erstellung, Pflege und Konfiguration von Filtern, welche über die eigenen zugewiesenen Aufgaben hinausgehen, sehr komplex und aufwendig sein kann: Wenn bei der Konfiguration von Filtern in JIRA falsche Parameter ein- oder notwendige Parameter ausgeschlossen werden, kann dadurch ein falsches Gesamtbild entstehen. Auch durch die Änderung von in den Tools abgebildeten Prozessen oder Änderungen in der Personalstruktur und damit einhergehende neue Parameter, welche nicht in den bestehenden Filtern berücksichtigt sind, können zu fehlerhaften Auswertungen führen.

Dasselbe Problem trifft auch für die Suchfunktion zu, indem falsche oder zu viele Ergebnisse angezeigt werden, bzw. Ergebnislisten nicht ausreichend eingegrenzt werden. Probleme bei der Suche treten vor allem auch bei der in Abschnitt 5.5.1.4 beschriebenen Verwendung von JIRA zur Ablage von Dokumenten zu Tage.

5.5.2.5 Ursachen

Eine Ursache dieser Probleme liegt in der Strukturierung von Projekten nach Meilensteinen: Viele Themen sind nicht nach einem Meilenstein abgeschlossen, sondern müssen aufgrund ihres Umfangs über mehrere Meilensteine hinweg entwickelt und bearbeitet werden. Dadurch ist es nötig, immer wieder auf vermeintlich bereits „abgeschlossene“ Fachkonzepte (welche im untersuchten Projekt in JIRA-Tasks abgelegt sind, siehe Abschnitt 5.5.1.4) zurück zu greifen. Solange die betreffenden Tasks noch der bearbeitenden Person zugewiesen und nicht abgeschlossen sind, werden sie über die vorkonfigurierten Filter wiedergefunden, sobald die Tasks aber geschlossen oder einer anderen Person zugewiesen werden, verschwinden sie aus dem „Blickfeld“ den ein Filter darstellt (siehe Interviewausschnitt 27).

B: (...) ich weiss nur, dass ich mit den Konzepten Schwierigkeiten habe die zu finden. Und zwar einfach, wenn ich einen Task habe, dann finde ich das natürlich. Und dann, wenn es dann neue Sachen dazugibt, dann finde ich die auch. So, wenn ich jetzt aber keinen Task habe, und ich soll ein Konzept finden, dann gebe ich manchmal die Nummer ein, was weiss ich, "KO_FU irgendwas", oder "PD_KO_FU", und manchmal findet er das, und manchmal

findet er das nicht. Und er findet es halt eben auch manchmal NICHT. Obwohl es diese Nummer gibt, obwohl es unter "PD_KO_FU was weiss ich, 1234" abgelegt ist, findet er es manchmal nicht. Ich weiss nicht warum, aber es ist manchmal so. Und das finde ich sehr ärgerlich. Und das könnte bei..., ich weiss nicht woran das liegt, aber so ist es halt eben.

Interviewausschnitt 27 [03_ENTW_DE_2: 79) Wiederfinden von Konzepten in JIRA

Eine weitere Ursache des Problems liegt in der hohen Anzahl an vorhandenen Tasks in JIRA. In Abb. 29 ist zum Beispiel ersichtlich, dass für das teamübergreifende LFS-Projekt mehr als 3500 offene und geschlossene Tasks existieren, diese sind mehr als 50 Personen zugeordnet. Ausserdem wurden, bevor die in 5.5.1.4 beschriebene teamübergreifende und standardisierte Projektstruktur eingeführt wurde, bereits andere, für das Kompositsteam relevante JIRA-Projekte angelegt, die durch die lange Bearbeitungszeit von Themen aber noch immer relevant sind für die gegenwärtige Arbeit.

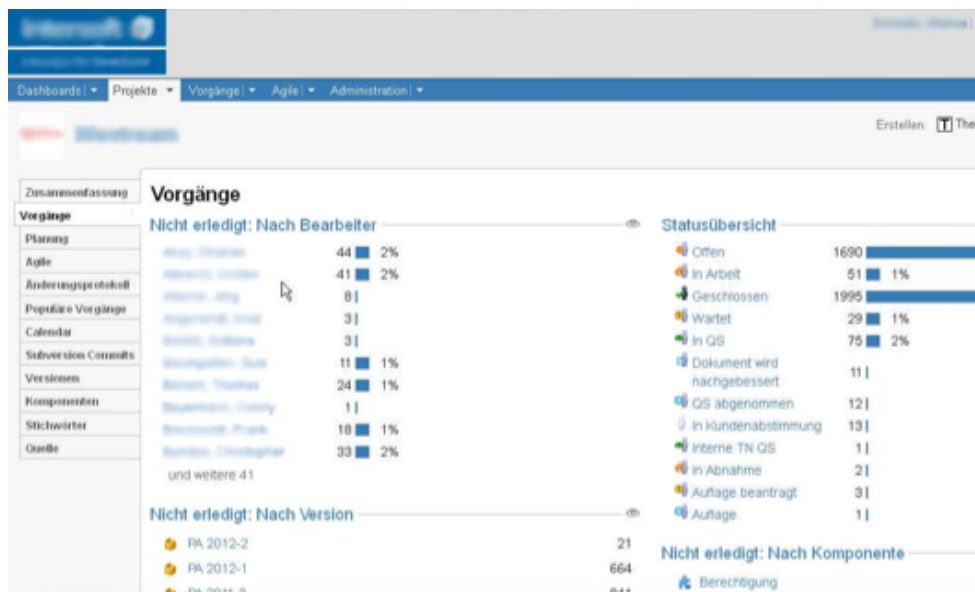


Abb. 29 Screenshot der LFS-Übersichtsseite in JIRA: Mehr als 50 Mitarbeiter, mehr als 3500 offene und geschlossene Tasks.

Auch die Abbildung von Prozessen in JIRA, bzw. deren Änderung und Anpassung, das hinzukommen neuer Meilensteine, Lösungsversionen oder Tasktypen, oder auch Änderungen im Teamgefüge erhöhen die Anzahl an Filter- und Suchkriterien die zu berücksichtigen sind, und somit die Komplexität in der Bedienbarkeit des Systems. Immer wenn es Änderungen oder Ergänzungen an dieser Systemkonfiguration gibt, müssen die Filter ebenfalls entsprechend aktualisiert werden, um korrekte und vollständige Ergebnisse zu zeigen.

Eine weitere, wesentliche, Ursache für Fehlinterpretationen in JIRA liegt darin, dass Tasks häufig nicht – oder erst verspätet – aktualisiert werden: Der Status von Aufgaben wird nicht angepasst, Verschiebungen von Terminen und Anpassungen von Zeitaufwänden werden nicht im Tool nachgezogen, freie Textbeschreibungen werden nicht aktualisiert. Dieses Verhalten wird in Interviewausschnitt 28 und Interviewausschnitt 29 deutlich. Die Ursachen für dieses Nicht-Anpassen liegen ebenso in der hohen Anzahl an (gleichzeitigen) Tasks und einem zu hohen Aufwand für die Pflege und Aktualisierung von Tasks und Abhängigkeiten. Besonders in stressigen Situationen bleibt keine Zeit, um inhaltlich aufschlussreiche Angaben zum Status von Aufgaben zu erfassen.

I: Und wie ist das dann mit den Terminen, wenn da irgendetwas nicht eingehalten wird? Angenommen du brauchst jetzt für die Implementierung.. sagen wir die ist mit einer Woche angesetzt und dann stellt sich heraus, dass das viel länger dauert?

B: Naja, das ist eigentlich nicht so problematisch. Wir haben ja, zum Beispiel wenn man jetzt so eine Implementations-Task hat - (klickt) ich kann mal hier da drauf kucken - zum

Beispiel 'Allgemeine Abzüge': da sind wir mit (unv. **ENTW_DE_TL_STV_1**) wie ich mit angefangen habe, die Felder eingearbeitet, und es gab eine erste Konzeptversion. Und da waren Sachen unklar oder nicht beschrieben, und dann.. ja dann dauert das schon mal länger. Aber das ist dann auch nicht so wild. Wir sind ja immer dabei dass die, dass Prozesse vorangetrieben werden, dass die Qualität gut werden soll. Ja, Hauptsache zum Release ist es dann fertig, so wie jetzt.

I: Ok. Aber wenn, dann machst du das quasi, (unv. meldest) du das dem **ENTW_DE_TL** oder so?

B: Nein (gedehnt). **ENTW_DE_TL** hat damit recht wenig eigentlich zu tun. Eigentlich.. die Implementations-Tasks die werden eigentlich immer erst zum Schluss geschlossen.. Da gibt es immer ein paar Abhängigkeiten dazwischen die man so nicht sieht, weil das ja recht getrennt ist.. Ja und dann werden Sachen jetzt so jetzt dann fertig, und dann schliessen wir die. Also meistens öffne ich die auch nicht wieder wenn die mal geschlossen sind. Wenn da noch Nacharbeiten sind, ja.. dann arbeite ich da einfach an dem Feld dran. Ich weiss ja letztendlich was zu tun ist, aber...

Interviewausschnitt 28 [09_ENTW_DE_3: 122-125] Anpassen von Terminen von Unteraufgaben in JIRA

I: Ok. Und welchen Task hast du jetzt zum Beispiel momentan in Arbeit?

B: Also eigentlich alle (schmunzelt). Das Problem ist, ich arbeite wahrscheinlich nicht so wie ein Entwickler damit muss man dazu sagen. Das Problem ist, ich habe sehr viele Fachkonzepte die einfach immer nur angepasst werden müssen. Also die werden pro Meilenstein, wenn da irgendwelche Fehler auftreten - das weisst du ja noch von KFZ sicherlich - dann werden die einfach nur nochmal angepasst, oder wird halt geschaut was falsch ist oder so. Und dadurch ist jetzt auch so eine Masse an Tasks geworden. Und ich nehme mir den Task dann meistens - muss ich dann auch gestehen - und nehme den "in Arbeit" wenn ich eigentlich schon fertig bin damit. Nehme den dann also quasi kurz in Arbeit, schliesse den dann sofort wieder, und gebe dann das ganze entsprechend zur QS.

Interviewausschnitt 29 [06_FK_2: 74-75] "In Arbeit"-Nehmen von Tasks

Alle diese Probleme führen letztlich dazu, dass die im Tool vorhandenen Informationen nicht immer vollständig oder aktuell sind (in dem Sinne, dass sie dem Wissensbedürfnis von interessierten Awarenesskonsumten entsprechen) oder korrekt sind, wodurch immer ein zusätzlicher Informationsaustausch in Form direkter Kommunikation nötig ist:

I: Nachdem du ja auch stellvertretender Abteilungsleiter bist musst du ja wahrscheinlich auch am Laufenden bleiben, was die Mitarbeiter so machen, also die anderen Entwickler im Komposit-Team. Wie machst du das, oder wie kriegst du das mit?

B: Da versuchen wir natürlich sehr viel über JIRA eben abzudecken. Ist nicht ganz einfach. Wir müssen ja in der Iterationsplanung die wir immer im zweiwöchentlichen Abschnitt machen, wird halt geschaut "ja was habe ich jetzt für die Iteration angemeldet gehabt, wie ist der Stand dabei, und was ist da fertig". Das erfolgt auswertungstechnisch über JIRA natürlich, die Tasks die ich dafür angemeldet habe, die werden in JIRA abgefragt ob die eben schon geschlossen sind, oder ob die warten auf irgendwas, so dass man dann den Status da abgeben kann. Nichts desto trotz müssen wir aber immer noch dann in unseren Teammeetings da auch den Zwischenstand abfragen. Denn alles, jeden Schritt kannst du nicht so in JIRA dokumentieren, und da ist es halt besser, wenn man direkt vom Entwickler das Feedback kriegt "ja ich arbeite zwar daran, aber ich kann nicht weitermachen weil irgendwas anderes mich jetzt davon abhält". Und deshalb ist es halt so eine Kombination aus Statusabfragen aus JIRA, und die Mitarbeiter interviewen wie es da mit den Arbeiten aussieht.

Interviewausschnitt 30 [05_ENTW_DE_STV_1: 18] Awareness als Kombination von Toolverwendung und direkter Kommunikation

5.5.2.6 Lösungsversuche

Im untersuchten Projekt werden unterschiedliche Ansätze versucht, um die Probleme, bzw. deren Ursachen zu verhindern oder zu kontrollieren. Die Standardisierung von JIRA-Tasks nach der in 5.5.1.4 beschriebenen Struktur soll eine bessere Eingrenzung von Tasks ermöglichen, erhöht damit aber gleichzeitig die **Komplexität** für die Erstellung von Filtern und den Aufwand für die Pflege von Filtern und Tasks. Eine weitere Möglichkeit um Tasks wieder auffindbar zu machen, ist die Verschlagwortung von Aufgaben und Projekten mittels der in JIRA-Tasks zu Verfügung stehenden Attribute (siehe dazu Abschnitt 4.3.4). Diese Verschlagwortung ist aber sehr zeitaufwendig und bedarf großer Sorgfalt, weshalb dieser Ansatz im untersuchten Projekt auch wieder aufgegeben wurde. Beide Ansätze bedingen ausserdem, dass Aufgaben nur entsprechend der vorgesehenen **Konventionen** angelegt werden. Im untersuchten Team wurde daher vorgesehen, dass nur noch wenige Personen (TeamleiterInnen und deren StellvertreterInnen) neue Tasks anlegen dürfen, damit die Einhaltung der Konventionen sichergestellt ist.

Das Eclipse-Plugin *MyLyn* welches die Ansicht und Bearbeitung von JIRA-Tasks aus Eclipse heraus ermöglicht, und zusätzlich den Bearbeitungskontext in Form eines Protokolles von verwendeten Sourcecode-Fragmenten erfasst, steht den EntwicklerInnen zu Verfügung, wurde aber nicht verwendet. In der Befragung gaben die EntwicklerInnen an, dass sie die Bedienung als *zu umständlich* empfinden und daher die Weboberfläche von JIRA für die Taskbearbeitung vorziehen:

I: Verwendest du das MyLyn?

B: Nein, das benutze ich überhaupt nicht. Als das mal eingeführt wurde habe ich das mal anfänglich benutzt, aber mache ich nicht mehr. Das war mir ein bisschen zu umständlich. Da ist man dann ja... In Eclipse dieses Plugin, nicht? Ich sehe das noch gelegentlich dass manche Leute damit einchecken, dann gibt es ja immer diese typischen Eincheck-Commit-Kommentare, aber sonst...

Interviewausschnitt 31 [10_RMM_1: 47-48] Verwendung von Mylyn

I: Dann eine kurze Frage zu Mylyn und Eclipse: Hast du das Mylyn aktiviert, oder verwendest du es?

B: Ich habe es mal probiert, fand es auf jeden Fall cool, meine Tasks zu sehen in Eclipse, habe die dann aber darüber nicht weiter verwaltet. Also ich habe mich da nicht richtig eingearbeitet. Also ich gehe immer über die Weboberfläche, und nicht über Eclipse. Ich habe auch mal probiert, dass man da die Tasks anschaltet, und dann so und so lange nur die Sourcen bearbeitet, und da wird na dann auch automatisch getrackt. Beim Einchecken ist ja dann das Task-Kürzel, quasi die ID, vorbelegt. Das hatte ich alles mal probiert, aber wahrscheinlich aus mangelnder... oder aus Bequemlichkeit wahrscheinlich nicht durchgezogen. Es ist auch keine Vorgabe für uns, dass wir das machen sollen, und dann macht man es so wie immer.

I: Und legst du in Mylyn private Tasks oder Todos an? Die du nicht mit JIRA synchronisierst, sondern nur innerhalb von Eclipse?

B: Also mittlerweile hab ich das Mylyn gar nicht mehr offen.

Interviewausschnitt 32 [01_ENTW_DE_1: 80-83] Verwendung von Mylyn in Eclipse

In der Gegenüberstellung von TTS und JIRA in Abschnitt 5.3.1.6 wird ersichtlich, dass TTS durchwegs für die Betrachtung der Aufgaben anderer Personen verwendet wird, während in JIRA hauptsächlich eigene Aufgaben angezeigt werden. Wie im betreffenden Abschnitt erwähnt, muss JIRA extra für diese Ansichten konfiguriert werden, man muss über die abgebildeten Tasks und Prozesse Bescheid wissen, und Veränderungen in JIRA (welche durchaus häufig stattfinden) müssen auch in den Filtern nachgezogen werden. Dazu können in JIRA nur eine bearbeitende Person und der/die AutorIn des Tasks eingetragen werden. Im Gegensatz dazu benötigt TTS zur Ansicht keinerlei Konfiguration,

und es können mehrere Personen und zumindest ein für die Bearbeitung verantwortliches Team eingetragen werden. Diese Kombination aus einfacher Konfiguration und der Erfassung mehrerer beteiligter Personen scheint sich also positiv auf die Awareness über Aufgaben und Personen auszuwirken.

5.6 Kategorie Faktoren

In den vorigen Abschnitten wurden eine Reihe von Faktoren identifiziert, die die Awarenessbedürfnisse der beteiligten Personen, und auch die Arbeit mit den untersuchten Systemen beeinflussen. Diese können unterschieden werden in Faktoren, die eher durch die Organisation oder das Arbeitsumfeld beeinflussbar sind, und solche die eher den verwendeten Tools und der zu kommunizierenden Information zuzuschreiben sind. Das Arbeitsumfeld – also eine Firma oder Organisation und ihr Arbeitsauftrag – ist ein wesentlicher und bestimmender Faktor, welche Awarenessbedürfnisse bestehen, welche Awarenessmechanismen eingesetzt werden können, und welche Tools zu Verfügung stehen, um Awarenessinhalte zu formalisieren und Mechanismen umzusetzen. Diese Faktoren sollen in diesem Abschnitt nochmals eigenständig betrachtet werden.

5.6.1 Organisatorische Faktoren

Rollen: Die Rolle einer Person bestimmt im wesentlichen darüber, ob sie eher über eigene oder fremde Aufgaben und Inhalte Bescheid wissen muss, darüber, wie detailliert sie Bescheid wissen muss und zudem, ob sie in Bezug auf dieses Wissen über Inhalte eher Informationsproduzent oder Konsument ist. In der untersuchten Firma konnte zwischen Rollen mit und ohne unmittelbarer Verantwortung für andere MitarbeiterInnen unterschieden werden: Team- und AbteilungsleiterInnen und „normale“ MitarbeiterInnen wie Entwicklerinnen und FachkonzepterInnen. Bei Team- und AbteilungsleiterInnen ist es deren Hauptaufgabe Übersicht über das Team und das Projekt zu behalten, das umfasst die fristgerechte Fertigstellung von Aufgaben, die Auslastung von Mitarbeitern um gegebenenfalls Aufgaben umverteilen zu können, das Einhalten von Terminen und auch das Erkennen von Risiken. Dazu sind sie auch für das Funktionieren der verschiedenen Prozesse an denen sie teilhaben verantwortlich, indem sie Probleme und Reibungspunkte in den Prozessen erkennen. „Normale“ Entwicklerinnen und Fachkonzepterinnen haben ihren Fokus hauptsächlich auf ihren eigenen Tasks, und interessieren sich für Aufgaben von Kolleginnen meist nur dann, „wenn sie einen selber betreffen“, d.h. da, wo die eigene Arbeit durch die Arbeit anderer beeinflusst wird, oder wenn sie ausreichend Zeit und keinen Arbeitsstress haben. Das Erkennen und Kommunizieren von Problemen und Risiken ist auch ein wesentlicher Teil ihrer Arbeit, allerdings haben sie weniger Einfluss auf die Gestaltung von Prozessen.

Vorgehensmodell: Das verwendete Vorgehensmodell zeigt sich in der Untersuchung als ein wesentlicher Faktor dafür, welche Awarenessbedürfnisse es gibt, und wie mit den vorhandenen Tools gearbeitet wird. Bei einem agilen Vorgehen sind die Zyklen aus Konzeption, Entwicklung und Test sehr viel kürzer als bei einem Wasserfallmodell, es wird eher parallel als sequentiell gearbeitet und einzelne Arbeitspakete sind kleiner. Die interviewte Person des agil vorgehenden Teams gibt an, stets sehr genau über die Arbeit der vorgelagerten Teams informiert sein zu müssen, dieser Austausch findet sehr stark in direkter Kommunikation statt. Im Wasserfallmodell sind die Arbeitspakete wesentlich grösser, diese werden im Prinzip bloß einmalig übergeben, allerdings zeigt sich, dass es im Lauf der Entwicklung und des Testens häufig zu Korrekturen und Ergänzungen der Konzeption kommt, welche dann auch parallel mit neu begonnenen Aufgaben behandelt werden. Durch die grösseren Arbeitspakete bleiben somit auch die Artefakte mit denen sie formalisiert werden länger aktiv, was wiederum die zu filternde und zu verarbeitende Informationsmenge in einem Informationssystem erhöht.

Teamgröße: Je kleiner ein Team ist, und je weniger verschiedene Teams oder Abteilungen an der Umsetzung einer Aufgabe beteiligt sind, desto einfacher ist es, über relevante Aktivitäten anderer Personen und Teams zu informieren und informiert zu sein. Das zeigt

sich in der Untersuchung im Vergleich von Releasemanagement und Kompositteam. In Abschnitt 5.3.2 wurde festgestellt, dass es eine Reihe von Abhängigkeiten zwischen den an einer kooperativen Arbeit beteiligten Entitäten (also Personen, Aufgaben, Artefakten, Tools, Terminen etc.) gibt. Mit der Anzahl an Personen die an einer Aufgabe oder einem Projekt mitwirken, steigt somit auch die Anzahl an zu beachtenden, abhängigen Entitäten.

Teamgrenzen und Hierarchien können in Bezug auf Awareness einen positiven Effekt haben, indem sie einen Filter darstellen und somit die Informationsmenge einschränken, sie können aber auch einen negativen Effekt haben, indem sie ein Kommunikationshindernis darstellen. Zu Beginn der Verwendung von JIRA in der untersuchten Firma wurden bestehende Teamgrenzen auch im Tool abgebildet, indem die Fachkonzeption- und Entwicklungsabteilung jeweils eigene Projekte für die Abwicklung derselben Themen angelegten. Zur teamübergreifenden Zusammenarbeit wurden Aufgaben nicht zwischen EntwicklerInnen und FachkonzepterInnen hin- und herbewegt, sondern auf Seiten der Entwicklung wurden eigene Tasks angelegt. Dadurch kam es einerseits zu einer doppelten Aufwand in der Administration, aber auch zu fehlender Awareness und dadurch zusätzlichem Kommunikationsaufwand in der Zusammenarbeit zwischen EntwicklerInnen und FachkonzepterInnen. Auch das in Abschnitt 5.3.2, Abb. 24 skizzierte „Aufmerksam machen auf ein Problem“ läuft bedingt durch Teamgrenzen und Hierarchien nicht von der / dem EntwicklerIn zur Teamleitung der Fachkonzeption, sondern nimmt den Weg über den / die FachkonzepterIn, bzw. die Teamleitung der Entwicklung, welche somit einen Filter darstellen. Bei der Abbildung von Projekten und Aufgaben in (Awareness-)Tools sollten diese also teamübergreifend abgebildet werden, so bleiben Kommunikations- und Awarenessmöglichkeiten bestehen, mittels individuell konfigurierbarer Filter kann die Informationsmenge anschließend auf eine gewünschte Gruppe von Personen eingeschränkt werden.

Konventionen: Konventionen, also ein gemeinsames Verständnis unter den Teammitgliedern wie mit den vorhandenen Systemen gearbeitet wird und werden muss, ist wichtig für die Arbeit mit Informationssystemen, da ansonsten pullbasierte Mechanismen, die auf einer korrekten, und vor allem auch zeitnahen Eingabe von Status- und sonstigen Informationen in den Systemen durch die BenutzerInnen basieren, nicht funktionieren, bzw. falsche oder sogar irreführende Informationen liefern können.

Distanz: Die Distanz zwischen den handelnden Personen ist ein wesentlicher Faktor für Awareness. Sie bestimmt wesentlich darüber, ob direkte Kommunikation möglich ist, und somit wie detailliert der Austausch von Information von statten gehen kann. In der Untersuchung zeigte sich, dass die Erfassung von Informationen in Tools (also eine Form von indirekter Kommunikation) einerseits zeitaufwendig ist und andererseits auch keine Garantie gibt, dass die Information auch tatsächlich zeitnah registriert wird, weshalb daher vor allem in dringenden Fällen mündliche Kommunikation bevorzugt wird.

5.6.2 Tool- und Informationsfaktoren

Transparenz von Information und Kommunikation: Transparente, d.h. für alle einsehbare, Information und Kommunikation zeigt sich in der Untersuchung als Grundlage für die teamübergreifende Zusammenarbeit im Kompositteam. Durch das Setzen des Status und Zuweisung von Aufgaben an andere Personen, gemeinsam mit Awarenessmechanismen, wird eine effiziente Zusammenarbeit zwischen Personen und Teams ermöglicht. Durch Übersichtsmöglichkeiten über Teamaufgaben wird verhindert dass Aufgaben „liegenbleiben“ und ermöglicht dass Arbeitslast gleichmässig verteilt werden kann und dass Verzögerungen rechtzeitig erkannt werden können. Die Abbildung von Aufgaben und Prozessen in Artefakten und Tools stellt eine Kommunikationsgrundlage für die Arbeit an Aufgaben zwischen Personen dar und lässt den Fortschritt in der Bearbeitung von Themen verfolgen. Die Betrachtung der Aufgaben anderer erlaubt es, sich auf kommende Aufgaben einzustellen, und eine Übersicht über bereits weitergereichte Themen und deren Abschluss zu behalten. Das Erfassen von Status- und Hintergrundinformation strukturiert nicht nur die eigene Arbeit, sondern macht diese auch für andere Personen, auch zu späteren Zeitpunkten, nachvollziehbar.

Informationsmenge: Je mehr Daten in einem System erfasst sind, desto schwieriger und aufwendiger wird es, die für eine Person relevanten Informationen aus der Menge zu filtern und zu suchen. In der Untersuchung zeigt sich, dass ein „zu-viel“ an (irrelevanten) Informationen dazu führt, dass Awarenessmechanismen die auf *Subscription* basieren nicht mehr funktionieren und ausgeschaltet oder ausgeblendet werden, da sie durch die Informationsmenge eher störend als hilfreich werden. Ausserdem zeigt sich, dass auf *Polling* basierende Mechanismen wie Suche und Filter zu viele (irrelevante) Ergebnisse liefern, bzw. dass die Konfiguration von Filtern und Suchanfragen einer sehr guten Kenntnis des Informationssystems und der zugrundeliegenden Datenstrukturen bedarf, um zum einen eine überschaubare Ergebnismenge zu bekommen, zum anderen um eigentlich relevante Ergebnisse nicht aus Unkenntnis auszuschliessen. Nicht nur bei der Abfrage, auch bei der Erfassung von (Awareness-)Informationen spielt die Datenmenge eine Rolle, da eine höhere Anzahl an Artefakten aufwendiger zu administrieren ist. In der Untersuchung zeigt sich, dass bei zu vielen gleichzeitigen Aufgaben in JIRA deren Status aus Zeitgründen häufig nicht mehr gepflegt wird, wodurch Dritte diesen Status nicht mehr korrekt erfassen können.

Aufwand / Nutzen: In der Untersuchung zeigt sich, dass der Aufwand für die Erfassung von einfachen, expliziten Attributen in den Tools, welche den Status einer Aufgabe anzeigen, ansich relativ gering ist, jedoch bei einer Vielzahl von (gleichzeitigen) Aufgaben häufig nicht zeitnah aktualisiert wird. Besonders in stressigen Situationen wie Ticketbearbeitung reichen zudem diese Attribute nicht aus, um den Fortschritt der Bearbeitung korrekt wiederzugeben, wodurch in diesen Situationen zumeist direkte Kommunikation verwendet wird, um den Status von Aufgaben abzufragen. Die Erfassung und Auswertung von Informationen in freien Textbeschreibungen und Kommentaren ist zeitaufwendiger, zeigt sich aber als gewinnbringend für die längerfristige Nutzung von Artefakten, als dadurch der Zustand, oder das Zustandekommen eines Status auch im Nachhinein noch nachvollzogen werden kann.

In den Interviews wird festgestellt, dass EntwicklerInnen im Kompositteam in JIRA hauptsächlich ihre eigenen Aufgaben betrachten, bzw. die Filter so konfiguriert sind, dass hauptsächlich die eigenen, zugewiesenen Aufgaben angezeigt werden. Im Gegensatz dazu geben alle befragten Personen an, im Ticketsystem TTS, welches eine einfache, kaum konfigurierbare, sondern bloss sortierbare Übersichtsliste über alle Tickets der Firma bereitstellt, nicht nur die eigenen Tickets anzusehen, sondern ganz bewusst alle Tickets des Teams zumindest überblicksmässig betrachten, und teilweise auch die von anderen Teams. Hier liegt also der Schluss nahe, dass prinzipiell Interesse an den Aufgaben von Teammitgliedern besteht und wenn von den Tools angeboten auch genutzt wird, das aber mit der **Einfachheit der Benutzbarkeit** der Systeme einhergeht, bzw. dass dafür eine gewisse **Einarbeitungszeit** erforderlich ist: Während TTS in der Firma schon jahrelang in Gebrauch ist und außer Sortierungen und einiger vorkonfigurierter Filter (aktive und geschlossene Tickets) keine Konfigurationsmöglichkeit für die BenutzerInnen vorsieht, ist JIRA erst seit Kurzem in Verwendung. Prozesse, die in JIRA abgebildet sind, werden noch laufend angepasst, und die Erstellung von Filtern – welche hauptsächlich die eigenen, momentan zugewiesenen Tasks anzeigen – benötigt Kenntnis des Systems und der darin abgebildeten Prozesse und Parameter. Viele der untersuchten EntwicklerInnen und FachkonzepterInnen geben an, sich noch nicht sehr stark mit den gebotenen Funktionalitäten von JIRA auseinandergesetzt zu haben, und bloss die vorkonfigurierten Filter zu verwenden. Im Gegensatz dazu scheinen sich die TeamleiterInnen wesentlich mehr mit den Filter- und Suchmöglichkeiten auseinandergesetzt zu haben und haben – da sie selbst für die Erstellung und Abbildung der Prozesse verantwortlich sind – auch bessere Kenntnis der verschiedenen Parameter.

Appropriation: Die Möglichkeiten von *Appropriation* (das Ändern, Anpassen und Konfigurieren von Tools), die ein Enterprise 2.0-System wie JIRA bietet, erlauben eine sehr vielseitige Nutzung der Systeme: JIRA in der untersuchten Firma wird nicht nur für teaminterne Zuweisung und Verwaltung von Aufgaben verwendet, sondern ermöglicht durch Anpassung von Workflows, Taskstypen und Attributen die Abbildung von teamübergreifenden Prozessen. Durch Enterprise 2.0-Funktionalitäten wie Suche und Links lässt sich das Tool auch als Dokumentenmanagementsystem verwenden. Diese Vielseitigkeit ist auch nötig, da die Untersuchung zeigt, dass einzelne Teams und handelnde Personen durch ihre Rollen durchaus unterschiedliche Awarenessbedürfnisse haben.

Diese freien Konfigurationsmöglichkeiten bringen aber auch Probleme mit sich. In der Untersuchung zeigt sich, dass, je stärker ein Tool konfiguriert und auf einen bestimmten Zweck oder Prozess zugeschnitten wird, das Tool dadurch unflexibler wird. Bei Änderungen von Prozessen, Vorgehensweisen oder Teamstrukturen müssen eine Vielzahl von Parametern und Filtern geändert werden, durch Erstellung neuer Tasktypen und Attribute vergrößert sich die **Komplexität** der Datenstrukturen wodurch wiederum Such- und Filterfunktionen aufwendiger werden und unter Umständen schlechtere Ergebnisse liefern. Besonders wenn die in einem Tool erfassten Aufgaben und Artefakte über einen längeren Zeitraum verwendet werden müssen – wie das im Kompositteam durch das Vorgehen nach dem Wasserfallmodell und die Verwendung von JIRA als Dokumentenablage der Fall ist – wird bei Änderung der Prozesse entweder die in den Tools vorhandene Information wertlos, oder es müssen Such- und Filterkriterien erweitert werden.

Einheitliche Tools: Im untersuchten Kompositteam werden zum Zeitpunkt der Untersuchung teilweise mehrere Tools für die Abbildung und Administration einzelner Aufgaben und Projekte verwendet: In der Planungsphase wird mit Projektplanungs- und Tabellenkalkulationsprogrammen gearbeitet, später werden daraus JIRA-Tasks – teilweise wiederum separat je Abteilung – erstellt. Die Ursache für diese Vielzahl an verwendeten Tools liegt darin, dass neu eingeführte Tools zusätzlich zu den bestehenden verwendet werden, ohne diese komplett abzulösen, da der Funktionsumfang nicht identisch ist. Dadurch sind Informationen zu einzelnen Aufgaben in mehreren verschiedenen Tools oder Artefakten gleichzeitig erfasst. Dies bedeutet einen erhöhten Administrationsaufwand, da Änderungen in einem Artefakt in den anderen Artefakten (händisch) nachgezogen werden müssen, und es werden auch Awarenessmechanismen behindert, da durch diese Medienbrüche Kommunikation und Übersicht zwischen den Tools verhindert wird. Andererseits kann aber die Erhöhung des Funktionsumfangs in einem Tool auch eine Erhöhung der Komplexität in der Verwendung des Tools bedeuten und somit entgegenstehen zum Faktor Aufwand / Nutzen.

5.7 Modell von Awareness

Awareness im Arbeitsumfeld einer Softwareentwicklungsfirma lässt sich mit einem Modell aus vier Komponenten beschreiben. Diese vier Komponenten entsprechen den in den Abschnitten 5.3 bis 5.6 vorgestellten Kategorien „Awarenessbedürfnisse“, „Awarenessmechanismen“, „Toolverwendung“ und „Faktoren“. Das Modell wurde bereits in Abschnitt 5.2 einführend vorgestellt, und soll an dieser Stelle nochmals abschliessend diskutiert werden. Abb. 30 zeigt das Beziehungsgefüge der vier Kategorien *Awarenessbedürfnisse*, *Awarenessmechanismen*, *Toolverwendung* und *Faktoren* in einer schematischen Übersicht:

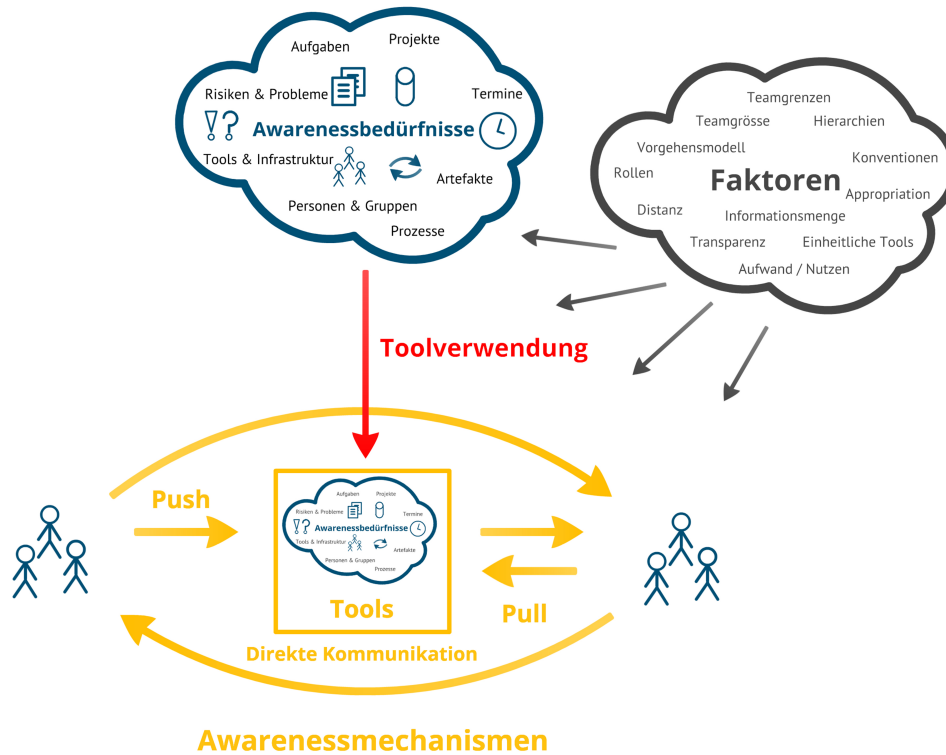


Abb. 30 Modell von Awareness: Beziehungsgefüge von Awarenessbedürfnissen, Awarenessmechanismen, Toolverwendung und Faktoren

Betrachtet man das Thema Awareness, stellt sich als erstes die Frage, worüber man eigentlich Awareness benötigt. Die Kategorie „**Awarenessbedürfnisse**“ umfasst die Inhalte, über die die handelnden Personen Awareness erlangen wollen oder müssen. In der Untersuchung identifizierte Inhalte von Awareness sind Aufgaben, Projekte, Termine, produktions- und administrative Artefakte, Tools und technische Infrastruktur, Personen und Gruppen, Prozesse, Probleme und Risiken. Der Arbeitsauftrag des Untersuchungsfeldes stellt den gemeinsamen Kontext dieser Inhalte dar, allerdings weisen die Inhalte auch untereinander starke **Abhängigkeiten** auf. So gibt es komplexere Entitäten wie Projekte und Aufgaben, die sich wiederum aus anderen (Teil-)Aufgaben zusammensetzen, einer zeitlichen Beschränkung in Form von Terminen unterworfen sind, von bestimmten Personen oder Gruppen mit einer Reihe von Tools und innerhalb von mehr oder weniger formellen und formalisierten Prozessen umgesetzt werden sollen. Administrative Artefakte werden als Hilfsmittel für diese Prozesse verwendet. Als Ergebnis der Umsetzung wird eine Reihe von Produktionsartefakten geschaffen, die wiederum aufeinander Bezug nehmen. Probleme und Risiken die sich aus einzelnen Entitäten ergeben und die Produktion und Kommunikation behindern gilt es zu erkennen, das Ausmaß und Auswirkungen auf andere Inhalte abzuschätzen und zu kommunizieren.

Von Interesse über die Inhalte sind **Ereignisse** wie neue oder geänderte Aufgaben, Artefakte oder Termine, Probleme mit Tools, (kurzfristige) Anpassungen von Prozessen, An- und Abwesenheiten und (geänderte) Zuständigkeiten von Personen. Fragen über die Inhalte und deren **Zustand** werden auch ohne das Vorliegen von konkreten Ereignissen gestellt, da die handelnden Personen Übersicht über ihren Arbeitsbereich behalten müssen. Gefragt wird nicht nur **Übersichtswissen**, sondern auch **Details** zu Hintergrundinformationen bezüglich des Kontextes der einzelnen Entitäten, insbesondere wenn nicht geplante Ereignisse oder Zustände auftreten.

In der Praxis werden eine Reihe von Tools und Kommunikationsmechanismen eingesetzt, um Awarenessinformationen auszutauschen, und letztlich Awarenessbedürfnisse zu befriedigen. Diese reichen von Enterprise 2.0-Tools wie JIRA über „klassische“ Tools wie das in Lotus Notes implementierte Ticketsystem TTS bis hin zu direkter, mündlicher Kommunikation in Gesprächen und Meetings. Die in der Firma zu Verfügung stehenden Tools werden von den einzelnen Teams, abhängig vom Vorgehensmodell und der

Teamgrösse sehr unterschiedlich eingesetzt. Der Einsatzzweck reicht dabei von einer vorwiegend teaminternen Nutzung, über die Abbildung von komplexen teamübergreifenden Prozessen bis hin zur Verwendung als Dokumentenmanagementsystem. Mittels der Tools und Kommunikationsmechanismen lassen sich die Awarenessmechanismen Informationspulling- und Pushing, Subscription und Notification, Monitoring und periphere Awareness umsetzen. Diese ermöglichen es, über den Zustand und Ereignisse von Entitäten in der Arbeitsumgebung informiert zu bleiben, sich Übersicht und Detailwissen über einfache und komplexe Vorgänge im Arbeitsumfeld zu holen.

Die **Formalisierung von Inhalten** in Tools und Artefakten bildet die Grundlage, um Awarenessinformationen zu sammeln und zwischen den handelnden Personen mittels technischer Unterstützung austauschen zu können. Administrative Artefakte wie JIRA-Tasks oder TTS-Tickets machen Aufgaben operationalisierbar, indem sie das Ziel und den Inhalt einer Aufgabe festlegen, den Fortschritt bei der Bearbeitung einer Aufgabe und mit der Aufgabe verknüpfte Entitäten wie Termine, Projekte, Artefakte, Personen, Prozesse und auch Probleme in expliziten und impliziten Attributen und Informationen erfassbar und nachverfolgbar machen. Die Tools stellen Mechanismen zu Verfügung, um Ereignisse wie Änderungen an betroffene und interessierte Personen zu kommunizieren und den Zustand von Aufgaben und Projekten zu visualisieren und auszuwerten. Filter stellen eine wichtige und wesentliche Quelle dar, um Übersicht über eigene Aufgaben zu haben und um über Aufgaben und Themen anderer Personen informiert zu sein. In der Untersuchung zeigt sich, dass die Erstellung von Filtern aufgrund der Vielzahl von Parametern und der abgebildeten Prozesse nicht ganz einfach ist, und eine gute Kenntnis des Systems und der abgebildeten Entitäten benötigt. Zudem müssen Filter auch gepflegt und an sich ändernde Parameter und Inhalte angepasst werden. Ein weiteres Problem bei der Arbeit mit Informationssystemen ist die Tatsache, dass die zugrundeliegenden Daten oft nicht, oder erst spät aktualisiert werden.

Organisatorische Faktoren wie Rollen und Vorgehensmodell bestimmen neben dem eigentlichen Arbeitsauftrag einer Organisation wesentlich darüber, welche Awarenessbedürfnisse die handelnden Personen haben. Die Rolle bestimmt ausserdem darüber ob eine handelnde Person in einer bestimmten Situation eher (Awareness-) Informationsproduzent oder Konsument ist. Teamgrösse, Teamgrenzen, Hierarchien und Konventionen bestimmen die Kommunikationsweisen und Kommunikationsmöglichkeiten, ausserdem ist auch die Informationsmenge als wichtiger Faktor in der Arbeit mit Tools durch die Teamgrösse beeinflusst. Generell zeigt sich die Komplexität bei der Bedienung von Tools als entscheidender Faktor darüber, ob Awareness mittels Tools umgesetzt werden kann. Die Untersuchung zeigt, dass einfach zu benutzende Systeme Awareness über die Aufgaben von Teammitgliedern fördern können.

5.7.1 Verbindung zu bestehenden Modellen von Awareness

Das beschriebene Modell baut auf den in Abschnitt 2.3 vorgestellten Modellen von Awareness auf. Das von Gross und Koch (2007) angeführte Modell für Awarenessinformationsverarbeitung und das Ereignismodell wird um den Begriff des „Zustandes“ erweitert. Das bedeutet, dass in Bezug auf Awareness von Aufgaben und Aufgabenkontext nicht nur Ereignisse von Interesse sind, sondern der Zustand von Aufgaben auch ohne dem Vorliegen von konkreten Ereignissen abgefragt wird.

Abgeleitet aus dem Kontextmodell stellen auch im hier entwickelten Modell die Abhängigkeiten der einzelnen Inhalte untereinander eine wichtige Komponente von Awarenessbedürfnissen dar. Die Untersuchung zeigt, dass bei Ereignissen an einzelnen Entitäten immer auch die abhängigen Entitäten betrachtet werden. Die Formalisierung von Inhalten mit ihrem Kontext in Tools zeigt sich als wichtige Quelle für die Erfassung von Awarenessinformationen. Der von Gross und Koch (2007) angesprochene erhöhte Aufwand in der Bedienung von Tools durch die Eingabe von Kontextinformationen wurde auch in der vorliegenden Untersuchung beobachtet: Detaillierte Angaben zum Zustand von Aufgaben sind über die in den Tools verfügbaren expliziten Attribute häufig nicht abbildbar (z.B. der Zwischenstand einer Implementierung), eine Beschreibung des Zustandes mittels freier Textbeschreibungen lässt Details zu, allerdings ist die Erfassung solcher

Informationen sehr zeitaufwendig, und wird daher besonders in stressigen Informationen nicht durchgeführt. Auch die Erfassung/Erschließung von freien Textbeschreibungen, oder ein Schliessen auf den Zustand von Aufgaben durch implizit vorhandene Informationen (beispielsweise aus der Historie von Aufgaben oder Tickets) benötigt Zeit und Expertise durch die InformationskonsumentInnen. Aus diesem Grund werden in der Praxis detaillierte Informationen besonders in zeitkritischen Situationen bevorzugt mündlich kommuniziert.

Gross und Koch (2007) nennen einige Faktoren wie Privatsphäre und organisatorische Regularien, die für die Arbeit mit Awarenessinformationssystemen beachtet werden müssen. Das hier vorgestellte Modell übernimmt diesen Ansatz und erweitert diese Faktoren um eine Reihe von organisatorischen, tool- und informationsbasierten Faktoren. Der von Gross und Koch (2007) angesprochen Faktor Privatsphäre konnte in der Untersuchung nicht direkt beobachtet werden, organisatorische Regularien und Rahmenbedingungen zeigen sich aber als überaus wichtig dafür, welche Awarenessbedürfnisse bestehen und welche Tools und Awarenessmechanismen eingesetzt werden.

5.7.2 Anwendbarkeit des Modells

In der hier durchgeführten Arbeit wurde eine mittelständische Softwareentwicklungsfirma untersucht. Es ist anzunehmen, dass sich deren Awarenessbedürfnisse, Arbeitsabläufe und Tools beispielsweise von denen von ÄrztInnen und Pflegepersonal in einem Krankenhausbetrieb unterscheiden. Die grundlegenden Parameter aber – dass es bestimmte Awarenessbedürfnisse gibt, die verschiedene Inhalte haben, dass man über deren Zustand und Ereignisse informiert sein muss und dass es verschiedene Mechanismen und Umsetzungsmöglichkeiten gibt, um diese Awarenessbedürfnisse zu befriedigen – könnten jedoch auch auf andere Arbeitsumfelder übertragen werden. Auch viele der identifizierten Faktoren sollten sich auf andere Arbeitskontexte übertragen lassen.

6. Zusammenfassung

Der Begriff *Awareness* in der CSCW-Forschung wird durch technische Neuerungen stetig vorangetrieben. Die Verwendung von Web 2.0-Technologien in Unternehmen – auch bekannt als *Enterprise 2.0* – stellt eine solche Neuerung dar. Das Ziel der vorliegenden Diplomarbeit ist es zu zeigen, wie Enterprise 2.0-Systeme gemeinsam mit klassischen Informationssystemen und Kommunikationsmechanismen im Arbeitsumfeld eingesetzt werden, um Awareness zwischen ArbeitskollegInnen herzustellen. Durch eine theoretische und empirische Analyse soll ein Modell von Awareness im Enterprise 2.0 konstruiert werden, welches darstellt, welche Variablen, Mechanismen und Faktoren bei der Arbeit mit Awarenessinformationssystemen wirken und welche Beziehungen zwischen diesen Variablen und Faktoren bestehen.

Die theoretische Analyse in Kapitel 2 zeigt, dass der Awarenessbegriff sehr breit gestreut ist und für eine Vielzahl von (technischen) Konzepten verwendet wird, die den Begriff unterschiedlich prägen. Gross und Koch (2007) beschreiben verschiedene abstraktere Modelle von Awareness, welche die Erfassung, Verarbeitung und Präsentation von Ereignissen durch technische Systeme darstellen. Eine weitergehende theoretische Analyse dieser Modelle und verschiedener Ausprägungen von Awareness resultiert in drei Betrachtungskategorien, die sich auch in der späteren empirischen Analyse wiederfinden: *Awarenessmechanismen* (2.4) beschreibt verschiedene technische, organisatorische und kommunikative Mechanismen, die den Austausch von Awarenessinformationen ermöglichen. *Inhalt von Awareness* (2.5) betrachtet die eigentliche Information die in Awarenesssystemen ausgetauscht wird: Was ist das Thema des Austausches und welche Fragen werden gestellt. *Faktoren für Awareness* (2.6) sammelt schließlich eine Reihe von Aspekten aus der Literatur über Awareness, die die Inhalte von Awareness und die Arbeit mit Awarenesssystemen beeinflussen. Als Abschluss der theoretischen Analyse wird diskutiert, wie sich Enterprise 2.0-Funktionalitäten für die Umsetzung von Awareness nutzen lassen.

Für die empirische Untersuchung wurde ob der Komplexität des Themas ein qualitativer Forschungsansatz im Stile der Grounded Theory gewählt. Die ethischen Kriterien *Offenheit*, *theoriegeleitet*, *regelgeleitet* und *Schutz der untersuchten Personen* leiteten das Untersuchungsdesign. Die Untersuchung wurde in einer Softwareentwicklungsfirma in der Versicherungsbranche mit ca. 140 MitarbeiterInnen durchgeführt. Zur Datenerhebung wurden in der betreffenden Firma zuerst am Standort in Österreich über einen Zeitraum von ca. sechs Wochen eine Intranetrecherche und teilnehmende Beobachtung durchgeführt. Anschliessend wurden am Standort in Deutschland in einem Zeitraum von eineinhalb Wochen eine Reihe von Experteninterviews mit MitarbeiterInnen aus verschiedenen Abteilungen und Hierarchieebenen durchgeführt. Die erhobenen Daten wurden in Form von Beobachtungsprotokollen, Interviewtranskripten und Screenshots festgehalten. In Kapitel 3 werden die angewandten Untersuchungsmethoden und deren Umsetzung ausführlich beschrieben.

Kapitel 4 enthält die Ergebnisse der Intranetrecherche, welche den Interviews voranging. Darin werden der wirtschaftliche, organisatorische und technische Kontext des Untersuchungsfeldes dargestellt. Zuerst wird kurz die untersuchte Firma und die Branche vorgestellt, gefolgt von einer kurzen Beschreibung des in der untersuchten Projektgruppe verwendeten Softwareentwicklungsprozesses und der damit einhergehenden organisatorischen Aufteilung. Anschliessend werden die wichtigsten in der Firma verwendeten Softwaretools und Technologien beschrieben. Bei den beschriebenen technischen Systemen handelt es sich um jene Tools, die zur alltäglichen Zusammenarbeit zwischen den in der Untersuchung beobachteten und interviewten Personen verwendet werden. Dabei kommen sowohl „klassische“, also intranetbasierte, Informations- und Kommunikationssysteme zur Anwendung, als auch webbasierte Issuetrackingsysteme die der Definition von Enterprise 2.0 entsprechen. Es werden der Aufbau und die wichtigsten Funktionalitäten der Tools besprochen, sowie deren Möglichkeiten zur Umsetzung von verschiedenen Awarenessmechanismen.

In Kapitel 5 werden die in den Interviews und der teilnehmenden Beobachtung erhobenen Daten mittels der Grounded Theory Methodik analysiert und dargestellt. Das Ergebnis

besteht aus einem Modell von Awareness, das aus vier Kategorien *Awarenessbedürfnisse*, *Awarenessmechanismen*, *Toolverwendung* und *Faktoren* besteht.

Die Kategorie *Awarenessbedürfnisse* umfasst mit ihren Subkategorien jene Begriffe, auf welche die in dem untersuchten Projekt handelnden Personen ihre Aufmerksamkeit richten und worüber sie Awareness erlangen wollen und müssen. Die in den Interviews und der teilnehmenden Beobachtung identifizierten Entitäten sind Awareness über Aufgaben, Projekte, Termine, Artefakte, technische Systeme, Personen und Gruppen, Prozesse, Risiken und Probleme. Diese Inhalte von Awareness entstehen aus dem Arbeitskontext der Firma und weisen eine starke Abhängigkeit untereinander auf. Die Untersuchung zeigt, dass nicht nur Ereignisse an bestimmten Entitäten als Awarenessinformation ausgetauscht werden, sondern dass auch ohne dem Vorliegen von Ereignissen der Zustand der Entitäten überblicksmässig und im Detail abgefragt wird.

Die Kategorie *Awarenessmechanismen* umfasst Beobachtungen von Handlungsweisen, die eingesetzt werden, um die verschiedenen Awarenessbedürfnisse zu befriedigen. Interviewaussagen wie „Bescheid sagen“, „Erwarte, dass ich informiert werde“ oder „Regelmässig nachschauen“ können den im Theorieteil dargestellten Awarenessmechanismen Pushing und Pulling oder Subscription zugeordnet werden. Der Austausch von Awarenessinformationen erfolgt entweder in Form direkter Kommunikation, oder indirekt über den Weg eines Tools oder Artefaktes. Die Untersuchung zeigt, dass häufig ein Mix, bzw. eine Abfolge von direkter und indirekter Kommunikation verwendet wird.

In der Beobachtungskategorie *Toolverwendung* wird zuerst die unterschiedliche Nutzung des untersuchten Issuetrackingsystems durch verschiedene Teams dargestellt. Es zeigt sich, dass die vielfältigen Konfigurationsmöglichkeiten des Enterprise 2.0-Systems verschiedene Kooperationsweisen auf teaminterner und teamübergreifender Ebene unterstützen, dass mit zunehmender Komplexität des Team- und Prozessgefüges aber auch die Anforderungen an das Tool steigen. Zur Umsetzung von Awareness mittels technischer Unterstützung ist eine Formalisierung von Awarenessinhalten in den Tools nötig. Das untersuchte Issuetrackingsystem bietet eine Reihe von Attributen, um diese Inhalte und ihre Abhängigkeiten auf explizite oder implizite Weise abzubilden. Dadurch wird das Tool zu einer Informations- und Kommunikationsplattform, die die verschiedenen Awarenessmechanismen ermöglicht. Allerdings treten bei der Toolverwendung auch Probleme auf. Diese liegen einerseits in der großen Informationsmenge und der Komplexität und Vielfalt der abgebildeten Inhalte, die wiederum die Komplexität der Konfiguration und Bedienung der Tools erhöhen. Ein weiteres Problem besteht darin, dass die Datenbasis aus verschiedenen Gründen nicht immer aktualisiert wird, wodurch letztlich die toolvermittelte Gewinnung von Awareness be- oder verhindert wird und somit die direkte Kommunikation ein wichtiger Bestandteil der täglichen Zusammenarbeit bleibt.

In der Kategorie *Faktoren* werden die wichtigsten der in der empirischen Analyse entdeckten Faktoren, welche die Awarenessbedürfnisse der beteiligten Personen, Awarenessmechanismen und deren Umsetzung in Tools beeinflussen, zusammengefasst. Diese Faktoren lassen sich grob in organisatorische Faktoren wie Rollen, Vorgehensmodell oder Teamgrösse und tool-, bzw. informatorische Faktoren wie Informationsmenge oder Aufwand und Nutzen gliedern.

Die vier Kategorien bilden schliesslich ein Modell von Awareness, mit dem sich die Zusammenarbeit von KollegInnen und Teams in einem Ökosystem von Anwendungen beschreiben lässt. Abb. 31 zeigt das Beziehungsgefüge der vier Kategorien *Awarenessbedürfnisse*, *Awarenessmechanismen*, *Toolverwendung* und *Faktoren* in einer schematischen Übersicht.

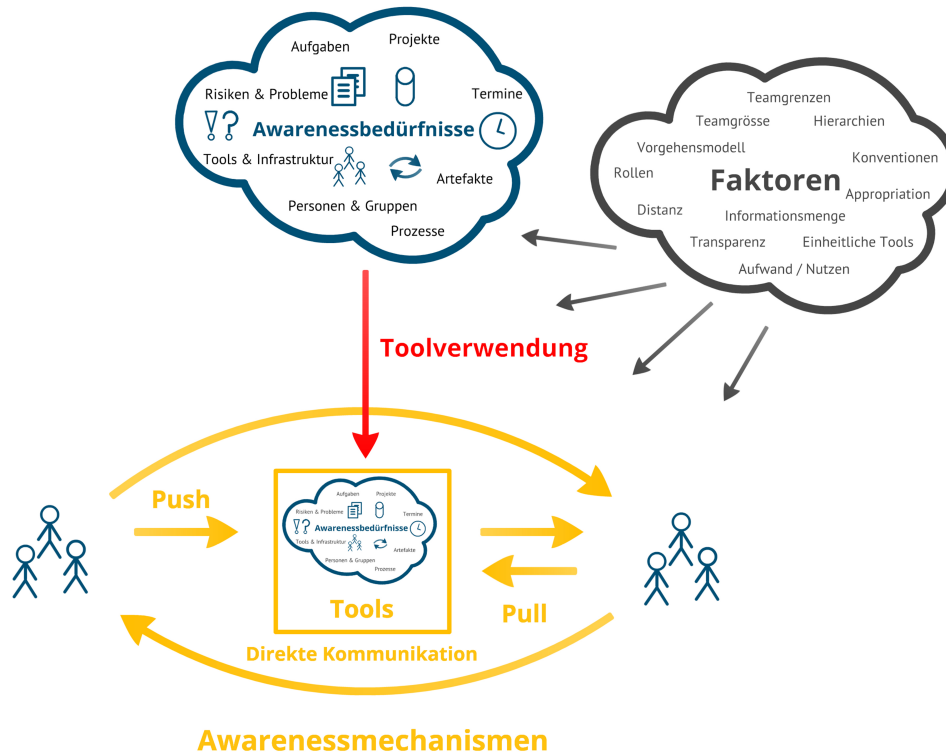


Abb. 31 Modell von Awareness: Beziehungsgefüge von Awarenessbedürfnissen, Awarenessmechanismen, Toolverwendung und Faktoren

Das erarbeitete Modell baut in seinem Ursprung auf den vorhandenen Modellen von Awareness auf, erweitert diese aber in einigen Punkten: Ereignis- und Kontextmodelle werden in der Kategorie Awarenessbedürfnisse durch die *Inhalte von Awareness* verfeinert und um den Begriff des *Zustandes* erweitert. Die Erfassung, Verarbeitung und Präsentation von Ereignissen (und Zustand) wird durch verschiedene *Awarenessmechanismen* und die *Formalisierung von Awarenessinhalten* ausdifferenziert. Die bestehenden Modelle nennen bereits einige *Faktoren*, die bei der der Arbeit mit Awarenesssystemen beachtet werden müssen, diese werden durch organisatorische, tool- und informationsbasierte Faktoren ergänzt.

6.1 Future Work

Der Faktor Distanz konnte in der vorliegenden Arbeit nicht zufriedenstellend untersucht werden. Die Methode der teilnehmenden Beobachtung eignet sich gut, um „Nähe“ zu untersuchen, um klare Aussagen über Distanz zu treffen, benötigt man aber andere Mittel. Durch zeitliche Beschränkungen bei der Durchführung und Auswertung der Interviews konnten keine Aussagen zu dem in 5.4.4 angesprochenen Thema von toolvermittelter, peripherer Awareness – das durch den Faktor Distanz geprägt wird – gemacht werden. Hierzu wären also geeignete Datenerhebungs- und Auswertungsmethoden zu finden.

Die Untersuchung zeigt einige Probleme bei der Umsetzung von Awareness mit Tools auf: Schwierigkeiten in der Verwendung und Konfiguration der Tools, verursacht durch große Informationsmengen, die Komplexität der Tools und der darin abgebildeten Prozesse sowie die Vielzahl von Aufgaben, die die einzelnen MitarbeiterInnen zu erledigen haben bei zeitlichen Beschränkungen. Dies führt dazu, dass die in Awarenessinformationssystemen vorhandenen Daten nicht zeitnah aktualisiert werden, die Auswertung sehr aufwendig wird und bei falscher Konfiguration sogar irreführend sein kann und letztlich Awareness und Übersicht verloren gehen. Auch Treude und Storey (2010) stellen fest, dass viele AnwenderInnen mit der Fülle an Informationen und Funktionalitäten in CDEs überfordert sind und fordern daher übersichtlichere und besser angepasste Awarenessinformationen in kollaborativen Systemen. Eine zukünftige Forschungsaufgabe sollte es also sein, basierend

auf den erarbeiteten Erkenntnissen, verschiedene Verbesserungsmöglichkeiten auszuarbeiten und in der Praxis zu erproben.

Die Untersuchung zeigt, dass organisatorische Faktoren wie die Teamgrösse und das Vorgehensmodell in der Softwareentwicklung großen Einfluss auf Awareness haben können. Das in dieser Arbeit hauptsächlich untersuchte Team war zum Zeitpunkt der Untersuchung nach einem klassischen Wasserfallmodell organisiert. Eine Nachbetrachtung des Untersuchungsfeldes zeigt, dass die Firma mittlerweile organisatorisch nach einer Projektstruktur ausgerichtet wurde und dass vermehrt agile Methoden eingesetzt werden. Eine umfangreichere Untersuchung von agil arbeitenden Teams könnte erörtern, welche Awarenessbedürfnisse dort herrschen, welche Auswirkungen auf die Toolverwendung und Awarenessmechanismen bestehen und ob oder wie sich durch geänderte organisatorische Strukturen die in dieser Arbeit identifizierten Probleme beheben lassen.

6.2 Reflexion

Die Durchführung einer qualitativen, empirischen Studie stellte sich als sehr herausfordernd, aber auch spannend und gewinnbringend dar. Eine Schwierigkeit bestand im Zurechtfinden in der Terminologie der sozialwissenschaftlichen Methoden, da Begriffe wie „Konzepte“, „Kategorie“ oder „Codieren“ in den unterschiedlichen Lehrbüchern und Forschungsmethoden sehr vielfältig verwendet werden.

Danach zeigte sich die Aufbereitung und Auswertung der gesammelten Daten als sehr aufwendig. Die verwendeten Tools zur qualitativen Analyse sind zwar hilfreich bei der Codierung und Indizierung der gesammelten Daten, führen aber besonders bei der verwendeten Methode der Grounded Theory, die die Offenheit von Forschung sehr stark betont, auch leicht zu einer sehr großen Anzahl an Codes und Kategorien. Die Übersicht über Codes, Memos, Kategorien und deren Beziehungsgeflecht zu behalten ist auch mit den verwendeten Tools nicht einfach.

Letztlich musste beim Schreiben des Forschungsberichtes erst ein Weg gefunden werden, die Komplexität in den Daten geeignet zu abstrahieren und zu reduzieren, ursprüngliche Forschungsfragen und neu entdeckte Aspekte in den Daten in Einklang zu bringen oder auch abzugrenzen, und schließlich die Ergebnisse geeignet darzustellen.

7. Bibliografie

Antunes, P., et al. (2010). Awareness checklist: reviewing the quality of awareness support in collaborative applications. Collaboration and Technology - 16th International Conference, CRIWG 2010, Maastricht, The Netherlands, September 20-23, 2010: 202-217.

Aschenbrenner Michael, et al. (2010). Informationsverarbeitung in Versicherungsunternehmen. Springer.

Bardram, J. E. & T. R. Hansen (2004). The AWARE architecture: supporting context-mediated social awareness in mobile cooperation. Proceedings of the 2004 ACM conference on Computer supported cooperative work, ACM. 192-201.

Breuer, F. (2010). Reflexive Grounded Theory - Eine Einführung in die Forschungspraxis. VS Verlag.

Brush, A., et al. (2002). Notification for shared annotation of digital documents. Proceedings of the SIGCHI conference on Human factors in computing systems: Changing our world, changing ourselves, ACM. 89-96.

Brush, A. J. B., et al. (2009). Exploring awareness needs and information display preferences between coworkers. Proceedings of the 27th international conference on Human factors in computing systems, Boston, MA, USA, ACM: 2091-2094.

Cataldo, M., et al. (2006). Identification of coordination requirements: implications for the Design of collaboration and awareness tools. Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work, Banff, Alberta, Canada, ACM: 353-362.

Convertino, G., et al. (2010). Clorg: collective intelligence in organizations. Proceedings of the 16th ACM international conference on Supporting group work, Sanibel Island, Florida, USA, ACM: 355-358.

Convertino, G., et al. (2008). Articulating common ground in cooperative work: content and process. Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems, ACM. 1637-1646.

Convertino, G., et al. (2009). Supporting content and process common ground in computer-supported teamwork. Proceedings of the 27th international conference on Human factors in computing systems, ACM. 2339-2348.

Diekmann, A. (2000). Empirische Sozialforschung - Grundlagen, Methoden, Anwendungen. Rowohlt Taschenbuch Verlag GmbH.

Dix, A., et al. (2004). Human-Computer Interaction. Pearson Education.

Dourish, P. (2003). The appropriation of interactive technologies: Some lessons from placeless documents. Computer Supported Cooperative Work (CSCW) **12**(4): 465-490.

Dourish, P. & V. Bellotti (1992). Awareness and coordination in shared workspaces. Proceedings of the 1992 ACM conference on Computer-supported cooperative work, ACM. 107-114.

Espinosa, A., et al. (2000). Coming to the wrong decision quickly: why awareness tools must be matched with appropriate tasks. Proceedings of the SIGCHI conference on Human factors in computing systems, ACM. 392-399.

Espinosa, J. A., et al. (2007). Team knowledge and coordination in geographically distributed software development. Journal of Management Information Systems **24**(1): 135-169.

- Girtler, R. (1992). Methoden der qualitativen Sozialforschung. Böhlau Verlag - Wien, Köln, Weimar.
- Gläser, J. & G. Laudel (2009). Experteninterviews und qualitative Inhaltsanalyse. VS Verlag für Sozialwissenschaften.
- Gross, T. & M. Koch (2007). Computer-supported cooperative work. Oldenbourg Wissenschaftsverlag.
- Gutwin, C., S. Greenberg & M. Roseman (1996). Workspace awareness in real-time distributed groupware: Framework, widgets, and evaluation. People and Computers: 281-298.
- Hinchcliffe, D. (2007). The state of Enterprise 2.0. [Internet] Letzer Zugriff: 09.07.2013, <http://www.zdnet.com/blog/hinchcliffe/the-state-of-enterprise-2-0/143>
- Holmes, R. & R. J. Walker (2010). Customized awareness: recommending relevant external change events. Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering - Volume 1, Cape Town, South Africa, ACM: 465-474.
- King, A. & K. Lyons (2011). Automatic status updates in distributed software development. Proceeding of the 2nd international workshop on Web 2.0 for software engineering, Waikiki, Honolulu, HI, USA, ACM: 19-24.
- Kuckartz, U. (2007). Einführung in die computerunterstützte Analyse qualitativer Daten. VS Verlag für Sozialwissenschaften.
- Larsson, A., et al. (2010). Engineering 2.0: exploring lightweight technologies for the virtual enterprise. From CSCW to Web 2.0: European Developments in Collaborative Design: 173-191.
- Lim, B. Y., O. Brdiczka & V. Bellotti (2010). Show me a good time: using content to provide activity awareness to collaborators with activityspotter. Proceedings of the 16th ACM international conference on Supporting group work, Sanibel Island, Florida, USA, ACM: 263-272.
- MacLean, A., et al. (1990). User-tailorable systems: pressing the issues with buttons. Proceedings of the SIGCHI conference on Human factors in computing systems: Empowering people, ACM. 175-182.
- MacMillan, J., et al. (2004). Questionnaires for distributed assessment of team mutual awareness. Handbook of human factors and ergonomics methods: 51-51.
- Mark, G. (2002). Conventions and commitments in distributed CSCW groups. Computer Supported Cooperative Work (CSCW) **11**(3): 349-387.
- Mayring, P. (2000). Qualitative Inhaltsanalyse. Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, 1(2), Art. 20. <http://www.qualitative-research.net/index.php/fqs/article/view/1089/2383>
- McAfee, A. P. (2006). Enterprise 2.0: the dawn of emergent collaboration. MIT Sloan management review **47**(3): 21-28.
- Naaman, M., J. Boase & C. H. Lai (2010). Is it really about me?: message content in social awareness streams. Proceedings of the 2010 ACM conference on Computer supported cooperative work, ACM. 189-192.
- Prinz, W., E. Hinrichs & I. Kireyev (2010). Anticipative Awareness in a Groupware System. From CSCW to Web 2.0: European Developments in Collaborative Design. Selected Papers from COOP08. Springer.: 3-20.
- Rogge, K.-E. (1995). Methodenatlas für Sozialwissenschaftler. Springer.

Schmidt, A., et al. (1999). Advanced interaction in context. Lecture Notes in Computer Science Volume 1707, 1999, Springer. 89-101.

Schmidt, K. & T. Rodden (1996). Putting it all together: Requirements for a CSCW platform. Human Factors in Information Technology **12**: 157-175.

Schwabe, G., N. Streitz & R. Unland (2001). CSCW-Kompodium. Springer.

Souza, C. R. d., et al. (2007). Supporting collaborative software development through the visualization of socio-technical dependencies. Proceedings of the 2007 international ACM conference on Supporting group work, Sanibel Island, Florida, USA, ACM: 147-156.

Spriestersbach, A., et al. (2001). Integrating context information into enterprise applications for the mobile workforce - a case study. Proceedings of the 1st international workshop on Mobile commerce, Rome, Italy, ACM: 55-59.

Su, N. M. & G. Mark (2008). Communication chains and multitasking. Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems, ACM. 83-92.

Tam, J. & S. Greenberg (2006). A framework for asynchronous change awareness in collaborative documents and workspaces. International Journal of Human-Computer Studies **64**(7): 583-598.

Tang, D. & J. Birnholtz (2010). Sharing awareness information improves interruption timing and social attraction. Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems, Atlanta, Georgia, USA, ACM: 3685-3690.

Tellioglu, H. (2010). "Coordination 2.0. Using Web-based Technologies for Coordination Support". Proceedings of the International Workshop on Web Intelligence and Virtual Enterprises 2 (WIVE 2010), October 11-13, 2010, Saint-Etienne, France.

Treude, C. & M. A. Storey (2010). Awareness 2.0: staying aware of projects, developers and tasks using dashboards and feeds. Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1, ACM. 365-374.

Turner, T., et al. (2010). Exploring the workplace communication ecology. Proceedings of the 28th international conference on Human factors in computing systems, Atlanta, Georgia, USA, ACM: 841-850.

Veiel, D., J. Haake & S. Lukosch (2009). Extending a shared workspace environment with context-based adaptations. Groupware: Design, Implementation, and Use. Springer Berlin Heidelberg: 174-181.

Warta, A. (2011). Kollaboratives Wissensmanagement in Unternehmen - Indikatoren für Erfolg und Akzeptanz am Beispiel von Wikis. VWH - Verlag Werner Hülsbusch.

Appendix A – Einverständniserklärung Firma

Technische Universität Wien

Einverständniserklärung zur Teilnahme an einem Forschungsprojekt

Projektinformation:

An der Technischen Universität (TU) Wien wird eine Diplomarbeit zum Thema „Awareness im Enterprise 2.0“ durchgeführt. *Awareness* wird in der Forschung über computerunterstützte kollaborative Arbeit („CSCW“) als „das Gewährsein der Anwesenheit und des Verhaltens anderer Gruppenmitglieder“¹³ definiert. Ein klassisches Beispiel für ein System mit Awarenessinformation stellen Instant Messaging Programme dar, die Informationen über den Onlinestatus von Chatpartnern anzeigen. Web 2.0-Technologien stellen ein neues Beispiel für eine Technologie dar, die potentiell fähig ist, Awarenessinformationen zu generieren und anzuzeigen. Die Verwendung von solchen Technologien in Unternehmen wird in McAfee (2006) auch als *Enterprise 2.0* bezeichnet. Moderne webbasierte Issuetracking- und Projektmanagement Systeme wie *JIRA*¹⁴ oder *TRAC*¹⁵ erfüllen mit ihren Anwendungsmöglichkeiten und ihrer technologischen Basis die Definition von Enterprise 2.0, und stellen auch mehrere Arten von Awarenessinformationen gleichzeitig zur Verfügung. In der Unternehmenspraxis werden üblicherweise beide Formen von Awareness-Informationssystemen – also klassische und Enterprise 2.0 – verwendet. In der Forschungsliteratur gibt es allerdings wenige empirische Daten über die tatsächliche Verwendung und Auswirkung von Awarenessinformationen in Enterprise 2.0-Systemen: zum einen fehlen anwenderzentrierte Studien, welche die Verwendung und Auswirkungen mehrerer Arten von Awareness in einem Enterprise 2.0-System gemeinsam untersuchen, zum anderen fehlen vergleichende Daten über die gemeinsame Verwendung von „klassischen“ und Enterprise 2.0 Awareness-Informationssystemen. Es gibt keine Definition davon, welche Art von Awareness durch die Verwendung mehrerer Arten von Awarenessinformation in einem Enterprise 2.0 Informationssystem entsteht.

Mit dieser Diplomarbeit soll die Verwendung von Web 2.0-Technologien, welche Awarenessinformationen anzeigen und die in Unternehmen eingesetzt werden, untersucht werden. Ein besonderer Fokus soll dabei auf der Verwendung von webbasierten Issuetracking- und Projektmanagementtools in grossen Unternehmen mit mehreren, verteilten Teams liegen. Andere webbasierte Kommunikations- und Awarenessanwendungen, die in Unternehmen eingesetzt werden können, wie beispielsweise *Facebook*¹⁶, sollen ebenfalls Eingang in die Untersuchung finden. Die Verwendung von „klassischen“ Awareness-Informationssystemen wie desktopbasierten Chatprogrammen soll dem gegenübergestellt werden.

Als Grundlage für diese Diskussion sollen anhand einer konkreten Unternehmenssituation empirische Daten über die Verwendung und Auswirkungen von Awarenessinformation durch und auf AnwenderInnen gesammelt, und quantitativ und qualitativ ausgewertet werden. Daraus soll sich ein Modell von Awareness im Enterprise 2.0 ergeben, das definiert, welche Arten von Awarenessinformationen von den neuen Technologien angeboten werden, welche Art von Awareness dadurch entsteht, und wie sie sich von bisherigen Definitionen und Spielarten unterscheidet, bzw. weiterentwickelt hat.

Forschungsmethoden:

Für die Datenerhebung des empirischen Teiles der Diplomarbeit sollen folgende Techniken angewendet werden: Teilnehmende Beobachtung: Mittels einer teilnehmenden Beobachtung sollen die Arbeitsweise, verwendete Tools sowie konkrete Anwendungsfälle dokumentiert werden. Die Beobachtung soll oder kann über mehrere Iterationen hin vorgenommen werden. Beobachtet werden sollen Individuen an ihrem eigenen Arbeitsplatz, aber auch beispielsweise Teams bei Besprechungen.

Interviews: Interviews mit einzelnen AnwenderInnen sollen Aufschluss über deren Arbeitsweise geben. Beobachtungen oder Unklarheiten aus der teilnehmenden Beobachtung können eingehend besprochen werden.

¹³ Schwabe, G., Streitz, N. and Unland, R., (2001). *CSCW-Kompendium*. Springer, p. 184.

¹⁴ www.atlassian.com/jira

¹⁵ <http://trac.edgewall.org/>

¹⁶ www.facebook.com

Interviews mit Projekt- oder IT-Verantwortlichen können Aufschluss über die Beweggründe für das Einführen bestimmter Technologien bringen.

Fragebögen: Fragebögen eignen sich, eine größere Zahl von BenutzerInnen, bei der einzelne Interviews zu zeitaufwändig wären, zu befragen und im Fall von nicht-offenen Fragen quantitativ auszuwerten. Fragen könnten sich nach der quantitativen Nutzung von bestimmten Programmen (z.B. „benutze ich häufig/selten/nie“), oder aber auch nach der Güte von bestimmten Anwendungen (z.B. „finde ich sehr übersichtlich/übersichtlich/unübersichtlich“) richten.

Intranet-Recherche: Im Intranet der Firma könnte Information über die Organisationsstruktur herausgefunden werden.

Projektbearbeitung:

Die Diplomarbeit wird an der Technischen Universität (TU) Wien, am Institut für Gestaltungs- und Wirkungsforschung, Arbeitsgruppe Multidisziplinäres Design, unter Betreuung von Assoc. Prof. Dipl. Ingⁱⁿ. Drⁱⁿ. Hilda Tellioglu, von Hrn. Simon Diesenreiter durchgeführt.

Assoc. Prof. Dipl. Ingⁱⁿ. Drⁱⁿ. Hilda Tellioglu

Email: [hilda.tellioglu\[at\]tuwien.ac.at](mailto:hilda.tellioglu[at]tuwien.ac.at)

Institut für Gestaltungs- und Wirkungsforschung, Arbeitsgruppe Multidisziplinäres Design

Argentinierstr. 8 / 187, 1040 Wien

Simon Diesenreiter Bakk. techn.

Email: [e9926534\[at\]student.tuwien.ac.at](mailto:e9926534[at]student.tuwien.ac.at)

Lorenz Bayer Platz 9/12, 1170 Wien

Einverständniserklärung:

Die Firma **Versicherungs GmbH** erklärt sich dazu bereit, dass Simon Diesenreiter im Rahmen seiner Diplomarbeit Daten über die **Versicherungs GmbH** erheben darf. Diese empirische Untersuchung umfasst folgende Tätigkeiten und Datenerhebungen:

Teilnehmende Beobachtung: Anwesenheit im Büro der **Versicherungs GmbH** in **Österreich**. Erstellung von Beobachtungsprotokollen und Fotos über die Arbeits- und Kommunikationsweise von MitarbeiterInnen und Teams.

Interviews mit MitarbeiterInnen: Erstellung von Interviewprotokollen und Screenshots der Bildschirmarbeitsplätze.

Fragebögen: verschicken von Fragebögen an MitarbeiterInnen.

Intranet-Recherche: Sammeln von Informationen über die Firmen-, Team- und Organisationsstruktur.

Sämtliche Beobachtungen, Interviews und Befragungen werden nur nach Absprache und auf freiwilliger Basis mit den betroffenen MitarbeiterInnen durchgeführt. Namen von InterviewpartnerInnen oder unternehmenskritische Daten in Mitschriften oder Bildmaterial werden anonymisiert. Alle gesammelten Daten werden unter Wahrung des Betriebsgeheimnisses der **Versicherungs GmbH** vertraulich behandelt, und nicht an Dritte weitergeben. Die Daten dienen ausschließlich der wissenschaftlichen Forschungsarbeit. Das Ergebnis der Forschungsarbeit wird in Form einer Diplomarbeit und eines öffentlichen Vortrages – mit ausschließlich anonymisierten Daten – veröffentlicht.

Einverständnis zu Erhebung und Verarbeitung wie beschrieben:

Ort, Datum

Unterschrift **Versicherungs GmbH**

Die Daten werden wie beschrieben erhoben und verarbeitet:

Ort, Datum

Unterschrift Simon Diesenreiter

Die Firma **Versicherungs GmbH** erklärt sich weiters dazu bereit, die Verarbeitung der im Rahmen der Diplomarbeit erhobenen Daten in anonymisierter Form unter Wahrung des Betriebsgeheimnisses der

Versicherungs GmbH für weitere Forschungsarbeiten am Institut für Gestaltungs- und Wirkungsforschung, Arbeitsgruppe Multidisziplinäres Design, zu erlauben.

Einverständnis zu Erhebung und Verarbeitung wie beschrieben:

Ort, Datum

Unterschrift **Versicherungs GmbH**

Die Daten werden wie beschrieben verarbeitet:

Ort, Datum

Unterschrift Institut für Gestaltungs- und Wirkungsforschung

Appendix B – Projektinformation MitarbeiterInnen

Technische Universität Wien

Information für Interviewpartner im Forschungsprojekt „Awareness im Enterprise 2.0“

Projektinformation:

An der Technischen Universität (TU) Wien wird eine Diplomarbeit zum Thema „Awareness im Enterprise 2.0“ durchgeführt. *Awareness* wird in der Forschung über computerunterstützte kollaborative Arbeit („CSCW“) als „das Gewährsein der Anwesenheit und des Verhaltens anderer Gruppenmitglieder“¹⁷ definiert. Ein klassisches Beispiel für ein System mit Awarenessinformation stellen Instant Messaging Programme dar, die Informationen über den Onlinestatus von Chatpartnern anzeigen. Web 2.0-Technologien stellen ein neues Beispiel für eine Technologie dar, die potentiell fähig ist, Awarenessinformationen zu generieren und anzuzeigen. Die Verwendung von solchen Technologien in Unternehmen wird in McAfee (2006) auch als *Enterprise 2.0* bezeichnet. Moderne webbasierte Issuetracking- und Projektmanagement Systeme wie *JIRA*¹⁸ erfüllen mit ihren Anwendungsmöglichkeiten und ihrer technologischen Basis die Definition von Enterprise 2.0, und stellen auch mehrere Arten von Awarenessinformationen gleichzeitig zur Verfügung. In der Unternehmenspraxis werden üblicherweise beide Formen von Awareness-Informationssystemen – also klassische und Enterprise 2.0 – verwendet.

Mit dieser Diplomarbeit soll die Verwendung von Web 2.0-Technologien, welche Awarenessinformationen anzeigen und die in Unternehmen eingesetzt werden, untersucht werden. Ein besonderer Fokus soll dabei auf der Verwendung von webbasierten Issuetracking- und Projektmanagementtools wie beispielsweise *JIRA* in grossen Unternehmen mit mehreren, verteilten Teams liegen. Die Verwendung von „klassischen“ Awareness-Informationssystemen wie desktopbasierten Chatprogrammen soll dem gegenübergestellt werden.

Anhand einer konkreten Unternehmenssituation sollen empirische Daten über die Verwendung und Auswirkungen von Awarenessinformation durch und auf AnwenderInnen gesammelt, und quantitativ und qualitativ ausgewertet werden. Daraus soll sich ein Modell von Awareness ergeben, das definiert, welche Arten von Awarenessinformationen von den neuen Technologien angeboten werden, welche Art von Awareness dadurch entsteht, wie sie sich von bisherigen Definitionen und Spielarten unterscheidet, bzw. weiterentwickelt hat, und wie sich diese Awareness letztlich auf die Arbeit in einem Unternehmen auswirkt.

Forschungsmethode:

Mittels Interviews mit MitarbeiterInnen in einem Softwareunternehmen sollen Informationen über die Verwendung und Auswirkungen von webbasierten Issuetracking- und Projektmanagementsystemen in einem Unternehmen gesammelt werden.

Zur späteren Analyse der Interviews sollen Tonaufzeichnungen, Bildschirmvideos und Screenshots angefertigt werden. Die Tonaufzeichnungen werden für die Analyse transkribiert und anonymisiert. Screenshots werden anonymisiert. Bildschirmvideos werden für die Analyse gesichtet, und zur Erstellung von anonymisierten Screenshots verwendet.

Datenschutz und Veröffentlichung:

Sämtliche Beobachtungen, Interviews und Befragungen werden nur nach Absprache und auf freiwilliger Basis mit den betroffenen MitarbeiterInnen durchgeführt.

Ton- oder Videoaufzeichnungen werden für die Analyse und zur Erstellung von Transkripten und Screenshots ausschließlich durch Simon Diesenreiter gehört und gesichtet, und nicht an Dritte weitergegeben. Die weitere

¹⁷ Schwabe, G., Streitz, N. and Unland, R., (2001). *CSCW-Kompodium*. Springer, p. 184.

¹⁸ www.atlassian.com/jira

Verarbeitung und eventuelle Veröffentlichung von Ton- oder Videoaufzeichnungen erfolgt ausschließlich in Form von Transkripten und Screenshots die aus diesen Aufzeichnungen erstellt werden.

Sämtliche Ton- und Videoaufzeichnungen werden nach Fertigstellung der Transkripte und Screenshots, und spätestens mit Ende Dezember 2011, datenschutzgerecht nach den Empfehlungen des Bundesamtes für Sicherheit in der Informationstechnik¹⁹ durch mehrmaliges komplettes Überschreiben mit Zufallszahlen gelöscht. Die Firma **Versicherungs GmbH** wird zusätzlich schriftlich per Email über die Löschung informiert. Namen oder unternehmenskritische Daten in Mitschriften oder Bildmaterial werden anonymisiert. Alle gesammelten Daten werden unter Wahrung des Betriebsgeheimnisses der **Versicherungs GmbH** vertraulich behandelt, und nicht an Dritte weitergeben. Die Daten dienen ausschließlich der wissenschaftlichen Forschungsarbeit.

Die Ergebnisse der Untersuchung werden in Form einer Diplomarbeit und eines öffentlichen Vortrages – mit ausschließlich anonymisierten Daten – veröffentlicht. Die Geschäftsleitung der Firma **Versicherungs GmbH** hat einer solchen Veröffentlichung unter Wahrung des Betriebsgeheimnisses der Firma **Versicherungs GmbH** zugestimmt.

Die Firma **Versicherungs GmbH** hat sich weiters dazu bereit erklärt, die Verarbeitung der im Rahmen der Diplomarbeit erhobenen Daten in anonymisierter Form unter Wahrung des Betriebsgeheimnisses der Firma **Versicherungs GmbH** für weitere Forschungsarbeiten am Institut für Gestaltungs- und Wirkungsforschung, Arbeitsgruppe Multidisziplinäres Design, zu erlauben.

Projektbearbeitung:

Die Diplomarbeit wird an der Technischen Universität (TU) Wien, am Institut für Gestaltungs- und Wirkungsforschung, Arbeitsgruppe Multidisziplinäres Design, unter Betreuung von Assoc. Prof. Dipl. Ingⁱⁿ. Drⁱⁿ. Hilda Tellioglu, von Hrn. Simon Diesenreiter durchgeführt.

Assoc. Prof. Dipl. Ingⁱⁿ. Drⁱⁿ. Hilda Tellioglu

Email: [hilda.tellioglu\[at\]tuwien.ac.at](mailto:hilda.tellioglu[at]tuwien.ac.at)

Institut für Gestaltungs- und Wirkungsforschung, Arbeitsgruppe Multidisziplinäres Design
Argentinierstr. 8 / 187, 1040 Wien

Simon Diesenreiter Bakk. techn.

Email: [e9926534\[at\]student.tuwien.ac.at](mailto:e9926534[at]student.tuwien.ac.at)

Lorenz Bayer Platz 9/12, 1170 Wien

Sonstiges:

Die Zeit für die Interviews kann in der Zeiterfassung unter „Weiterbildung“ mit dem Kommentar „Diplomarbeit Simon Diesenreiter“ verbucht werden.

Die Informationen zu „Datenschutz und Veröffentlichungen“ wurden vom Datenschutzbeauftragten der Firma **Versicherungs GmbH** geprüft und freigegeben.

¹⁹ <https://www.bsi.bund.de/ContentBSI/grundschutz/baustein-datenschutz/html/m07015.html>

Appendix C – Einverständniserklärung MitarbeiterInnen

Technische Universität Wien

Einverständniserklärung für Interviewpartner im Forschungsprojekt „Awareness im Enterprise 2.0“

Die TU Wien verpflichtet sich zur Einhaltung der Grundsätze guter wissenschaftlicher Praxis²⁰. Zu diesen Grundsätzen zählt insbesondere der Schutz von TeilnehmerInnen an Forschungsprojekten. In diesem Sinne soll sie das Dokument „Information für Interviewpartner im Forschungsprojekt „Awareness im Enterprise 2.0““ (im Folgenden „Projektinformation“ genannt) über den Zweck der Forschungsarbeit und des Interviews, über die dabei gesammelten Daten und deren Verarbeitung, sowie über geplante Veröffentlichungen informieren.

Mit ihrer Unterschrift dieser Einverständniserklärung bestätigen sie, dass sie vor Beginn des Interviews das Dokument „Projektinformation“ erhalten haben, und ausreichend Zeit hatten es zu lesen und zu bedenken.

Insbesondere verpflichtet sich auch der Interviewer mit seiner Unterschrift, ihre Daten vertraulich, und so wie in dem Dokument „Projektinformation“ beschrieben, zu behandeln.

Ich bin damit einverstanden, dass (zutreffendes bitte ankreuzen)	Ja	Nein
... eine Tonaufzeichnung des Interviews angefertigt, und wie in der Projektinformation beschrieben verarbeitet wird		
... Screenshots von meinem JIRA-Account angefertigt, und wie in der Projektinformation beschrieben verarbeitet werden		
... ein Bildschirmvideo von meinem JIRA-Account angefertigt, und wie in der Projektinformation beschrieben verarbeitet wird.		

Einverständnis zu Erhebung und Verarbeitung wie im Dokument „Projektinformation“ vereinbart:

Ort Datum Name des Interviewpartners Unterschrift

Die Daten werden wie im Dokument „Projektinformation“ vereinbart erhoben und verarbeitet:

Ort Datum Unterschrift Simon Diesenreiter

²⁰ http://www.tuwien.ac.at/dle/recht/sonstige_informationen/code_of_conduct/

Appendix D – Interviewleitfaden

Zuerst: einige Fragen unabhängig von JIRA

- Zur Person, Rolle, Aufgabe im Team, Kooperationen, Tools:
 - „Was ist deine Aufgabe hier im Team“
 - Aufgaben im Großen (z.B. Teamleiter, Entwickler)
 - Aufgaben im Kleinen (z.B. Termine vereinbaren, Implementieren, QSen, Tickets bearbeiten, FK-Klärung mit Kunde, FK-Klärung mit Entwickler, etc.)
 - „Mit wem arbeitest du dabei zusammen?“
 - Zimmer
 - Team (Komposit)
 - Teamübergreifend
 - **Österreich? / Süddeutschland?**
 - Wie unterschiede sich die Kommunikation / Zusammenarbeit mit Kollegen im selben Büro, Kollegen in anderen Räumlichkeiten im Haus, und Kollegen in **Norddeutschland / Österreich?**
 - Wie organisierst du deinen Arbeitstag, welche Programme verwendest du dazu?
 - Lotus Notes, Sametime, JIRA,
 - „Hast du Notes / Sametime immer geöffnet?“
 - „Wie oft siehst du in JIRA?“
 - „Welche Tools verwendest du sonst?“
 - Über welche Themen wird kommuniziert (formell / informell) – mit wem?
- Awareness-Bedürfnisse, Awareness allgemein
 - „Interessierst du dich dafür, was deine Kollegen im Zimmer machen? Hörst du z.B. mit was die telefonieren, und schaltest du dich gegebenenfalls ein?“
 - „Beteiligst du dich zB an Diskussionen die deine KollegInnen führen“
 - „Spielt es für dich eine Rolle, woran deine KollegInnen im Projekt gerade arbeiten?“
 - Vorgelagerte Teams
 - Nachgelagerte Teams
 - Im gleichen Team
 - **„Wie erfährst du darüber?“**
 - „Spielt es für dich eine Rolle, wie deine KollegInnen ausgelastet sind / Probleme mit einem bestimmten Thema / Ticket haben?“
 - Vorgelagerte Teams
 - Nachgelagerte Teams
 - Im gleiche Team
 - **„Wie erfährst du darüber?“**
 - „Teamwork-Awareness / „Social Awareness“:
 - Ort / Onlinestatus
 - Z.B. sieht man in Sametime nach ob KollegIn anwesend bevor man anruft?
 - Stimmung von KollegInnen
 - Probleme im Projekt / in der Zusammenarbeit
 - **„Wie erfährst du darüber?“**
 - Artefakte: „Wie wichtig ist es für dich über bestimmte Dokumente am Laufenden zu sein?“
 - Dokumente (z.B. FKs)
 - Cruise Controll, Trunk, Branch, ... → **CODE**
 - Tickets
 - JIRA-Tasks
 - **„Wie erfährst du darüber?“**

Fragen speziell zu JIRA / Software

- Vergleich Sametime
 - „Sind die Leute die du in JIRA siehst die selben, wie jene die du in der Kontaktliste in Sametime hast?“
 - Wie oft am Tag siehst du in Sametime-Kontaktliste?
 - Verwendest du Sametime-Kontaktliste um über den Status von Kollegen Bescheid zu wissen?
- Vergleich TTS

- „Wo siehst du die Unterschiede zwischen TTS und JIRA?“
- „Was sind Vorteile und Nachteile eines jeden Systems?“
- „Wie geläufig sind dir JIRA-IDs im Vergleich zu TTS-IDs / „Ticketnummern““
- MyLyn / Eclipse: (Nur Entwickler)
 - Wird MyLyn (konsequent) verwendet?
 - Immer geöffnet?
 - Wechsel zwischen Tasks?
 - Anhängen des Kontextes?
 - Werden private TODOS / Tasks in MyLyn angelegt?
 - Werden private TODOS / Tasks in JIRA angelegt?

→ Einloggen in JIRA

- Dashboard: selbst konfiguriert oder vorgefertigtes?
 - Welche „Widgets“ werden verwendet?
 - Diagramme, Filter, Aktivitäten-Stream, ...
 - Warum oder wozu verwendest du bestimmte Widgets
 - Welche Filter werden verwendet? Default oder eigene? Wozu eigene?
- Tasks:
 - Wie behältst du Übersicht über Tasks? → Navigation
 - Nur innerhalb von JIRA?
 - Bookmarks im Browser?
 - Email-Benachrichtigungen?
 - Bedienelemente:
 - Oben: „Dashboards“ / „Projekte“ / „Vorgänge“: werden diese Buttons tatsächlich benützt?
 - Wird „Überwachen“-Funktion verwendet?
 - Wird „Aktivität“ (unten, zeigt „Kommentare“, „Änderungshistorie“) verwendet, angesehen? Warum sieht man sich das an?
 - Wird Kommentarfunktion statt Email für Zusammenarbeit verwendet?
 - Fälligkeiten, Aufwandsschätzung:
 - Werden Fälligkeiten immer eingehalten?
 - Wie akkurat ist die angegebene Aufwandsschätzung?
 - Wie wird vorgegangen falls Fälligkeiten / Aufwände nicht eingehalten werden können?
 - → welche Kommunikationswege (Email, Telefon, etc.)
 - Wer ist für Administration von Aufwänden / Zeitabschätzung verantwortlich?
 - Was passiert mit geschlossenen Tasks:
 - → Dokumente in JIRA:
 - Wie werden diese Tasks / Dokumente wiedergefunden? (zB um bestimmtes FK wiederzufinden?)
 - Was passiert bei Meilensteinwechsel mit Tasks die nicht geschlossen wurden, bzw. die für später „aufgehoben“ werden sollen (z.B. VVs)?
 - Tasks anderer Leute:
 - Sieht man sich die an?
 - Wenn ja: welche Tasks sind das (eigenes Team, andere Teams, eigenes Projekt? Vorgesetzte – Untergebene – gleiche Ebene?)?
 - Was genau sieht man sich dabei an? (Auslastung?)
 - Welche Infos fehlen?

Abschliessend:

- Präferenzen bei den Tools?
- Eigene Meinung zu JIRA?
- Anmerkungen?
- Fragen an Interviewer?
- Feedback für Interviewer?
- Vielen Dank!

Appendix E – Teilnehmende Beobachtung

Anmerkungen zur Tabelle: Aus Gründen der Übersichtlichkeit wurden hier die Spalten für Datum und Uhrzeit, Tags / Markierungen und Kommentare entfernt. Die grauen Balken zwischen den Einträgen markieren einen neuen Tag. Farblich hinterlegte Fall-IDs markieren Teambesprechungen.

Fall-ID	Beschreibung
1	ENTW_OE_2 : DB funktioniert nie, schemas müssen immer per hand eingespielt werden; Workspace ist langsam, funktioniert nicht.
2	ENTW_OE_3 : „weißt du wer fachkonzept für hausrat schreibt?“
3	ENTW_OE_3 (an alle) „wer schaut sich denn [Komponente auf CruiseControl] an?“ ENTW_OE_1 „im prinzip ich, aber ich hab mir das schon angeschaut und das ist ein Problem wo wir nix dafür können“
4	ENTW_OE_1 : „wer ist noch mal von Releasmanagement?“ ENTW_OE_2 : „ RMM_2 , RMM_3 ... [Kürzel für diese Usergruppe in Lotus Notes]“
5	Situation: Donnerstag Vormittag, „Eincheckdeadline“ war Mittwoch Nachmittag (gestern).
5A	ENTW_OE_TL [geht zu ENTW_OE_1 und ENTW_OE_4 (reden ohnehin gerade miteinander)-> mündlich]: „ ENTW_DE_TL_STV_2 hat gerade angerufen, der ist gerade beim Kunden in Süddeutschland , die haben einen Fehler bei [Funktionalität] gefunden, das funktioniert nicht mehr so wie früher. Sie werden zwei TTs eröffnen und euch zuweisen. Die TT-Nummern sind xxxx und xxxx.. Dann haben die beiden jetzt mal Priorität“
5B	ENTW_OE_1 und ENTW_OE_4 machen sich an die Arbeit (an ENTW_OE_4s Rechner), TT-Beschreibung ist in ihren Augen schlecht, und unklar woher der Fehler kommt. TT-Beschreibung (Screenshots) werden ausgedruckt, TT-Nummern o.Ä. werden noch über Chat von ENTW_OE_4s Rechner an ENTW_OE_1s Rechner geschickt (für Verwendung eines bestimmten Tools das nur auf ENTW_OE_1s Rechner ist?). Anschliessend gemeinsam Arbeiten an ENTW_OE_4 -Rechner. Nach einer Weile versuchen sie ENTW_DE_TL_STV_2 telefonisch zu erreichen, erwischen ihn nicht. Machen weiter mit Fehleranalyse. Nach einer Weile Anruf von ENTW_OE_4 an ENTW_OE_3 (dieser ist gerade in Norddeutschland wegen fachlicher Frage. ENTW_OE_4 erklärt ENTW_OE_3 anhand eines Beispiels (dem TT?) die Ausgangslage und fragt wie das verarbeitet werden muss. Nach einer Weile ist Telefonat beendet, ENTW_OE_4s Fazit (zu ENTW_OE_1 , der daneben sitzt): „dieser Fall ist im Produktkern nicht vorgesehen“. ENTW_OE_4 geht rüber zu ENTW_OE_TL und sagt auch dem sein Fazit. Danach kommt er wieder zurück zu ENTW_OE_1 . Sie überlegen und diskutieren weiter welche Werte rauskommen sollten.
5C	ENTW_DE_TL_STV_2 ruft an, ENTW_OE_4 geht ran. ENTW_OE_4 : "Punkt eins: das hat nie funktioniert, Punkt zwei: wird von Produktkern nicht unterstützen, aber wir haben jetzt eine Idee wie man es machen könnte. Aber wir müssen hier das FK anpassen, weil es gibt kein eigenes Kapitel dafür. Es steht zwar was drin, aber das ist total verstreut, sollte geordnet werden, und auch an den Kunden kommuniziert werden." [ENTW_DE_TL_STV_2 spricht...]. ENTW_OE_4 erklärt nochmal den Fehler und wie das zustande kommt. [...diskutieren weiter...]. ENTW_OE_4 -Frage "wo ist ENTW_DE_TL ?", ENTW_OE_4 -Frage "wie sollen wir das einchecken", ENTW_OE_4 -Frage "kannst du ankündigen lassen dass FK geändert werden muss? sonst kann ich das auch machen" "OK, danke"
5D	ENTW_OE_4 und ENTW_OE_1 machen weiter. ENTW_OE_1 (nach einer Weile): "und wie sollen wir jetzt vorgehen?" ENTW_OE_4 : " du machst das..., ich mach das..., das so und so...". ENTW_OE_4 : " ok, ich schick dir das dann", ENTW_OE_1 geht zurück zu seinem Platz.
5E	ENTW_OE_4 verlässt auch seinen platz, trifft ENTW_OE_TL am Gang: ENTW_OE_TL "und? was war?" ENTW_OE_4 erklärt kurz vorgehensweise (FK muss geändert werden). ENTW_OE_TL "und was ENTW_DE_TL_STV_2 dazu?" ENTW_OE_4 : "für den ist das ok" (+ weiteres erklären der Vorgangsweise).
5F	Ende. Zeitrahmen: ca 1-1,5h
6	Uhrzeit: 12:30,
6A	ENTW_OE_4 von Arbeitsplatz zu ENTW_OE_1 : (liest Fehlerbeschreibung eines neuen TTs vor), ENTW_OE_1 : "was? was soll das?" ENTW_OE_4 : "ja das können sie sich abschminken, weil das ist ja von Produktebene nicht schreibbar", "Können TT ev ENTW_OE_3 zuweisen"
6B	ENTW_OE_4 versucht ENTW_OE_3 anzurufen, erwischt ihn nicht
6C	ENTW_OE_4 + ENTW_OE_1 reden weiter übers Problem

6D	ENTW_OE_3 ruft zurück, ENTW_OE_4 : "ein noch grösseres Problem, (haha...), TT-Nr ist xxxxxx" [...warten..] ENTW_OE_4 erklärt Fehler/erwartetes Ergebnis weiter.... ENTW_OE_3 und ENTW_OE_4 sprechen weiter am Telefon. ENTW_OE_1 (steht neben ENTW_OE_4) schaltet sich ab und zu dazwischen. ENTW_OE_4 "gut, ich seh mir das an".
6E	ENTW_OE_1 geht zurück zum Platz, ruft rüber zu ENTW_OE_4 : "Datenbank ist gerade nicht verfügbar, achso, moment ich muss das noch konfigurieren..." ENTW_OE_1 : "wie heisst 3.0er Schema?" ENTW_OE_4 "da brauchst du 2.9er Schema!" reden weiter (über Tische hinweg) über Schemas und wo bestimmte Getter sind. ENTW_OE_1 verlässt Platz und geht zu ENTW_OE_4 rüber, besprechen Codestelle (ENTW_OE_4 codet, ENTW_OE_1 sieht zu).
6F	12:43: Telefon kommt rein (ENTW_OE_3 an ENTW_OE_4), ENTW_OE_1 geht eben zurück an den Platz. ENTW_OE_4 spricht am Telefon. Problem und mögliche Lösung wird diskutiert. ENTW_OE_4 : "ist ENTW_OE_2 da?", ENTW_OE_7 : "nein, die ist eben mittagessen gegangen". ENTW_OE_4 zu ENTW_OE_3 "ist grad mittagessen". Telefon Ende. ENTW_OE_1 ist mittlerweile wieder bei ENTW_OE_4 , die Beiden besprechen wieder. ENTW_OE_4 "ok schreiben wir Analyse" ENTW_OE_1 geht wieder zurück, ENTW_OE_4 fragt nach "wie siehst du aus mit Tests?" ENTW_OE_1 "ja, die sind schon wieder grün", geht zurück an Arbeitsplatz (12:50).
6G	Fazit: laut ENTW_OE_1 erster Fehler behoben, zweites TT geht an Produktkern (ENTW_OE_2). (12:52)
6H	13:00 ENTW_OE_1 berichtet ENTW_OE_TL in dessen Büro vom Stand zu den beiden TTs, und noch von einer weiteren Sache an der er arbeitet
6I	13:26: ENTW_OE_TL fragt bei ENTW_OE_2 (diese bekam eines der beiden TTs zugewiesen) wegen dem Stand nach
6J	13:30: ENTW_OE_4 erklärt ENTW_OE_TL Fehler/Situation am Whiteboard im Besprechungsraum.
6K	ENTW_OE_TL [geht zu ENTW_OE_1 und ENTW_OE_4 (reden ohnehin gerade miteinander)-> mündlich]: „ ENTW_DE_TL_STV_2 hat gerade angerufen, der ist gerade beim Kunden in Süddeutschland , die haben einen Fehler bei [Funktionalität] gefunden, das funktioniert nicht mehr so wie früher. Sie werden zwei TTs eröffnen.
7	14h: ENTW_OE_TL sucht ENTW_OE_1 , frage an ENTW_OE_7 wo er ist, ENTW_OE_7 vermutet Mittagessen, Antwort von ENTW_OE_2 oder ENTW_OE_5 dass er NICHT Mittagessen ist. Zwei minuten später sucht ENTW_OE_2 ebenfalls nach ENTW_OE_1 (fragt ebenfalls ENTW_OE_7)
8	ENTW_OE_2 fragt ENTW_OE_1 welchen Workspace (Auslieferungsversion) er hat (bzw. weiss sie eh dass er einen Workspace für Auslieferungsversion hat), bittet ihn, ein TT auf seinem Rechner fixen zu dürfen.
9	zweites TT vom Vormittag wird weiterhin bearbeitet: ENTW_OE_2 wird von ENTW_OE_4 nach Status gefragt, ENTW_OE_2 sagt, dass es nur bestimmte Produkte betrifft, aber dass sie erst im FK schauen muss welche. ENTW_OE_2 s Problem: im JIRA hatte sie früher einen Filter mit allen ihren Konzepten, seit einer neuen Version oder sonstigem Problem ist ihr Filter weg und sie findet die FKs nicht mehr. Während ENTW_OE_2 an dem dringenden TT (bzw. der offiziellen TT-Analyse-Dokument) arbeitet, fragt ENTW_OE_5 (arbeitet an Hausrat, ist erst zu Mittag (ca 13:00 gekommen und hat "Aufregung" des Vormittags deshalb nicht mitgekriegt) dazwischen ob sie grade was zu tun hat. ENTW_OE_2 fragt ob es lang dauert, unterbricht ihre Arbeit und hilft ihm kurz.
10	ENTW_OE_5 fragt in Gruppe: weiss wer was ZN-Sätze sind? ENTW_OE_4 antwortet "Zuschläge/Nachlässe". ENTW_OE_4 fragt nach "warum?". ENTW_OE_5 : "hab da so ein blödes TT".
11	ENTW_OE_2 bekommt Anruf von ENTW_OE_3 , erklärt ihm "ich mach grad TT-Analyse und ENTW_OE_1 macht gerade xy, ich wollte vorhin fragen in welche Version das geliefert werden muss" (-> ihr Problem: in TT-Beschreibung war falsche Liefer-Version angegeben)
12	ENTW_OE_5 braucht also Hilfe von ENTW_OE_4 , ENTW_OE_1 braucht ebenfalls Hilfe von ENTW_OE_4 (erklärt ihm seine Situation: "ich muss heute noch 4 Nachttest fertigmachen, etc...", --> hoher workload), --> ENTW_OE_4 (sitzt neben ENTW_OE_1) sagt "ich komm erst zu dir, ENTW_OE_5 , und dann zu dir, ENTW_OE_1 ". Drei Minuten später fragt ENTW_OE_4 bei ENTW_OE_1 nochmal nach was er jetzt noch zu tun hat, Antwort ENTW_OE_1 "ein TT und 4 Nachttests". Im weiteren Gespräch wird über 3 weitere (oder die selben?) Probleme gesprochen, am Branch funktionieren Tests nicht. ENTW_OE_2 kommt von Pause zurück, fragt nach ob ENTW_OE_1 schon fertig ist, da beide am selben TT zu arbeiten haben. ENTW_OE_1 sucht für ENTW_OE_4 noch TT-Nummer raus, schickt sie ihm via Chat, ENTW_OE_4 übernimmt Nachttest des TTs. ENTW_OE_TL kommt ebenfalls dazu, fragt bei ENTW_OE_2 nach wie es mit TT-Analyse steht, sie erklärt ihm dass ENTW_OE_1 das TT noch nachtetestet und dass sie dann die Analyse abschickt.
13	ENTW_OE_5 meldet sich nochmal bei ENTW_OE_4 für Hilfe (ENTW_OE_4 hat in der

	Zwischenzeit wieder vergessen), "hast du zeit?", ENTW_OE_4 kommt zu ihm, sie besprechen Problem.
14	ENTW_OE_4 und ENTW_OE_1 arbeiten gemeinsam an ENTW_OE_1s Rechner
15	ENTW_OE_5 in Runde, bzw. ENTW_OE_4 : "18er (Anm.: Version 18) wird am Montag gebaut, oder?", ENTW_OE_4 : "ja, aber da ist eh mal ne Email gekommen, oder?", ENTW_OE_5 : "ja, wollte nur sichergehen dass wir gerade in 17er sind".
16	ENTW_OE_TL kommt zu ENTW_OE_4 , "Kunde lehnt TT-Analyse [siehe erstes Problem-TT heute] ab, weil der Anwendungsfall früher schon mal funktionierte, und will jetzt wissen, wodurch (i.e. mit welchem TT-Fix) der Fehler zustande kam." --> ENTW_OE_4 bekommt Auftrag, die betreffende Codestelle, bzw. wann, wo und warum die eingeecheckt wurde, zu finden.
17	ENTW_OE_4 fragt bei ENTW_OE_1 nach Status, ENTW_OE_1 sagt dass er schon von ENTW_DE_TL gefragt wurde (per Chat!?) wie es aussieht, ausserdem hat er ein Problem mit einem Nachtest, weil er die Applikation lokal nicht bauen kann. ENTW_OE_4 fragt ihn nach der "Build-Reihenfolge", die von ENTW_OE_1 stellt sich als falsch heraus, ENTW_OE_4 sagt ihm die richtige Reihenfolge (5 Schritte: cleanxy, cleanxyz, build1, build2, build3)
18	ENTW_OE_TL kommt zu ENTW_OE_1 , fragt nach Status, sagt ihm dass das heute (bzw. morgen früh) rein muss. Soll sich mit XY von Releasemanagement absprechen, der ist aber heute nicht mehr da, deshalb soll er dessen Kollegen RMM_2 fragen. ENTW_OE_1 chattet daraufhin RMM_2 an, fragt ihn zur Vorgehensweise (wechselt dazu ins TTS um die TT-Nummer zu suchen). Sieht kurz später (via Email von Releasemanagement) dass sein Nachlieferungsantrag (für SQL-Datenkorrektur im Produktionsschema) abgelehnt wurde, --> Problem für ihn: jetzt kann er wieder nicht nachtesten. Anschliessend Wechsel in Eclipse zwecks lokalem Nachtest.
18A	ENTW_OE_4 kommt zurück zu ENTW_OE_1 , fragt nach Status.
19	ENTW_OE_TL kommt zu ENTW_OE_1 , fragt nach Status, sagt, dass er laut ENTW_DE_TL einen REA stellen soll und dazu Releasemanagement und ENTW_DE_AL ein Email dazu schreiben soll (gemeinsam mit dem REA). Diese sollen dann entscheiden was weiter passiert.
20	ENTW_OE_2 fragt ENTW_OE_1 und ENTW_OE_4 ob sie schon nachgetestet haben, weil sie würde dann gern heimgehen. Sie hat aber gehört dass ENTW_OE_1 heute noch 4 Nachtests machen muss, und fragt ihn ob er jetzt noch bis 10h sitzen muss. Daraufhin beschliessen ENTW_OE_4 , ENTW_OE_1 und ENTW_OE_2 die 4 Nachtests plus einen fünften Test unter sich aufzuteilen.
21	ENTW_OE_TL kommt zu ENTW_OE_1 , sagt ihm dass er eine Stunde weg muss aber dann wieder kommt.
22	aus Nebenraum hört man ENTW_OE_4 telefonieren (privat) dass er seine Verabredung absagt, und heute länger in der Arbeit bleibt.
23	ENTW_OE_4 kommt zu ENTW_OE_1 , erzählt ihm dass er ein TT geschlossen hat. Fragt ENTW_OE_1 nach Status. ENTW_OE_1 : "muss grad noch xy machen, weiss aber nicht auf welches Schema. Releasemanagement ist aber nicht mehr da, jetzt frag ich grad ENTW_OE_3 wo das liegt"
24	ENTW_OE_4 und ENTW_OE_1 unterhalten sich über Team Releasemanagement: "RMM_2 kenn ich eigentlich nicht, mit dem hab ich normal nix zu tun".
25	ENTW_OE_2 fragt ENTW_OE_1 und ENTW_OE_4 ob sie schon nachgetestet haben. ENTW_OE_1 sagt er macht grad noch was anderes. ENTW_OE_2 ist sauer, weil sie extra auf ENTW_OE_1 wartet, und seinen Test für ihre TT-Analyse braucht. ENTW_OE_4 beschreibt ihr den Testfall, damit sie nicht auf ENTW_OE_1 warten muss.
26	ENTW_OE_TL kommt zu ENTW_OE_1 , fragt nach Status von gestrigen TTs, ENTW_OE_1 erklärt Situation
27	ENTW_OE_TL fragt ENTW_OE_1 weiter nach ENTW_OE_4 (ist noch nicht im Büro, gestern abend wurde bis halb 9 gearbeitet, das weiss ENTW_OE_TL nicht?) und ob dieser gestern gesagt hätte dass er heute später kommt. Begründet die Nachfrage damit dass dessen TT-Analyse (siehe gestern: Fehler war in früherer Version nicht vorhanden) vom Kunden ganz dringend gefordert wird (diese Info wiederrum kommt von ENTW_DE_TL der momentan beim Kunden in Süddeutschland ist), dass das heute geschehen muss, und der Kunde sonst ziemlich schwere Konsequenzen (Aussetzung der Verwendung des Softwareproduktes) androht.
28	ENTW_OE_4 kommt ins Büro, wird von ENTW_OE_1 über ENTW_OE_TLs Anfrage informiert -> ENTW_OE_4 geht zu ENTW_OE_TL ins Büro um die Lage zu besprechen
29	ENTW_OE_4 ruft ENTW_OE_3 (in Norddeutschland an und fragt ihn ob er etwas geändert hat das zu dem von ihm untersuchten Fehler geführt hat. ENTW_OE_3 sagt er

	sieht sich das mal an.
30	ENTW_OE_2 sagt ENTW_OE_1 dass sie ein Email von BL bekommen hat, und dass sie es an ENTW_OE_1 weitergeleitet hat. ENTW_OE_7 macht grosse Augen, ENTW_OE_2 beruhigt, dass es eh nichts Schlimmes ist, sondern dass die Nachlieferung eines TTs auf später/nächste Woche verschoben wird.
31	ENTW_OE_2 erhält einen Anruf (von ENTW_OE_3 ?) dass alle Test auf CC fehlschlagen (die anderen im Raum können das ev hören, reagieren aber nicht). Sie vermutet dass vlt irgendetwas mit der DB nicht stimmt. Fragt ihn weiters, ob er einen bestimmten Nachttest schon gemacht hat.
	Situation: Alle richten ihre Arbeitsumgebung ein, plaudern über Wochenende, sowie über ENTW_OE_3s Erlebnisse beim Auftraggeber in Norddeutschland letzte Woche.
32	ENTW_OE_TL kommt zu ENTW_OE_4 , fragt wegen (Problem-)TTs von letzter Woche (Do, Fr) --> einige Minuten später kommt ENTW_OE_4 zu ENTW_OE_1 , fragt wer von Releasemanagement für irgendwas verantwortlich ist. ENTW_OE_1 antwortet:" RMM_2 ist verantwortlich", sieht kurz in Messenger (der schon offen ist) und ergänzt:"der ist eh grad online"
33	ENTW_OE_2 (aus dem Nichts, an die Kollegen):"Ach, das muss man ja doch in 18er nachliefern" ENTW_OE_3 und ENTW_OE_1 (wissen anscheinend worum es geht) "ja, sicher". ENTW_OE_1 geht zu ihr, und sieht sich mit ihr den REA an. ENTW_OE_1 und ENTW_OE_3 schlagen vor, sie soll bei Releasemanagement oder ENTW_DE_TL nachfragen wie sie vorgehen soll. Die Drei diskutieren den Zustand der vorhandenen Code-Stränge und welche Nachlieferung (in welchen Branch) möglich wäre. Schlagen abermals vor dass man am besten nochmal nachfragen soll (bei Releasemanagement).
34	ENTW_OE_2 zu ENTW_OE_1 :" auf normalem 18er hast du das eh oben, deine Änderungen, oder?" ENTW_OE_1 :"ja, hab ich"
35	ENTW_OE_4 kommt zu ENTW_OE_2 :"was brauchts du von mir?" ENTW_OE_2 :"machst du das aus mit dem Releasemanagement?" ENTW_OE_4 :"ja, da bin ich gerade dabei." und erklärt die Vorgehensweise für Einchecken und Nachliefern für diesen Tag. ENTW_OE_4 zu ENTW_OE_2 weiter:"und schreib RMM_2 (von Releasemanagement) noch ein Email wegen Einchecken." ENTW_OE_4 zurück zu seinem Platz. Gleich darauf zu ENTW_OE_2 :"du, ich schreibe RMM_2 das Email selbst.". ENTW_OE_4 wieder zurück zu ENTW_OE_2 , bereden nochmal welches TT in welche Version eingchecked wird.
36	ENTW_OE_4 zu ENTW_OE_1 : "hast du den Branch ausgecheckt?" Besprechen weitere TTs von ENTW_OE_1 . ENTW_OE_4 :"ok, dann schau du weiter auf den 3.0er)
37	ENTW_OE_2 und ENTW_OE_4 besprechen etwas, ENTW_OE_2 :"... oder ENTW_DE_TL zeigt sich nur 'Abwesend', manchmal macht er das"
38	ENTW_OE_3 zu ENTW_OE_2 : hast du die Tests im Auge? ENTW_OE_2 :"nein, aber ich glaube nicht dass die laufen"
39	ENTW_OE_1 zu ENTW_OE_3 :" hast du in 3.0 etwas an den Tarifmerkmalen geändert?" ENTW_OE_3 :"nein, eigentlich nicht". ENTW_OE_1 :"weil hier im Dialog passt etwas nicht, und ich weiss nicht wieso?"
40	ENTW_OE_1 geht zu ENTW_OE_4 , erzählt ihm was er jetzt als nächstes vorhat zu tun (nämlich eine Codeänderung in einem bestimmten Branch, bzw. am Trunk zu testen)
41	ENTW_OE_1 zu ENTW_OE_4 : "Du, wir müssen noch REAs für 3.16 und 3.018 stellen." ENTW_OE_4 :"hab ich schon gemacht" ENTW_OE_1 "super" ENTW_OE_3 schaltet sich dazwischen:"aber für xxx müsst ihr auch noch stellen!" ENTW_OE_4 und ENTW_OE_1 :"achja, stimmt!". Danach machen sich ENTW_OE_1 und ENTW_OE_4 (jeder an seinem Recher) an die Arbeit Codes in die verschiedenen Branches zu mergen, fragen dazu wechselseitig ob bestimmte Code-Fragmente schon in bestimmte Branches gemerged wurden, schicken sich REA-Nummern via Chat zu, etc... ENTW_OE_TL kommt auch kurz mal dazu, fragt?/gibt Anweisung? zu einer bestimmten Branch-version und geht gleich wieder. Zeitrahmen: REAs müssen bis 12:00 beantragt/nachgeliefert sein --> nur 20 min Zeit!
42	ENTW_OE_4 und ENTW_OE_1 diskutieren über FK-Anforderung. ENTW_OE_3 schaltet sich ein, da er die Diskussion periphär mithört, und korrigiert sie in einer Aussage.
43	ENTW_OE_4 und ENTW_OE_1 diskutieren über FK-Anforderung. Sie telefonieren mit FK_2 , und besprechen wie es genau umgesetzt gehört. Eventuell handelt es sich dabei um eine Änderungsanforderung. ENTW_OE_3 sitzt gegenüber an seinem Platz und diskutiert ebenfalls mit, bzw. schaltet sich ab und zu ein. ENTW_OE_3 schickt einen Auszug aus seinem Konzept an FK_2 mittels Chat, und gibt ENTW_OE_4 und ENTW_OE_1 Bescheid dass er ihm das geschickt hat. Sie kommen zu dem Schluss dass eine Änderung im Code erforderlich ist, ENTW_OE_4 sagt FK_2 dass sie noch "ein, zwei" Sachen umbauen müssen, und dass er ihm am nächsten Tag eine Liste mit den Änderungen schicken werde, damit

	dieser das ins FK einbauen kann, bzw. davon Kenntniss hat. Sie machen auch einen Screenshot den sie für die Liste mit Änderungen mitschicken.
44	ENTW_OE_TL kommt zu ENTW_OE_4 , fragt um den TT-Status.
45	ENTW_OE_5 hat Probleme mit Workspace (JARs sind verschwunden) - ENTW_OE_2 hatte letzte Woche selbe Probleme. ENTW_OE_5 hat mit Norddeutschen deswegen gechattet, aber noch keine Lösung gefunden.
46	ENTW_OE_3 zu ENTW_OE_2 : "die Emails von ... (irgendwer in Norddeutschland hast du aber auch bekommen, oder?" ENTW_OE_2 :"ja, hab ich, hab ich mir aber nicht angeschaut."
47	ENTW_DE_TL_STV_2 ruft ENTW_OE_3 an, ENTW_OE_3 erklärt ihm dass ein bereits geschlossenes TT in JIRA als TASK (Todo-Task) aufscheint, und fragt ihn ob das da irrtümlich reinkam. ENTW_DE_TL_STV_2 entfernt daraufhin das TT/Task aus JIRA.
48A	ENTW_OE_TL zu ENTW_OE_3 : wie siehts aus mit deinen zwei Tickets in [Projekt B] ? Hab gesehen dass du das in JIRA noch hast. ENTW_OE_3 :"ja, die krieg ich heut noch hin" ENTW_OE_TL :"heisst das fertig oder realisiert?" ENTW_OE_3 :"realisiert"
48B	ENTW_OE_TL zu ENTW_OE_1 : "Du hast auch zwei TTs in [Projekt B] ? Weil ENTW_DE_TL muss das wissen" ENTW_OE_1 :"ja muss ich schau" - Schaut in Lotus TTS - ENTW_OE_1 :"ja die sind schon geschlossen".
48C	ENTW_OE_3 spricht über den Tisch ENTW_OE_TL an:"ich hab da auch noch ein drittes TT zu dem Thema" erklärt ihm kurz worum es geht. ENTW_OE_3 :"ok, ich schick dir dann ein Mail von JIRA"
49	ENTW_OE_2 bemerkt Problem:"'xyz', wer ist das?"
50	Scrum-Meeting
50A	ENTW_OE_TL fasst Probleme letzte Woche zusammen, macht auf neuen Meilenstein aufmerksam, erklärt seine eigenen TTs und Tasks
50B	ENTW_OE_4 : TTs, TTs letzter Woche, haben Tests angelegt, auch für ENTW_OE_2 , verschiedene Branch-Versionen. ENTW_OE_TL macht ENTW_OE_4 darauf aufmerksam dass für neuen Meilenstein schon ein Task für Hausrat fällig wird, ENTW_OE_4 weiss davon noch nichts.
50C	ENTW_OE_2 : TTs, fährt auf Urlaub (ab übermorgen), hat noch Sachen zu implementieren, soll sich mit ENTW_DE_6 absprechen
50D	ENTW_OE_1 : TTs
50E	ENTW_OE_3 : TTs (viele -> ENTW_OE_TL kurz erschrocken); TT von Kunde_Mitarbeiter_1 , schlecht beschrieben, hängt bei Releasemanagement fest; Meiste TTs von Wohngebäude (Projekt), wenige von KFZ (Projekt B); Glas muss für neuen Meilenstein bearbeitet werden.
50F	ENTW_OE_5 : TTs, Implementierungstask; hat einige TTs an ENTW_OE_6 (momentan noch auf Urlaub) abgegeben, weiss über dessen TT-Status nicht Bescheid; ein TT an TST_1 abgegeben; ENTW_OE_TL sagt dass es zwei ENTW_OE_5s in JIRA gibt, ENTW_OE_5 soll ihm sagen welches das Richtige ist; ENTW_OE_3 wirft mal dazwischen, dass ENTW_DE_TL_STV_1 ebenfalls aus Urlaub zurück ist;
50G	
51	ENTW_OE_2 trägt Stunden für TT von letzter Woche ein, fragt in die Runde wer [Nachname] ist, da diese Person eine bestimmte Anzahl von Stunden für etwas veranschlagt hat was ENTW_OE_2 betrifft. "Wer ist denn zuständig für PAZ?" --> Antwort ENTW_OE_1 :" ENTW_DE_TL ". Problem: TT war eigentlich Änderungsanforderung, deshalb ist unklar wohin die Zeit geschrieben werden soll.
52	ENTW_OE_3 zu ENTW_OE_1 :"ist FK_2 denn heute nicht da?" ENTW_OE_1 :"Keine Ahnung, ich hätte selber schon was gebraucht von ihm. Gestern war er am Vormittag da, aber am Nachmittag auch nicht mehr". ENTW_OE_3 :"Hat der Urlaub oder was?". Kurz darauf: ENTW_OE_3 zu ENTW_OE_1 :" FK_2 hatte gestern Geburtstag. Ich hab grad FK_6 [Büro-Nachbar von FK_2] angechattet. Und Grüsse von FK_6 soll ich ausrichten". ENTW_OE_4 kommt ebenfalls in den Raum (zufällig), ENTW_OE_3 erzählt ihm ebenfalls dass FK_2 Geburtstag hatte. ENTW_OE_4 :"Ahaaa."
53	ENTW_OE_TL kommt ins Zimmer zu ENTW_OE_1 :"Du ENTW_OE_1 , ich hab dir gerade ein TT zugewiesen." ENTW_OE_1 :"Ja, ich schau es mir gerade an." ENTW_OE_TL :"Ok, das ist nämlich wichtig." ENTW_OE_TL geht wieder.
zu 52	ENTW_OE_3 zu ENTW_OE_1 und ENTW_OE_4 :" FK_2 hat sich krankgemeldet heute".
54	ENTW_OE_TL zu ENTW_OE_1 :"Wegen dem TT heute" - Zwischenfrage ENTW_OE_1 :"Das neue oder das wiedereröffnete?" - ENTW_OE_TL :"Nein, das wiedereröffnete. Bevor wir da die TT-Analyse machen, bzw. bevor wir da irgendwas sagen wie 'das hat doch eh schon funktioniert' möchte ich dass wir uns dann kurz mal

	zusammensetzen und das besprechen."
55	ENTW_OE_TL kommt ins Zimmer. ENTW_OE_TL zu ENTW_OE_1 :"Na, wie schauts aus?" "Ja, ich hab das und das getestet, allerdings weiss ich nicht auf welchem DB-Schema das getestet wurde. Die Screenshots die im TT mitgeschickt wurden sagen aber etwas anderes. Mitarbeiter_Kunde_1 erwische ich aber gerade nicht." ENTW_OE_TL :"Ja, den erwische ich auch nicht. Ich hab nämlich grade ein ähnliches Problem. In den DBs verwenden sie nämlich oft die gleichen Daten. Schau mal nach ob das ein Vertrag ist der diese Bedingungen erfüllt."
56A	ENTW_OE_3 zu ENTW_OE_1 :" macht grade jemand Spässchen mit unserer DB?" ENTW_OE_1 :"Ja, offenbar." [beide sitzen wartend vor ihren Rechnern]
56B	ENTW_OE_3 :"Hm, in Squirrel krieg ich 'closed connection'". ENTW_OE_1 :"Hm, sagst du das den DB-Admins oder soll ich es ihnen sagen?" ENTW_OE_3 :"ja kannst du das machen?" ENTW_OE_1 :"ja, klar.". [ENTW_OE_3 ab, geht Mittagessen holen]
57	ENTW_OE_1 zu ENTW_OE_4 :"Du, weisst du die Nummer vom RMM_DB_1 ?" ENTW_OE_4 :"nein, weiss ich nicht. Wieso?" ENTW_OE_1 :"ja weil die DB ist grad voll langsam. Hast du ihm gesagt er soll spiegeln [Daten neu einspielen]?" ENTW_OE_4 :"ja hab ich. Aber spielt das eine Rolle?" ENTW_OE_1 :"ja, weil jetzt geht nichts mehr." ENTW_OE_4 :"hast du RMM_DB_1 angechattet?" ENTW_OE_1 :"nein, der ist nicht online. Ich würde ihn anrufen wenn ich seine Nummer hätte." ENTW_OE_1 sucht die Nummer raus, wohl aus Messenger, ruft RMM_DB_1 an:"hallo, hier ist ENTW_OE_1 aus Österreich , unsere DB ist gerade down. (...)". ENTW_OE_1 legt wieder auf, zu ENTW_OE_4 :"ja, er schaut sich das mal an." Die beiden reden informell weiter über Mittagessen, und ein Problem das ENTW_OE_4 hat - er muss einen bestimmten Anwendungsfall für Hausrat nachstellen, dieser ist sehr zeitaufwändig, ENTW_OE_4 erwähnt, dass er Kunde_Mitarbeiter_1 deswegen kontaktieren muss. ENTW_OE_1 sagt ihm dass er auch schon versucht hatte ihn zu kontaktieren, er aber nicht da war [siehe 55]. Später ENTW_OE_4 zu ENTW_OE_1 :"So, was machen wir jetzt eigentlich?" ENTW_OE_1 :"Wir warten dass die DB wieder läuft." ENTW_OE_4 wieder zurück an seinen Platz, ENTW_OE_1 in Küche. Uhrzeit: 12:35)
58 (siehe 52)	ENTW_OE_4 zu ENTW_OE_1 :"Weisst du eigentlich schon was von FK_2 ?" ENTW_OE_1 :"Der ist krank heute."
59	ENTW_OE_TL kommt ins Zimmer:"mir ist gerade eingefallen, weil ENTW_DE_TL da letztens ein Email geschrieben hat, dass ihr bei REAs den Status nur auf 'teilweise behoben', und nicht auf 'komplett behoben' stellt, da die TTs sonst auf 'realisiert' gesetzt werden."
60	ENTW_OE_3 zu ENTW_OE_1 :"'Unterbrechungszeit', wo find ich die eigentlich?" ENTW_OE_1 :"Bei den Stufungsständen zum Beispiel, das ist TabelleXY."
61	ENTW_OE_1 kommt aus Küche, zu ENTW_OE_3 :"Du hast in der Küche eine Überschwemmung (--> Kaffemaschine) angerichtet!". ENTW_OE_3 eilt daraufhin in die Küche.
62 - Besprechung	ENTW_OE_TL , ENTW_OE_6 , ENTW_OE_1 , ENTW_OE_4 , ENTW_OE_3 . ENTW_OE_5 kommt später dazu.
62A	ENTW_OE_TL : eigener Status
62B	ENTW_OE_TL : Problem bei [Eingabemaske], VersicherungsGmbH - und Kunde-1 -Version sind verschieden (Listbox statt Eingabefeld). Fragt ob das ENTW_DE_8 gehört, bzw. ob der zuständig ist. ENTW_OE_4 :" nein, das sollte bei uns sein. ENTW_DE_8 hat damit nichts mehr zu tun. Das hat er auch letzte Woche zu ENTW_OE_5 gesagt". ENTW_OE_6 erwähnt dass er ein TT hatte, wo die Umstellung von Eingabefeld auf Dropbox verlangt wurde. ENTW_OE_TL :"na vielleicht wurde das nicht von der VersicherungsGmbH -version auf die Kunde-1 -Version nachgezogen. Ich seh mir das Freitag an."
62B	ENTW_OE_6 : erklärt Status
62C	ENTW_OE_TL fragt ENTW_OE_6 wie er mit REA vorging, macht ihn auf 'Teilbehebung' aufmerksam.
62D	ENTW_OE_6 : hat TT wegen Absturz auf bestimmter Eingabemaske, kann es lokal nicht nachstellen (auf Testversion schon). ENTW_OE_3 fragt nach, wo das genau auftritt, ENTW_OE_6 antwortet auf Z/N, ENTW_OE_3 sagt ihm dass er und ENTW_DE_2 das selbe Problem hätten.
62E	ENTW_OE_1 : 4 TTs, ein wiedereröffnetes, erklärt Status, ENTW_OE_TL fragt ob er es noch in Branch-Version xy kriegt. ENTW_OE_3 schaltet sich ein wegen REA-Vorgehen.
62F	ENTW_OE_1 : hat Produktiv-TT, Anwendungsfall ist aber im FK nicht beschrieben, d.h. sie brauchen wohl einen Change Request. Er hat deshalb schon FK_2 geschrieben, der ist heute aber krank.
62G	ENTW_OE_1 : letztes TT (ebenfalls produktiv) hab ich noch nicht geschafft, muss aber nicht in Produktiv-Version rein.
62H	ENTW_OE_TL erwähnt dass morgen in Süddeutschland ebenfalls Feiertag ist, in

	Norddeutschland wird aber gearbeitet.
62I	ENTW_OE_4 : "mache Summenanpassung in Hausrat, warte dazu aber auf DB-Aktualisierung von RMM_DB_1 ." Erklärt warum Schema kaputt ist (DB-Update-Job ist während Einspielens der Daten abgebrochen), meint dass es dazu eine neue/andere/bessere Vorgehensweise geben sollte, da im Problemfall Norddeutschland dieses Problem nicht registriert, und sich erst Österreich melden muss. RMM_DB_1 muss sich ebenfalls Log-Dateien der Jobs ansehen, darauf hat in Österreich keiner Zugriff. ENTW_OE_4 schlägt vor dass sich jemand aus Österreich dauerhaft um DB-Schemas kümmern sollte. ENTW_OE_4 erwähnt weiters, dass RMM_DB_1 ein Email wegen neuer DB-Hardware für Österreich schreibt.
62J	ENTW_OE_4 hat noch ein TT, dieses ist auch ein Changerequest, bedarf allerdings keiner Implementierung mehr. Muss mit FK_2 noch abklären, ob TT dadurch erledigt ist, oder noch nachgetestet werden muss.
62K	ENTW_OE_3 : hat TT, erklärt Status (Analyse geschickt, REA genehmigt, Email an ENTW_DE_TL_STV_1 , ...)
62L	ENTW_OE_3 : sonst noch 9 TTs, ENTW_OE_TL freut sich ("langsam aber sicher..."), da TT-Anzahl letzte Woche noch doppelt so hoch lag.
62M	ENTW_OE_TL : " Norddeutschland hat gerade mehr Stress als wir, schauen wir nochmal wie es am Freitag [morgen, Donnerstag ist Feiertag] steht.
62N	ENTW_OE_4 zu ENTW_OE_TL : "hab gesehen, dass du Dialogänderungen [in JIRA] erst sehr spät [geplant] hast, vielleicht sollten wir das noch vorziehen, da das Ende ohnehin erst sehr spät ist, und wir auch noch die Tests durchkriegen müssen"
62O	ENTW_OE_4 zu ENTW_OE_TL : "hast du Norddeutschland auf Süddeutschland angesprochen, das worüber wir letzgens geredet haben?" ENTW_OE_4 meint, dass Kunde sich nicht um korrekte Versionierung für TTs kümmert, und dadurch bestimmte Fehler nicht in allen Versionen gefixt werden. ENTW_OE_TL meint dass das der Kunde wohl schon korrekt und bewusst macht. ENTW_OE_4 gibt aber das Beispiel, dass ein bestimmtes TT vom Kunden für Version 3.0 bestellt wurde, und hinterher wurde er vom Kunden gefragt, wieso das TT nicht auch für Version 2.90 gefixt wurde.
62P	ENTW_OE_5 (ist später gekommen), wird von ENTW_OE_TL auf REA-"Teilbehebung" aufmerksam gemacht.
62Q	ENTW_OE_5 erklärt Status
62R	ENTW_OE_TL macht ENTW_OE_5 auf Eincheckdeadline (heute, 16h) aufmerksam.
62S	ENTW_OE_TL fragt ENTW_OE_5 wegen [Eingabemaske], ENTW_OE_5 weiss nichts davon.
63	ENTW_OE_3 zu ENTW_OE_4 : habt ihr Tests zu [Funktionalität]? ENTW_OE_4 : "ja", geht zu ihm, und zeigt im den Test auf ENTW_OE_3s Rechner
64	ENTW_OE_1 gibt laut von sich: "jetzt schickt der Kunde schon leere Nachtests" --> bedeutet für ihn dass er Kunden deswegen wieder kontaktieren muss. Gibt kurz darauf ENTW_OE_4 deswegen bescheid [für den zur Info].
65	ENTW_OE_1 ruft ENTW_DE_TL_STV_1 in Norddeutschland an: "Habe Analyse-Dokument für TT fertig, wollte fragen ob du das gesehen hast, weil das ist ziemlich dringend, muss noch in Patch 3 rein [bis heute, 16h]". [ENTW_DE_TL_STV_1 muss Analyse begutachten]
66	ENTW_OE_4 zu ENTW_OE_5 : "In welchem Schema bist du gerade?"
67	ENTW_OE_TL bei ENTW_OE_6 , machen gemeinsam TT.
68	Telefon läutet, ENTW_OE_1 geht ran, spricht kurz. Später kommt ENTW_OE_3 aus Küche zurück, ENTW_OE_1 sagt ihm dass Telefon war, ENTW_OE_3 sucht Nummer raus und ruft zurück.
69	ENTW_OE_TL kommt ins Zimmer zu ENTW_OE_1 , ENTW_OE_3 ist nicht anwesend: "Sagst du ENTW_OE_3 dass er seinen REA checken soll, weil in Norddeutschland warten die gerade drauf, dass alles fertig wird.". 5 Minuten später kommt ENTW_OE_3 zurück, ENTW_OE_1 zu ihm: "ich soll dir ausrichten, dass du deinen REA-Antrag einchecken sollst oder so..". ENTW_OE_3 : "war das RMM_2 ?" ENTW_OE_1 : "Nein, ENTW_OE_TL ". ENTW_OE_3 : "Das hab ich doch eigentlich schon gemacht!"
70	ENTW_OE_5 zu ENTW_OE_1 und ENTW_OE_3 : "wie erstell ich ein TT? welchen Status geb ich bei Intern/extern an?" ENTW_OE_1 u ENTW_OE_3 : "kommt darauf an. wenn du es selber entdeckt hast intern, wenn es vom Kunden ist extern."
71	ENTW_OE_3 zu ENTW_OE_1 : " ENTW_DE_AL hat mich gerade auf ein bestimmtes Vorgehen für die REA-Anträge [angabe einer bestimmten Kalenderwoche -> 25 statt 26] aufmerksam gemacht, und ich soll dir sagen dass du es auch so machen sollst."
	Fenstertag, nur wenige Leute im Büro (ENTW_OE_4 , ENTW_OE_5 , ENTW_OE_6 , ENTW_OE_TL)
72 -	

Besprechung	
72A	ENTW_OE_TL erklärt Vorgehensweise für neuen Branch/Abspaltung da ab nächster Woche Version 3.0.19 produktiv gesetzt wird.
72B	ENTW_OE_TL hat mit ENTW_DE_TL telefoniert, dieser ist sauer weil TT nicht ordentlich nachgetestet wurden, und vom Kunden wieder-eröffnet wurden. Es wird beschlossen, dass Nachtests immer von jemand anderem als dem TT-Beheber gemacht werden. ENTW_OE_4 fragt nach der Nummer der TTs, ENTW_OE_TL sagt ihm, er soll sich ENTW_OE_1s TTs ansehen, dann sollte er sie finden.
72C	ENTW_OE_TL erklärt eigenen Status und TTs
72D	ENTW_OE_4 erklärt Status
72E	ENTW_OE_5 erklärt Status
72F	ENTW_OE_6 erklärt Status
73	ENTW_OE_4 kommt zu ENTW_OE_1 :"und, wie gehts dir?" ENTW_OE_1 :"Naja, ich mach grad die Fehleranalyse [Anm.: für ein Produktivticket von letzter Woche]". Anschliessend besprechen sie recht detailliert das Problem. Im Folgenden besprechen sie auch in welchen Branch das TT geliefert werden muss. ENTW_OE_1 :"in der Mail [Anm.: welche Email??] steht dass es in Version 3.18 gehört, aber eigentlich gehört es ja jetzt in die neuere Version, oder?" ENTW_OE_4 :"ja chatte mal den ENTW_DE_TL an". ENTW_OE_1 :"Der ist grad offline". ENTW_OE_4 :"hm, vielleicht sollen wir Kunde_Mitarbeiter_1 anrufen..". Anschliessend besprechen sie weiter das Problem [Anm.: ein Absturz im Hintergrundjob], ENTW_OE_4 setzt sich dazu zu ENTW_OE_1 an den Platz. Den Fehler finden sie recht bald [Zeitpunkt: 9:36], anschliessend testen sie den Fehler in verschiedenen Versionen des Branches durch, und besprechen den Inhalt der TT-Analyse. ENTW_OE_1 :" aber was ist wenn es jetzt wieder eine Konfiguration gibt die wieder fehlschlägt? Ich teste jetzt einfach alle Fälle durch, auch wenn das vier Stunden dauert." ENTW_OE_4 :"ja, mach das". Dann machen sie gemeinsam weiter mit Testen. ENTW_OE_4 zwischendurch:" wieviele TTs hast du jetzt dann noch?" ENTW_OE_1 :"zwei TTs und zwei Analysen" ENTW_OE_4 :"Pfff [Anm.: "das ist aber viel"]" Anschliessend erklärt ENTW_OE_1 die TTs. ENTW_OE_1 :"bei dem einen hat Kunde_Mitarbeiter_1 zwei Zeilen geschrieben und 'passt das?'" ENTW_OE_1 :"das Problem ist, dass sie [der Kunde und Auftraggeber] uns das TT immer wieder aufmachen, auch wenn der Fehler aber wegen eines anderen Problems auftritt." [Zeitpunkt: 9:44]. Weiter besprechen sie Branch und Zielversionen für das TT. ENTW_OE_1 :"in der neuen Version müssten wir es ja gar nicht mehr fixen, weil da passt es ja schon." ENTW_OE_4 :"ja, mach am besten einen Screenshot von allen fällen mit allen Versionen, das dauert halt, aber egal." ENTW_OE_4 kann ich mal schnell meinen REA anschauen, ob der schon durch ist, wenn ich schon hier sitz?" ENTW_OE_1 :"ja klar" ENTW_OE_4 :"hm.. welche Version ist der jetzt noch mal [Branch-Version, Anm.]." ENTW_OE_1 :"ich frag jetzt mal den ENTW_OE_TL wie das mit den Versionen genau ist." steht auf und geht rüber zu ihm [Zeitpunkt: 9:48]. ENTW_OE_1 kommt zurück [Zeitpunkt 9:51]:"ok, es muss eh nur auf 3.0.19 nachgeliefert werden. Da geht es ja eigentlich schon, dann können wir es ja gleich auf 'testbereit' setzen.". ENTW_OE_4 mach grosse Augen:"Hm... wir sollten aber schon noch verschiedene Tests machen! Das hab ich ja vorhin gesagt.". Kurze Pause. ENTW_OE_1 :"ja, ich muss dann eh mit Kunde_Mitarbeiter_1 telefonieren, wegen eines Tests."
74	ENTW_OE_1 :"ach ja, wegen des TTs zum Refactoring der [Funktionalität] , hast du das jetzt schon gemacht? Weil ENTW_OE_TL hat gefragt." ENTW_OE_4 :"hm, das weiss ich jetzt gar nicht, das ist ja schon ein halbes Jahr her. Da müsste ich nachschaun, aber eingchecked wurde es ja unter dem Namen." ENTW_OE_1 :"ja, vielleicht sollten wir das auch nicht 'refactoring' nennen, sondern einfach 'TT'." ENTW_OE_4 sieht etwas im Rechner nach, findet die Eincheck-ID des Refactoring-TTs:"hier ist sie, da hab ich eingchecked." ENTW_OE_1 :"ok, dann schreib ich rein, dass es mit diesem TT gelöst wurde." ENTW_OE_4 :"ok." Sie sehen sich die Stelle mit dem einchecken an, ENTW_OE_1 :"ist das irgendwo gekennzeichnet, dass es sich um ein Change-Request oder Refactoring handelt?" ENTW_OE_4 :"ja, das steht hier irgendwo..". ENTW_OE_1 :"ok, dann muss ich eigentlich eh nichts mehr einchecken." Sie sprechen wieder über verschiedene Liefer-Versionen, und dass das immer der Auftraggeber festlegt [Zeitpunkt: 10:01]. Anschliessend behandeln sie wieder das Produktiv-TT, bzw. verschiedene Testfälle und Konfigurationen.
75	ENTW_OE_1 und ENTW_OE_4 machen Witze wer als nächster Probleme kriegt.
76	ENTW_OE_1 und ENTW_OE_4 besprechen Test-Prozeduren der Firma, bemängeln dass es zu wenige Tester gibt, und dass sich eine Testabteilung darum kümmern sollte, dass alle Test-Konstellationen abgehandelt werden. Nebenbei lassen sie Tests für das Produktiv-TT von vorhin laufen [bis 10:12].
77	ENTW_OE_1 und ENTW_OE_4 sprechen über ein Framework das für bestimmte Berechnungen verwendet wird, dabei gab es Fehler, weil ENTW_DE_TL_STV_2 und

	ENTW_DE_TL_STV_1 dort Änderungen vornahmen. Sie überlegen wer welche Änderungen vorgenommen hat (" ENTW_DE_TL_STV_2 hat da [in dem Framework] was gemacht, und ENTW_DE_TL_STV_1 macht viel in Bestand."
78	ENTW_OE_TL kommt zu ENTW_OE_1 ins Zimmer:"wie ist der Status von dem TT?" ENTW_OE_1 :"Analyse ist fertig, aber welche Version soll ich nachliefern?" ENTW_OE_TL :"Hm, lass mich mal sehen was es da gibt [Anm.: im Analyse-Formular]. Hm.. schau mal im Entwicklerkalender."
79	ENTW_OE_1 zu ENTW_OE_3 :" hast du die Nummer von ENTW_DE_TL ?" "xxx" ENTW_OE_1 ruft ENTW_DE_TL an, erwischt jemand anderen (Anm.: ENTW_DE_TL_STV_1):"Hallo hier spricht ENTW_OE_1 ich habe ein Frage zu einem TT. Da hat ENTW_DE_TL eine Email geschrieben [liest vor], welche Version soll ich da jetzt nehmen? Version xy ist schon vorbei. Ursprünglich hatte das TT die Zielversion 2.90, das TT ist in 3 aber schon behoben. Mhmh... Ok, danke, Ciao."
80	ENTW_OE_3 , greift zum Telefon:"Hi RMM_2 , ich hab ne Frage zu der Version 3.0.19. Wenn ich da was einchecke muss ich da dann nichts mehr extra antriggern, oder. Wegen der Scripte. [...] OK, danke, Ciao."
81	ENTW_OE_TL kommt zu ENTW_OE_1 , sehen sich gemeinsam TT im TTS an. ENTW_OE_TL :"hast du es schon lokal nachgestellt?" ENTW_OE_1 :"ja, auf 3.19." Besprechen weiter das TT, meinen dass den Test auf der Testversion eigentlich der Nachtester machen sollte. ENTW_OE_TL :"und warum wird das da auf den Sachbearbeiter ausgesteuert?" ENTW_OE_1 erklärt warum. ENTW_OE_TL :"und hast du da schon mit dem Kunde_Mitarbeiter_1 gesprochen?" ENTW_OE_1 :"ich hab ihm eine Mail geschrieben." ENTW_OE_TL :"ok, klär das eben noch mit Kunde_Mitarbeiter_1 ab, ob das hier auch die Situation ist, und er soll dir noch das Absturzprotokoll schicken. Und wenn das dann OK ist, würd ich es noch dem ENTW_DE_TL_STV_1 schicken."
82	ENTW_OE_TL zu ENTW_OE_3 :" ist der eine REA schon durch?" ENTW_OE_3 :"ja ist schon, und der andere ist auch schon beantragt" ENTW_OE_TL :"welcher andere?" ENTW_OE_3 erklärt kurz. ENTW_OE_TL :"ist das jetzt neu reingekommen?" ENTW_OE_3 :"ja, genau. aber REA ist schon beantragt." Besprechen weitere TTs etc..
83	ENTW_OE_TL zu ENTW_OE_4 , fragt Status.
84	ENTW_OE_1 ruft Kunde_1 an:"Hallo, hier spricht ENTW_OE_1 aus Österreich , ich hab hier das TT mit der Ziffer xyz, und ich wollte nachfragen wie das jetzt aussieht. Ich hab ihnen deshalb auch ein Email geschrieben. [hört zu] Ok, Ok. Ja, ich hab die Analyse jetzt gemacht, und das für die Version xy gefixt. [hört zu]. Ok, dann werde ich die Analyse dann abschicken. Danke, wiederhören."
85	ENTW_OE_3 laut:"Ach, wieso funktioniert denn das auf einmal?. Ich wollte grade ein TT in Wohngebäude fixen wo immer eine Exception kam, und jetzt gehts aber wo ich es nachstellen möchte. Setz ich das jetzt einfach auf realisiert?"
86	ENTW_OE_3 nimmt Telefon:"Hallo, ich hab da ein TT für Wohngebäude, und ich hab gesehen du hast das selbe TT für ein anderes Projekt, und ich hab dazu letzts was gefixt, und jetzt geht das anscheinend auch wieder. Also ich werde das mal auf realisiert setzen."
87	ENTW_OE_TL kommt zu ENTW_OE_1 :"Wie siehst jetzt aus mit dem TT? Hast du mit Kunde_Mitarbeiter_1 telefoniert?" ENTW_OE_1 :"Ja, ich hab mit ihm telefoniert, und es war eh der Fehler den er gemeint hat." ENTW_OE_TL :"ok, dann kannst du das nachher auch mir zuweisen auf 3.0.19 fürs nachtesten."
88	ENTW_OE_3 laut:"wer kümmert sich eigentlich zur Zeit um [Komponente auf CruiseControl]?" ENTW_OE_1 :"Hm, wahrscheinlich niemand." ENTW_OE_4 :"niemand?" ENTW_OE_3 zu ENTW_OE_1 :"hast du dir mal die 3.19er angeschaut? Sieh mal her." ENTW_OE_1 geht zu ihm rüber, sie sehen es sich gemeinsam an. ENTW_OE_1 :"hm, da hat wohl irgendwer was schlechtes eingecheckt. Weiss auch nicht genau, aber ich kanns mir bei Gelegenheit ansehen. Wenn viele Sachen fehlschlagen hat wahrscheinlich irgendwer irgendwas geändert, das liegt dann meistens nicht an uns. Das renkt sich dann mit der Zeit meistens wieder ein."
89	ENTW_OE_1 sitzt bei ENTW_OE_4 . Diskutieren Problem bei 'Terminen'
90	ENTW_OE_5 zu ENTW_OE_3 :"Kennst du eine Tabelle für 'Nachtragsgründe'?" ENTW_OE_3 :"Gibt wahrscheinlich schon eine, aber wo genau weiss ich auch nicht."
90A	ENTW_OE_5 fragt ENTW_OE_4 zu Hausrat-FK, und welche Funktionalität eine bestimmte Eingabemaske haben muss. Dazu hat er das FK geöffnet und erklärt ENTW_OE_4 die Lage. Anschliessend öffnet er Chat und fragt FK_1 . Wartet auf Antwort. Geht zu ENTW_OE_1 und fragt nach der Tabelle für Nachtragsgründe (negativ), geht zu ENTW_OE_4 und fragt nach Tabelle für Nachtragsgründe.
90B	ENTW_OE_5 bekommt Antwort im Chat.
91A	ENTW_OE_3 laut:"ich hab gerade mal wieder eine falsche TT-Beschreibung gekriegt" ENTW_OE_1 :"haha, das hab ich auch letzte Woche gekriegt."

91B	<p>ENTW_OE_3 greift zum Telefon [ENTW_DE_TL_STV_1?]:"hallo, ich hab da bei einem TT einen falschen Nachtest angehängt bekommen", sagt die TT-Nummer dazu.</p> <p>ENTW_OE_3:"Die haben da ja einen neuen Nachtester beim Kunden." (...)</p> <p>ENTW_OE_3:"Ist die da? im Chat ist sie nämlich offline. (...) Ok, dann versuch ich die mal."</p> <p>ENTW_OE_3 wählt neue Nummer:"Hallo, (...) Problem mit TT, die Nummer ist xyz. (...) Ok, danke, ciao." [Problem mit falschem Anhang wird weitergegeben, und von anderer Person (KollegIn in Norddeutschland) überprüft und gelöst].</p>
91C	<p>ENTW_OE_3 laut:"so, jetzt haben die den falschen Anhang ausgebessert. Und jetzt seh ich, dass die da in Süddeutschland die Tabellen ja ganz falsch verwendet haben bei ihren Tests..."</p>
92 - Besprechung	<p>ENTW_OE_TL: "Mail von ENTW_DE_TL von Montag mit Aufgaben für diese Woche, das waren glaub ich drei TTs, da sind wir auf nem guten Weg was ich so gesehen habe. Eines war bei ENTW_OE_4, eines bei ENTW_OE_1 und eines bei ENTW_OE_3. Das von ENTW_OE_1 ist geschlossen, hab ich gesehen. Wie ist der Status bei dir, ENTW_OE_3?"</p> <p>ENTW_OE_3 erklärt Status. ENTW_OE_4 ist auch erledigt.</p>
93	<p>ENTW_OE_TL:"Mail von ENTW_DE_TL wegen Schreibens von Emails an Kunde: da gabs Emails wo die Form nicht passte, Rechtschreibfehler, etc... Das soll in Zukunft passen, auch mit Signatur, und MFG. Signatur-Email von Marketing sollte jeder haben, falls nicht schick ich das nochmal rum.</p>
94	<p>ENTW_OE_TL:"TTs von [Projekt B] [diese werden doppelt in TTS und JIRA gepflegt] sollen in JIRA aktualisiert werden. hab ein bisschen drübergeguckt, aber bin nicht sicher ob die noch aktuell sind. Da soll sich jeder um seine TTs kümmern."</p>
95	<p>ENTW_OE_TL erklärt eigenen TT-Status ("Berg, aber nicht arg"), 5 Stück, mache diese und sonst momentan nichts [keine IMPL oder Nachtests].</p>
96	<p>ENTW_OE_5: "mache Summenanpassung-Druckoutput, wenn ich fertig bin werde ich das mit ENTW_OE_4s Code mergen." Ansonsten 4 TTs</p>
97	<p>ENTW_OE_3: "10 TTs, kriege derzeit Trunk-TTs, die sind kritisch. Habe ein Problem [TT], das betrifft auch andere Sparten. Habe das Gefühl das war ein Problem wegen einem Refactoring. Habe es jetzt in Absprache mit RMM_2 auf realisiert gesetzt". ENTW_OE_TL meint, dass das schon viel ist, und dass er versuchen solle, wieder auf 5 oder weniger zu kommen. ENTW_OE_3:"naja, hab ja schon viele gefixt, 8 bis 10 Stück oder so, aber jetzt kommen halt andere Sachen hoch. Wahrscheinlich wegen Merge von Branch auf Trunk in Eile ohne ordentliches Testen auf Trunk-Version. ENTW_OE_TL versteht [ist etwas beruhigt], und meint dass man da wohl mal zwei ruhige Tage ohne frische TTs bräuchte.</p>
98	<p>ENTW_OE_1: bei ihm fehlen 3 TT-Analysen. "Eines ist ein Produktkern-Problem (Anm.: dafür sind andere Kollegen verantwortlich), das hab ich schon mit ENTW_OE_3 besprochen. Zweite Analyse betrifft Stufenanpassung, da brauche ich Zeit das fachlich zu verstehen [siehe auch Whiteboard-Aktion zu Mittag mit ENTW_OE_4]. Dritte Analyse muss ich noch nachtesten, konnte ich noch nicht genau lokalisieren. Da gab es einen Absturz den ich jetzt nicht mehr hinkriege." ENTW_OE_TL fragt nach ob sich bei dem Absturz um eine Nullpointer-Exception handelt. ENTW_OE_1 erklärt den Absturz genauer, erklärt weiter dass der Fehler in Version 3.0.16 auftrat [veraltete Version, aktuell 3.0.19] und dass es dafür mittlerweile gar keine Testumgebung mehr gibt. ENTW_OE_TL fragt nach weiteren Details zu dem Fehler und der Version nach. ENTW_OE_1 meint, er werde die Version lokal auschecken, da meint ENTW_OE_3 dass er nicht sicher ist, ob diese Version noch auf dem lokalen Server gespiegelt ist. ENTW_OE_TL meint, dass ENTW_OE_1 wegen des Fehlers und genauerer Daten ev den Kunden anrufen sollte.</p>
99	<p>ENTW_OE_4 macht Hausrat-Summenanpassung, erwähnt dass es jetzt auch Tests dazu gibt. ENTW_OE_TL findet das gut, und meint, dass die dann auch in die QS 'reingehängt' werden.</p>
100	<p>ENTW_OE_TL zu ENTW_OE_4:"hast du in den Hausrat-Aufgaben [JIRA] geschaut, ob es da überfällige gibt?" ENTW_OE_4:"das wollte ich am Freitag machen und auch mit dir besprechen. Die bisher überfälligen haben wir eingeholt, am Freitag wollte ich schaun, und ev müssen wir dann was streichen." ENTW_OE_4 zu ENTW_OE_5:"Du hast deine Aufgaben bis 8.Juli, oder?" ENTW_OE_5 bejaht.</p>
101	<p>ENTW_OE_6 hat zwei TTs, erklärt TTs und sein Vorgehen. ENTW_OE_TL sagt ihm noch wie er sie nachtesten soll.</p>
102	<p>[Runde fertig]</p>
103	<p>ENTW_OE_TL:"hab mit ENTW_DE_TL telefoniert, es passt alles soweit, er macht einen entspannteren Eindruck. Liegt vlt auch daran dass er gestern frei hatte. Nachtests sehen auch gut aus."</p>
104	<p>ENTW_OE_TL:"hatte gestern ein unangenehmes TT, nämlich bei der Vertragssuche." ENTW_OE_4:"das kenn ich, dieses TT ist glaub ich schon bei jedem mal gelegen!"</p>

	<p>ENTW_OE_TL: "ja, ich muss es jetzt auch nochmal liegen lassen, werd mir das Freitag nochmal ansehen." ENTW_OE_4: "Problem ist, dass das falsch implementiert ist." ENTW_OE_TL: "weiss auch nicht, ich hab alles probiert, hab XMLs ausgetauscht, etc.." weitere Fehlermöglichkeiten werden von ENTW_OE_TL und ENTW_OE_4 erörtert.</p>
105	[Projektspezifische Erörterungen]
106	ENTW_OE_3 laut: "hat jemand schon mal migrations-notes gemacht? ENTW_OE_4 , hast du schon mal Migrationsnotes gemacht?" geht rüber zu ENTW_OE_TL um zu fragen.
107A	ENTW_OE_3 laut: "was, 'Error'?" ENTW_OE_1 : "Hä?" ENTW_OE_3 : "Error bei durchführung von deinem Test. Aber nicht der Error den du haben wolltest...". Ansonsten keine grossartigen weiteren Reaktionen, beide arbeiten weiter an ihren Sachen. (Anm: ENTW_OE_3 testet ein TT nach, von dem ENTW_OE_1 meinte dass es den Produktkern betrifft, und das früher funktionierte)
107B	ENTW_OE_3 fragt ENTW_OE_1 wegen Fehlerdetail, bzw. äussert er eine Vermutung was den Fehler betrifft. ENTW_OE_1 : "hm, nein, das wars nicht, warte mal, ich muss das kurz nachschauen, ich hab das eh hier offen." ENTW_OE_1 sieht in Email nach, kopiert eine Stelle raus und schickt es im Chat an ENTW_OE_3 . ENTW_OE_3 : "aber schliesst ihr da ein Element aus?" ENTW_OE_1 : "nein, machen wir nicht. Ich habs nur so gemacht wie ichs immer gemacht hatte, da wird nichts ausgeschlossen." Die beiden erörtern weitere Details über die Fehler-Rekonstruktion, ENTW_OE_3 fragt genauer nach wie die Vertrags-Konfiguration war ("automatische Beginnverlegung oder händisch?"). Wo der Fehler jetzt genau herkommt ist noch nicht ganz klar (ob aus dem Produkt oder der Bestandsverwaltung).
107C	ENTW_OE_1 : "Achso, da wird ja tatsächlich etwas ausgeschlossen! Von wem ist das?" [sieht im Code nach] " ENTW_DE_TL_STV_2 , am 15.12. Das ist ja schon uralt! Aber ich hab auch keine Ahnung warum er da etwas ausschliesst". Kurz darauf ENTW_OE_1 : "ich glaube da muss ich mich an ENTW_DE_TL_STV_2 wenden."
108	ENTW_OE_5 chattet mit ENTW_DE_TL_STV_1 wegen Problem in Hausrat ("Erzeugung von Vertragsterminen in Dialog oder Abschlussbatch?". ENTW_DE_TL_STV_1 antwortet, aber ENTW_OE_5 kann damit nichts anfangen, bzw. ist für ihn das im FK falsch angegeben. ENTW_OE_5 macht sich gestisch bei ENTW_OE_4 bemerkbar, damit dieser ihn hört da er Kopfhörer drin hat, daraufhin setzt er sich zu ENTW_OE_5 an den Platz und liest den bisherigen Chatverlauf. ENTW_OE_4 versteht die Antworten von ENTW_DE_TL_STV_1 besser, und erklärt ENTW_OE_5 was er meint.
109	ENTW_OE_1 und ENTW_OE_4 sehen sich gemeinsam ein wiedereröffnetes TT in TTS an. Da das TT bereits seit 9 Monaten immer wieder wiedereröffnet wird sehen sie die Liste von Kommentaren durch, welche Personen für das TT implementierten und den Verlauf bzw. die "Geschichte" des TTs. Die Liste enthält ca 40 Einträge. ENTW_OE_4 und ENTW_OE_1 sehen sich auch die gebuchten PT [Personentage] an (3.84), was ihrer Meinung nach zu wenig ist. ENTW_OE_4 bemerkt, dass er nicht immer alle Arbeitszeiten auf das TT gebucht hat.
110	ENTW_OE_TL kommt ins Zimmer zu ENTW_OE_3 : "Du, das eine TT steht jetzt auf testbereit!?" ENTW_OE_3 : "ja, das hab ich vorhin gesetzt" Anschliessend diskutieren sie wie mit dem TT weiter verfahren wird, wo das Problem liegt und wer und wie der Nachtest durchgeführt wird. ENTW_OE_TL schlägt vor, dass auch ein Email an ENTW_DE_TL geschrieben wird mit einer Erklärung über das Vorgehen. ENTW_OE_TL : "wenn nämlich TST_1 den Nachtest durchführen würde, würde er das TT wieder aufmachen. Wenn wir es aber so wie der Kunde machen, dann sollte zumindest ENTW_DE_TL davon wissen."
111	ENTW_OE_TL kommt zu ENTW_OE_1 , er fragt ihn wegen eines TTs, bzw. der TT-Analyse. Dazu setzt er sich zu ihm, sie suchen eine Email die von ENTW_DE_TL geschickt wurde [siehe auch Team-Meeting gestern wg Todo-Liste von ENTW_DE_TL], ENTW_OE_TL zeigt auf das TT, sie öffnen es und öffnen die TT-Analyse die am TT angehängt ist. ENTW_OE_TL macht ENTW_OE_1 auf kleinere Fehler aufmerksam die im aufgefallen [schon vorher] waren, und bittet ihn die auszubessern, bzw. fragt nach ob es tatsächlich so gehört. ENTW_OE_TL betont, dass die Analyse und der Testverlauf für den Kunden nachvollziehbar sein müssen.
112	ENTW_OE_1 arbeitet an Nachtest für ein TT, die Schwierigkeit besteht darin, den genauen/richtigen/komplexen Testablauf nachzustellen. Kunde, Tester, ENTW_OE_3 , ENTW_OE_4 und ENTW_OE_1 haben das TT schon auf verschiedene Arten nachgetestet. ENTW_OE_1 fragt ENTW_OE_3 auf welche Art er nachgetestet hat, welche Elemente eingeschlossen wurden, etc..
113	ENTW_OE_3 bekommt einen Rückruf von ENTW_DE_TL , ENTW_OE_3 erklärt ihm dass es um ein TT geht, gibt ihm die Nummer durch. Erklärt ihm weiter das Problem, dass er es gemeinsam mit ENTW_OE_1 debuggt hat, und dass er das TT jetzt an ENTW_OE_1 abgegeben hat, da der Fehler in dessen Zuständigkeit liegt.
114	ENTW_OE_3 und ENTW_OE_1 besprechen kurz ob für ein TT eine Analyse gemacht

	werden muss, das TT hat schon 'Go-Hotfix', aber nicht sicher ob Analyse erforderlich ist und in welchen Branch es gehört. ENTW_OE_3 meint, er werde ENTW_DE_TL_STV_2 anrufen. Dieser ist gerade in Süddeutschland , ENTW_OE_3 ruft in Süddeutschland an, da er von ENTW_DE_TL weiss, dass ENTW_DE_TL_STV_2 bei Kunde_Mitarbeiter_1 im Büro sitzt, und er von ENTW_DE_TL dessen Telefonnummer bekam. In Süddeutschland ist ENTW_DE_TL_STV_2 momentan nicht im Zimmer.
115	ENTW_OE_TL kommt zu ENTW_OE_1 ins Zimmer, will den Status für ein TT wissen, bzw. ob er den Fehler schon gefunden hat. ENTW_OE_1 zeigt ihm die fehlerhafte Stelle, ENTW_OE_1 , ENTW_OE_3 und ENTW_OE_TL besprechen den Fehler. ENTW_OE_TL meint, dass man dafür ein eigenes TT erstellen sollte, ENTW_OE_3 sagt ihm, dass sie das schon gemacht hätten. Anschliessend bereden ENTW_OE_TL und ENTW_OE_1 Vorgehen für Nachtest (zuerst Lokal, anschliessend REA, dann auf Testversion), Problem ist, dass es lokal funktioniert, auf der Testversion aber nicht.
116	ENTW_OE_5 kommt zu ENTW_OE_3 und ENTW_OE_1 , fragt sie wer für eine bestimmte Sache zuständig ist. Die beiden Antworten "Releasemanagement" und nennen einige Kollegen aus dieser Abteilung.
117	ENTW_OE_TL kommt ins Zimmer: "ich hab jetzt auch grade noch ein 9er TT gekriegt." Erklärt kurz, dass es schon kritisch ist weil Bestands-Projekte betroffen sind, aber es relativ einfach zu lösen ist. ENTW_OE_TL : "Ich hatte grade ENTW_DE_TL angerufen wegen des Hotfix-Tickets, und dann fragt er 'welches?'" (Lachen)
118	ENTW_OE_3 erzählt laut dass er mit ENTW_DE_TL_STV_2 telefoniert hat, ENTW_OE_4 wird hellhörig, möchte wissen worüber sie gesprochen haben. ENTW_OE_3 erklärt dass es wegen einiger TTs mit ihm telefoniert hatte, ENTW_OE_4 : "wieso 'einige?'" ENTW_OE_1 schaltet sich ein: "beim Testen eines TTs sind drei neue TTs aufgetaucht."
119	ENTW_DE_TL_STV_2 ruft bei ENTW_OE_3 an, möchte Problem-Beschreibung eines TTs wissen. ENTW_OE_3 erklärt ihm und sie diskutieren die Ursache/Lösung. Dauer bis ca 18:15.
120 - Besprechung	Anwesend: ENTW_OE_TL , ENTW_OE_5 , ENTW_OE_4 ; ENTW_OE_1 in Norddeutschland , ENTW_OE_2 Urlaub, ENTW_OE_3 krank.
121	ENTW_OE_TL : "TT-Nachlieferungen (v. 3.0.019) sind momentan ma wichtigsten, da kam auch eine Email von ENTW_DE_AL ". ENTW_OE_5 : "War das eine Email an alle?" ENTW_OE_TL : "Nein, nur an die Betroffenen [Anm: ENTW_OE_5 macht Hausrat, bei diesem Projekt muss nichts nachgeliefert werden]. Er hatte Freitag Abend noch Fixes [Anm: Bugfixes] gemailt, die heute rausgehen sollen." ENTW_OE_4 hat ja irgendein Problem mit einem TT.
122	ENTW_OE_5 : "Habe 4 TTs, sonst Summenanpassung [Anm: Implementierungs-Aufgabe für Hausrat]. Wollte Freitag auf Trunk mergen, aber hatte Probleme mit [dem mergen von] XML-Dateien. Da wurden anscheinend Sachen auf Bausteinen geändert - ist das möglich? Da bin ich nicht sicher. Ich hatte jedenfalls deswegen Probleme, und musste alles händisch mergen." ENTW_OE_TL : "Ok. Ich würde mir deine Sachen dann gerne Donnerstag oder Freitag im Rahmen eines Code-Reviews ansehen [Anm: Code und Funktionalität von Summenanpassung]. Es wäre auch gut, wenn bis dahin auch alles fertig ist."
123	ENTW_OE_4 : 1TT, ENTW_OE_4 erklärt kurz: "Weiss noch nicht wo der Fehler ist, tritt bei einem Hintergrundjob auf. Das ist total schlampig implementiert . Problem tritt auch nur beim Kunden auf, nicht lokal und nicht in der VersicherungsGmbH -Testversion." ENTW_OE_TL : "Was sind deine nächsten Schritte?" ENTW_OE_4 : "keine Ahnung" ENTW_OE_TL macht Lösungsvorschlag ["Liegt es vielleicht an der DB-Konfiguration?"], beide diskutieren möglich Schritte. Ev soll der Kunde angerufen werden, ENTW_OE_TL fragt nach den Tests beim Kunden. ENTW_OE_4 sagt, dass es beim Kunden eben nicht funktioniert, und dass er letzten Freitag auch deswegen ein Email an den Kunden geschrieben hatte. ENTW_OE_TL müssen halt den Fehler oder die Fehlerursache einschränken, und verschiedene Konstellationen ausprobieren - wenn der Fehler nicht im Coding gefunden werden kann. Ev müssten wir den Kunden um weitere Tests bitten." ENTW_OE_4 meint er würde sich die Kunden-DB ansehen, "ich kann aber nicht vom Kunden verlangen, jetzt viele Tests deswegen zu machen. Ich weiss auch nicht, auf welchem DB-Schema der Kunde gearbeitet hat. Schema xy [Anm: Kunden-Testschema] ist von uns aus ja auch gar nicht erreichbar. Ich mache das über ENTW_OE_1 [Anm: der ist gerade in Norddeutschland], dort haben sie eine bessere Verbindung."
124	ENTW_OE_TL : "Ok, ich selbst hab 3 oder 4 TTs, das meiste davon Hausrat. Das hat auch mit einem Hotfix von ENTW_OE_8 zu tun - er hatte da was einchecked bevor er auf Urlaub ging, und das funktionierte noch nicht richtig. Den anderen Fehler in der Vertragssuche [Siehe Fall-ID 104] hab ich gefunden, das lag an xy [kurze Beschreibung des Fehlers]. Das war eigentlich im VWB-XML. ENTW_OE_3 ist heute krank, der hat ein kritisches TT falls jemand fragt. Da wird eine falsche Prämie berechnet." ENTW_OE_4 : "Ist

	das in [Dialogmaske] [Anm: diejenige wo der Fehler auftritt]?" ENTW_OE_TL :"Ja genau, aber das werde ich mir mit ENTW_OE_3 morgen ansehen."
125 - Besprechung	ENTW_OE_TL, ENTW_OE_4, ENTW_OE_5, ENTW_OE_3, ENTW_OE_1, ENTW_OE_2
126	ENTW_OE_TL :"Wir haben Nachlieferungen zu machen, ENTW_DE_TL schreibt ja auch immer Mails dazu aus, ich weiss nicht ob ihr die bekommen habt, aber jedenfalls sieht das grade ganz gut aus, die Ampel ist auf 'grün'. PL1 wird nächste Woche abgespalten, da sind noch ein Hotfix und ein paar TTs nachzuliefern."
127	ENTW_OE_TL :" ENTW_OE_1 macht ja diese Woche seine Sun-Zertifizierung, ich habe heute selber schon dagegen verstossen, aber eigentlich sollten wir ihn diese Woche nicht stören [Anm: ENTW_OE_1 hat zum Lernen Bildungs-Urlaub, ist aber im Büro und lernt dort. Am Morgen beim Kaffeetrinken erzählte er ENTW_OE_4 dass er seinen Sametime-Account auf 'bitte nicht stören' gestellt hat. Zum Lernen sitzt er nebenan im Besprechungsraum.]
128	ENTW_OE_TL :"Ich habe noch ein TT von ENTW_DE_TL_STV_2 , da ist ein Fehler bei der Schwebel, deswegen hatte ich auch ENTW_OE_1 gefragt [siehe Fall-Id 127]. Ansonsten habe ich aber nix 'Arges'".
129	ENTW_OE_TL zu ENTW_OE_2 :"Bei Wohngebäude hatte ich ein TT von ENTW_OE_6 übernommen, ich habs durchgespielt, aber vielleicht kannst du dir das nochmal ansehen." ENTW_OE_2 :"Hast du schon was gemacht dazu?" ENTW_OE_TL :"Was ENTW_OE_6 mir gegeben hatte habe ich eingesehen, aber das TT ist noch nicht geschlossen." ENTW_OE_2 :"Ok, ich seh mir das an."
130	ENTW_OE_TL zu ENTW_OE_1 :" ENTW_DE_TL hat mich angerufen wegen des Hotfix" ENTW_OE_1 :"Das hab ich ENTW_OE_4 gegeben"
131	ENTW_OE_3 :"Ich hab zwei Analysen zu machen." ENTW_OE_TL :"hast du nicht nächste Woche Urlaub?" ENTW_OE_3 :"Ja, da müssen wir eh schauen wegen einchecken." [Anmerkung: Analyse muss diese Woche erledigt sein, i.e. Fehler muss diese Woche gefixt werden, Code kann aber erst nächste Woche - sobald Branch PL1 freigegeben ist - eingesehen werden.] ENTW_OE_TL :"Sollen wir das so machen dass du die Sachen vorbereitest, und wir es dann einchecken?" ENTW_OE_3 :"Ja, so irgendwie. Eventuell frage ich aber auch ENTW_DE_1 oder ENTW_DE_2 ob die das machen können."
132	ENTW_OE_2 :"Ich hab 30 Tasks bis Ende Juli!" Zu ENTW_OE_TL :"Damit du dich aber nicht schreckst: die sind zwar bis Ende Juli, aber ENTW_DE_TL weiss schon Bescheid, und wahrscheinlich werden die eh verschoben. ENTW_DE_TL meint ich soll halt mal machen, und wir verschieben dann am Ende des Monats, aber eigentlich könnten wir das auch gleich verschieben." [Anm.: ENTW_OE_2 war eben drei Wochen auf Urlaub, bzw. Zeitausgleich] ENTW_OE_2 weiter:"Also ENTW_DE_TL weiss Bescheid, und es wird wohl verschoben. Das ist halt auch weil ich weniger Stunden mache [Anm.: 30 Stunden / Woche], und ENTW_DE_6 in ein anderes Projekt kommt. Die meisten TTs kommen dann wieder zurück zu ihr. Ich dachte ich werde meine TTs diese Woche liegen lassen und mich erst noch auf die Implementierungs-Tasks kümmern, weil sonst führt das nur zu weiteren Verzögerungen. Ich muss auch nächste Woche nach Norddeutschland ." ENTW_OE_3 :"Machst du da Wohngebäude?" ENTW_OE_2 :"Ja"
133	ENTW_OE_5 :"Ich habe 7 TTs und Summenanpassung ." ENTW_OE_TL :"Oh, das ist wieder hochgegangen?" ENTW_OE_5 :"Ja, ich hab auch eine Email wegen Summenanpassung geschrieben." ENTW_OE_4 :"Ja, hab ich gesehen" ENTW_OE_5 erklärt das Problem, und sagt dass er nach wie vor an der Sache arbeitet. ENTW_OE_TL :"Gibts dafür einen Ansprechpartner?" ENTW_OE_5 :"keine Ahnung, ich weiss niemanden. Das müsste ich erst herausfinden." ENTW_OE_4 :" AH meinte er sieht sich das an, da sollte heute noch ein Ergebnis kommen." ENTW_OE_TL :"Ok, und wann machen wir dann Codereview?" ENTW_OE_5 :"Mittwoch?" ENTW_OE_4 :"Mittwoch geht bei mir nicht. Das macht jetzt keinen Sinn. Frühestens Freitag."
134	ENTW_OE_2 macht einen Einwurf wegen Review: ENTW_OE_2 :"Ist Review jetzt ein 'Impl-Task'?" ENTW_OE_TL :"Ja, das ist jetzt als Impl-Task eingestellt [Anm: in JIRA]. Ich weiss aber nicht wie das bei dir ist - ich frag mal ENTW_DE_TL deswegen." ENTW_OE_2 :"Es war ja eigentlich so geplant, oder? War die letzten drei Woche jemand aus Norddeutschland hier in Österreich ? Wegen dem Projektmanagementtool oder wie das heisst - das sollte ich euch mal zeigen. Hat sich das keiner von euch angeschaut?" ENTW_OE_TL :"Also bei mir klingelt es da nicht.." ENTW_OE_2 :"Da wurden doch mal Folien geschickt. ENTW_DE_TL_STV_1 meinte, ich soll euch das mal zeigen." ENTW_OE_TL :"hm, ich frag mal ENTW_DE_TL oder ENTW_DE_TL_STV_1 deswegen."
135	ENTW_OE_5 erklärt eines seiner TTs weiter (eines das 'wieder aufgegangen' ist), erzählt, das jemand aus Norddeutschland mit ENTW_DE_TL_STV_1 darüber spricht. Laut ENTW_OE_5 ist aus den Kommentaren im TTS ersichtlich dass es wieder aufgegangen ist, und dass er sich die Historie angeschaut hätte, wo er auch gesehen hat, dass er selbst schon

	daran gearbeitet hätte.
136	ENTW_OE_4 : "Habe jede Menge TTs, auch von ENTW_OE_1 und zum Thema [Funktionalität XY] ." ENTW_OE_TL : "Der Reihe nach - da ist jetzt erstens mal das Hotfix-TT, und zweitens dann die [Funktionalität XY] ?" ENTW_OE_4 : "Ja" ENTW_OE_TL : "Anzahl?" ENTW_OE_4 : "4" ENTW_OE_3 zu ENTW_OE_4 : "Du sollst auch den ENTW_DE_TL zurückrufen [Anm.: zu Beginn des Meetings läutete das Telefon, ENTW_OE_3 ging ran]. ENTW_OE_4 : "Was?" ENTW_OE_3 und ENTW_OE_1 : "wegen dem Hotfix-TT"
137	ENTW_OE_TL : "Na ich freu mich jedenfalls dass wir wieder die ganze Mannschaft das sind, obwohl nächste Woche gehen ja auch ENTW_OE_3 , ENTW_OE_5 und Mitte August ENTW_OE_4 auf Urlaub."
138	Im Hinausgehen ENTW_OE_4 zu ENTW_OE_3 : "Wie ist ENTW_DE_TL s Telefonnummer?" ENTW_OE_3 : "xyz"
139	ENTW_OE_3 wählt Telefon: "Hallo ENTW_DE_TL_STV_2 , hab da ein TT auf das ich dich aufmerksam machen wollte. [Beschreibt Fehler]. Also das betrifft da ziemlich viele Sachen, drum wollte ich dir Bescheid geben. [Beschreibt Fehler und mögliche Auswirkungen weiter]. Ok, bitte. Ciao."
140	ENTW_OE_TL zu ENTW_OE_3 : " ENTW_DE_TL meinte, wenn du 'nicht da' bist, solltest du dich im Sametime abmelden." ENTW_OE_3 : "ja, das hab ich eh gemacht, aber ich könnte noch eine 'out-of-office' Nachricht machen."
141	ENTW_OE_3 bekommt Rückruf von ENTW_DE_TL_STV_2 (ENTW_OE_2 macht ihn aufs Telefon aufmerksam - ' ENTW_DE_TL_STV_2 ruft an!'), sie sprechen kurz. Danach ENTW_OE_3 zu ENTW_OE_2 : "Das TT wird wahrscheinlich ein Hotfix."
142	ENTW_OE_3 am Telefon: "Hallo, hast du noch Zeit? Ich hab noch ein anderes TT. (...) Es geht um KFZ. (...) Das ist TT-Nummer xyz. (...) Wenn du da unten in die Kommentare reinschaust, siehst du, dass die da auch mit einem Fremdsystem reingehen. (erklärt kurz weiter was geschieht) (...) Ja genau (...) Meinst du ich soll denen sagen sie können das nicht machen, oder soll ich da das Interface anpassen? [reden über mögliche Auswirkungen]. Ok, dann mach ich das mal so. Ciao."
143	ENTW_OE_TL kommt zu ENTW_OE_3 : "Kann ich dich kurz was fragen?" ENTW_OE_3 : "Ja" Die beiden gehen rüber in ENTW_OE_TL s Büro.
144 - Besprechung	ENTW_OE_TL , ENTW_OE_1 (lernt), ENTW_OE_5 , ENTW_OE_3 , ENTW_OE_4 , ENTW_OE_2 - eher kurzes Meeting, da wenige Probleme momentan
145	ENTW_OE_TL erzählt dass er keine TTs hat, ausser ein paar Nachlieferungen. Fragt Status von ENTW_OE_5 zu einer bestimmten Aufgabe (Terminverarbeitung in Hausrat), die beiden diskutieren das Vorgehen. Dabei werden auch andere Entwickler und Personen aus Norddeutschland erwähnt, die daran beteiligt sind. Auch ENTW_OE_4 schaltet sich kurz dazu ein, um seinen Stand von der Entwicklung zu erzählen.
146	ENTW_OE_5 hat 7 TTs, erklärt kurz.
147	ENTW_OE_3 : "komme gegen 10 TTs" - ENTW_OE_TL : "von oben oder von unten?" ENTW_OE_3 erklärt kurz. ENTW_OE_TL : "Was hältst du für realistisch bis Ende der Woche [Anm.: Reduktion der TT-Anzahl]? Ansonsten gib mir Bescheid wenn es mehr wird."
148	ENTW_OE_4 erklärt kurz seine Aufgaben, keine TTs. ENTW_OE_TL sagt, er könne ihm TTs geben.
149	ENTW_OE_2 : "Ich mache technische Implemetierung, da hab ich auch einen Task dazu, aber die sind zu grob aufgeteilt, und ich kann sie nicht einzeln schliessen, sondern erst wenn diese eine fertig ist, da alle Ausfertigungsgründe [Anm.: und somit mehrere andere Tasks] davon abhängig sind." Ansonsten habe ich 5 TTs, aber die kann ich erst in Norddeutschland machen.
150	ENTW_OE_2 zu allen noch in der Besprechung: "Mein Office wird gerade installiert [Anm.: Remote-Installation bzw. Update von MS-Office], deshalb kann ich jetzt am Nachmittag nichts machen." ENTW_OE_1 : "wird das bei allen gemacht?" ENTW_OE_4 : "Nur wenn dein Computer aufgedreht ist". Kurz später (alle sind wieder an ihrem Platz), ENTW_OE_1 zu ENTW_OE_3 : "Geht das automatisch mit der Office-Installation?" ENTW_OE_3 : "Nur wenn dein Rechner aufgedreht ist. Hast du einen Termin vereinbart?" Daraufhin greift ENTW_OE_1 zum Telefon und ruft eine Person in Norddeutschland an [Anm.: ohne eine Nummer nachzufragen]. Er erklärt dass sein Rechner diesen Nachmittag frei wäre für das Update.
151	ENTW_OE_3 greift zum Telefon: "hallo, du ich weise dir ein TT zu, es geht dabei um TT-Nr xyz." Er erklärt kurz das Problem, und warum er es dem Kollegen/der Kollegin zuweist.
152	ENTW_OE_4 schickt eine Chatnachricht an ENTW_OE_TL : "Guten Morgen, hast du ein Minütchen für mich? Vertragsnummer XY." Kurz darauf läutet bei ENTW_OE_4 das Telefon,

	<p>er hebt ab:"Hallo ENTW_OE_TL, (...), ich hörte du schreibts gerade die Testfälle für den Anwendungsfall xy (...)." Im Folgenden diskutieren sie diesen Anwendungsfall und stimmen sich darüber ab. Insbesondere müssen Daten für einen existierenden Vertrag im Testsystem befüllt werden, ENTW_OE_4 meint, dass ENTW_DE_TL_STV_1 ebenfalls schon Daten für diesen Vertrag eingetragen hat. ENTW_OE_4:"Vertrag 1, der hat funktioniert. Ich schicke dir die Nummer. (...)" ENTW_OE_4:"Habt ihr eigentlich schon die Fälle für den Changerequest?" (...) ENTW_OE_4:"Der TST_1 macht ja diese Fälle als automatisierten Test, oder?" (...). Die beiden reden noch eine Weile weiter.</p>
153	<p>Während des Telefonats hat ENTW_OE_4 ein Excel-File geöffnet das in einem Ordner auf einem Teamlaufwerk liegt. Dieses Verzeichnis wurde für [Projekt B] angelegt, darin werden Informationen und Daten für Performance- und fachliche Tests, sowie Anwendungsfälle abgelegt. Zugriff darauf haben ENTW_OE_4, ENTW_OE_1, ENTW_DE_TL_STV_1, ENTW_OE_TL, ENTW_DE_TL_STV_2, ENTW_DE_TL und TST_1.</p>

Appendix F – Interviews

Interview 01_ENTW_DE_1

I: So, Tonband läuft. (...) So jetzt erste Frage einmal: was ist deine Aufgabe hier im Team? Ich glaube du bist Entwickler. 00:00:14-5

B: Genau. Softwareentwickler im Bereich Produktentwicklung, Dialog-Entwicklung so ein bisschen, und Leistung mach ich auch mal ein bisschen was. Also hauptsächlich Softwareentwicklung. 00:00:26-9

I: OK. Und was sind jetzt zum Beispiel deine alltäglichen Aufgaben im Kleinen, also. 00:00:35-8

B: Ja unterscheidet sich im wesentlichen erst mal in Neuentwicklung und das Beheben von Tickets. Also am Anfang vom Meilenstein wird neu entwickelt, das heißt die Anforderung in Form von einem Konzept, dann setzt man die um, schreibt Tests dafür und am Ende vom Meilenstein häuft es sich dann dass die Software natürlich getestet wird. Es kommen Fehler, und die werden dann halt in Form von Tickets aufgegeben, und ich behebe die Tickets. 00:01:14-5

I: Und mit welchen Leuten arbeitest du dabei zusammen? 00:01:14-5

B: Ja, für fachliche Fragen mit verschiedensten Fachkonzeptern, und ansonsten hauptsächlich mit **ENTW_DE_TL_STV_1**, ja und Teamkollegen halt wenn man Fragen hat. Aber ansonsten arbeite ich eigentlich alleine, aber mit der Fachkonzeption hat man schon mehr zu klären, auch wirklich. 00:01:40-2

I: Und zum Beispiel mit der Testabteilung oder so? Hast du mit denen viel zu tun? 00:01:42-6

B: Ja, da gehts dann meistens um irgendwie so Klärungssachen, wenn Tickets nicht verständlich sind, oder so Sachen. Aber, auch nicht so viel. Aber da läuft viel über dieses Ticketsystem. Da gibts dann wohl Fragen, aber sonst. Eher weniger. 00:02:00-8

I: Und sind die Leute mit denen du zusammenarbeitest, sitzen die bei dir im Zimmer, oder sitzen die hier in irgendwo in einem anderen Zimmer, oder in **Süddeutschland** oder so irgendwo? 00:02:15-0

B: Nein, die sind alle hier in **Norddeutschland**, und auch alle in meinem Raum oder irgendwie zwei Räume weiter, und die Fachkonzeption das sind so eine Minute Fussweg. Also genau am gegenüberliegenden Gang. 00:02:29-6

I: Und gehst du zum Beispiel öfter hinüber, wenn du irgendetwas, wenn jetzt im Fachkonzept etwas unklar ist oder so? 00:02:32-5

B: Also tendenziell bin ich jemand, der ungerne Sachen über Emails und über Messenger und so weiter klärt, wenn es wirklich um fachliche Probleme geht, nicht, wenn es wirklich um die Arbeit geht, weil da finde ich bringt so ein Gespräch einfach mehr, und man, das ist viel eindeutiger, und man kriegt auch einfach mehr Feedback. Also ich lauf schon ziemlich oft. Also wenn ich wirklich was zu klären habe, laufe ich auf jeden Fall hin. Oder rufe an. 00:02:59-9

I: Ja und was würdest du sagen, wie unterscheidet sich das, eben mit Leuten, wie zum Beispiel, ich weiss nicht, ob du mit Releasemanagement oder so zu tun hast... 00:03:16-1

B: Genau, mit denen hab ich in jedem Fall auch Kontakt. 00:03:16-5

I: Gehst du dann zum Beispiel, wenn du von denen etwas brauchst, gehst du dann hinauf? 00:03:20-8

B: Nein. 00:03:23-3

I: Also das bleibt quasi schon eher auf den Stock hier beschränkt. 00:03:25-7

B: Genau. Da ist der Weg einfach schon zu weit, da müssen schon große Probleme sein. 00:03:31-4

I: Und mit denen kommunizierst du über Email, oder rufst du an? 00:03:33-9

B: Ja, Telefon, Email. Kommt auch immer darauf an, wenn man irgendetwas verbindliches braucht, also wenn

Leute etwas für einen entscheiden sollen, irgendwie so, dann in Schriftform, per Email, weil man dann einfach etwas hat, und die Entscheidung auf dem Papier steht. Ja und wenn es etwas Kleineres, belangloseres ist, dann per Telefon. Oder via Chat auch. 00:03:58-8

(Pause wegen Mikrofoncheck) 00:04:00-8 bis 00:04:27-1

I: Ja, dann, welche Programme verwendest du so in deinem Arbeitsalltag? 00:04:31-5

B: Lotus Notes, für Emails. Darin integriert teils selbst-, also viele Selbstentwicklungen, viele Eigenentwicklungen von uns. 00:04:44-3

I: Also Ticketsystem und so. 00:04:44-3

B: Ticketsystem, REA, Dokupool und das Intranet, alles in Notes. Emails komplett mit Notes. Und den Messenger. Ja. Zum Entwickeln Eclipse, da braucht man eigentlich nicht viel mehr. Und ja, Office halt. Und JIRA natürlich. 00:05:18-0

I: Ja und also Notes und Sametime nehme ich an hast du immer geöffnet, oder? Das ist bei den meisten wohl das erste was sie aufmachen. 00:05:27-8

B: Ja, genau. Emails checken, nachsehen wer schon da ist, wem man schon "Guten Morgen" sagen muss (lacht). 00:05:37-4

I: Achso, das macht man schon. Und ins JIRA, wie oft siehst du da rein? 00:05:41-6

B: Hm, eigentlich fast nur wenn ich Emails bekomme. Also immer wenn irgendwelche Aufgaben zugewiesen werden oder geändert werden kriegts du (...). Teilweise, also meistens ist es so, du kriegst eine ganze Reihe von Tasks, "Testnachweis", "Konzept" und so weiter, irgendwie eine ganze Menge, und anstelle die alle durchzulesen (Anm.: die Emails) ist dann meiner Meinung nach nicht so sinnvoll. Ich lösche die alle, und denke dann, "ok, es ist irgendetwas in JIRA passiert, ich kuck mal", und dann habe ich das eigentlich im Kopf was zu wann fertig sein muss. Dann mache ich es meistens so, dass ich immer wenn ein Email kommt ich da rein gehe, und ja, Abends dann meistens auch noch, nur um zu kucken, um den nächsten Tag so zu planen. Was liegt denn eigentlich an, was ist denn noch offen bei mir, habe ich vielleicht irgendwelche Tasks schon erledigt aber noch nicht geschlossen? Ja, einmal am Abend kann man sagen so. 00:06:34-3

I: Ok. Diese Emails die du angesprochen hattest - ist das quasi ein aggregiertes Email, wo dann von allen Tasks etwas drinnenstehen, oder bekommst du etwas von jedem Task? 00:06:43-2

B: Nein, ich bekomme zehn Emails "irgendetwas wurde getan", und das macht halt auch überhaupt keinen Sinn in der Form. Kann man sich auch wegfiltern unter Umständen. Ich hab halt einen Filter eingebaut, wie gesagt so "schupp" (Anm.: Wischgeste) - weg damit. 00:07:01-4

I: Ja dann, wenn du jetzt so im Büro bist, worüber wird da gesprochen, wenn du zum Beispiel jetzt mit deinen Kollegen redest, also jetzt einerseits informelle Gespräche, wo du jetzt nur sagst, ich weiss nicht, "wie geht es dir", oder auch mehr formelle Sachen? 00:07:24-6

B: Ja beides. Also du kennst ja den Mittagsplan von mir (Anm.: Planung des Mittagessens - wer geht mit, wohin geht man). Jeder läuft irgendwie herum, und es läuft immer wer herum und fragt was es zu Mittag gibt, das geht von elf bis zwölf halt. Ja und ansonsten geht man halt mal wo einen Kaffee trinken, meistens in andere Büros. Und bei uns im Büro rede ich mittlerweile ziemlich viel mit **ENTW_DE_4** weil ich ihn gerade einarbeite, beziehungsweise für den Ausbildungsbetreuung mache. Ja und wirklich für die Arbeit rede ich eigentlich nur mit **ENTW_DE_2**, so fachlich, und dann geht es einfach um Problemlösung, wenn man Feedback will für ein entweder fachliches oder technisches Problem. 00:08:09-2

I: Und die Leute mit denen du Mittagessen gehst, das sind jetzt nicht unbedingt die selben mit denen du arbeitest, also am Projekt arbeitest? 00:08:18-6

B: Nein, schon. Eigentlich doch. Also vor allem, es gibt halt einfach Leute die relativ viel Wissen haben. Also Dialogentwicklung bin ich noch nicht so viel drin, ich renne andauernd zu **ENW_DE_TL_STV_1** und zu **ENW_DE_TL_STV_2** und zu Leuten die das halt einfach wissen, einfach um da Informationen zu bekommen. Also das ist auch glaube ich das, was am meisten gemacht wird. 00:08:43-3

I: Das du fragst "wie implementierst du irgendetwas"? 00:08:43-3

B: Ja genau, oder "wie funktioniert etwas", das ist ja meistens ja nicht wirklich rein technisch, also nicht

Softwareentwicklungsmässig. Also du hast hier keine Fragen wie du irgendetwas strukturierst oder wie du Klassen baust, das kriegt man alles hin. Aber zu fragen, wo passiert was, wenn man gar keinen Ansatz hat, dann läuft man auf jeden Fall, also zum großen Teil zu den Dialogentwicklern, weil ich da eben gerade anfangen zu arbeiten, und da nicht so viel weiß darüber. (...) Genau. so das meiste ist was fachlich irgendwie gefragt wird. 00:09:16-5

I: Dann andere Frage: Interessiert du dich dafür was deine Kollegen im Zimmer machen? Also wenn zum Beispiel **ENW_DE_2** mit jemandem telefoniert und irgendein Problem hat, schaltest du dich dann ein und sagst "ich habe dieses selbe Problem schon gehabt"? 00:09:37-8

B: Ja. Mit **ENW_DE_2** mache ich halt ziemlich viel gemeinsam. Dieser komplette Produkt-Teil. Den hat eigentlich **ENW_DE_2** alleine gemacht, und immer wenn es irgendwie knapp wird, oder irgendwie, **ENW_DE_2** nicht mehr weiter weiß, oder es einfach zu viel ist für einen, nur dann mache ich eigentlich Produkt mit, ansonsten mache ich auch immer Dialogentwicklung oder Leistung oder irgendwas, und dann interessiert es mich schon was da geredet wird. 00:10:06-6

I: Spielt das für dich eine Rolle, woran deine Kollegen im Projekt gerade arbeiten. Also wenn du jetzt - also nicht nur Kollegen wie **ENW_DE_2**, die mit dir im Büro sitzt - sondern jetzt auch Fachkonzepter oder Tester oder so, ist das für dich wichtig, dass du weisst woran die gerade arbeiten? Also wenn du zum Beispiel ein Fachkonzept bekommst, zum Beispiel von **FK_1**, ist das für dich von Bedeutung ob sie gerade das eine macht oder das andere? 00:10:50-0

B: Eigentlich nicht, weil ich von dem nicht direkt abhängig bin. Also wenn ich anfangen zu arbeiten, dann ist deren Arbeit eigentlich schon passiert, grösstenteils, und dann gibt es bloss die Fälle dass sich irgendwann herausstellt, technisch ist etwas nicht so umsetzbar, oder nicht so einfach, oder unverständlich. Und dann ist es meistens so, dass man dieses Problem direkt zu dem hingibt, und einfach wartet und Druck macht, bis die Lösung kommt. Das sind immer so Kleinigkeiten. Genau. Das funktioniert dann meistens auch über diese Ticketsystem. Also wenn es ein Problem gibt, dann gibt es dann zu 90 Prozent in Form eines Tickets. Also ausser es sind wirklich Unklarheiten. Aber meistens ist es irgendwie so etwas wie "der Kunde will dann irgendetwas anders als es beschrieben ist". Und das sind alles so Klärfälle die gehen wirklich zur Konzeption, und das ist sehr oft in Form von einem Ticket. Und genauso ist es zum Testteam rüber, da ist es genauso. Wenn ich ein Ticket realisiere, dann ist es das einzige was ich über deren Arbeit wissen will, der Punkt wo ich sage "ich habe es gefixt, es ist testbereit, es gibt Zeitdruck, bitte mach doch mein Ticket zu. Bitte teste das Ticket nach." Das sind so die einzigen beiden Punkte, wo ich interessiert an der Arbeit anderer bin. Auf welchem Punkt die stehen. 00:12:12-0

I: Aber weisst du zum Beispiel wie **TST_1**, oder jemand von den Testern, wie bei dem gerade die Lage ist? 00:12:21-1

B: Ja, aber eher über Raucherpausen. Sonst kriegt man eigentlich nicht so viel mit. 00:12:27-5

I: Ok, du sagst dann aber schon auch zu ihm "Bitte, das ist sehr dringend". 00:12:39-2

B: Genau. 00:12:40-3

I: Und da schreibst du ihm dann eine Chatnachricht, oder was? 00:12:42-9

B: Zu den Testern laufe ich hin weil es so dicht ist. Das sind drei Räume. Und Fachkonzeption per Email oder per Chat. 00:12:58-1

I: Schaust du zum Beispiel, bevor du jemanden anrufst, schaust du da in die Kontaktliste von Sametime oder der online ist? 00:13:17-2

B: Ja. 00:13:17-2

I: Spielt das für dich eine Rolle, wie deine Kollegen gelaunt sind? Wenn du zum Beispiel in der Früh siehst einen Kollegen, und der zieht so ein Gesicht, spielt das eine Rolle, ob du den dann, ich weiss nicht, direkt hingehst zu ihm wenn du etwas wissen willst, oder ob du ihm dann lieber ein Email schreibst, oder ihn vielleicht in Ruhe lässt? 00:13:48-0

B: Nein. Da muss man privates und Arbeit trennen können. Also eigentlich nicht. Bei uns hat auch selten mal jemand so richtig schlechte Laune, von daher... 00:14:01-0

I: Wenn man sich ansieht, wenn irgendwelche Dokumente geändert werden, seien es Fachkonzepte oder irgendwelche Stellen im Code, wie wichtig ist das für dich, dass du da Bescheid weisst. Dass es vom Fachkonzept zum Beispiel eine neue Version gibt. 00:14:26-2

B: Das ist super wichtig. [00:14:29-2](#)

I: Und das erfährst du wie? [00:14:29-2](#)

B: Das läuft auf jeden Fall über JIRA. [00:14:34-9](#)

I: Und kriegst du da eine Emailbenachrichtigung? [00:14:34-9](#)

B: Genau. Da krieg ich die Emailbenachrichtigung und denk mir so, ok, was ist das, und überfliege halt, ok, zehn neue Tasks oder zwei geänderte Tasks, kuck mal nach, und dann, ja wenn es nur zwei sind, dann kannst du natürlich schon mal auf die Email kucken was ist es denn, und dann hast du schon mal eine Idee was es sein könnte. Und dann gehe ich aber ins JIRA und sieh nach wie sehen meine Aufgaben aus, wieviele habe ich. Da hab ich ungefähr einen Stand wieviel von welcher Art von Tasks ich noch zu tun habe, und dann seh ich einfach, ok, da ist einer dazugekommen, kuck da rein und find die relativ fix (rasch). [00:15:08-6](#)

I: Und zum Beispiel irgendwelche Änderungen im Code? [00:15:11-0](#)

B: Machen wir gar nicht. Also nicht über JIRA oder Ticketsystem oder irgendwelche anderen Sachen. [00:15:21-6](#)

I: Das geht also rein über Eclipse. [00:15:20-9](#)

B: Genau. SVN, du checkst irgendwas aus und gehst davon aus ... [00:15:28-5](#)

I: (Unterbricht) Sprichst du dich zum Beispiel mit Kollegen ab, wenn du weisst, dass der an dem selben Problem arbeitet, oder in der selben Klasse unterwegs ist? Oder weisst du das? Oder arbeitest du einfach drauf los? [00:15:40-1](#)

B: Mach ich mir eigentlich keinen Kopf darüber, weil das Mergen mit SVN eigentlich prima geht, und weil es eigentlich auch so ist, dass ich selten mal mit jemandem an dem gleichen Problem arbeite, so dass man wirklich mal so drei, vier, fünf Klassen hat die man zusammen bearbeitet. Das passiert super selten. Wenn dann sind es mal so Sachen dass Produkt und Dialog sich verständigen müssen. Wann checkt Produkt seine Änderungen ein, und wann müssen die Datenbankänderungen durch sein. Dass man sich da einfach abspricht. Läuft aber auch Teamintern, das heisst, ich renn auch über den Flur, lieber als das anders zu machen. [00:16:21-7](#)

I: Ja dann ein paar Fragen zu JIRA und zu anderen Programmen. Und zwar: sind, wenn du die Kontaktliste im Sametime hast, wie viele Leute hast du da drinnen, und hast du diese Leute auch in JIRA auf dem Radar? [00:16:52-0](#)

B: Ich habe in meinem Sametime zwei Gruppen. Einmal das Team, inklusive **ENTW_DE_AL**. [00:16:58-4](#)

I: Team sind die Entwickler? [00:16:58-4](#)

B: Genau. Also mein Team, Komposit. Und dann hab ich eine Liste von allen die bei **VersicherungsGmbH** irgendwie erfasst sind. Und da scrolle ich dann einfach. Wenn ich irgendjemanden suche, und ansonsten, die im Team liegen, mit denen man öfter mal chattet irgendwie. Die hab ich da noch extra. Ja. Und im JIRA ist es eher so, dass wenn ich mal, also auf meiner Startseite sehe ich wer zuletzt welche Tasks bearbeitet hat, aber da kuck ich auch nicht gezielt nach. Wenn ich es mal sehe, dann denke ich so "ach, kuck mal der arbeite da und da dran". Ist aber eher interessant für Leute ausserhalb des Teams, dass man sieht, die arbeiten auch mit JIRA. Das ist immer ganz interessant, dass man sieht was tun die gerade, und da einfach so aus Neugierde, aber Teamintern kuck ich da nicht drauf wer welche Tasks bearbeitet, oder wer welche Tasks schliesst. [00:17:56-8](#)

I: Ok. Und JIRA im Vergleich zum Ticketsystem. Was sind da die Vorteile und Nachteile von jedem System? Das ist jetzt vielleicht eine etwas grosse Frage. [00:18:13-4](#)

B: Ja. Es wird halt für zwei verschiedene Aufgaben genutzt meine ich. Prinzipiell glaube ich, kann man das Ticketsystem ablösen mit JIRA, allerdings weiss ich nicht, wie man das dann... . Das kriegt man ja alles nicht in ein Projekt, weil da wirklich so viel passiert. Das es mit der Oberfläche bestimmt gut zu bewältigen wäre. Man kann da Screenshots dranhängen, und eigentlich alles was unser Ticketsystem kann, kann JIRA sicherlich auch, wäre auch irgendwie schöner finde ich, wenn man seine neu, also die Tasks die man hat die sind ja meistens für Neuanlage und Tests, die Tickets sind ja eher für Fehlerbehebung. Momentan ist es getrennt, und ich denke es würde auch zusammenpassen wenn man das vernünftig strukturiert, also einfach mal ein neues JIRA-Projekt aufsetzt für die Tickets, vielleicht für jedes Team, oder irgendwie. Wenn man das vernünftig macht (...) [00:19:21-1](#)

I: Entschuldige, das hattest glaube ich du gesagt, dass am Ticketsystem so viele andere Systeme dranhängen,

PAZ und so, dass das ja irgendwie verknüpft ist. [00:19:32-2](#)

B: PAZ weiss ich glaube ich gar nicht. Nein, es ist glaube ich nicht direkt mit PAZ verknüpft. Du gibts nur, wenn du ein Ticket fixt, die Ticketnummer im PAZ an, aber es ist nicht direkt. So etwas wäre möglich, ich glaube wenn man das über Eclipse macht, und dann eincheckt, das hatten wir glaube ich gestern. Das man dann, je nachdem auf welchem Task man arbeitet, die Tasks wenn man ausknipst, dann die Zeit erfassen könnte. Aber wird momentan nicht so gemacht, und macht auch keinen Sinn. Ich glaube der einzige Sinn, dass wir das Ticketsystem noch haben, ist, dass die Tickets ja übertragen werden an **Kunde_1**, und dass es dann technisch einfach nicht so schön wäre das rauszulösen. Und du hast dann... Das Ticketsystem besteht ja glaube ich schon seit zehn Jahren, und das zu migrieren nach JIRA wäre so ein Aufwand, dass es glaube ich nur deswegen nicht gemacht wird. Ansonsten würde das glaube ich funktionieren. (...) Weil ich glaube, wenn man JIRA vernünftig konfiguriert und sich da ein bisschen Mühe gibt, dann würde es glaube ich auch einfach noch schöner als das Ticketsystem. [00:20:37-9](#)

I: Und wie gut kennst du dich auch selbst schon aus mit JIRA, weil das Ticketsystem, also JIRA gibt es ja jetzt noch nicht so lange. Seit einem Jahr vielleicht? [00:20:47-7](#)

B: Ja, JIRA. Ich kann es so benutzen wie ich es brauche. Konfigurieren tu ich es überhaupt nicht, die Suche ist grottenschlecht. [00:20:59-9](#)

I: Wie kommst du zum Beispiel klar mit diesen Task-IDs die es da gibt? Im Ticketsystem hast du ja immer zum Beispiel "TT" und dann immer eine sechsstellige Nummer, das ist den meisten ja glaube ich ziemlich geläufig. Und im JIRA da hat man ja so "LFS-1375"? [00:21:19-3](#)

B: Ja, das ist ja genau das gleiche eigentlich. Du hast einen Task, und hast eine ID dran. Also, da kuck ich aber nicht drauf. Nein, nein. Ich unterhalte mich nicht über Tasknummern mit irgendwelchen Leuten. Da gibt es auf jeden Fall einen Link dazu, und dann so mit der Nummer hat man eigentlich nichts. [00:21:34-9](#)

I: Aber bei Tickets macht man das ja schon, oder? [00:21:35-1](#)

B: Genau. [00:21:35-1](#)

I: Das man am Telefon sagt "Ticket soundso", das geht bei den Tasks nicht, oder? [00:21:39-5](#)

B: Ja vor allem: du kannst nicht auf jedem Weg - also das ist auch der Vorteil von JIRA, es ist eine Webanwendung und du kannst den Link einfach verschicken, und dann hast du ihn genau da hin, und du brauchst die Nummer nicht mehr. Und wir merken uns eben an allen möglichen Stellen "TT" und dann die Nummer. Es gibt auch zwei verschiedenen Nummern, eine für **Kunde_1**, eine für uns, im Ticketsystem, und wenn du mit einem Kunden telefonierst, und dich über ein Ticket unterhalten willst, dann hilft es halt einfach nicht mehr. Ja. Ansonsten brauche ich die Nummer eigentlich nicht. [00:22:09-9](#)

I: Dann eine kurze Frage zu Mylyn und Eclipse: Hst du das Mylyn aktiviert, oder verwendest du es? [00:22:20-6](#)

B: Ich habe es mal probiert, fand es auf jeden Fall cool, meine Tasks zu sehen in Eclipse, habe die dann aber darüber nicht weiter verwaltet. Also ich habe mich da nicht richtig eingearbeitet. Also ich gehe immer über die Weboberfläche, und nicht über Eclipse. Ich habe auch mal probiert, dass man da die Tasks anschaltet, und dann so und so lange nur die Sourcen bearbeitet, und da wird na dann auch automatisch getrackt. Beim Einchecken ist ja dann das Task-Kürzel, quasi die ID, vorbelegt. Das hatte ich alles mal probiert, aber wahrscheinlich aus mangelnder... oder aus Bequemlichkeit wahrscheinlich nicht durchgezogen. Es ist auch keine Vorgabe für uns, dass wir das machen sollen, und dann macht man es so wie immer. [00:23:12-2](#)

I: Und legst du in Mylyn private Tasks oder Todos an? Die du nicht mit JIRA synchronisierst, sondern nur innerhalb von Eclipse? [00:23:22-4](#)

B: Also mittlerweile hab ich das [Mylyn] gar nicht mehr offen. [00:23:22-4](#)

(JIRA Login)

I: Ok, also das ist dein Dashboard. [00:25:05-6](#)

B: Genau. Hier hab ich so eine Monatsübersicht, welche Tasks offen sind. [00:25:13-1](#)

I: Das sind jetzt nur deine eigenen Tasks. [00:25:13-1](#)

B: Genau. Ja, das ist hier so die Hauptübersicht mit der ich arbeite. Was ist offen, was in Arbeit und was in QS.

Beziehungsweise wie viele. Und mit den Zahlen merke ich mir einfach ungefähr "ok, wo stehst du", ist was neues dazugekommen. Also wenn ich abends reinkomme, dann weiss ich "Hä?" 25 Implementierungstasks hatte ich doch heute morgen nicht, da ist was dazugekommen. So ungefähr. Und das war es dann auch fast schon. Das ist so die Übersicht für mich, und zum Scrollen... da schaue ich schon nicht mehr. [00:25:49-9](#)

I: Das sind vom ganzen Komposit-Team die Aufgaben? [00:25:53-0](#)

B: Genau. Die Fälligen. Also einfach die Monatsanzeige, und hier ist die (...) [00:26:06-3](#)

I: **KMP**-Benutzer. [00:26:05-9](#)

B: Ich glaube sogar das ist meine Liste, aber einfach nur mal als Liste. Und nicht sortiert hier nach welcher Art von Tasks, sondern einfach so. [00:26:20-1](#)

I: Ok. Und das ist das Standard-Dashboard das **ENTW_DE_TL** vorbereitet hat? Oder hast du das selbst konfiguriert? [00:26:27-5](#)

B: Nein, das hat **ENTW_DE_TL** gemacht. Ich weiss nicht wo die herkommt, die hab ich wahrscheinlich mal selber gemacht ... Nein, das wäre meine private. Ach genau. Da hatte ich früher mal ein paar Filter angelegt, das war noch in dem alten Projekt. Einfach so, welche sind mir zugewiesen, und gehören zu diesem Meilenstein. Ist dann, ja ich weiss gar nicht worauf es gefiltert war, achso genau. "Behebung für". So ein so ein Flag. [00:26:56-1](#)

I: Für einen Meilenstein, oder wie? [00:26:56-1](#)

B: Genau. Du kannst es einmal ganz gut filtern. Ja. benutze ich jetzt nicht mehr, ich finde die (Anm.: Default-Filter von **ENTW_DE_TL**) ganz gut. Und wenn ich dann mal irgendwie... . Hauptsächlich benutzt man es (Anm.: JIRA) auch einfach mal um Konzepte zu suchen. Momentan sind die ja bei uns in den Tasks angehängt. Da gehe ich über "Projekte", das ist unser **LFS**-Projekt, das alte **KMP**-Projekt gibt es auch noch, und kuck mir dann hier über "Vorgänge" einfach an:" ok, wozu gehört es denn", und wenn ich eben weiss, es geht um ein **Hausrat**-Konzept, dann finde ich es eben da drin sehr wahrscheinlich. Oder ich suche ein Konzept wo ich weiss, dass hat bestimmt **ENTW_DE_2**, dann schaue ich mir die hier an, und kuck was hat **ENTW_DE_2** denn so. Das mach bei einigen Leuten mehr Sinn, wenn du da bloss zehn Tasks hast, kommst du da fix durch, gibt aber auch Leute die haben, ... naja... (Anm.: befindet sich auf Übersichtsseite für Projekt "LFS" --> "Vorgangsnavigator" -> Filter "Nicht erledigt: nach Bearbeiter": Enthält eine Liste mit allen Mitarbeitern im LFS-Projekt, diese haben 0 - 94 Tasks zugewiesen, dazu gibt es mehr als 700 nicht-zugewiesene Tasks) [00:27:53-3](#)

I: Das ist aber im Prinzip die ganze Firma, oder? [00:27:57-1](#)

B: Das sind alle aus dem **LFS**-Projekt, genau. [00:28:01-8](#)

I: Aber wieso weisst du jetzt, wenn jetzt zum Beispiel, ich weiss nicht, du suchst das aktuelle **Hausrat**-Neuanlage-Konzept, wieso weisst du dass es dann hier bei den "Nicht-Erledigten" ist? [00:28:07-1](#)

B: Ah, stimmt. "Nicht erledigt". Warte. Wir hatten das mal so, dass, genau. Das ist ja so, dass wir, ich kann es ja einfach mal probieren. **Hausrat**-Storno ist ein **KO_FU**, das ist jetzt der Task für das technische Konzept - "TK" - und von da aus kommt man einfach über den Über-Task an alle anderen, und da gibt es auf jeden Fall den FK-Task den die Fachkonzeption jeweils hat wenn die Fachkonzept schreiben, und da finde ich dann auch, da ist das Dokument angehängt. [00:28:54-6](#)

I: Und das spielt immer der Teamleiter, lädt das rauf? [00:28:58-0](#)

B: Nein, das macht der Fachkonzepter selbst. **FK_1**. Also die fängt an, die erstellen da drüben auch die Task glaube ich. Das kann **ENTW_DE_TL_STV_1** wahrscheinlich besser erzählen. Und die "Impl-Task" und alle die wir zu erledigen haben, die werden dann zu uns zugewiesen, und zwar auch erst nachdem dieses Fachkonzept geschrieben ist, dieser FK-Task fertig ist. Wenn der hier fertig ist, dann gehts hier nochmal runter, dann wird das nochmal innerlich (...). Nein, genau. Das ist schon der nächste Task der dann an uns geht, so halb gelb, man sieht, kommt mal so "die Entwickler müssen was tun". [00:29:41-5](#)

I: Die Entwickler sind gelb, oder wie? [00:29:40-7](#)

B: Genau. Und grün ist dann (...) glaube ich das Testteam. Kucken ob es den TN-Task. Testliste (...) Na gut, da hat ja jeder was mit zu tun. Ja, ich glaube so war das. (...) Ach nein, Testlisten-QS macht (...). Ach, frag am besten **ENTW_DE_TL_STV_1** nochmal! [00:30:14-6](#)

I: Siehst du dir die JIRA im Browser an, oder über den Lotus Notes-Browser? [00:30:17-7](#)

B: Nein, immer über Mozilla. [00:30:21-4](#)

I: Und machst du dir Bookmarks für Tasks? [00:30:27-3](#)

B: Ja, mach ich. Wenn ich es irgendwie... Einfach so spontan, wenn man auch von etwas altem was sucht, und dann ein Fachkonzept findet an dem man die nächsten zwei Wochen noch arbeitet, dann einfach den FK-Task. [00:30:45-2](#)

I: Und da verwendest du dann den Bookmark, damit du schneller reinkommst? [00:30:44-4](#)

B: Genau. (...). Ja und sowas ist dann irgendwie, wo haben wir es - "Zusammenfassung" (Anm.: Projekt LFS --> "Zusammenfassung"). Soetwas ist dann eher immer irgendwie interessant, wenn man dann auch mal eine Grafik sieht, wieviele Tasks werden denn erstellt, wieviele gehen raus. Wenn man mal draufkuckt grob ist es auf jeden Fall interessant, und genau dieser Aktivitätsstrom hier, das meine ich auch. Mit **Sonstige_1** habe ich eigentlich NICHTS zu tun, aber einfach zu kucken, ok, "Überschussbeteiligung", ist mal interessant, was tun die denn. Oder wie arbeiten die auch mit JIRA ist auch interessant, wenn einer hier auf einmal durchgängig 30 Tasks schliesst, dann denkt man sich so "na, da ist irgendwas schief gelaufen, die hast du bestimmt nicht alle heute gefixt oder so." Oder die hätten bestimmt schon früher geschlossen gehört oder sowas. Aber das ist eher einfach mal interessant. Wer wirklich wie arbeitet, wie sowas hier aussieht, da kriegt man... [00:31:53-2](#)

I: Diesen Kommentar, liest man sich das dann durch? [00:31:54-8](#)

B: Genau, aber eigentlich so oft... [00:32:02-4](#)

I: (unterbricht) In welchem Projekt ist das hier jetzt drinnen, das habe ich nicht gesehen? [00:32:02-8](#)

B: Das ist in jedem Fall Fachkonzeption. Und es geht um ein "Limit Komposit", was immer das sein mag. [00:32:12-0](#)

I: Wenn du ein bisschen hier raufscrollst, das ist jetzt von **LFS**? [00:32:12-9](#)

B: Genau, vom kompletten Projekt. Das ist hier die Oberseite zu **LFS**. Zusammenfassung. Also alles was hier passiert. Ja. Das ist einer von uns, das ist einer von Leben, sieht man einfach was ist fällig. [00:32:26-8](#)

I: Und da sind jetzt auch alle Unterprojekte drin, also **Hausrat** und **Wohngebäude**? [00:32:27-9](#)

B: Genau. Also wie das genau im Hintergrund strukturiert ist, kenn ich mich nicht so aus, aber es gibt ja nicht beliebig viele Ebenen in JIRA. Es gibt ja nur zwei. Den Haupttask und Untertask. Und halt das Projekt. [00:32:50-2](#)

I: Könnten wir vielleicht mal wechseln auf deinen aktuellen Task den du gerade bearbeitest? [00:32:58-9](#)

B: "Nach Bearbeiter". Hm. Normalerweise würde ich da jetzt meinen Bookmark hier benutzen (Anm: um auf das Standard-Dashboard zu kommen). Genau. Hab ich nämlich auch hier oben drin, und dann komme ich immer wieder auf die Seite, die schon ganz gut ist für mich. Implementierung, das ist das, was ich wirklich am meisten tue. Und da sieht man hier: "Rechnungszins für die UVI", das mache ich gerade. Da klicke ich auch irgendeinen beliebigen an, ob das der Implementierungstask ist oder irgendwer, ich gehe über den Weg wieder zum Haupttask. Bevor ich mir lange suche "was ist der IMPL, was ist irgendwas". Und ja, je nachdem was da passiert, such ich mir die Testliste aus dem Testlisten-Task. Also hauptsächlich um Dokumente rauszulesen, oder um mitzukriegen (...) keine Ahnung. Hier das war ganz schön gewesen, da hab ich mit **SONSTIGE_2** ein bisschen Kommunikation, hab den QS-Task hierzu gekriegt, hab mir angekuckt "ok, was gibt es hier für Dokumente", das war eine ganze Menge, war dann so ein bisschen geschockt, und ich weiss jetzt nicht ob sie mir was draus geschrieben hat (...) Ja hier hat sie reingeschrieben, dass es gar nicht so schlimm ist wie es aussieht, dass es nur etwas kleines ist, komm doch rüber dann können wir kurz darüber reden. (...) Also hier war das glaube ich nicht, aber in einem von den Untertasks war das so. [00:34:27-0](#)

I: Diesen Kommentar, den hat sie da reingeschrieben, nachdem du sagtest "was ist da los", oder wie? [00:34:31-2](#)

B: Genau. Ich glaube ich hatte sie angerufen, dann hat sie das nochmal geklärt. [00:34:36-5](#)

I: Und sie hat dann den Kommentar da reingegeben. [00:34:36-5](#)

B: Genau. Oder das sind hier Bearbeitungsvermerke die du hier generell machst. Wenn du hier den Task öffnest, schliesst, öffnest, schliesst, oder irgendwie daran arbeitest, immer wenn du den Status von so einem Task änderst, sollte man auch etwas dazu schreiben. Also wenn er einfach nur geschlossen wird, dann schreibt man

irgendwie "ist" (...). Ja ok, da macht es keinen Sinn, aber wenn du dann irgendwas Besonderes damit machst, irgendwas das andere Leute auch interessieren könnte, einfach so als Kommentar dran. [00:35:13-6](#)

I: Diese Sachen hier, "Zeitverfolgung" und "Fälligkeitsdatum": stimmt das meistens? [00:35:20-1](#)

B: Das kommt hier... Die Tasks die werden von uns erzeugt, oder zumindest geben wir die Vorlage dafür, und zwar machen wir das so, wenn wir so ein Konzept schätzen - den Aufwand komplett schätzen für unsere fünf Tasks oder zwei Tasks die wir dann kriegen - machen wir ein Excelsheet fertig, indem wir es in einer bestimmten Form aufschreiben. Also das und das so, also die ganzen Unteraufgaben. Wenn du einen Implementierungs-Task hast, dann hast du einfach so sieben Sachen die du so in Tasks aufteilen könntest, die du einzeln auch abtesten könntest. Da machst du dir sieben Tasks, schreibst die in eine Excel-Liste, schreibst dahinter wieviele Tage brauchst du dafür. Also die Schätzung machen wir schon ziemlich konkret und ziemlich klein, und dann wird das über ein Importer-Tool importiert. Und da kommen dann diese geschätzten Zahlen die da aufgeschrieben wurden da dran. Also diese zwei Tage hier, die kamen mal von mir. Inwieweit das dann mit Fälligkeits-Datüern dann passt, das macht **ENTW_DE_TL**. Das ist dann auch nicht zwingend, also das ist dann auch nicht, ich muss das dann auch nicht zwingendermassen machen. Wenn ich jetzt sage, ich mache das am Ende vom Meilenstein, oder am Anfang, dann ist das egal. Das kann ich machen wie ich will. [00:36:36-4](#)

I: Gibst du dann vorher **ENTW_DE_TL** bescheid, dass er es auf ein anderes Datum legen soll? Oder hast du alle Termine und alle Fälligkeitsdaten am Ende vom Meilenstein? [00:36:46-4](#)

B: Nehmen wir mal an, ich hätte den Task am Anfang vom Meilenstein bekommen, und wüsste, ok, (...), ja das ist ja hier sogar so (Anm: Erstellungs- und Aktualisierungsdatum sind aus der Anzeige ersichtlich). Wann ist denn der fällig? (Anm.: Sucht das Fälligkeitsdatum) (...). Ach, hier muss man in den Unter-Task (Anm.: wechselt in den Implementierungs-Task). (...) Ich glaube, der war nämlich schon richtig lange fällig. (...) Nehmen wir einmal an, der ist bei mir Fälligkeit irgendwann am Anfang vom Meilenstein, irgendwann im März. Und ich denke mir "nein, ich mache das erst mal fertig", dann spreche ich das mit **ENTW_DE_TL** ab, sage im Scrum ich mache "das und das", und wenn ich dann bei meinen fälligen Tasks, wenn der mich dann nervt, dann verschiebe ich das einfach um zwei Monate nach hinten. [00:37:39-5](#)

I: Wie ist das bei verschiedenen Meilensteinen, es gibt ja glaube ich (...) Spielt das eine Rolle, weil früher war es ja glaube ich so, da war für einen Meilenstein ein Projekt angelegt [00:38:14-2](#)

B: Mittlerweile nicht mehr. Ich denke nicht. [00:38:20-1](#)

I: Das geht jetzt alles nur mehr über **LFS**? [00:38:19-2](#)

B: Genau. Es gibt ein Projekt **LFS** und es gibt keine Unterteilung mehr ins so (...). Schauen wir mal nach, ich bin jetzt gar nicht so sicher. Aber das müsste ja in so einem Einzel-Task drinstehen. [00:38:29-1](#)

I: Für welchen Meilenstein das ist, das steht ja immer bei den Einzel-Tasks dabei? [00:38:32-9](#)

B: Ich glaube das kriegt man nur vom Datum hier heraus. Ich weiss nicht, ob der Meilenstein wirklich noch dasteht. [00:38:39-0](#)

I: Es gibt ja glaube ich schon so eigene (...) "Lösungsversion" [00:38:46-6](#)

B: Ja gut. Meistens sind es nicht so viele Oberthemen. Du weisst, du hast im Meilenstein immer zwei bis drei Themen wirklich zur Neuentwicklung, da brauchst du nicht mehr wirklich darauf kucken. Du weisst schon am Anfang vom Meilenstein: Ok, ich habe die drei Themen, dann kriegst du die Tasks dazu, und dann weisst du schon dass die das ... Für den nächsten Meilenstein kriegst du auch gar keine Implementierungstasks jetzt. Die würdest du dann (...). Die kriegst du nur am Anfang. Also da überschneidet sich nicht viel. [00:39:16-7](#)

I: Ja, da wäre jetzt die Frage gewesen, wie ist das Früher war es ja so, da war das Konzept einem Meilenstein zugewiesen, dann war Meilensteinende, und dann musste man aber trotzdem nochmal irgendetwas im Konzept nachsehen, und dann war das unter Umständen schwierig wiederzufinden. Aber dann hat sich das ja erledigt. [00:39:34-4](#)

B: Ja, aber das finde ich prinzipiell nicht so schön, weil die Schnellsuche einfach meiner Meinung nach - also vielleicht bin ich auch zu doof - aber schön wäre es wenn ich hier, wie in dem Beispiel, "**PD_KO_FU_2704***" suche, dann möchte ich bitte hier die ".2" auch dabei haben. Ja und die Suche funktioniert einfach nicht so in die Richtung. Also entweder man gibt es komplett an, oder er findet es nicht. [00:39:59-1](#)

I: Und du suchst aber auch eher nach "**PD_KO_FU-Soundso**", und nicht nach "**LFS-1197**"? [00:40:06-7](#)

B: Ich kann es ja mal probieren (...) (Anm.: Tippt KO_FU-Nummer in Schnellsuche ein). (...). So, ich würde mir wünschen dass es gefunden wird. (...) (Anm.: Eine Liste mit 27 Einträgen wird zurückgegeben, die ersten Einträge sind **PMM_1** zugeordnet). Es wird irgendetwas mit "2704" gefunden, und mein "2704.2" ist da leider nicht dabei. Es funktioniert auch nicht mit "*" oder soetwas. Mag er nicht. Vielleicht kann man damit auch etwas mehr machen. Aber so wie ich die benutze, muss ich da ".2" hinschreiben, und dann finde ich meinen Task. Also nicht so schön. 00:40:55-9

I: Also das muss man momentan also wirklich ganz genau wissen. 00:40:57-0

B: Genau. Oder sich eben hier drüber "langhangeln". Über dieses **LFS**-Projekt, und dann entweder über "wem ist es zugewiesen", "wem könnte es zugewiesen sein", welcher Meilenstein war es, oder welche Sparte ist es. Also das sind so drei Sachen, an denen man sich "langhangeln" kann. An denen man suchen kann. Da suchst du teilweise in dreihundert Tasks. 00:41:30-8

I: Ok. Und dass du dir zum Beispiel ansiehst, was **ENTW_DE_2** gerade macht, das machst du nicht? 00:41:33-5

B: Nein. 00:41:48-3

I: Und von **ENTW_DE_TL**, was der so macht siehst du dir wahrscheinlich auch nicht an. 00:41:51-6

B: Nein. Ich weiss aber auch nicht, ob der seine eigene Arbeit mit Tasks verwaltet. Das glaube ich nicht. Seine Arbeit ist eher, die Tasks zu verwalten und zu "controllen". 00:42:15-0

I: Ja gut, dann herzlichen Dank.

Interview 02_FK_TL

I: Die erste Frage wäre einmal was deine Rolle hier bei **VersicherungsGmbH** ist. 00:00:09-5

B: Ich bin Teamleiter in der Fachkonzeption und für die Bereiche Komposit und Zentralsysteme, das heisst alles ausser Leben. 00:00:20-9

I: Ok. Und was sind da so deine täglichen Aufgaben? 00:00:22-3

B: Tägliche Aufgaben, also das ist einmal das Personalmanagement, Personalplanung. Für die Fachkonzepte Ausarbeiten der Machbarkeitsaussagen für die jeweiligen Meilensteine. Controlling des Projektfortschrittes in den jeweiligen Meilensteinen. Dafür zu sorgen, dass die Leute arbeiten können wenn Probleme entstehen, da biete ich eine entsprechende Eskalationsebene. Ich arbeite wenig bis gar nicht fachlich an den Themen mit, bin also eher auf der organisatorischen Ebene tätig. 00:01:01-1

I: Diese Machbarkeits-Sachen, da redest du dann mit dem Kunden, oder wie? 00:01:06-7

B: Machbarkeit: also der Kunde reicht eine Liste an Themen normalerweise ein die er bearbeitet haben möchte. Der Produktionsabschnitt ist fest terminiert, das heisst die Dauer des Produktionsabschnittes ist fest. Ich habe in diesem Produktionsabschnitt in meinem Team eine bestimmte Kapazität zur Verfügung die ich mir vorher ermittle, weiss wer welche Themen bearbeiten kann in meinem Team. Die Themen werden dann in der sogenannten Woche Null des Produktionsabschnittes den Fachkonzeptern zur Schätzung vorgelegt, die lesen sich dann ein in das entsprechende Thema, kriegen eine Funktionsumfangsbeschreibung (Anm.: im Folgenden auch „FUB“ genannt) vom Kunden meistens, versuchen darauf eine Fachkonzeptschätzung zu erstellen - wie lange brauche ich für das Fachkonzept, wie lange brauche ich für die Testliste dazu. Es gibt von meinem Produktmanagement, da kriege ich dann noch eine Liste wie die Priorisierung der Themen aussieht, und gemäss der Priorisierung und meiner Kapazitäten und den Schätzaufwänden die ich von meinen Leuten bekommen habe, kucke ich halt dann was in diesem Produktionsabschnitt von uns gemacht werden kann. Also wieviel passt rein, und daraus ergibt sich dann die Machbarkeit. Das heisst die Themen die wir auch zeitlich bearbeiten können, wo wir davon ausgehen, kriegen die Machbarkeit "ja". Wenn Themen dabei sind die vielleicht ein bisschen zu gross sind, oder wo die Anforderungen noch so schwammig sind dass man überhaupt nicht weiss wo es da hinlaufen könnte, kann es passieren, dass die erst mal eine Machbarkeit "offen" kriegen. Das heisst da muss man sich noch mal drüber unterhalten. Und Themen die jetzt einfach nicht mehr reinpassen in den Produktionsabschnitt kriegen gleich von vornherein eine Machbarkeit "nein". Das liefere ich meinem Auftraggeber zurück, das ist das Produktmanagement, und das Produktmanagement übernimmt über die Machbarkeit dann die Kommunikation mit dem Kunden. 00:03:00-7

I: Aha, ok. Das war mir nicht ganz klar. Und wieviele Leute sind bei dir im Team, oder für die du verantwortlich bist? 00:03:06-1

B: Hat jetzt ein bisschen geschwankt in der letzten Zeit, sagen wir mal in Spitzenzeiten waren wir inklusive aller externen Kräfte bei mir 13 Leute, und sind jetzt bei acht angelangt. 00:03:21-2

I: Und externe Kräfte sind aber Leute die schon hier in der Firma sitzen und hier mitarbeiten, oder? 00:03:24-3

B: Nein, da waren auch Leute aus externen Unternehmensberatungen die wir zusätzlich noch mit eingestellt haben, die bei mir im Team dann auch mitgearbeitet haben. 00:03:31-2

I: Ok, die aber irgendwo anders in einem Büro sitzen. Wie wird mit denen dann zusammengearbeitet? 00:03:37-0

B: Was meinst du genau? 00:03:39-2

I: Wird da über Email und Telefon, oder wie? 00:03:41-2

B: Ach nein, jetzt verstehe ich erst. Die waren durchaus auch bei uns hier im Büro anwesend für die Zeit. Also die konnten nicht bei sich im Büro arbeiten, sondern die haben bei uns dann einen Arbeitsplatz gekriegt. Und also zusammen mit dem Team auch in dem Büro hier der **VersicherungsGmbH** eingebracht. 00:03:57-1

I: Mit wem sitzt du zum Beispiel im Zimmer? 00:04:03-5

B: Mit wem ich im Zimmer sitze? Mit meinem anderen Teamleiterkollegen, der für das Leben-Team zuständig ist. Also auch Fachkonzeption, aber entsprechend die Themen für die Lebensversicherung. 00:04:16-4

I: Ok. Und deine Leute, die sind aber in den angrenzenden Büros. 00:04:18-9

B: Die sind in den angrenzenden Büros. Also ein Büro ist auf der einen Seite, und eins, zwei, drei Büros auf der anderen Seite. Nein, vier sogar. Vier auf der anderen Seite. [00:04:30-6](#)

I: Ok, und wie wäre dann zum Beispiel zum Produktmanagement, weil du vorhin gesagt hast dass das quasi dein Auftraggeber ist? [00:04:34-8](#)

B: Produktmanagement sitzt einen Stockwerk über uns. Aber auf derselben Seite des Gebäudes. Praktisch direkt über unseren. [00:04:43-8](#)

I: Und wenn du jetzt zum Beispiel mit Produktmanagement etwas besprechen musst, gehst du da einfach rauf, oder schreibst du ihnen ein Email, oder chattest du sie an? [00:04:52-9](#)

B: Kommt darauf an, ob es etwas grösseres, längeres ist. Wenn es irgendwie nur eine kurze Frage ist die mit einem Satz zu beantworten ist, mache ich es telefonisch meistens. Wenn es längere Sachen sind die man auch schriftlich haben will geht es per Email oder zum Termine einladen wenn es darum geht, erstmal überhaupt eine Diskussion irgendwie anzustellen um erst mal ein gemeinsames Ergebnis zu erarbeiten. Also was da gerade sich für den konkreten Fall sich am besten eignet. [00:05:23-4](#)

I: Und welche Programme verwendest du dann zum Beispiel jeden Tag bei der Arbeit? [00:05:30-6](#)

B: Natürlich unseren Lotus Notes Client, wo wir unsere Email und Kalenderverwaltung mit machen. Ich nutze den Sametime Client für Instant Messaging auch relativ viel. Project, MS Project benutzte ich meistens am Meilensteinanfang um die Initialplanung zu machen, weil es damit am einfachsten ist, die Kapazität entsprechend raus zu rechnen. Es gibt diverse Excel-Tools, Exceltabellen die wir in Verwendung haben. [00:06:10-1](#)

I: Von den internen Tools, Ticketsystemen, JIRA, was ... [00:06:12-9](#)

B: Ticketsystem... [00:06:14-2](#)

I: Spielt das bei dir eine Rolle? [00:06:14-2](#)

B: Ticketsystem auch. Zur, also einige der Tickets kommen bei mir zur weiteren Verteilung dann an, es gibt aber auch die Möglichkeit, dass Tickets direkt bei den Leuten auch ankommen. Wir haben in der Planung immer einen gewissen Puffersatz mit drin für eben Ticketbearbeitung oder soetwas. So dass das nicht unbedingt in jedem Einzelfall über meinen Tisch läuft. Wenn es dann mehr wird, arbeite ich auch mit dem Ticketsystem, und übernehme dann die Verteilung. Oder teilweise auch Bearbeitung der Tickets für einzelne Themen. JIRA benutze ich einmal um die Aufgaben in den Meilensteinen auch den Leuten zuzuteilen. Alle Themen die wir bearbeiten bekommen entsprechende Einträge in JIRA, da gibt es dann eine hierarchische Struktur von verschiedenen Aufgabentypen die immer zu erledigen sind wenn Fachkonzept zu erstellen ist. Die werden dann über JIRA entsprechend dem Bearbeiter in meinem Team zugewiesen. Und über JIRA wird auch entsprechend ein Workflow für den Qualitätssicherung dargestellt. Und ich überprüfe da auch, ob die Termine eingehalten werden, und wie da der Status gerade ist, um den Projektfortgang zu überwachen. [00:07:36-1](#)

I: Ist dieser Workflow bei JIRA entstanden, der hat wohl auch erst programmiert werden müssen, oder? Habt ihr das bei euch im Team selbst gemacht, oder macht das jemand in der Firma? [00:07:46-0](#)

B: Nein, wir hatten, wir haben einen Kollegen hier in der Firma die uns da unterstützt haben, und mit denen wir am Anfang eben den nötigen Workflow erarbeitet haben am Papier, und die haben das dann in JIRA für uns entsprechend umgesetzt und programmiert. [00:08:04-1](#)

I: Ok. Aber das war quasi dieser Workflow den es eigentlich schon vorher gegeben hat bevor das JIRA eingeführt worden ist. [00:08:08-4](#)

B: Genau. Den gab es grundsätzlich schon vorher, wir haben vorher per Email sehr viel hin- und hergeschickt. Also wenn wir eine QS haben wollten für ein Fachkonzept, haben wir das Fachkonzept in eine Email gepackt, die Email an einen Entwickler geschickt, der hat dann da seine Anmerkungen reingemacht, per Email ging es dann zurück, und das hatte halt den Nachteil, dass praktisch viel von der Dokumentation nur in den Mailboxen der einzelnen Leute gewesen ist. Wenn da einer mal nicht da war, keine Möglichkeit darauf zuzugreifen was da gerade diskutiert wurde. Und dann haben wir irgendwann entschieden, dass wir ein zentrales Medium brauchen wo einmal wir selbst Zugriff darauf haben, und eventuell auch mein Chef sich da Überblick verschaffen kann "wo stehen wir denn eigentlich gerade". Und die Kollegen im Implementierungsteam haben schon viel mit JIRA gearbeitet vorher, wir haben uns das dann angeguckt, haben gesagt das ist eigentlich ganz gut. Haben das für uns auch entsprechend adaptiert. Dann hatten wir alle unsere eigenen Projekte, also das Implementierungsteam hatte sein eigenes Projekt, wir hatten unser eigenes Projekt, die Kollegen von Leben hatten ihr eigenes Projekt,

Zahlungsverkehr und Provision hatten irgendwie ihre eigenen Projekte, und dann haben wir gemeint, dass das eigentlich relativ doof ist, weil man halt dann diese grundsätzlichen Themen, also den Kopf drüber, immer wieder neu anlegen muss, und alles duplizieren muss. Und nicht einfach, wie es vielleicht der Sinn von JIRA wäre, die Aufgabe einfach wandern zu lassen. Und dann haben wir uns mit dem letzten, Schrägstrich vorletzten Produktionsabschnitt entschieden, dass wir es vereinheitlichen wollen. Haben neue JIRA-Projekte aufgesetzt was dann für alle gültig ist, und wo wir dann wirklich die Aufgaben wandern lassen können. 00:09:50-8

I: Das ist jetzt das "**LFS**"-Projekt? 00:09:50-8

B: Genau. Das ist das **LFS**-Projekt das wir haben. 00:09:52-6

I: Ok. Und quasi, Fachkonzepte, die wurden ja früher auch immer auf Netzlaufwerken oder über Email hin- und hergeschickt. 00:10:03-0

B: Ja, also die sind früher immer einmal auf dem Netzlaufwerk in der Verzeichnisstruktur abgelegt worden. Das ist grundsätzlich jetzt eigentlich immer noch so, dass es die Leute den Arbeitsstand auf dem Netzlaufwerk haben sollen. Weil sonst haben wir wieder das Problem, wenn das nur lokal vorgehalten wird, aber noch keine Version war die schon verschickt werden konnte zur QS, dann hätten wir die wieder nicht zentral irgendwo auf dem Server. Und deswegen soll man den Arbeitsstand immer auf ein Laufwerk legen. Sonst wenn einer ausfällt, man trotzdem darauf zugreifen kann. Und erst wenn dann aber irgendwie eine QS durchgeführt werden muss, oder wenn jemand anders darüber informiert werden soll, dann läuft das über JIRA. 00:10:47-2

I: Sehr gut. (...) Ja dann eine Frage zu deinem Arbeitsalltag, und zu deinen Kollegen. Nachdem du ja Teamleiter bist, wie bleibst du da am Laufenden was jetzt deine Mitarbeiter gerade machen? 00:11:06-4

B: Also einmal bin, versuche ich relativ viel mit den Leuten direkt zu sprechen, also dass ich auch einmal am Tag die Büros alle abklappere und mir da einen aktuelle Stand hole. Zusätzlich machen wir einmal in der Woche ein Teammeeting, wo ich einmal über Information von oben berichte, andererseits mir aber auch einen Status einmal abhole in der Woche, wo alle zusammen in einem Raum sind, sodass man da irgendwie eventuell auch mal irgendwelche Abhängigkeiten erkennen kann, wo dann mehr Leute betroffen sind, und Themen sind, die mehr Leute betreffen auch. 00:11:45-8

I: Und wie ist das zum Beispiel mit der Arbeitsauslastung von einzelnen Leuten - die (unv.: verteilt?) sich über das Teammeeting, oder über das direkte Gespräch? Oder welchen Anteil spielt da zum Beispiel das JIRA, dass man sieht "ok, der hat jetzt 500 Tasks gerade offen und kommt vielleicht nicht mehr nach"?

B: (...) 00:12:12-9

I: Oder wo würdest du, was würdest du wichtiger einschätzen: das man zu den Leuten hinget, oder reicht das wenn man einfach ins JIRA schaut? 00:12:21-3

B: Ich persönlich bin Freund der persönlichen Kommunikation, weil man da meistens mehr Information, detailliertere Informationen rausziehen kann, und auch spontaner an Infos herankommt. In JIRA ist aber die gute Möglichkeit, sich erst mal einen schnellen Überblick über den Gruppenstatus zu verschaffen. Und vielleicht auch da schon Probleme die eventuell auftauchen zu erkennen. Also JIRA ist hervorragend dazu geeignet, um irgendwelche Fälligkeiten zu prüfen. Weil da kann ich zentral meine entsprechende Abfrage bauen, und sagen, "ja zeig mir alle Tasks die jetzt seit, keine Ahnung, mehr als drei Tage bei der Entwickler-QS liegen". Oder "zeig mir alle Tasks die jetzt gerade bei meinen Leuten im Status 'warten' liegen", aus welchen Gründen auch immer. Dafür ist es natürlich ganz gut geeignet sich erst mal ein Bild zu verschaffen, ich glaube aber, dass es auf jeden Fall die Kombination sein muss. Also das persönliche Gespräch ist wichtig, weil man daraus erstmal noch mehr Information rausziehen kann, und wie gesagt auch spontaner, wenn einer sagt "Mensch, ich hab gerade mit dem und dem telefoniert", und das ist etwas, das er vielleicht gar nicht dokumentieren würde in einem System wie JIRA. 00:13:39-7

I: Und wenn du mit anderen Teams kommunizierst, wie zum Beispiel Produktmanagement, läuft das dann auch zu Teil über JIRA, oder ...? 00:13:51-3

B: Eigentlich ... 00:13:52-4

I: Oder wie sehr ist das für dich wichtig? 00:13:57-5

B: Also einige Sachen laufen dann auch über JIRA. Es gibt auch sowas, also Produktmanagement ist ja mein Auftraggeber, und am Ende des Meilensteins muss ich meinem Auftraggeber natürlich auch bestätigen, dass wir mit dem Thema fertig sind, das wir das erledigt haben. Und da beantrage ich formal eine Abnahme vom Produktmanagement und sage "hier, Thema XY, Fachkonzept ist fertig, Entwickler haben es abgenommen, Kunde

hat es abgenommen, bitte nehmt mir das auch ab, und bestätigt mir, dass ich mit dem Auftrag, den Auftrag erledigt habe". Und das funktioniert auch über JIRA. Da haben wir auch einen entsprechenden Unteraufgabentyp eingerichtet, "Abnahme Produktmanagement" heisst der, den wir dann entsprechend raufschicken, der wird eventuell auch verwendet, um anzukündigen "boh, wir werden den Fertigstellungstermin der eigentlich geplant war nicht halten können", deswegen beantrage ich eine Auflage, dass ich dieses Konzept für, keine Ahnung, eine Woche oder zwei Wochen später fertigstellen kann. Auch das läuft über JIRA wo ich dann die entsprechende Antwort kriege "ja ist in Ordnung" oder "nein, ist nicht in Ordnung". Wenn es nicht in Ordnung ist, muss man sich halt nochmal in einem konkreten Termin zusammensetzen, und kucken, was man entsprechend noch machen kann. [00:15:13-1](#)

I: Wie ist das zum Beispiel, spielt das für euch eine Rolle was in der Implementierung geschieht? Weil oft ist es so, dass in der Implementierung dann noch irgendwelche Themen auftauchen, die vorher noch nicht ganz klar waren. Wo man dann wieder zurückgeht, also wo dann die Entwickler wieder zurückgehen zum Fachkonzepter und sagen: "Du, pass auf, irgendwie das und das ist mir noch nicht ganz klar, und gehört das vielleicht geändert, oder ist das vielleicht ein Fehler jetzt im Fachkonzept?" oder so. Inwieweit müsst ihr da am laufenden sein, was in der Entwicklung vor sich geht? [00:15:49-1](#)

B: Also unsere Konzepte benutzen wir auch zu Dokumentationszwecken. Das heisst, es ist eigentlich essentiell, dass das Konzept auch immer den Stand widerspiegelt der dann irgendwann umgesetzt wird. Und das kommt durchaus vor, dass entweder in der Umsetzung jemandem auffällt "das geht so aber gar nicht" oder "ich hab eine bessere Lösung, ich will das anders machen als das im Konzept beschrieben ist", weil unser Konzept das vielleicht schon relativ detailliert vorgegeben hat, den Lösungsansatz. Da sind wir natürlich darauf angewiesen, dass wir auch entsprechende Rückmeldungen dann zurückkriegen. "Wir machen hier gerade was anderes" oder lass es aus Zeitgründen sein, dass die Implementierung nicht das komplette Konzept in einem Produktionsabschnitt schafft fertigzustellen, und sagt, "ja ich mache jetzt den Teil", dann kann man schon einmal damit arbeiten und irgendwie die Funktionen und Plausibilitäten muss man in einem ersten Schritt mit einer Arbeitsanweisungen dann irgendwie abfangen. Das sind auch Informationen, die für uns natürlich wichtig sind. [00:16:54-6](#)

I: Und wie bekommt ihr dann, wie bekommst du das dann zum Beispiel mit? [00:16:53-9](#)

B: Das ist im Augenblick bei uns noch ein bisschen das Problem, dass es da keinen sauber beschriebenen Prozess gibt, und das mal funktioniert, und mal nicht funktioniert. Kann sein, wenn zum Beispiel ein Ticket hereinkommt, dann ist die Wahrscheinlichkeit relativ gross, dass wir da auch involviert werden, weil der Entwickler das Ticket nimmt und sagt "hm, das findet jetzt jemand falsch, ich hab aber das Fachkonzept so gelesen und so verstanden, das lasse ich jetzt den Fachkonzepter nochmal prüfen. Hier, der Kunde sagt der wollte das ganz anders haben, sag was dazu. Ich finde das passt so." Dann kriegen wir das über das Ticket mit. Kann auch sein dass einer der Entwickler einfach rüberkommt zu den Leuten und sagt "du, pass auf ich hab hier ein Problem, im Konzept steht dieses und jenes, das kann ich so nicht umsetzen. Kann ich es so oder so machen?" Das heisst, das würde erst mal nur in dem Gespräch halt stattfinden. Wenn das aber dazu führt, dass die Lösung anders ist als im Konzept beschrieben, dann muss das im Konzept auch angepasst werden. Wenn das dann gemacht wird, dass das Konzept angepasst wird, läuft es auch wieder komplett über JIRA, durch den QS-Prozess durch. [00:18:13-5](#)

I: Also wird dann das Fachkonzept wieder geöffnet? [00:18:16-0](#)

B: Wird wieder geöffnet, der Task, und wird komplett durch die Prozesskette dann wieder durchgeschleust. Im Normalfall muss auch die Testliste geprüft werden, ob die dann noch aktuell ist, und wird auch entsprechend dann durch die QS-Kette dann wieder durchgeschleust. [00:18:33-3](#)

I: Wie ist das bei euch, so auf persönlicher Ebene im Team. Spielt das eine starke Rolle, wenn jemand zum Beispiel, ich weiss nicht, schlechte Laune hat oder so? [00:18:48-9](#)

B: (lacht) Wir sind natürlich alle Profis, aber wir sind genauso auch alle Menschen, und natürlich immer wenn Menschen zusammenarbeiten, kommt es natürlich auch immer ein bisschen auf die Tagesform drauf an. Man hat vielleicht mal irgendwie einen schlechten Tag, und dann klappt irgendwie eine Diskussion über ein Thema schlechter als an einem anderen Tag. Das darf aber natürlich nicht irgendwie soweit eskalieren, dass das die Arbeit wirklich behindert. Also dass es da schon mal zu Diskrepanzen kommt schon, das ist ganz normal, ich persönlich halte meine Leute erst mal an, dass sie das Gespräch suchen, wenn irgendetwas nicht passt. Wenn das aber unter den Kollegen alleine nicht funktioniert, bin ich aber auch jederzeit eine Eskalationsebene. Um mich da entsprechend einzumischen. Aber wie gesagt, das ist jetzt nichts wo ich sage "das steht an der Tagesordnung dass da ständig Krawall ist", aber wenn Menschen zusammenarbeiten, ist es ganz normal dass es auch mal zu Konflikten kommt. [00:19:55-0](#)

I: Artefakte. Welche Dokumente (...) Fangen wir an bei den Fragen zu JIRA. Oder beziehungsweise zu anderen Sachen. Ich nehme an, du hast eine Menge Leute in deiner Sametime-Kontaktliste. Sind das die selben die du auch in JIRA siehst? [00:20:40-2](#)

B: Ich hab in Sametime mehr Leute drin. Also ich habe mir verschieden Gruppen gemacht. Ich habe einmal komplett den Standort **Norddeutschland** drin, mit allen Leuten, der ist im Normalfall aber zugeklappt. Den hab ich nur wenn ich mal ... [00:20:55-5](#)

I: (unterbricht) Das sind alle 150 Mitarbeiter? [00:20:55-5](#)

B: Also, wenn sie angemeldet sind. Klar. Ich habe eine Gruppe für unsere Vorstände, ich habe eine Gruppe für unsere Abteilungsleiter, ich habe eine Gruppe für alle unsere Teamleiter, ich habe eine Gruppe für mein Team, und jeweils eine Gruppe für die Teams mit denen ich dann auch mehr zusammen arbeite. [00:21:20-4](#)

I: Mhm. Welche sind das in deinem Fall? [00:21:20-4](#)

B: Wie gesagt, meine Teams sind Komposit einmal, also das Komposit Entwicklungsteam, ich habe die ganzen Umsysteme, das ist Zahlungsverkehr das Entwicklungsteam, Provision das Entwicklungsteam, ich habe Architektur als Gruppe mit drin, ich glaube ich habe auch die Kollegen von Leben jeweils mit drin, meine Gruppe für STQ, das jetzt PQM heisst - funktioniert jetzt glaube ich grade nicht mehr so gut (lacht) - die muss ich nochmal updaten wieder. Und ich kann mich gerade nicht erinnern was noch drin ist. Wir können aber gern später nochmal kucken. [00:22:03-2](#)

I: STQ ist die Testabteilung? [00:22:03-2](#)

B: Genau. [00:22:06-4](#) [00:22:12-7](#)

I: Und da schaust du aber schon laufend rein wahrscheinlich? [00:22:11-1](#)

B: Ich habe die Gruppe für mein Team normalerweise aufgeklappt. Das ist aber eigentlich die einzige die ich eigentlich immer aufgeklappt habe. Die anderen klappe ich mir halt auf wenn ich Bedarf habe, und wenn ich sie brauche. [00:22:29-3](#)

I: Und was ziehst du da für Information daraus? Also einfach welche Leute online sind oder ...? [00:22:32-5](#)

B: Ich werfe da einen kurzen Blick drauf wenn ich ein Gespräch mit jemandem haben will, ob er gerade als grün angezeigt wird, oder gar nicht am Platz ist, oder auf "Bitte nicht stören" steht. Und entscheide demnach ob ich rüberlaufe, oder ob ich ihn anrufe oder ihn anhatte oder anmaile. Je nachdem. Zusätzlich habe ich mir auch die Kalender meiner Kollegen auf eine Kurzwahl gelegt, sodass ich sie in meinem Kalender miteinblenden kann. Heisst, ich habe zwar keinen... [00:23:10-1](#)

I: (unterbricht) Das ist im Notes-Kalender? [00:23:10-1](#)

B: Genau. Heisst, ich habe zwar keinen Vollzugriff bei den meisten Leuten, dass ich die Inhalte auch alle sehen kann, aber ich kann zumindest die geblockten Zeiten entsprechend auch anzeigen lassen. [00:23:17-7](#)

I: Ok. Und was steht da drinnen von deinen Kollegen? Wann sie zum Beispiel beim Kunden sind, oder was tragen die Leute da ein? [00:23:22-5](#)

B: Da sind eigentlich alle Termine die sie so haben. Wenn sie irgendwie Termine mit den Entwicklern ausgemacht haben um über irgendein Konzept zu sprechen, oder mit den Kollegen aus dem Testteam ein Gespräch haben. Die Kundentermine sind meistens nicht so detailliert drin, weil wir die normalerweise nicht über einen Notes-Termin herstellen, sondern das sieht ja meistens so aus, dass wir komplett drei Tage nach **Süddeutschland** fahren, und dann die drei Tage vor Ort sind, und da dann einen Excel-Plan, Besprechungsplan aufstellen, und das ist dann meistens aber nicht mehr in den Kalendern mit drin. [00:24:06-4](#)

I: Das Ticketsystem, da hast du gesagt das verwendet ihr auch. Wie würdest du die grossen Vorteile und Nachteile jetzt sehen, wenn du dir ansiehst Ticketsystem und JIRA? Weil es ja im Prinzip irgendwie ähnliche Funktionalität hat. [00:24:26-2](#)

B: (lacht) [00:24:25-9](#)

I: Das ist vielleicht eine grosse Frage... [00:24:27-4](#)

B: Eine unqualifizierte Aussagen wäre jetzt, dass das Ticketsystem in Notes ist, und vielleicht manchmal so ein bisschen langsam ist. Aber letztenendes sage ich mal, kann man in beiden Systemen eigentlich alles abbilden was wir zumindest für Troubleshooting benötigen. Mein Workflow jetzt für Konzept-QS und Testlisten-QS würde ich in dem Ticketsystem nicht abbilden wollen, und dazu ist es auch nicht gemacht. Könnte mir aber durchaus

vorstellen, dass man die Funktionalität für die gefundenen Softwarefehler durchaus auch in einem JIRA abbilden könnte. Aber das glaube ich wäre relativ problemlos umzustellen. [00:25:11-2](#)

I: Wie findest du das zum Beispiel rein von der Übersichtlichkeit, wenn du dir das Ticketsystem (unv.) mit diesen ganzen Tickets. Im Gegensatz dazu eine Übersicht mit allen Tasks wie du sie... . Ist das vergleichbar? [00:25:24-3](#)

B: Also ich muss, ich habe bei JIRA mehr Möglichkeiten, weil ich mir meine Filter selbst bauen kann. Das heisst, ich kann die eingrenzen die mich jetzt gerade interessieren. Mir selbst herstellen. Das kann ich im Ticketsystem Notes nicht. Da bin ich auf die vorgefertigten Ansichten festgelegt. Die kann ich zwar von der Auswahl der angezeigten Spalten noch anpassen, ich kann aber die Eingrenzung nicht ändern. Ich kann jetzt nicht sagen "zeig mir wirklich nur alle für Kollege XY an". Kann ich zwar indirekt über einen zweiten Schritt mir herstellen, indem ich mir die ganze Ansicht nehme, den Kollegen XY markiere, und dann nur die markierten anzeigen lasse, dann habe ich natürlich auch die Menge soweit eingegrenzt, kriege ich im Normalfall vernünftig nur über mehrere Schritte hin. Und mit der Suchfunktion wäre es wahrscheinlich sogar auch möglich so eine Eingrenzung zu machen, finde ich aber ungemein komplizierter als die Abfragesprache in JIRA oder das schnelle Zusammenklicken von einfachen Filtern in JIRA. [00:26:36-7](#)

I: Ok. Ja das wäre dann eh schon die nächste Frage. Wieviele.. Wie du dir JIRA selbst konfiguriert hast. Ob du da eine Standardkonfiguration verwendest. [00:26:45-8](#)

B: Nein, ich habe keine Konfiguration gemacht, muss aber zugeben, dass ich da noch nicht wirklich happy bin darüber, und jetzt mir auch die Zeit noch nicht genommen habe, das nochmal sauber zu überarbeiten. Ich habe ganz am Anfang, als wir angefangen haben mit JIRA zu arbeiten, wie wir das **LFS**-Projekt eingerichtet haben, habe ich sowohl für mich, als auch für meine Mitarbeiter ein Dashboard eingerichtet mit verschiedenen Filtern. Das ganze musste aber relativ schnell gehen, und ich konnte mich aber vorher nicht wirklich intensiv mit den Leuten auch auseinandersetzen "was will man denn eigentlich sehen". Und habe einfach mal alles an Filtern gebaut, was ich mir in dem Moment vorstellen konnte, und habe relativ schnell gemerkt, dass es zu viel Information in zu vielen Boxen, was dann am Ende eigentlich mehr unübersichtlich ist, als das es dir in der Übersicht hilft. Das heisst, ich persönlich nutze MEIN Dashboard im Moment gar nicht mehr, ich habe mir ein, zwei, drei Filter gebaut die ich häufiger verwende, die dann in meinen Favoriten auch mit drin sind. Wobei da sind ein paar mehr jetzt auch noch drin. Die ich dann gegebenenfalls ein bisschen anpasse, weil das mit der Parametrisierung ich das noch nicht richtig hingekriegt habe ob das geht vernünftig, das heisst, die werden dann eventuell im laufenden Produktionsabschnitt mal angepasst, oder die Eingrenzung auf einen bestimmten Kollegen, dass ich sehen will "welche Aufgaben hat der gerade bei sich liegen", arbeite dann aber wie gesagt mehr mit einzelnen Filtern als mit dem Dashboard. Werde mir das aber durchaus nochmal so einrichten, das es funktioniert. [00:28:24-1](#)

I: Und da gibt es für die Mitarbeiter, gibt es auch keine Vorgabe welches Dashboard sie verwenden sollen? [00:28:29-6](#)

B: Ich habe, ich stelle es ihnen frei was sie verwenden wollen. Also wenn sie Lust haben, dürfen sie sich gerne ihr eigenes bauen. Ich habe den Leuten auch angeboten, dass wenn sie andere Informationen, oder die Informationen anders aufbereitet haben wollen, und das selbst nicht einstellen können, dass ich ihnen auch gerne helfe dabei. Haben relativ wenige genutzt, aber ein Kollege nutzt das Dashboard wohl, und da hatten wir auch schon ein paarmal festgestellt, dass irgendwelche Tasks da jetzt aus irgendwelchen Gründen nicht angezeigt werden, weil der Filter noch einen Fehler hatte. Das haben wir dann entsprechend angepasst. [00:29:07-8](#)

I: Und verwendet ihr das JIRA zum Beispiel auch für das Reporting nach oben, oder für das Team auch? Also ich weiss bei den Entwicklern... [00:29:16-8](#)

B: Nein, eigentlich gar nicht. (...) Nein. eigentlich nicht. [00:29:27-9](#)

I: Wie funktioniert das dann, macht ihr dann da eigene Übersicht, oder Excelliste oder was? [00:29:33-3](#)

B: Also den wöchentlichen Status haben wir kein schriftliches Reporting gehabt jetzt in der letzten Zeit. Es gab ein Statusmeeting jede Woche wo wir mündlich dann berichtet haben. Gab aber keine festen Berichte die wir nach oben gegeben haben. Es gibt dann zum Meilensteinende ... [00:30:02-4](#)

I: (unterbricht) oder eben auch zum Beispiel teaminterne Statusberichte oder so? Also ich kenne das von Komposit, da wird glaube ich alle zwei Wochen oder so, da schreiben sie irgendwas zusammen, dass jetzt die Anzahl der Tickets... [00:30:16-4](#)

B: Aber in der Fachkonzeption so gar nicht. [00:30:18-6](#)

I: Nein? [00:30:18-6](#)

B: Nein. [00:30:21-3](#)

I: Ok. Und Tickets sind jetzt wahrscheinlich bei euch auch gar nicht so das grosse Thema. [00:30:22-4](#)

B: Tickets sind bei uns meistens halt nur durchlaufende Posten. Also ein Ticket direkt fixen können wir ja selten. Bei uns liegt halt ein Ticket zur Bewertung, ob das im Konzept falsch beschrieben ist, ob die Fachlichkeit irgendwie missverstanden wurde, ob es eine Änderungsanforderung ist die Bewertung durchzuführen. Eventuell Gespräche mit dem Kunden um ihm ein Ticket auszureden weil zu aufwendig, und nach Alternativen zu suchen. Aber Tickets sind bei uns immer eigentlich nur durchlaufende Posten. [00:31:02-0](#)

(JIRA-Login)

I: Also, das ist dein Standard-Dashboard? [00:32:19-9](#)

B: Das ist das Dashboard für die Fachkonzepter das ich erstellt habe. [00:32:26-6](#)

I: Das ist für alle Fachkonzepter? [00:32:26-6](#)

B: Das ist für alle meine Fachkonzepter. Leben hat etwas ähnliches gemacht, wir haben es aber immer getrennt, damit die Listen etwas kürzer werden. Man sieht jetzt hier, dass relativ viel Information drauf ist. Filter sind einmal auf den gerade eingeloggten Kollegen bezogen, welche Aufgaben hat der überhaupt, offene und wiedereröffnete. [00:32:54-6](#)

I: Also das wären jetzt deine eigenen Sachen hier? [00:32:54-6](#)

B: Das ist immer für denjenigen der gerade angemeldet ist. Das ist das Dashboard was für jeden Fachkonzepter zu Verfügung steht. Und wenn der Meier angemeldet ist sieht der Meier seine Aufgaben, und wenn der Huber angemeldet ist, sieht der Huber seine Aufgaben an der Stelle. [00:33:16-5](#)

I: Und du selbst tust jetzt... [00:33:25-3](#)

B: Ich selber habe mir ein Dashboard gemacht "Teamleiter **KKZ**", genau. [00:33:32-6](#)

I: Möchtest du vielleicht selbst ein bisschen herumzeigen was du da hast. [00:33:34-5](#)

B: Ja. Es gibt einmal eine Box für alle wartenden Aufgaben, wo ich mir einen Überblick verschaffen kann welche Themen werden denn jetzt gerade nicht bearbeitet, weil sie aus irgendwelchen Gründen gerade aufgehalten werden. Da habe ich die Chance nachzufragen bei den Kollegen. Es gibt eine Box mit QS-Anforderungen die seit mehr als drei Tagen überfällig sind. Werte da also das Fälligkeitsdatum an dem entsprechenden JIRA-Task aus und die Regel ist normalerweise gewesen, innerhalb von drei Tagen muss eine QS zurückkommen. Im Normalfall bringt es uns nicht um wenn es einmal eine Woche dauert, das ist im Normalfall auch noch vertretbar. Wenn es aber länger als eine Woche dauert, sollte nochmal nachgefragt werden was denn da ist. Und eine Box grundsätzlich für alle noch offenen Fachkonzept-Tasks. Also das JIRA-Projekt ist ja so aufgezogen, dass wir einmal eine Task für das Fachkonzept haben, zudem gibt es jeweils ein Untertask für das Fachkonzept-QS und Kundenabnahme für das Fachkonzept. Und mit der Box will ich halt wissen wie ist denn die Bearbeitung von dem Fachkonzept. Die Ansage ist halt, die Anleitung sagt auch, wenn der Fachkonzepter das Fachkonzept erst mal fertiggestellt hat und zur QS gibt, dann soll er diesen Fachkonzepttask auch schliessen, auch wenn dann eventuell noch Nacharbeiten notwendig sind, soll hierüber angezeigt werden, dass das Thema grundsätzlich schon mal erledigt ist. [00:35:07-1](#)

I: Ich sehe, du hast da jetzt eigentlich nur die Tasks selber. Die Person die die bearbeitet, das ist... [00:35:13-1](#)

B: ... sehe ich ... [00:35:13-1](#)

I: ... das ist dir im Prinzip egal. [00:35:14-1](#)

B: Genau. Sehe ich in der Ansicht nicht, weil wenn da was ist vom Thema her, wo ich sehe "oh, das ist aber ein wichtiges Thema" gehe ich entsprechend in den Task rein, und habe dann die vollen Informationen. Das ist einfach auch geschuldet, dass zu viel hier angezeigt, das ist nicht wirklich übersichtlich hiermit. (...) Wie gesagt, QS ist immer ein spannendes Thema, weil vom zeitlichen Ablauf ist es eigentlich immer so, dass wenn wir von der Fachkonzeption soweit fertig sind dass wir Entwickler-QS brauchen, die Entwicklungsabteilung gerade schwer damit beschäftigt ist den vorhergehenden Meilenstein für das Release fertig zu machen. Und dann kommt es durchaus häufiger mal vor, dass so eine QS mal untergeht oder durchaus ein bisschen länger dauert. Deswegen habe ich mir hier auch nochmal eine Box gemacht wo ich alle Fachkonzept-QSen sehe. Dann entsprechend hier

auch mit einem Status ob "offen" oder "QS abgenommen" oder "in QS" oder "wartend" oder wie auch immer. Und da sehe ich dann auch beim Hinblick dass gerade das Thema... . Mit den Komponenten bilden wir ab welche, im Prinzip ist das eine Verschlagwortung die wir da machen, damit ich auch sehe in welchem Bereich dieses Konzept gerade irgendwie relevant ist. (...) Das ist im Prinzip eine Box "2-Dimensionale Filterstatistik" mit der ich vor relativ kurzer Zeit erst angefangen haben zu spielen, die kannte ich anfangs nicht, ist aber ganz spannend um zu sehen "welchen Zustand haben denn die Themen bei den verschiedenen Bearbeitern". Also ich sehe hier jetzt zum Beispiel dass bei den meisten Kollegen keine offenen Themen mehr sind. Ich habe einen Kollegen der ein offenes Thema hat, und ich habe einen Kollegen der hat vier Themen. Ich sehe hier wieviel geschlossen ist, und ich sehe auch wieviel in Arbeit ist. Also "offen" heisst, das ist noch gar nicht angefangen. "In Arbeit" ist klar, da ist er gerade dran, und ich kann hier auch sehen wie viele Themen er hat. Wenn ich dann hier sage der Kollege mit vier, will ich wissen was sind das für Themen die noch offen sind kann ich hier über den Link auch direkt draufgehen. (...) [00:37:38-8](#)

I: Das ist "alle nicht-Leben Themen" [00:37:40-6](#)

B: ..."inklusive Machbarkeit". Ja in der Box, ich hab mir teilweise mit den Filtern und den Abfragen ein bisschen schwer getan. Wie kriege ich einen Filter hin, dass ich wirklich nur genau die Themen sehe, für die mich mein Produktmanagement auch beauftragt hat. Das Ding ist... [00:38:03-1](#)

I: (unterbricht) darum "nicht-Leben" [00:38:03-1](#)

B: Genau. Ich mache alle Themen die nicht "Leben" sind, das heisst es ist für mich die einfachste Eingrenzung zu sagen "zeige mir alles an was nicht Leben ist" dann sind sie alle, dann müssten die Themen eigentlich mich interessieren. War als Filter einfacher zu bauen als irgendwie andersrum. Da hätte ich die Liste von allen Komponenten irgendwie aufzählen müssen. (...). Also es gibt noch ein Dokument wo ich mit meinem Produktmanagement auch hin- und herschicke, das sind die Schätzraster. Wie gesagt, am Anfang vom Produktionsabschnitt werden die Themen geschätzt, dafür gibt es eine Excel-Vorlage wo diese Schätzung auch eingetragen wird. Die Schätzungen werden dem Kunden dann auch zu Verfügung gestellt von dem Produktmanagement. Und am Anfang müssen eben alle Themen geschätzt werden, werden dann ans Produktmanagement das Schätzraster übergeben, und dort wird es dann entsprechend weiterverarbeitet. Und hier sehe ich dann eben auch nochmal welche Schätzraster denn entsprechend auch da sind, bearbeitet sind. (...) Anfangs dachte ich das ist auch eine gute Idee zu sehen welche geschlossenen Fachkonzepte es gibt, diese Ansicht benutze ich aber so gut wie nie. Weil es ist eigentlich eher spannender welche Themen noch offen sind, als zu kucken welche schon fertig sind. [00:39:20-4](#)

I: Aber wie findet man zum Beispiel ein Fachkonzept das schon geschlossen ist und wiedereröffnet werden muss? [00:39:25-6](#)

B: Über die Suche meistens. [00:39:31-0](#)

I: Wie bist du mit der zufrieden? [00:39:30-5](#)

B: (lacht) [00:39:31-6](#)

I: Ich habe schon gehört dass die zum Teil nicht so super ist. [00:39:33-5](#)

B: Die Suche und ich, wir lernen langsam uns zu arrangieren. [00:39:39-4](#)

I: Man muss halt genau wissen wonach man sucht, oder? [00:39:41-9](#)

B: Ja. Es wird langsam besser. Ich habe anfangs bei den Filtern auch nur die einfache Ansicht verwendet. Ich bin mittlerweile soweit, dass ich sehr viel von den Filtern auch einfach in der erweiterten Ansicht mir handschriftlich praktisch erstelle. Und seit ich das mache komme ich mit der Suche auch besser klar. Weil ich dann gezielter einfach auf ein bestimmtes Attribut abfrage. Ich habe mir mittlerweile auf meinem Browser die erweiterte Suche die Ansicht auf einen Bookmark gelegt, so dass wenn ich etwas suche ich auch einfach reinspringe, und gar nicht die Schnellsuche oben verwende, sondern direkt über den Filter praktisch gehe, den ich aber nicht speichere, sondern jetzt nur für die aktuelle Abfrage verwende. Das heisst ich habe mich da eher so arrangiert, dass ich es ähnlich benutze wie auf einer Datenbank, dass ich per SQL mir einfach meine, meine... [00:40:37-7](#)

I: ach das verwendest du, oder was? [00:40:34-4](#)

B: ... meine Information. Bitte? [00:40:39-1](#)

I: Eben, es gibt ja dieses SQL, diese Eingabemaske. [00:40:39-7](#)

B: Genau. Ich weiss nicht wie die Sprache hier heisst bei JIRA, aber diese Eingabemaske benutze ich mittlerweile häufiger als die Schnellsuche. Hat ein bisschen den Grund auch, dass es nicht nachvollziehbar ist wie die Wildcards funktionieren, oder ob sie funktionieren. Ich hab den Eindruck mal geht es und mal nicht, und ich habe noch nicht herausgefunden wann es funktioniert und wann es nicht funktioniert. [00:41:06-5](#)

I: Ok. Und diese Widgets, schaust du da aber überall jeden Tag rein? [00:41:13-0](#)

B: Also wie gesagt, das Dashboard benutze ich mittlerweile nicht mehr so häufig, weil ich es mir einfach zu vollgepfropft auch habe. Es ist zu viel drauf und noch zu wenig strukturiert. Ich habe, warte jetzt bin ich eingeloggt, sehe ich dann hier, ja ich sehe meine, ich habe mir ein paar Filter gebaut, die ich einzeln ausführe. Ich muss nur gerade kucken wo er ist (sucht). Das müsste er sein. Das ist einmal die Fachkonzept-QS-Tasks, da müsste ich wahrscheinlich ein bisschen weiter ausholen. Wir benutzen, also das Ding ist: ich kriege einmal von meinem Produktmanager eine Liste von Aufträgen die ich bearbeiten soll. Das interessiert mich für einen konkreten Produktionsabschnitt über den berichte ich auch mündlich wie wir da stehen. Für den muss ich auch zusehen, dass wir für den Termin alle Konzepte fertig und abgenommen haben. Zur gleichen Zeit arbeitet die Entwicklung aber an einem anderen Produktionsabschnitt. Und die Software ist in Produktion auch schon. Mit einer anderen Version wieder. Das heisst es könne auch Tickets reinkommen für andere Produktionsabschnitte als den gerade zu bearbeitenden. Und es kann passieren, dass aus diesen Tickets Änderungen im Konzept notwendig sind. Wir hatten vorhin schon gesagt jede Konzeptänderung wird auch per JIRA wieder durch den Prozess durchgeschleust. Das sind aber unter Umständen Konzepte die jetzt nicht auf meiner Liste stehen für den aktuellen Meilenstein. Tickets laufen bei uns normalerweise nicht über das Produktmanagement, sondern die werden direkt hier in der Entwicklung und der Fachkonzeption bearbeitet, und nur wenn es wirklich grössere Sachen werden, dass der Aufwand sehr hoch, würde man nochmal eine Eskalation hochgehen und sagen "oh, ich habe hier aber ein Ticket was ich aber so nicht unterkriege". Das heisst, im Normalfall läuft die Ticketbearbeitung in einem kleineren Kreis als die normale Konzeptbearbeitung. Wie unterscheide ich das jetzt in JIRA? Irgendwie gar nicht. Weil ich muss irgendwie als Lösungsversion auch für die Tickets einen Produktionsabschnitt angeben. Wir sind augenblicklich aber so aufgestellt, dass wir Themen - also ein Thema wird NIE in nur einem Produktionsabschnitt bearbeitet, es wird immer Nacharbeiten geben in einem Folge-Produktionsabschnitt noch geben, das ist einfach so. Wir legen das dann aber normalerweise nicht neu an, weil wir JIRA auch ein bisschen zu Dokumentationszwecken benutzen - JIRA ist das Medium wo am Ende das gültige Fachkonzept liegt in der neuesten Version, und wo es auch zu suchen ist. Das heisst, wir machen ein bisschen Dokumentenmanagement auch mit JIRA, unter Anführungszeichen. Das führt aber dazu, dass ich unter Umständen bei einem Fachkonzept-Task hier nicht nur eine Lösungsversion habe, sondern zwei, drei, vier. Da tue ich mir mit meiner Eingrenzung schwer. Wie kriege ich jetzt die Liste, über die ich für mein Produktmanagement reporten muss? Das heisst, ich habe mich jetzt diesmal einmal bei einem Meilensteinanfang hingesetzt, und habe mir die Themen einfach alle gemerkt, habe mir angekuckt welche sind die Themen die wir haben, habe mir die Identen von den Elterntasks gesucht, habe die in eine Liste gepackt, und habe meine Abfrage so gebaut, dass ich für diese Meilensteinbetrachtung wirklich nur die Themen auch betrachte, indem ich hardcodiert einfach nur die Identen hier verwendet habe. Ich habe keine andere Möglichkeit gefunden als die Liste soweit einzugrenzen dass ich nur die Themen sehe die mich interessieren. So. Und dementsprechende habe ich hier so den QS-Task. Das ist für mich so der beste Indikator "wo stehen wir mit dem Thema", weil wenn eine QS offen ist, dann heisst das das Fachkonzept ist noch überhaupt nicht fertig, das ist nie zur QS gegangen. Wenn es in QS ist weiss ich, das Konzept ist grundsätzlich erst mal fertig, das ist jetzt praktisch schon im Dialog mit Entwicklung und Fachkonzeption ob das so in Ordnung ist. "Nachgebessert" heisst, die Entwicklung hat es sogar schon einmal QSt. Und ich weiss das liegt jetzt wieder bei meinen Leuten, und die müssen nochmal was tun. [00:45:48-6](#)

I: Das ist wenn ein Entwickler bei der QS sagt, irgendwie "ich hab noch eine Frage". [00:45:52-6](#)

B: Genau. Hat noch Fragen, oder es muss noch irgendetwas korrigiert werden, muss noch anders beschrieben werden, funktioniert so nicht. Dann wird nachgebessert und geht zurück an den entsprechenden Konzepter. Das heisst, die Ansicht über die Fachkonzept-QS-Tasks habe ich jetzt für mich herausgefunden dass es eigentlich die ist, die für mich am meisten Aussagekraft einfach hat. Und in die kucke ich auch täglich rein und mache mir ein Bild "wo stehen wir denn da". [00:46:16-3](#)

I: Kurze Frage noch: die QS, da gibt es ja nach wie vor dieses Word-Dokument das zum... [00:46:25-5](#)

B: Nein, in den Fachkonzepten sind wie so, ja jetzt habe ich falsch geklickt ([00:46:33-1](#)), gehe ich nochmal zurück. In den Fachkonzepten sind wir mittlerweile so, dass die QS nicht mehr in einem eigenen Word-Dokument abgebildet wird, sondern das wird hier in JIRA - da haben wir ein Template reingemacht - wo man so ein Frage-Antwort Spiel praktisch machen kann ([00:46:57-0](#)). Die ganze Box ist als Template hinterlegt, einmal in jedem Fachkonzept-QS-Task schon mit der Anlage des Tasks in JIRA, ist der automatisch schon mit drin, und kann dann entsprechend von den Kollegen kopiert werden, und an den zu verändernden Stellen mit Leben befüllt werden. Das heisst, der Fragesteller trägt sein Kürzel ein, trägt das aktuelle Datum ein, die Versionsnummer des Konzepts auf die es sich bezieht, stellt seine Frage, und der Fachkonzepter hat dann hier die Möglichkeit entsprechend eine Antwort zu einer Frage mit aufzunehmen. Es gibt hier oben noch den Bereich wo so ein bisschen die

Abnahmeinformationen verwaltet werden, einmal um welche Fachkonzept-Version geht es, dann sind Anforderungen drin die die technische Plattform betreffen, sprich Architektur müsste hier auch nochmal tätig werden, "ist eine Schätzung zum jetzigen Zeitpunkt schon möglich für eine Umsetzung" und "könnte der Entwickler jetzt schon anfangen die Umsetzung vorzunehmen" mit dem Stand den er jetzt gesehen hat. (...) Wir hatten vorher, anfangs hatten wir überlegt ob wir das Dokument auch noch benutzen wollen, aber wir haben gesagt "warum eigentlich". Also eigentlich reicht es die Aufgabe hin- und herzuschieben, und hier entsprechend die Dokumentation auch vorzunehmen. Dann muss ich nicht extra auch nochmal ein Worddokument öffnen, sondern sehe hier schon wo steht es ([00:48:22-9](#)) [00:48:23-0](#)

I: Wenn da jetzt zum Beispiel der Entwickler fertig ist mit der QS, wird dann der Task wieder dem Fachkonzepter, der Fachkonzepterin zugewiesen? [00:48:37-9](#)

B: Ja. Grundsätzlich immer, also sowohl wenn der Entwickler sagt "ja es passt", als auch wenn der Entwickler sagt "nein da ist noch was zu tun", geht der Task zurück an den Fachkonzepter, damit der auch die Möglichkeit hat zu sehen "das ist jetzt fertig" oder "ich muss noch was nacharbeiten". Es gab in der Anfangsphase so ein paar Schwierigkeiten dass die Leute vergessen haben mit „QS-Abnehmen“ das Ding auch wieder zurückzuweisen, und das sind dann immer die Dinge, die dann erst mal liegen bleiben. Weil dann ist nämlich keiner mehr zuständig, und keiner weiss was gerade der Status ist, weil der Entwickler sagt "ich bin ja fertig", und der Konzepter weiss es aber einfach nicht, weil er keine Informationen mehr darüber kriegt. [00:49:40-2](#)

I: Also du weist es zu, und der Fachkonzepter bekommt dann wahrscheinlich über das JIRA ein Benachrichtigung dass sich der Status geändert hat? Und wie läuft das dann? [00:49:47-7](#)

B: Die Email-Benachrichtigungen weiss ich gerade nicht wie die eingestellt sind. Ich filtere da relativ viel in Notes weg, weil ich finde das Benachrichtigungssystem... ich brauche es im Normalfall nicht. Ich sehe den Workflow mit JIRA so, das ist für mich wie ein schwarzes Brett. Ich muss hinlaufen und kucken. Also ich brauche keine aktive Information, ich weiss welche Informationen für mich relevant sind. Das sind hier entweder Sachen die mir zugewiesen werden, die kann ich mir filtern, oder ich habe bestimmte Filter mir angelegt für Informationen die bei Dritten liegen, die mich aber interessieren. Da habe ich mir auch einen Filter angelegt. Das heisst, diese aktive Benachrichtigung ist mir persönlich zu inflationär bei JIRA. Deswegen filtere ich die weg, und benutze es einfach indem ich reingehe und mir die Information herausschleife die ich gerade brauche. [00:50:43-3](#)

I: Wie filterst du die weg, kann man das einstellen bei JIRA dass man keine Email bekommt? [00:50:46-4](#)

B: Nein, ich kann das in Notes einstellen. Ich mache mir nur einen Filter, Sortierfilter, und sage "wenn Absender JIRA", oder "wenn in dem Thema irgendwo JIRA steht, dann sortier sie mir weg". [00:51:02-5](#)

I: Ja, sehr gut. Ich glaub das wars dann. Vielen Dank. Eine kurze Frage noch: vielleicht weil vorher die Sache war mit QS: wenn du jetzt siehst die QS liegt noch bei dem und dem Entwickler, und die ist schon drei Tage überfällig, schreibst du dann direkt dem Entwickler, oder gehst du rüber, oder wie wird das dann gehandhabt? [00:51:34-6](#)

B: Ich wende mich dann normalerweise an den Teamleiter, und beschwere mich bei dem dass die QSen zu lange dauern, und versuche mit ihm einen Termin zu vereinbaren bis wann wir es denn kriegen. Man kann bei manchen Themen auch noch ein bisschen jonglieren und sagen "he, das ist jetzt nicht super dringend, das ist ein kleines Thema", wenn die QS dann durch ist und wenn noch Nacharbeiten sind dann dauert es trotzdem höchstens einen Tag oder so, dann ist das kein Problem wenn es mal länger dauert. Wenn aber eventuell die Gefahr besteht, dass es etwas Grösseres sein könnte, dann muss man halt kucken wann kriegt man das dann, weil ich dann ja auch eventuell Risikomeldungen nach oben geben muss und sagen muss ich kriege aber bis zu einem festgelegten Termin keine Abnahme für das Konzept weil die Entwickler-QS noch nicht da ist. Und dementsprechend wende ich mich dann normalerweise an den Teamleiter drüben, im Entwicklungsteam, und sage "hier, bei dem Thema XY". Läuft meistens so, dass ich persönlich vorbeilaufe, weil einige von den Teamleitern JIRA-E-mails auch grundsätzlich wegfiltern, und ich damit Gefahr laufe, dass meine Email leider ins Nirvana verschwindet und nicht beantwortet wird. Insofern bringt mir das nichts. Weil ich kann ja einzelne Tasks direkt als Email verschicken, und sagen "ja was ist denn damit", das würde nicht funktionieren. Ich kopiere es mir entweder in eine eigene Email rein, oder ich laufe rüber und sage "hier, Task XY123, bräuchte ich bitte eine entsprechende QS". [00:53:10-0](#)

I: Ok, dann möchte ich mich herzlich bedanken. Hast du vielleicht noch irgendwelche Fragen? [00:53:27-3](#)

B: Ich bin mir noch nicht ganz sicher was das Ziel dann ist? Du willst den Nutzen im Prinzip herausfinden. Was sind sinnvolle Mittel die ich nutzen kann. [00:53:39-1](#)

I: Genau. [00:53:39-1](#)

B: Hast du speziell auf JIRA festgezogen, dass du sagst du willst JIRA an der Stelle betrachten? Oder ist das ganz

allgemein gehalten? [00:53:43-3](#)

I: Schon hauptsächlich JIRA. Also schon auch ein bisschen den Vergleich zu Email oder Sametime oder so, und ja. [00:53:52-1](#)

B: Also JIRA gibt einem halt die Möglichkeit dass jeder reinkucken kann der die entsprechenden Zugriffsrechte hat, und nicht nur die Leute die gerade an den Tasks arbeiten. Und Email oder Sametime sind nur die zwei Akteure drin. Ein anderer sieht es nicht. Und das ist etwas, was ich jetzt auch gemerkt hat, wie wir dieses **LFS**-Projekt in JIRA eingeführt haben, wo wir alles drin haben, kam mein Chef öfters mal und sagt der findet das ganz toll, weil da kann er jetzt plötzlich sehen was da gerade passiert. Er hat da ein bisschen reingekuckt und so und kann sich da selber auch ein Bild machen. Eine Gefahr die natürlich besteht ist, wenn derjenige vorher nicht bei den Prozessen der Erstellung - also wie ich das in JIRA abbilde - mit dabei gewesen ist, kann es passieren, dass er Sachen fehlinterpretiert. Wenn er das Handbuch noch nicht gelesen hat wie das jetzt gerade zu verstehen ist, was bedeutet das wenn ein Task "wartet". Weil wir uns natürlich für uns bestimmte Regeln natürlich überlegt haben, und gesagt haben, keine Ahnung, er wird auf warten gestellt wenn es länger als drei Tage dauert. [00:55:01-5](#)

I: Ja. Da gibt es ja auch dieses Dokument dazu. [00:55:04-2](#)

B: Da gibt es Handbücher auch entsprechend dazu, die sind auch über die JIRA-Seite direkt... [00:55:06-4](#)

I: Die sind glaube ich gleich beim Login [00:55:05-9](#)

B: Weiss ich gar nicht ob da ein Link drauf ist, man kann sie glaube ich auch über ... [00:55:15-0](#)

I: ... Ich werde dich hier mal ausloggen, das ist glaube ich gleich hier ganz am Anfang, da sieht man diese Handbücher. [00:55:22-3](#)

B: Richtig, da ist ein Link drauf, da sind die Handbücher. [00:55:28-4](#)

I: Ja, eure Abteilung ist ja sehr interessant, weil es gibt halt eine Arbeit, die untersucht dieses "Jazz" von IBM, das ist auch so ein ähnliches Tool wie JIRA, das ist das einzige das es zu dem Thema gibt, aber die haben halt hauptsächlich Entwickler befragt. Und eben Fachkonzeption zu sehen, und die haben halt wahrscheinlich einen ganz anderen Zugang dazu als es ein Entwickler hat. [00:56:04-1](#)

B: Das ganz sicher. Also, (seufzt) wir machen jetzt auch mehr mit JIRA als mir persönlich eigentlich recht ist. Das ist aber einfach auch ein bisschen aus der Not geboren, dass wir kein anderes Tool haben, dass wir in absehbarer Zeit kein anderes Tool kriegen werden. Ich halte JIRA nicht wirklich geeignet um Dokumentenmanagement mit zu machen. Also ich kann das sicherlich auch alles machen, ich kann das verschlagworten, ich kann das verknüpfen verschiedene Tasks hintereinander, ich kann mir sicherlich auch Hilfsmittel herstellen um Dokumente auch vernünftig wiederfinden zu können. Was ich schon nicht machen kann, ist eine Volltextsuche auf die Dokumente, was für ein Dokumentenmanagement für mich zwingend mit dazu gehört, dass das möglich ist. Und ich bin mir noch nicht sicher, wie die Performance auf Dauer aussehen wird. Bei der Anzahl an Dokumenten die wir reinhängen - jedes Word-Dokument hat dann doch schnell mal ein paar MB auch - und dann weiss man jetzt auch noch nicht so genau wie es dann reagieren wird. Aber es hat uns auf jeden Fall geholfen deutlich mehr Transparenz reinzukriegen. Wo welche Themen liegen, wer gerade irgendwie am Ball ist bei den Themen, welche Version erstellt wurde, welche Version QSt wurde, das war vorher alles sehr mühsam herauszufinden. Das geht jetzt deutlich einfacher. Du hattest vorhin schon die Suchfunktion angesprochen, die finde ich noch relativ schlecht, funktioniert dann gut, wenn man sich genau überlegt hat wie man die Dinger benennt, und da entsprechend auch auf eine Suche Rücksicht genommen hat. Oder wenn man vielleicht zusätzliche Attribute einführt über die ich dann in der Suche... [00:57:47-8](#)

I: Das ist dann die Verschlagwortung die du gemeint hast? [00:57:49-6](#)

B: Zum Beispiel, genau. [00:57:50-2](#)

I: Das ist, was ich bei den Tasks... [00:57:50-6](#)

B: Die benutzen wir... Verschlagwortung benutzen wir im Augenblick nur die Komponenten zu, es gibt ja auch die Möglichkeit, direkt Stichworte zu erfassen zu jedem Task, das nutzen wir im Augenblick gar nicht bis wenig. Also wenig bis gar nicht. Ist, einfach aus dem Grund, weil ich es am Anfang nicht gesehen habe, und nicht wusste dass es da ist, und es dann einfach auch ein bisschen zu viel war. Wir mussten überhaupt erst mal kucken dass wir den Workflow vernünftig abgebildet kriegen, ist aber etwas wo man sich unbedingt nochmal Gedanken drüber machen sollte, weil je länger das Projekt lebt, und je mehr Tasks dann irgendwann drin sind - weil wir archivieren sie auch nicht, die sind dann erst mal auf alle Ewigkeit erst mal drin - umso schwieriger wird es auch wieder eine

Eingrenzung vernünftig zu machen. Wie schon gesagt, grundsätzlich arbeiten wir eigentlich Produktionsabschnitts-bezogen, benutzen JIRA aber für Dokumentationszwecke übergreifend, thematisch eigentlich. Und das macht es dann auch wieder schwer, weil ich ein Thema über mehrere Produktionsabschnitte mit dem selben Task immer weiter schleppe, und es dadurch immer weiter aufblähe. Weil die Historie von diesem einen Task dann immer länger wird. Wenn ich mir dann soetwas wie einen Fachkonzept-QS-Task vorstelle, bei einem Thema was jetzt über drei oder vier Meilensteine gelaufen ist, der wird halt irgendwann auch lang. Und dann ist die Übersichtlichkeit und Wiederfindbarkeit von Informationen auch nicht mehr so gegeben. Also der Weisheit letzter Schluss ist es meiner Meinung nach noch nicht, hat uns im Vergleich zu vorher aber deutlich mehr Transparenz reingebracht. Auch für Aussenstehende die sich aber für die Themen interessieren die wir bearbeiten. 00:59:36-5

I: Na gut. Dann herzlichen Dank.

Interview 03_ENTW_DE_2

I: Also erste Frage einmal ist zu deiner Aufgabe hier im Team, was machst du im Grossen? Also du bist Softwareentwicklerin? 00:00:20-0

B: Ja, ich programmiere hier den Produktkern. 00:00:22-7

I: Produktkern, ok. Und wie sieht bei dir der, also welche Aufgaben machst du im kleinen, also jetzt zum Beispiel QSen, Implementieren... 00:00:31-3

B: Ja, ich, also, hauptsächlich QSen von Produkt-Konzepten und den Testlisten dazu und die Implementierung. Das sind halt Themen. 00:00:42-9

I: Und mit welchen Leuten arbeitest du für diese Sachen zusammen? 00:00:51-5

B: Hauptsächlich mit **ENTW_DE_1**,... 00:00:55-0

I: Der ist bei dir im Team? 00:00:57-8

B: Der ist bei mir. Ja, mit (unv. **ENTW_DE_TL**) natürlich, und so ein paar Berührungspunkte mit den Testern, aber ansonsten auch nicht viel. Und bei mir im Team natürlich auch teilweise zum Nachfragen, und so. 00:01:10-4

I: Na eben, zum Beispiel gestern war es ja so da... 00:01:12-7

B: Ja, da natürlich mit den Fachkonzeptern (Anm.: am Vortag waren zwei Leute aus der Fachkonzeption bei ihr im Büro für eine fachliche Besprechung). Da muss man dann ja nochmal einige Sachen klären. Dafür war das gestern da. 00:01:26-5

I: Ok. Wie du zum Beispiel gestern diese Besprechung mit den Fachkonzeptern gemacht habt, wie ist das, wie wurde das ausgemacht? Hast du die angechattet, oder angerufen? 00:01:36-1

B: Ich habe sie eingeladen und einen Termin dafür ausgemacht. 00:01:38-8

I: Ach direkt über Lotus Notes? 00:01:42-1

B: Mit Lotus Notes. Ich habe allerdings vorher kurz gefragt ob sie noch da sind, ich habe das nicht gleich, nicht gleich den Termin ausgemacht, sondern die erst mal gefragt, und dann habe ich den Termin eingetragen. 00:01:50-0

I: Und gefragt über Chat, oder Email? 00:01:51-4

B: Nein, ich habe sie angerufen. 00:01:53-7

I: Angerufen, ok. 00:01:55-2

B: Angerufen, ganz normal. 00:01:57-1

I: Hast du mit den **Österreichern** auch öfter zu tun zum Beispiel, oder mit **Süddeutschland** oder so? 00:02:01-7

B: Beides relativ wenig. Da dann vor allem mit **ENW_OE_3** oder mit **ENW_OE_2**, wenn ich da irgendwelche Nachfragen habe, was weiss ich, was sie zu tun haben, was, wo bei uns eben auch vorkommen, da natürlich schon. Und bei den **Süddeutschen** so ein bisschen was manchmal Tickets betrifft, oder ja eher Referenzwerte, da hatte ich mal eine Nachfrage zu. Also relativ wenig. 00:02:29-1

I: Welche Projekte machst du jetzt bei Produkt? 00:02:37-7

B: Im Moment habe ich die ganze letzte Zeit mit Hausrat zu tun gehabt, jetzt fange ich wieder mit Glas an. Fange ich wieder an, ich glaube das ist jetzt das fünfte Mal. Ich habe mal angefangen, und nach ein paar Tagen wurde es dann wieder abgebrochen und ich habe was anderes tun müssen. Und ich hoffe dass es jetzt mal endlich länger dauert. 00:02:50-5

I: Ok, und bist du für das Produkt-Entwickeln bei diesen zwei Projekten Glas und Hausrat, bist du da die einzige,

oder macht da der **ENTW_OE_3** auch noch was? [00:03:01-1](#)

B: Bei Glas war eigentlich geplant dass **ENTW_OE_3** da auch was mitmacht, aber ich weiss es noch nicht. Ich meine im Juni war es geplant dass er da mitmacht. Da wurde das ganze wieder..., bin ich ja auch nicht dazugekommen, und er ist auch nicht dazugekommen. Jetzt fange ich das erst mal wieder an, ich weiss nicht ob er da jetzt mitmacht, oder ob ich das alleine weitermache, oder ob **ENTW_DE_1** da später noch ein bisschen was macht, das sehen wir erst mal. Also ich fange da jetzt wieder an, und dann mal kucken. [00:03:26-9](#)

I: Und bei Hausrat, wie war das da? [00:03:31-0](#)

B: Da hat **ENTW_DE_1** ein bisschen mitgemacht. [00:03:35-3](#)

I: Also du und **ENTW_DE_1** gemeinsam. [00:03:35-4](#)

B: Gemeinsam kann man nicht sagen, das meiste habe ich gemacht, und ein bisschen hat **ENTW_DE_1** gemacht. Er hatte hauptsächlich das mit den Tabellen und so gemacht. [00:03:44-2](#)

I: Und was würdest du sagen, wie unterscheidet sich da die Kommunikation zum Beispiel, wenn du mit **ENTW_DE_1** irgendwie etwas beredest, im Gegensatz dazu wenn du mit den **Österreichern** etwas bereden musst. [00:04:02-8](#)

B: (lacht) Das ist ganz einfach, ich meine, bei den **Österreichern** frage ich eigentlich nur an wenn es irgendwas zu klären oder abzustimmen gibt was, oder Fragen wo wir nicht weiterkommen. Zum Beispiel bei **xxx** kennen **ENTW_OE_3** und **ENTW_OE_2** immer noch besser aus als wir denke ich mal. Und wenn es da zu Fragen kommt, frage ich da natürlich mal. Oder wenn ich merke dass irgendwo ein Fehler ist den ich weitergeben sollte, dann sage ich natürlich auch Bescheid. So mit **ENTW_DE_1**, ich meine der sitzt mir gegenüber, da kann ich auch einfach auf einmal mit dem Stuhl oder irgendwie, oder wenn etwas erklärt ist, oder wenn schon wieder was fehlt oder so (lacht). Oder wenn ich sage "hm, ich kann das ja eingeben, aber es gibt bestimmt einen Fehler" solche Sachen. Also da redet man ja ganz anders drüber. Bei den Fachkonzeptern da (...) ja, wenn es was zu klären gibt, oder nachzufragen gibt, das ist dann teilweise auch relativ häufig, manchmal recht selten. Ja mit **ENTW_DE_1** wir sprechen nicht das ganze Konzept durch, aber wir reden schon mal drüber, oder wenn einem etwas auffällt, dann sagt man das einfach. Das ist dann eher alltäglich. Oder wenn irgendeine Frage ist, manchmal hat man ja auch eine Frage im Kopf, man kommt nicht drauf, und dann fragt man, und dann weiss man "ok, das hätte ich auch gewusst". Das hätte man die **Österreicher** ja gar nicht gefragt, sondern man hätte einfach noch ein bisschen rumprobiert, dann hätte man das auch gehabt. Also man fragt ja ganz anders. Oder redet ganz anders. [00:05:39-2](#)

I: Ja, die nächste Frage ist, interessiert du dich dafür, was deine Kollegen im Zimmer jetzt gerade machen. Wenn du zum Beispiel hörst **ENTW_DE_1** telefoniert mit irgendwem über irgendein Problem, und ich weiss nicht, möglicherweise tangiert dich das Problem, möglicherweise aber nicht, aber fragst du dann zum Beispiel nach was denn los ist, oder du ihm irgendwie vielleicht helfen könntest, oder solche Sachen? [00:06:08-1](#)

B: Eigentlich nur wenn ich merke dass es mich auch irgendwie berührt. Weil **ENTW_DE_1** macht ziemlich viel mit Dialog, und da habe ich nicht viel Ahnung von. [00:06:18-9](#)

I: Ok. Spielt das für dich eine Rolle, woran deine Kollegen im Projekt gerade arbeiten? ... [00:06:23-9](#)

B: Ja natürlich [00:06:29-0](#)

I: ... Welche Task die jetzt gerade machen? [00:06:30-3](#)

B: Vielleicht nicht ganz so genau immer im Einzelnen, aber es interessiert mich schon was sie machen, oder was allgemein gerade gemacht wird, ja. [00:06:36-1](#)

I: Und interessiert dich das nur, also wenn man jetzt anschaut die Ebenen, zum Beispiel also nur mit den Leuten die bei dir im Büro sitzen, mit den Fachkonzeptern, die quasi vorgelagertes Team sind, oder mit den **Österreichern** jetzt, oder mit den Testern - interessiert es dich bei denen auch noch woran die gerade arbeiten? [00:07:02-8](#)

B: Jein. Ich meine, es kommt darauf an wie genau. Ich meine bei den Fachkonzeptern interessiert es natürlich, zu wissen was die gerade machen. Wenn es darum geht dass irgendwas noch nicht ganz fertig ist, oder um einfach so ganz grob zu wissen. Ich weiss im Moment aber jetzt auch nicht was **FK_2** hier jetzt gerade macht. Es gab zum Beispiel bei Hausrat ein paar kleine Änderungen, bei Glas gab es noch ein paar Sachen, da standen noch QSen von mir aus, die brauchte er natürlich auch irgendwann wieder... Und, aber was er nun ganz genau im Moment macht weiss ich nicht, was im Moment sein Hauptthema ist. Das ist ja nun, das... [00:07:39-2](#)

I: Aber es würde dich prinzipiell schon interessieren ob er jetzt, keine Ahnung, zum Beispiel gerade, ich weiss nicht, ... [00:07:45-6](#)

B: Ich weiss es nicht... [00:07:45-6](#)

I: Ob er jetzt Glas oder KFZ irgendwas [00:07:51-9](#)

B: ... (unsicher) Ja, aber (...) es ist immer so die Frage wie ich das im Einzelnen, wie es mich vielleicht berührt. Aber ansonsten weiss ich nicht genau. [00:08:01-8](#)

I: Und mit der Testabteilung zum Beispiel, interessiert dich das irgendwie? [00:08:08-5](#)

B: Ja da kriegt man das ja mit, welche Tickets sie bearbeiten und, da weiss man ja auch schon so ungefähr was sie testen. [00:08:15-5](#)

I: Aber kennst du zum Beispiel den Zeitplan von den Testern, weil da gibt es ja immer so Testwellen oder so. [00:08:17-5](#)

B: Nein, kenne ich nicht. [00:08:23-7](#)

I: Spielt es für dich eine Rolle wie deine Kolleginnen zum Beispiel ausgelastet sind? [00:08:32-2](#)

B: Ja natürlich. Also ich finde es nicht in Ordnung, also das ist einfach rein menschlich. Also ich fände es zum Beispiel nicht in Ordnung wenn - ich habe einen 30 Stunden Vertrag, **ENTW_DE_1** hat einen 40 Stunden Vertrag - wenn ich merke, ich... Also es wäre nicht in Ordnung wenn ich sage "ich arbeite meine 30 Stunden", und das was bei mir anfällt und ich es nicht schaffe, wenn **ENTW_DE_1** das erledigen muss und 50 Stunden arbeiten muss. Erstmal wäre das nicht in Ordnung, einfach aus Gerechtigkeitsgründen, weil **ENTW_DE_1** kann ja nur Sachen von mir abnehmen. So, wenn jetzt jemand, was weiss ich, wenn **ENTW_DE_3** zum Beispiel viel zu tun hätte, dann könnte ich ihm natürlich nichts abnehmen. Und wenn ich dann merke, dass alle Dialogleute fürchterlich viel zu tun haben, dann würde ich schon mal nachfragen warum das denn im Moment gerade so ist. Aber ich glaube auch **ENTW_DE_TL** hat das ziemlich gut im Griff im Moment, dass das nicht mehr so auftritt. Als ich angefangen habe, da kam das nämlich ganz schön vor dass Dialogleute ziemlich viel zu tun hatten. Und das ist jetzt so ein bisschen entspannter geworden. Weil da eine Zeitlang viel zu viele Überstunden angefallen sind. Aber ich meine da kann ich dann auch nicht viel machen. Wenn **ENTW_DE_1** viel zu tun hat, ok, dann muss ich eben auch mehr machen, damit einfach der Brocken weggeht. Aber mehr kann ich dann auch nicht machen, weil ich kann ja nichts von Dialog einfach so nehmen. [00:09:52-7](#)

I: Und gesehen auf die anderen Teams zum Beispiel, wenn ein Fachkonzepter sehr viel zu tun hat - kriegst du das mit, oder interessiert, oder tangiert dich das? [00:10:03-3](#)

B: Das kriegt man eigentlich kaum mit. [00:10:03-5](#)

I: Muss man aber auch nicht wissen. Oder will man es wissen? [00:10:09-1](#)

B: Ich denke, man sollte es schon wissen. Ich meine, wir können da nicht viel machen, aber ich meine, es wäre schon interessant wenn die viel machen müssen. Wenn man, das sollte man schon irgendwie wissen. [00:10:27-6](#)

I: Aber in welcher Form würde man das zum Beispiel mitkriegen dass das, ich weiss nicht, wenn man ab und zu, zum Beispiel, ich weiss es von mir, von der Testabteilung, der **TST_1** sitzt ganz alleine dort oder so und muss 10000 verschiedene Sparten testen... [00:10:46-7](#)

B: Ja bei der Testabfertigung merkt man das dann ja auch, dass wir dann manchmal ein paar... mehr nachtesten... was dann die Tester... Also da können wir ja auch so ein bisschen (langgezogen) entlasten. Aber ... [00:10:59-4](#)

I: Nun, mir geht es jetzt zum Beispiel auch darum, um die Art wie man das mitkriegt? Ob man das irgendwie am Gang hört, oder von einem Kollegen hört, oder ob man jetzt einfach sieht, keine Ahnung, da sind 10000 nicht nachgetestete Tickets? [00:11:14-7](#)

B: Also wenn, dann besprechen wir das im Scrum. Und also ich kann nicht einfach sagen - und ich mache das auch nicht einfach - wenn ich jetzt weiss, die haben ganz viel zu tun, dann sage ich "ok, gib mir meine ganzen Tickets, dann teste ich die selber nach" - das kann ich nicht machen. Weil,... aber wenn wir jetzt im Scrum besprechen "es ist im Moment so" und da müssen wir halt im Moment eben sehen dass wir die Tickets selber nachtesten, damit wir da ein bisschen von dem Berg herunterkommen, das ist was anderes. Aber ich kann ja

nicht einfach eine Arbeit machen die ich nicht, für die ich eigentlich nicht eingeteilt bin, und wo eigentlich andere Arbeiten anstehen. Ich meine, Glas zum Beispiel, wenn jetzt ein riesen Berg wäre bei den Testern, und wir sagen bei uns "ja, ich soll jetzt aber bei Glas weitermachen", da kann ich nicht sagen "ok, ich mache jetzt aber lieber mal ein paar Nachtests und dann mach ich bei Glas weiter" - da bin ich plötzlich gleich wieder eine Woche hinterher. Das geht ja nicht. Aber wenn wir jetzt im Scrum, also wenn wir absprechen "ok, jetzt machen wir mal zwei Tage nur Tickets, oder drei Tage", oder wir hatten auch mal gemacht, dass wir vormittags nur Tickets machen oder nachtesten, dann kann man das machen. Aber ich kann ja nicht einfach eigenständig bestimmen was ich mache - dann handle ich ja gegen das Team, das geht ja nicht. Man könnte höchstens, wenn man das mitkriegt, mal sagen, aber meistens wissen sowieso **ENTW_DE_TL_STV_1** oder **ENTW_DE_TL_STV_2** oder **ENTW_DE_TL** bescheid. [00:12:31-7](#)

I: Ja dann wäre die Frage, ob das wichtig für dich ist, und wie du darüber erfährst, ob es zum Beispiel Probleme im Projekt irgendwie gibt. Also dass zum Beispiel innerhalb von Komposit, dass man merkt irgendwie, da können, dass es schlecht läuft irgendwie, dass irgendwie Termine nicht eingehalten werden können, oder dass total viel los ist, oder dass die Tester nicht fertig werden, oder dass das Fachkonzept nicht fertig wird. (...) Wie wird soetwas kommuniziert, oder wie kriegt man das mit. Wahrscheinlich eh über das Scrummeeting und so? [00:13:18-3](#)

B: Ja wir reden ja eigentlich im Scrum über solche Sachen. Wir treffen uns zweimal die Woche, ja zweimal die Woche ist Scrum, und einmal in der Woche so ein Wochenmeeting, wo eigentlich hauptsächlich so alles mögliche gesagt wird, oder eigentlich... . Ja, alles mögliche wird eigentlich da gesagt. Und da wird das eigentlich verbreitet was wichtig ist. [00:13:41-0](#)

B: Das ist nur das Kompositteam. [00:13:41-0](#)

I: Wieviele Leute seid ihr da? [00:13:42-5](#)

B: Elf im Moment. [00:13:45-2](#)

I: Und da sind auch von **Produkt**, und von Dialogentwicklern alles dabei. [00:13:51-4](#)

B: Ja. [00:13:51-4](#)

I: Wie wichtig ist es für dich, über bestimmte Dokumente informiert, oder am Laufenden zu sein. Zum Beispiel Fachkonzepte. Wie erfährst du da, dass sich irgendetwas ändert? [00:14:09-6](#)

B: Ja, eigentlich über JIRA eben, (lacht) aber es kommt immer noch vor, dass ich sie einfach per Email bekomme. Das ist immer so eine Sache. Wenn man jetzt, eine Zeile in einem Fachkonzept geändert wird, dann ist es blöd, finde ich, dafür eine neue Version zu machen. Und **FK_2** hat es bis jetzt meistens auch immer so gemacht dass meistens so ein paar Sachen gesammelt sind, und dass es dann erst eine neue Version gibt. Manchmal, also heute hat er gerade eine in JIRA reingestellt, ein etwas anderer Rechenweg - nein, es nicht ein anderer Rechenweg, es ist anders gerechnet als es vorher war - und da hatte ich ja die QS schon abgenommen, und hatte ich auch gesagt, dass es so in Ordnung ist, also, man muss da nicht wieder den ganzen Weg laufen. Manchmal ist der aber auch gerade am arbeiten, und dann kommen noch Sachen dazu, die hat er direkt am Tisch liegen vielleicht, aber ich brauche das was er gerade hat, dann schickt er sie manchmal einfach per Email, und ein paar Tage später ist sie dann halt in JIRA. Aber ich habe sie dann schon mal, und kann schon damit arbeiten. Also das, die kriege ich alle über JIRA, und dann speichere ich sie bei mir zur Sicherheit nochmal ab, damit ich nicht so lange rumsuchen muss. Denn wenn ich keinen JIRA-Task mehr habe, dann muss ich das ja erst mal irgendwie wieder finden, und das ist nicht immer so ganz einfach. [00:15:34-8](#)

I: Ja und zum Beispiel über andere Sachen, über Sourcecode-Änderungen oder so? Ist bei dir viel Code den du gemeinsam mit anderen Entwicklern bearbeitest? [00:15:57-8](#)

B: Nein. [00:15:57-8](#)

I: Dann eine Frage zum Sametime: und zwar, wieviele Leute hast du in deiner Sametime-Kontaktliste, und hast du die irgendwie organisiert in Gruppen oder so? [00:16:13-4](#)

B: Ich habe, ja einmal natürlich unser Team, dann habe ich die Konzepte einmal in einer Gruppe extra. Und dann hatte ich, oder habe ich immer noch, noch eine Gruppe wo ich so alle drinstehen habe die ich kenne. Und dann habe ich festgestellt, dass man so, oder dass es so Obergruppen gibt für ganz **VersicherungsGmbH Norddeutschland**, und die habe ich mir dann auch noch eingerichtet. [00:16:40-0](#)

I: Das sind alle 150 Mitarbeiter? [00:16:42-5](#)

B: Ja, genau. Wobei mich stört, dass - ich glaube die Reihenfolge ist immer Alphabetisch in diesem Sametime. Das finde ich immer ein bisschen unschön. Ja weil das ist unwichtig. Die 150 Leute, ich finde es gut zu wissen, dann sehe ich mit, was weiss ich, mit jemandem mit dem ich wenig zu tun habe, dass er DA ist, ok? Aber die Gruppe mit wo ich drin habe, die ich kenne, das interessiert mich viel mehr, die hätte ich vielleicht lieber in einer anderen Reihenfolge gehabt. [00:17:10-0](#)

I: Aber hast du die immer aufgeklappt, diese Gruppe mit allen? [00:17:12-9](#)

B: Ja, die ist aufgeklappt. [00:17:15-5](#)

I: Und wenn du jetzt zum Beispiel deine Sametime-Kontakte vergleichst mit den Leuten die du im JIRA irgendwie siehst - entspricht sich das irgendwie? [00:17:28-5](#)

B: Ja in JIRA habe ich ja nur die Konzepter. Das ist meistens **FK_2**, manchmal **PMM_1** und ganz selten **FK_1**, und ich meine, die habe ich auch unter meinen Konzeptern stehen. Und ansonsten überschneidet sich das ja nicht weiter. [00:17:44-9](#)

I: Dann, Vergleich Ticketsystem und JIRA. Grosse Frage. Wo siehst du die Vorteile und die Nachteile bei... [00:18:02-7](#)

B: Also ich finde... [00:18:02-7](#)

I: (unterbricht) Also in Bezug darauf dass man jetzt irgendwie über den Ticketstand bescheid weiss, oder über den Task-Stand. [00:18:13-8](#)

B: Ich finde es hat beides Vor- und Nachteile. Also bei dem Ticketsystem finde ich es gut, dass man alle Tickets auf einen Schlag hat. Man kann, man hat alle auf einen Schlag, man kann nach den Namen sortieren, ich kann sie nach dem Datum sortieren, ich kann da so ein bisschen rumsuchen, und ich sehe auch, wenn ich ein bisschen blättere, wieviele Tickets bei **ENTW_DE_1** liegen, oder bei **ENTW_DE_5** liegen, oder wieviele nachgetestet sind. Also das sehe ich alles, ohne dass ich da gross suchen kann. Ich gehe da rein und habe das gleich. (...) Ich weiss, es ist irgendwie nicht so fürchterlich beliebt, aber ich finde es gut, dass da alles, dass da viel zu sehen ist. Das finde ich, weiss ich nicht, mag ich lieber. In JIRA habe ich ja nun kein Ticketsystem. Und ich könnte mir vorstellen, dass man da so ein bisschen - ich weiss nicht wie das aufgebaut werden sollte - ich weiss nur, dass ich mit den Konzepten Schwierigkeiten habe die zu finden. Und zwar einfach, wenn ich einen Task habe, dann finde ich das natürlich. Und dann, wenn es dann neue Sachen dazugibt, dann finde ich die auch. So, wenn ich jetzt aber keinen Task habe, und ich soll ein Konzept finden, dann gebe ich manchmal die Nummer ein, was weiss ich, "KO_FU irgendwas", oder "PD_KO_FU", und manchmal findet er das, und manchmal findet er das nicht. Und er findet es halt eben auch manchmal NICHT. Obwohl es diese Nummer gibt, obwohl es unter "PD_KO_FU was weiss ich, 1234" abgelegt ist, findet er es manchmal nicht. Ich weiss nicht warum, aber es ist manchmal so. Und das finde ich sehr ärgerlich. Und das könnte bei..., ich weiss nicht woran das liegt, aber so ist es halt eben. [00:19:57-0](#)

I: Und wie ist es für dich zum Beispiel - das JIRA gibt es jetzt auch noch nicht sooo lange - sind diese JIRA-Nummern die da verteilt werden, sind dir die ähnlich geläufig wie die Ticketnummern? [00:20:14-6](#)

I: Nein? [00:20:14-6](#)

B: Keine Ahnung. Also die kucke ich mir an und vergesse die wieder. Mit denen habe ich nichts zu tun. [00:20:23-4](#)

I: Aber Ticketnummern merkt man sich glaube ich schon, oder? Die werden glaube ich häufiger so ins Gespräch eingebaut, oder dass man sagt, am Telefon, "du, Ticket soundso"? [00:20:32-9](#)

B: Ich merke sie mir aber nur kurze Zeit. Und meistens schreibe ich sie mir auf. Und. Also ich merke sie mir nicht richtig. Also ich schreibe sie mir auf, dann habe ich sie irgendwo stehen, ich benutze sie, und dann sind sie wieder weg. So. Und diese JIRA-Nummern die da hinter stehen, die benutze ich ja überhaupt nicht. Also was... bringt mir nichts. [00:21:00-1](#)

I: Ok. Dann noch eine Frage zu Mylyn. Hast du das... [00:21:09-7](#)

B: Keine Ahnung was das ist. [00:21:09-7](#)

I: Das ist dieses Eclipse-Tool wo man sich die JIRA-Sachen auch anzeigen lassen kann. [00:21:14-1](#)

B: Kenne ich nicht. [00:21:16-5](#)

I: Kennst du nicht? Ok, dann hat sich das erledigt. [00:21:21-0](#)

(JIRA-Login --> Video vergessen!) [00:21:21-0](#)

I: Ok, ist das deine Standard... [00:27:20-7](#)

B: Das ist meine Standardseite, ganz genau. [00:27:23-0](#)

I: Und die hast du selber konfiguriert, oder ist das die, die der **ENTW_DE_TL** vorbereitet hat? [00:27:25-4](#)

B: Nein, das hat **ENTW_DE_TL** gemacht. Ich hatte mir mal hier Übersicht vom Link, ich weiss nicht, das ist das was immer angezeigt wird. Ich hatte mir etwas hier für **ENTW_DE_1** mal gemacht gehabt, aber das ist alt. Das gibt es gar nicht mehr. Ich hatte mir mal, also **ENTW_DE_TL** hatte das jetzt noch mal neu organisiert, und seitdem wir diese neuen (unv. Punkt) Dinger hier haben, ich hatte mir das hier mal so gemacht. Ich hatte das nicht gross sortiert, sondern ich hatte mir hier eben alles was ich hier drauf hatte. Und, das hatte ich aber noch gemacht, bevor es diese ganzen Aufteilungen gab mit "Implementierung", "Implementierung mit interner QS" und solche Sachen. Diesen ganzen gelben Punkten die es hier gibt. (...) Hm, ... wieso geht das jetzt nicht? (...) Das ist weg was wir vorhin hatten, oder? (...) Deswegen meine ich ja, also das ist ein bisschen seltsam. Vorhin hatte ich 58, ... achso das ist die gesamte Liste. Das ist weg, die äussere Seite. [00:29:02-1](#)

I: Die von **ENTW_DE_1** war jetzt weg, oder? [00:29:02-1](#)

B: Nein, ich war vorhin, vorhin war die angezeigt. Vorhin war Übersicht für **ENTW_DE_2** offen, und jetzt steht das offen, ist ja merkwürdig... . Also das ist meine normale Ansicht hier, und ich habe normalerweise auch noch irgendwie für die Testlisten-QS einen, und für Testlisten-Implementierung einen, Testnachweise stehen hier, das sind die allgemeinen Implementierungs-Sachen. Die müssen auch irgendwann geschlossen werden. Das war bei Hausrat so mein letzter Anker um in Sachen reinzukommen, jetzt hab ich glaub ich gar keinen mehr bei Hausrat. Da muss ich also doch über die Schnellsuche gehen und hoffen dass ich das finde. Oder manchmal schickt mir **FK_2**, wenn er eine Änderung gemacht hat, einen Link, dann komme ich über den Link da rein. [00:29:54-3](#)

I: Hast du zum Beispiel, ich weiss nicht, gehst du im Browser in das JIRA, oder über das Lotus Notes wo man auch einen Browser hat? [00:30:04-3](#)

B: Nein, über Lotus Notes gehe ich nicht. Also ich gehe, ja über Mozilla, oder über diesen Internet Explorer. [00:30:15-7](#)

I: Und hast du dir zum Beispiel Bookmarks gemacht, Lesezeichen? [00:30:18-1](#)

B: Ja natürlich, um auf diese Seite zu kommen. [00:30:25-2](#)

I: Ok, um auf diesen Vorgangsnavigator zu kommen. [00:30:31-0](#)

B: Ja, hierauf. [00:30:31-0](#)

I: Und das ist aber dein einziger Bookmark den du dafür gemacht hast, oder? [00:30:36-8](#)

B: Das ist der einzige. Ich hab noch einen, den alten Einstieg, den hab ich immer noch nicht weggeschmissen, aber den benutze ich nicht mehr. Aber das ist mein einziger Einstieg hier. (...) Weil ich das einfach sinnvoll sehe wie viele Aufgaben hier sind. [00:30:48-8](#)

I: Und da schaust du dir dann quasi an, das sind alles deine eigenen Aufgaben hier. Also von deinen Kollegen siehst du hier nichts. [00:30:59-3](#)

B: Genau. Das sind nur meine. [00:31:03-2](#)

I: Ok, und runterscrollen tust du hier auch nicht wirklich? [00:31:02-6](#)

B: Kommt drauf an. [00:31:09-1](#)

I: Oder von welchem Team sind diese Aufgaben? [00:31:11-6](#)

B: Das sind alles immer noch meine Aufgaben. Das sind hier, man sieht das sind hier 54, und hier sind, warum die 58 sind weiss ich nicht. [00:31:20-0](#)

I: Ok. [00:31:21-9](#)

B: Das ist merkwürdig. Eigentlich müssten das auch 54 sein. Aber vielleicht steht hier noch irgendetwas, über Hausrat oder sowas. Ja, vielleicht stimmt das hier nicht ganz überein. Kann ja sein. Keine Ahnung warum da vier mehr sind. [00:31:45-5](#)

I: Verwendest du zum Beispiel da oben diese Buttons hier zum Navigieren? Dieses Dashboard-Projekt? [00:32:01-0](#)

B: Das Dashboard? Ja, das benutze ich immer. Weil wenn man vor- und zurückgeht, dann kriegt man ja nur den Stand den es da vorher hatte. Also wenn ich es bearbeitet habe und geschlossen habe, und ich gehe zurück, dann steht es hier, also da ist es ja immer noch offen. Also muss ich über das Dashboard gehen, damit es den aktuellen Stand hat. Hat es aber auch nicht immer. Da ist es, da ist er dann manchmal ein bisschen langsam. Also wenn ich sage "ich schliesse was", oder ich sage, nein eigentlich reicht es ja hier wenn ich das schliesse, oder auch wenn ich jetzt die QS gemacht habe, und es wieder zurück an **FK_2** gegeben habe meinetwegen, und ich gehe da über Dashboard zurück, dann müsste der ja eigentlich weg sein. [00:32:39-5](#)

I: Dann ist hier noch, bei diesem zweidimensionalen Filter, der alte Stand, oder wie? [00:32:43-7](#)

B: Auch wenn ich über mein Dashboard gehe, ist hier manchmal noch der alte Stand, ja. Ich dachte, eigentlich müsste der dann ja weg sein, oder nicht? [00:32:52-5](#)

I: Eigentlich schon. [00:32:52-5](#)

B: Ist er aber nicht. Wenn ich aus JIRA rausgehe und wieder reingehe, dann ist er weg. Das ist manchmal ein bisschen lästig, weil... manchmal reicht es aber auch, zwei oder dreimal hin und her zu laufen dass dann irgendwas akutelles... [00:33:05-8](#)

I: (unterbricht) wenn man es aktualisiert oder so, macht er dann auch nichts? Mit "F5"? [00:33:15-2](#)

B: Kann man das hier mit F5 aktualisieren? Das wusste ich nicht. [00:33:16-0](#)

I: Ja, wahrscheinlich schon, oder? Dann müsste er die Anfrage nochmal senden. [00:33:29-7](#)

B: Da steht nichts. Hat er irgendetwas gemacht? [00:33:36-0](#)

I: Ich glaube schon. Nein, keine Ahnung. Und verwendest du dort oben zum Beispiel diese Bedienelemente? Dieses "Vorgänge" oder... [00:33:45-6](#)

B: Nein, noch nie benutzt. [00:33:49-7](#)

I: "Projekte"... Euer Projekt ist immer das "**LFS**"-Projekt, oder? [00:33:55-2](#)

B: Ja. [00:33:52-1](#)

I: Und diese anderen Sachen? KMP und so? [00:33:57-2](#)

B: KMP auch? [00:34:05-4](#)

I: Ist KMP nicht das alte? [00:34:05-4](#)

B: Aber da kuck ich gar nichts. Weil hier ist ja alles mögliche, aller mögliche Kleinkram. Das bringt mir ja nicht viel. [00:34:14-9](#)

I: Was, das hier zum Beispiel, "**ENTW_DE_6**"? [00:34:19-0](#)

B: Ja genau. [00:34:19-0](#)

I: "**ENTW_DE_TL_STV_1**". Das liest du dir nicht durch? [00:34:19-7](#)

B: Ja, also, das bring mir nichts. Was soll ich damit. [00:34:27-6](#)

I: Könntest du mir vielleicht noch den Task zeigen den du momentan in Arbeit hast, und wie du... [00:34:36-1](#)

B: Da wäre ich hier, dann gehe ich irgendwo rein, in der Implementierung oder irgendwie. "Interne QS", da suche

ich mal hier irgendwo... [00:34:48-6](#)

I: Die interne QS, das ist das was mit dem Fachkonzept abgestimmt wird? [00:34:53-7](#)

B: Nein, das ist... Ich suche jetzt hier "Neuanlage Glas", genau. Da muss ich jetzt erstmal wieder reingucken, weil ich muss ja jetzt von vorne wieder aufsetzen. Ich habe dazwischen ziemlich viel gemacht, aber inzwischen hat sich das Konzept auch schon wieder geändert, deswegen muss ich dann auch einfach nochmal rüberkucken. Hilft ja nichts. So. Dann gehe ich da rein und kuck mal hier drauf, dann kucke ich "Neuanlage Glas", Fachkonzept, sehe hier Version sechs, ist das neueste, dann gehe ich normalerweise... . Öffne ich das, und, ja. Sagen wir jetzt gerade fange ich mit Glas an, jetzt habe ich das geöffnet, hab es mir angekuckt (Anm.: Dokument wird geöffnet) (...) Das ist ja eben auch etwas, warum das so lange dauert. Das, bei dem Ticketsystem, wenn ich mir aussuchen könnte ob es in JIRA gemacht wird oder so wie bisher, würde ich es so wie bisher machen, weil ich bin viel schneller da dran was ich haben will. (...). [00:36:24-1](#)

I: Verwendest du zum Beispiel in Word diese Markups, oder wie weisst du welche Sachen geändert wurden? Wenn ein neues Fachkonzept kommt. [00:36:31-7](#)

B: Woran ich das sehe? Ja das ist dann meistens in einer anderen Farbe gemacht, oder die alten sind... [00:36:43-9](#)

I: Aber das kann man eben auch ein- und ausblenden, also... Also das verwendest du schon bewusst die Markup-Funktion. [00:36:43-5](#)

B: Ja. Ansonsten wäre ich ja verloren. Wie soll ich sonst feststellen, dass sich da ein Elementattribut geändert hat? Da müsste ich ja alles durchkucken. Das geht ja gar nicht. [00:37:04-2](#)

I: Das dauert ja. (Anm.: zum Laden des Dokumentes) [00:37:06-6](#)

B: Ja, das dauert allerdings. (Anm.: Dokument ist fertig geöffnet) So, dann kann ich das jetzt durchkucken, meistens kuck ich dann auch hier drauf (auf die Änderungshistorie am Beginn des Dokumentes). **FK_2** ist sehr penibel in dem was er hier ändert. Ist nicht jeder. Bei einigen stehen dann hier auch einige Sachen, und dann, ich habe es mir angewöhnt die wirklich durchzublätern. Also ich gehe nicht dahin und kucke mir das an, sondern ich blättere einfach immer durch. Weil bei einigen ist da plötzlich hier irgendwie doch was anderes eingetragen. Oder zum Beispiel, was hatte ich hier vorne... Es gibt ja diese Modellisten. Bei Hausrat standen da plötzlich zwei Zeilen hier, das stand oben gar nicht drin. Wenn ich es nicht durchgeblättert hätte, hätte ich es hier nie gefunden. Also kucke ich durch wo was, ob sich irgendwas geändert hat. Hier müsste jetzt bei Glas... das zum Beispiel, so das ist in der neuen Version. Aber wie gesagt es hat sich inzwischen, ich bin ja gar nicht auf dem aktuellen Stand. Deswegen muss ich einfach das einmal durchsteppen, durchgehen. Ich habe da Anfang Mai etwas zuletzt gemacht, ich meine das ist zu lange her. [00:38:06-8](#)

I: Da hat dann dazwischen jemand anderes an Glas weitergearbeitet? [00:38:06-8](#)

B: Nein, es hat auf Eis gelegen, da hat auch keiner was gemacht. Nur ich glaube, es sind zwei neue Versionen dazugekommen. Erst mal musste ich die eine mit der anderen nochmal wieder abgleichen, und zweitens musste ich einfach selber sehen, was ich genau gemacht habe. Ich habe zwar eigentlich aufgeschrieben, aber ich kann ja letztendlich nicht einfach so tun als wär das wirklich genau alles... Ich hab das aufgeschrieben, es wird schon richtig sein, aber ich muss jetzt trotzdem nochmal einmal rüberkucken, ob das richtig implementiert ist. Weil, weiss ich nicht, sonst hätte ich kein gutes Gefühl dabei. Dazu ist mir das zu lange her. Wenn das ein, zwei Wochen herumgelegen hätte... Aber das sind ja schon wieder gute drei Monate. [00:38:48-3](#)

I: Ja. [00:38:48-3](#)

B: So, dann mache ich das wieder zu. Und meistens, meistens war dann noch "Ziel speichern unter", dann hab ich dann noch meinen eigenen Speicher, da hab ich das dann auch noch. Da habe ich dann immer die ganzen Versionen drin stehen um nicht so lange herumsuchen zu müssen. Obwohl es vielleicht sinnvoller wäre hiermit zu arbeiten, aber du hast ja gesehen wie lange es gedauert hat um es einfach nur mal eben zu öffnen. Und ich will damit, man muss auch nochmal den Stand zwischen zwei verschiedenen Versionen vergleichen. Sagen wir, ich weiss nicht genau mit welcher ich zuletzt gearbeitet habe. Wahrscheinlich mit der vierer, mit der fünfer habe ich gar nicht gearbeitet, und mit der sechser so... Wenn ich jetzt sagen wir vor zwei Wochen zuletzt damit gearbeitet hätte, zwei Wochen Urlaub bekommen, und inzwischen sind die beiden dazugekommen. Dann hätte ich die fünfer auch genauer verglichen. So kuck ich lieber was ich gemacht habe mit der sechser, und dann reicht das. Aber wenn ich die hier alle aufmachen müsste, das hätte ja echt gedauert jetzt. Und es dauert bei mir am Rechner auch manchmal relativ lange. Dann wäre mir das zu tütelig. Also, da pflege ich das besser in meinem eigenen Ordner, und hab mir das einmal abgespeichert. Und wenn es dann was Neues gibt, dann speichere ich das bei mir ab. [00:40:05-8](#)

I: Hast du das bei dir selber... Ich weiss nicht, hast du wahrscheinlich für jedes Projekt einen Ordner? 00:40:08-0

B: Ich hab einfach für Hausrat, Glas, Unfall, Haftpflicht und Wohngebäude hab ich auch einfach einen Ordner, alles was da dazugehört schiebe ich da rein und dann habe ich da alles zusammen. (...) Das ist am Anfang recht übersichtlich, am Ende wird es weniger übersichtlich, weil da eben diese ganzen, weil dann habe ich hier, was weiss ich, bei Hausrat waren es jetzt zwanzig Versionen oder so stehen, aber es ist wenigstens alles da. 00:40:34-3

I: Aber wie jetzt eben zum Beispiel **FK_2** diese neue Version reingestellt hat ins JIRA... 00:40:42-1

B: Ja? 00:40:42-1

I: Also er hat dir vorher schon Bescheid gegeben über Email und hat dir vorher schon eine Version geschickt. 00:40:45-7

B: Also meistens, wenn ich da bin, schickt er mir per Email eine Benachrichtigung. Und wenn es losgeht dann kucke ich natürlich über JIRA. Und wenn sich da was ändert sagt er mir meistens Bescheid. 00:41:00-1

I: Und wie weisst du, dass es dann in JIRA ist? Kriegst du da so eine JIRA-Benachrichtigung, oder wird das im Meeting besprochen? Oder wie läuft das ab? 00:41:07-2

B: Mal ja mal nein. Also ich kriege immer dann und wann bei JIRA Nachrichten, ich weiss nicht genau welche das sind. Ich hab zum Beispiel ganz viele KFZ-JIRA-Nachrichten gekriegt, also hab ich überhaupt nichts damit zu tun, aber da hab ich ganz viele Emails gekriegt wenn da etwas geändert worden ist. Und ich hab mir doch grade in JIRA gekuckt, ob bei mir noch irgendetwas ist - war aber nichts. Und jetzt bei Glas, da hat er das reingestellt, da hat er mir ein Email geschrieben, aber automatisch benachrichtigt bin ich nicht worden. Also nur weil mir er Bescheid gesagt hat dass da was - nein, bei Hausrat war das, da hat er was geändert - da hat er mir bescheid gesagt, und ich bin nicht automatisch benachrichtigt worden. Also das ist, darauf kann ich mich nicht verlassen. Ich kucke auch nicht jeden Tag ob es was neues in JIRA gibt, denn das sieht man ja auch nicht so einfach. Weil wenn ich dieses Dashboard, wenn ich hier raufkucke (Anm.: zweidimensionaler Filter aus Standard-KMP-Dashboard), dann sehe ich zwar, "8, 6, 18, 8, 8, 6, 54", (...) 00:42:13-9

I: Aber das müsste man sich quasi im Kopf behalten was der alte Stand war. 00:42:17-0

B: Ja. Ich meine, woher weiss ich... Hieran sehe ich ja nicht ob etwas Neues gekommen ist. Eigentlich wäre es ganz schön, wenn hier einfach eine Spalte wäre mit Datüchern, einfach in denen neuester Eintrag und ein Datum drinstehen würde. Also so sehe ich ja nicht, ob sich da etwas getan hat. Und wenn ich jetzt nächste Woche kucke wo die Zahlen... Auch selbst wenn ich sie mir aufschreibe, sehe ich das doch nicht. Weil, selbst wenn die Zahlen alle gleich sind, wenn hier jetzt irgendwie... . Wenn **ENTW_DE_1** auf die Idee kommt "ach das mach ich eben mit", dann nimmt er mir hier einen weg, und **FK_2** schiebt mir einen wieder zu, dann hat sich doch da nichts geändert. Und wenn ich nicht automatisch benachrichtigt werde, dann kriege ich das nicht mit, und hier kriege ich das auch nicht mit. Und hier kriege ich das nur mit, wenn ich da zufällig mal reinkucke. Und genau nachkucke, ob da was Neues ist, oder was neues sein könnte. Aber das sieht man ja auch nicht. Auch nicht wenn da hier jetzt reingehe, so, dann seh ich da auch nicht, ob da was Neues ist. Woran soll man das sehen? Ich sehe hier "aktualisiert", ok, ich kann überall durchkucken... 00:43:20-6

I: Das ist überall dasselbe Datum... 00:43:20-6

B: Ja. Ich meine, daran kann ich kucken ob da was aktualisiert wurde, aber das Datum, das ist ja auch falsch. (...) Ok, 26... 00:43:29-8

I: Und was war das vorhin, das wollte ich wissen, "Neuanlage Glas, Produkt-Verfahren". 00:43:34-5

B: Genau 00:43:35-8

I: Das hatten wir uns vorhin angeschaut? 00:43:38-0

B: War das jetzt wirklich vom 26.? 00:43:40-8

I: Achso, nein warte mal, das war hier der 00:43:43-6

B: Wenn ich da jetzt reingehe, da steht 25., 26., ... 00:43:44-9

I: Achso, und das ist das übergeordnete Ding wo du dann das Fachkonzept... 00:43:55-4

B: Das war das falsche, das ist jetzt für die QS. (...) Da. 18.8.. (...) Aber bei den anderen steht 26.. (...) Quatsch. Da steht 18.8., so, bei dem anderen stand aber immer 26.5.. Woher soll ich denn jetzt wissen, dass am 18.8. sich etwas geändert hat? Woher soll ich das wissen, geht doch gar nicht. Wie gesagt, ich kriege nicht immer eine JIRA-Benachrichtigung. [00:44:34-0](#)

I: Ok. Also du bekommst eine Email-Benachrichtigung vom Fachkonzepter, und auf das hin schaust du dann ins JIRA, und suchst, navigierst dorthin in dieses "PD_KO_FU soundso"... [00:44:43-2](#)

B: Manchmal kriege ich eine Email mit einer Benachrichtigung, manchmal sagt er mir das, manchmal finde ich das zufällig, weil ich da reinkucke. [00:44:52-6](#)

I: Und wie ist das dann zum Beispiel mit den Tasks, ich meine, wenn sich da irgendetwas geändert hat, dann muss das auch implementiert werden. Wer stellt diese Tasks rein, das macht dann **ENW_DE_TL** wahrscheinlich? [00:45:01-5](#)

B: Ja. [00:45:03-9](#)

I: Oder du mit ihm gemeinsam? [00:45:05-9](#)

B: Nein, die macht er meistens. Also diese... [00:45:11-8](#)

I: Wie weiss **ENW_DE_TL** was geändert werden muss? Oder sagt er dann einfach nur generell, keine Ahnung, "Anpassung Version soundso"? [00:45:17-5](#)

B: Dass hier eine neue Version kommt? Ich weiss jetzt nicht genau wer was jetzt macht. Ich weiss, wenn **FK_2** hier was reinstellt, das geht ja nicht über **ENW_DE_TL**. Das stellt er rein, und wenn es dringend ist sagt er mir Bescheid. Hier hat er mir ja nicht immer Bescheid gesagt, das ist ja jetzt nur weil... Also hier da weiss ich nicht ob er Bescheid gesagt hat. Bei Hausrat hat er mir Bescheid gesagt. Weil es eine Änderung gab, ich glaube Donnerstag gab es eine Änderung, oder nein, Freitag gab es eine Änderung, und gestern (Dienstag) gab es schon wieder eine Änderung. So, da hat er mir natürlich Bescheid gesagt, weil dann geht es natürlich schnell. Wenn er mir das sagt, dann kucke ich mir das an, und dann implementiere ich das, und ich habe das von Freitag am Freitag implementiert und Montag eingchecked. Dann hat er mir gestern die Änderung gesagt, und das habe ich gestern eingchecked. Und heute habe ich das Konzept erst gesehen. Aber [00:46:12-8](#)

I: Aber wird dann schon noch ein eigener Task erstellt jetzt, zum Beispiel "Änderung von soundso implementieren", oder macht man das dann alles ohne Task einfach? [00:46:22-4](#)

B: Ich habe, also ich schreibe das ja in Lotus Notes unter PAZ. Und da schreibe ich ja rein, da habe ich eine Task "Hausrat Nacharbeiten" zum Beispiel. Da gebe ich die Sachen an, wo ich kein Ticket für habe, und ansonsten hat man ja hier eine Ticketnummer und gebe das Ticket an nachdem ich es bearbeitet habe. Aber ich nehme hier nicht... hier brauchst du irgendwas, und sage dazu dass... ich gebe es ja nicht DARUNTER irgendwie ein. (...). Ich finde es irgendwie schwierig mit dem JIRA, also irgendwie hört sich das gut an, man kommt überall ran - theoretisch - aber wie gesagt, ich kriege es nicht mit wenn sich etwas geändert hat, weil ich nicht immer eine automatische Benachrichtigung kriege. Selbst wenn ich regelmässig reinsehe, kriege ich das nicht, und theoretisch könnte **FK_2** hier jetzt locker, was weiss ich, mehrere Versionen reinstellen von denen ich nur etwas mitkriege, wenn ich da mal wieder reinkucke. Und damit zu arbeiten ist insofern umständlich, da es ziemlich lange dauert das zu finden. [00:47:34-7](#)

I: Aber verwendest du zum Beispiel - kannst du mal reingehen in diese Fachkonzept... das hier - da gibt es ja auch diese "Überwachen-Funktion" wo du dich quasi in eine Liste eintragen kannst, und wenn sich da irgendetwas ändert, dann sollte du eigentlich eine Email oder irgendetwas kriegen. [00:47:56-1](#)

B: Das habe ich noch gar nicht benutzt, weil ich noch gar nicht genau weiss wie das funktioniert. Bedeutet das "wenn sich auf dieser Seite etwas ändert", oder bedeutet das wenn... [00:48:08-2](#)

I: Ich glaube für diesen Task, wenn sich hier etwas ändert. Dann kriegst du eine Email-Benachrichtigung, oder sollte man kriegen. [00:48:14-7](#)

B: Aha, ok. Dann kann man das ja auch mal angeben (klickt). Wäre ja sinnvoll. [00:48:24-5](#)

I: Ich habe zum Beispiel auch schon gehört von Kollegen dass sie von JIRA so viele automatische Emails bekommen, dass sie das sowieso gleich herausfiltern. [00:48:34-6](#)

B: Ja, das war bei KFZ. Ja, aber ich meine ich bräuchte die, wo sich hier die Konzepte ändern. Und das

funktioniert jetzt, wenn man jetzt da... wenn da jetzt "watching" steht, dann kriege ich... und hier kommt ein neues Konzept, dann sollte ich eine neue Benachrichtigung kriegen, oder wie? [00:48:54-3](#)

I: Ja. [00:48:59-8](#)

B: Aber das bedeutet ja auch, dass ich überall wo ich, ja wo Konzepte sind,... Hier ist jetzt "Neuanlage" so. Wenn ich jetzt wieder zurückgehe, ... Neuanlage..., wenn sich was ändert, dann kriege ich eine Benachrichtigung. So, hier ist also alles nur "Neuanlage", das heisst, ich müsste in jeden einzelnen Vorgang reingehen, bevor ich überhaupt etwas mache (sichtbar ist eine Liste mit vielen Tasks zu verschiedenen Fachkonzepten). [00:49:37-1](#)

I: In "Änderung Glas" und "Storno" und... [00:49:38-3](#)

B: Das ist ein irrer Aufwand, was soll das? Wenn ich jetzt hier reingehe, dann müsste ich in jede dieser 18 Stellen reingehen, an die Stelle gehen, Fachkonzept und (unv.) eingeben, und dann müsste ich das auch noch hier machen - für Implementierung müsste ich das machen - und für alle diese Dinge. Für diese 54 Dinge müsste ich das überall eingeben. Das sind 54 mal, da bin ich ja mindestens zwei Stunden für beschäftigt. Das ist doch irre, was soll das? Wenn ich genau hierfür eine Benachrichtigung kriege wenn sich da etwas geändert hat, wenn ich HIER (Anm.: im 2-D Filter) jetzt "watching" eingeben würde, einmal für diese ganzen, das fände ich sinnvoll. Aber was ich da eben gemacht habe, das finde ich totalen Quatsch. Weil, ich meine was bringt mir das wenn ich dafür benachrichtigt werde? Ok, dann weiss ich an der Stelle hat sich etwas geändert, aber ich weiss nicht, ob ich hier eine neue QS für eine Testliste machen soll. Und das sehe ich hierdran auch nicht, und wenn ich hier reingehe sehe ich das auch nicht. Wie gesagt, das wäre so einfach, wenn man hier jetzt einfach ein Datum stehen hätte, "aktualisiert" oder was weiss ich, und da steht dann, oder "zuletzt aktualisiert", oder "zuletzt bearbeitet" irgendwas in der Art wenn das letzte mal etwas gemacht worden ist. Und da steht dann "23.8." dann kuck ich mal "ok, da hat sich ja was getan" oder selbst wenn sich da nichts geändert hat und ich sehe da diese Woche ist da etwas passiert, dann kucke ich da rein. Aber so... Oder am besten wäre "alter Stand - neuer Stand", wenn man da zwei Daten hat. Einmal wann es das letzte mal gemacht wurde, und einmal wann es das vorletzte Mal gemacht wurde. Dann sieht man ja, ob sich da etwas getan hat. [00:51:19-6](#)

I: Und wie..., das sind alles deine Tasks, oder? [00:51:25-6](#)

B: Ja, das hier sind die Obertasks, und das sind die einzelnen Sachen. Und "mit interner QS" bedeutet, wenn ich das eingegeben habe, dann gebe ich das an **ENW_DE_TL_STV_1** weiter eigentlich, und der kuckt dann nochmal rüber.. [00:51:37-8](#)

I: Ein Codereview, oder? [00:51:41-7](#)

B: Ja, genau. Ob es in Ordnung ist. Ob er es dann macht oder nicht weiss ich nicht, aber auf jeden Fall hat er die Möglichkeit es zu machen. Und das war es dann. Und er hat hier irgendwo die allgemeinen Aufgaben stehen. [00:51:54-7](#)

I: Und eben, diese Ansicht jetzt, das verwendest du dann auch, damit du weisst was du als nächstes machen musst? [00:52:02-4](#)

B: Naja, da hat man einfach, das ist ein bisschen ein Überblick einfach. Ein bisschen Überblick, und man hat auch die Möglichkeit, an die Konzepte ranzukommen. Aber ich meine, hier steht "aktualisiert Neuanlage Glas - **Produkt**-Verfahren", aktualisiert... hier steht der 17.5., vorhin hatten wir aber den 26.5, und der letzte Eintrag war der 18.8.. Warum steht hier nicht der 18.8.? Also hier könnte das doch zumindest drinstehen. Also es bring mit nicht viel, ich muss echt immer wieder mal durchsuchen. [00:52:35-0](#)

I: Aber das machst du dann eh quasi, dass du immer mal durchsiehst? [00:52:40-7](#)

B: Nein, das mache ich nicht. Ich erledige die Aufgaben die ich zu tun habe. Wenn dann eine QS ist die dringend ist - ich habe jetzt gerade glaube ich drei oder vier QSen, vier habe ich gemacht für Glas, zwei Konzepte und zwei Testlisten - die habe ich jetzt gemacht, obwohl da auch irgendwie "31.5." stand, aber das stand einfach nicht an, und Hausrat war wichtiger. Wenn da jetzt eine QS bei Hausrat gewesen wäre, also da kucke ich dann schon rein. Und das hätte ich gemacht. Aber ich durchsuche... wenn ich mit Glas nichts zu tun habe, dann kucke ich auch nicht nach ob sich da etwas geändert hat. Ich kucke erst wieder, wenn ich das bearbeite. [00:53:19-2](#)

I: Und wie ist das dann zum Beispiel mit diesen Fälligkeitsdatütern? Das setzt der **ENW_DE_TL**, oder wie wird das gehandelt? [00:53:29-6](#)

B: Ich weiss nicht genau. Ich glaube **FK_TL** teilweise, teilweise **ENW_DE_TL**. **Produkt**-Glas, da steht hier der 20.7., das war der ursprüngliche Termin, das ist auch nicht angefasst worden, das ist ja verschoben worden. So und normalerweise hätte man dann schon versucht das fertig zu machen, nur Hausrat waren ja doch ein paar

mehr Änderungen. Und es macht einfach keinen Sinn das alles so zu durchsuchen. Also ich finde das hier echt ärgerlich. Warum aktualisiert? Es stimmt wenn diese Seite hier aktualisiert würde, wann wird das überhaupt gesetzt? Ich meine, warum wird hier nicht ein Datum reingesetzt wenn sich in dem Bereich irgendetwas geändert hat? (...) Das finde ich unschön. (...) Man kommt immerhin an die meisten Sachen ran, (...) aber ich finde es... diese Schnellsuche: manchmal, was weiss ich, wenn ich jetzt "PD_KO_FU 2605" - ist glaube ich Hausrat... (Anm.: tippt am Computer) ich glaube es geht, ich glaube er findet das, aber wenn ich jetzt diese Nummer nicht weiss, sondern ich weiss nur es gibt jetzt ein neues Produkt, was weiss ich, "Tischeschrubben", dann kann ich ja nicht "Tischeschrubben" eingeben, sondern ich bräuchte ja die PD_KO_FU, die Nummer dafür. Und wenn ich die nicht weiss, dann finde ich das nicht. [00:55:05-9](#)

I: Also wenn du da oben jetzt eingibts "Glas Storno", dann findet er nichts? [00:55:08-0](#)

B: Doch, vielleicht findet er etwas. Ich weiss ja nicht genau wie er sucht. Vielleicht... Wir können es ja ausprobieren. Also einige Sachen findet er, einige nicht. Wenn ich jetzt "Glas Storno" weiss ich nicht... (tippt). So. (Eine lange Liste mit Tasks erscheint) [00:55:48-6](#)

I: Und ist das das was du gesucht hast? [00:55:48-0](#)

B: Was? Auf jeden Fall ist das alles hier mit "Glas Storno", ok, also hat er das auch gefunden. [00:55:58-8](#)

I: Aber ist da jetzt auch zum Beispiel der Task dabei der dir zugewiesen wäre? Oder wer ist zum Beispiel die **SONSTIGE_3**? [00:56:03-1](#)

B: Die arbeitet im... [00:56:08-3](#)

I: Fachkonzept? [00:56:10-6](#)

B: Also ich weiss nicht wie der sucht. Ich weiss nur, dass ich manchmal wirklich auch die Nummer eingegeben habe. Verstehe ich jetzt auch nicht. (...) Wenn ich jetzt "Glas" - findet er das so? Ja, findet er auch. Also manchmal findet er es gut, aber ich habe manchmal auch schon die richtige PD_KO_FU-Nummer eingegeben, hab sie eingegeben, habe nichts gefunden, habe mir das Konzept rausgesucht, habe nochmal genau die Nummer eingegeben - weil ich dachte ich habe mich irgendwie vertippt - und nicht gefunden. Und das, ich weiss es nicht genau, manchmal funktioniert es, manchmal aber auch nicht. Und ich habe auch schon jemand anders gefragt, dass er das eingibt, und er hat es auch nicht gefunden. Also... weil manchmal wenn man das ein paarmal macht und er findet nichts, dann denkt man das sei seine eigene Dummheit, aber das stimmt nicht. Manchmal findet er das und manchmal nicht. Und woran das liegt weiss ich nicht. [00:57:26-3](#)

I: Ja, das habe ich von anderen auch schon gehört. Also ich habe es ja selber schon erlebt. [00:57:31-3](#)

B: Gut. Also. Weil das finde ich ärgerlich. Deswegen fände ich das mit den Tickets - mir ist das lieber in Lotus Notes. Also es hat Vor- und Nachteile. Was ich zum Beispiel auch richtig unschön finde, ist (sucht in JIRA)... ich weiss nicht wo man jetzt die Seite findet wo man die QSen macht - ja genau, das hier finde ich auch sehr unschön. Das hier ist ja nicht schlecht, weil man gibt hier die Person ein, Dokument Version sechs, also für die ist die letzte QS fällig gewesen, hier ist die QS auch eingetragen. Das war die letzte Version für die das gemacht werden musste, für die ist das gemacht worden, hier steht es drin. Das ist ja in Ordnung. Was ich richtig unschön finde, wenn ich jetzt auf "Bearbeiten" gehe... geht das hier überhaupt? Ich hab gar keine Aufgabe dafür, deswegen geht das gar nicht... [00:59:29-7](#)

I: Ich glaube das kann man nicht mehr machen, weil man die QS... [00:59:33-4](#)

B: Ja, das geht gar nicht mehr zu ändern. So, was ich richtig unschön finde ist - wenn ich jetzt auf "bearbeiten" gehe, dann kriege ich ein Feld was ungefähr so breit ist von der Linie bis zu der. Und da stehen diese ganzen Sachen drin. [00:59:46-9](#)

I: Diese rosaroten... [00:59:49-1](#)

B: Nein, ab hier. Ab hier "Übersicht" bis da. So, und das Feld kann ich nicht vergrössern, das bleibt immer so. Das heisst, wenn ich hier jetzt.. [01:00:02-7](#)

I: ... mehr zu schreiben hast [01:00:02-7](#)

B: Also es wird sehr unübersichtlich. Hier ist jetzt immer "keine Fragen", wenn hier jetzt ein bisschen mehr steht - man weiss gar nicht genau wo man sich da befindet. Dann muss ich das ganze... Da gehe ich ganz zum letzten, ich gehe ganz zum Schluss runter, das letzte Teil hier, kopiere ich, setze es ans Ende, gehe wieder hoch, und wenn ich ein bisschen hochgehe, und das ist, da steht schon viel drin, ich gehe ein bisschen hoch, und dann

lande ich da mittendrin. So jetzt bin ich da am suchen, und habe nur dieses ganz kleine Fenster um das einzugeben. Ist ja kein Problem dass da ein Fenster ist, aber man könnte das ein bisschen verbreitern, und dann könnte man damit arbeiten. Aber das ist echt winzig klein, und man geht da an der Leiste so ein bisschen lang, und man ist schon wieder ganz wo anders... 01:01:13-6

(Unterbrechung des Interviews weil Besprechungsraum von anderen benötigt wird) 00:00:00-0

I: Also es ist unübersichtlich, und schwer zu navigieren. 00:00:23-2

B: Es ist nicht unbedingt schwierig, aber ich finde es nicht gut, dass man Änderungen so schlecht nach..., mitkriegt. Ich meine, ich müsste ja wirklich diese ganzen 18 - also bei der Implementierung und interner QS, weil das wäre ja das wichtigste vielleicht - wenn da jetzt was neues aufgemacht wird, sagen wir eine neue Testliste zu QSen, und ich habe wieder welche neue, dann ist ja da wieder das neue Feld, und dann finde ich das ja auch. Aber dann sieht man diese 18 Sachen da mit Implementierung und interner QS, wenn sich das was ändert, kriege ich das nur mit, wenn ich da entweder benachrichtigt werde, was nicht unbedingt automatisch geschieht, also wenn ich irgendwie Bescheid gesagt kriege, oder wenn ich alle 18 durchgehe. Und ich meine das mache ich natürlich nicht. Wenn sich da jetzt, sagen wir mal "technische Änderungen" habe ich noch nicht angefangen bei Glas, wenn sich da was ändert suche ich da jetzt nicht nach, sondern, ich würde da erst kucken wenn ich soweit bin das zu implementieren. Es sei denn, ich habe zwischedurch eine QS zu tun und kriege dafür einen Task. Aber wei gesagt, das mitkriegen dass sich da was geändert hat, ist nicht so einfach. Das wäre einfacher hinzukriegen. 00:01:41-3

I: Abschliessend eine Frage: schaust du dir zum Beispiel im JIRA an, welche Tasks beispielsweise **ENTW_DE_1** zugewiesen hat? 00:01:49-3

B: Das hatte ich mal gemacht, also bevor das jetzt mit dem neuen JIRA ist. Mit dem alten JIRA hatte ich mir ja so eine Suche gestellt weil ich teilweise dann einfach Sachen von **ENTW_DE_1** übernommen habe, und dann die gemacht habe. Aber im Moment habe ich genug zu tun, und deswegen mache ich das nicht. 00:02:13-2

I: Dann dankeschön.

Interview 04_TST_TL

I: Also sie sind Teamleiter von Team Qualitätsmanagement und Test? 00:00:21-4

B: Die Information ist nicht ganz richtig, ich betreue die Abteilung Prozess- und Qualitätsmanagement. Damals war ich Teamleiter Softwaretest und Qualität. Das hat sich ein bisschen verändert, unsere Abteilung hat sich vergrößert, und wir betreuen die gesamten Prozesse im Unternehmen. Und das Qualitätsmanagement besteht auch darin, dass man das Testmanagement aufzusetzen, vernünftig aufzusetzen. 00:00:54-7

I: Und was sind jetzt zum Beispiel ihre Aufgaben, was machen sie so jeden Tag? 00:00:56-9

B: Jeden Tag, also es ist so dass ich mich im Moment jeden Tag mit Aufgaben in Richtung Vorstand beschäftige, da wir die Organisation ein bisschen umbauen, optimieren wollen. Die Themen Prozesse sind da sehr wichtig, die haben eine grössere Betrachtung, weil, angefangen von der Anforderungserhebung bis hin zu Softwarelieferungen an die jeweiligen Kunden, müssen wir intern alle Prozesse optimieren. Und anpassen an die Realität der Aussenwelt. Im Bereich Test haben wir momentan auch ein paar Defizite, die Gründe sind eher darin zu finden, dass die Anforderungen momentan nicht so spezifiziert sind, dass man eine volle Abdeckung leisten kann, und von daher ist das auch Teil der Prozessdefinition, Prozessoptimierung und so weiter. 00:02:13-3

I: Mit welchen Leuten arbeiten sie jetzt täglich zusammen? 00:02:17-6

B: Wir haben drei Schnittstellen die als Input dienen für unsere Arbeit: die erste Schnittstelle ist die Fachkonzeption, also die konzipieren. Sie können sich das so vorstellen, ein Kunde hat eine Idee, einen Wunsch, manchmal einen Auftrag, oder eine Vermutung. Darüber wird diskutiert, erläutert, und zu einem späteren Zeitpunkt wird beauftragt wenn die Idee konkretisiert ist. Und dann werden die Konzepte erstellt. Und das ist dann die Grundlage für unsere Arbeit. Anhand dieser Konzepte definieren wir Testgegenstände, beziehungsweise Testobjekte, und zu einem späteren Zeitpunkt werden diese Testobjekte durch Testspezifikationen versehen, für die Überprüfung der Abdeckung zu einem späteren Zeitpunkt. Das ist Schnittstelle zur Fachkonzeption, und da ist der Austausch intensiv, weil die Konzepte werden von uns reviewt, "geQSt", da entsteht dann die Synergie, und die Leute kommen zusammen und besprechen die Themen, optimieren die Themen, und die werden dann vom Kunden abgenommen. 00:03:38-5

B: In der Entwicklung besteht die Zusammenarbeit darin, dass wenn Teilaufgaben abgeschlossen sind, dann müssen diese Teilaufgaben anhand von von uns erstellten Protokollen - in diesem Fall Testnachweise - dann werden die Testnachweise als Abnahmeprotokoll bei uns hinterlegt. Sprich, die sind im Vorfeld von dem jeweiligen Entwickler befüllt worden, diese Szenarien sind beschrieben - die Bedingungen, die Testumgebungen die dafür vorgesehen waren, und so weiter - sodass wir unter gleichen Bedingungen das ganze nochmal nachspielen können. Und zertifizieren können dass die Arbeit abgeschlossen ist. Erst dann können die Entwickler nach Hause gehen, also die Arbeit komplett abgeschlossen betrachten. 00:04:31-6

I: Und wie bleiben sie jetzt zum Beispiel für ihr Team am laufenden wie das Projekt abläuft, oder wie der Projektstatus ist? 00:04:40-1

B: Ja, das wird bei uns mit **Testtool xxx**, geregelt. Das ist ein Tool, was eine Anbindung hat zu gängigen Anforderungsmanagementsystemen, genauso wie eine Anbindung zur Entwicklung. Das heisst, von diesem Tool aus können wir auch die Teilkomponenten, oder auch Funktionalitäten, die in der Entwicklung entwickelt werden überwachen, laufen lassen, Metriken erstellen, und so weiter und sofort, und die Hauptarbeit besteht eigentlich darin, den gesamten Testprozess zu begleiten. Angefangen bei den Anforderungen, die Testanforderungen werden abgelegt, dann werden die Testspezifikationen definiert, dann werden die Testausführungen durchgeführt, Berichte werden erstellt, und so weiter. Das heisst, wenn die Grundarbeit im Tool abgelegt ist, dann braucht man kein zusätzliches Tool um zu arbeiten. 00:05:41-6

I: Das ist alles innerhalb von dem **Testtool xxx** abgebildet, oder wie? 00:05:46-1

B: Genau. Sogar die Automatisierung wird auch damit durchgeführt. 00:05:53-7

I: Ok. Und diese einzelnen Arbeitsschritte die im **Testtool xxx** passieren das wird aber.. Also das Fachkonzept, wie kriegen sie zum Beispiel das Fachkonzept? 00:05:58-8

B: Das wird importiert. Wir haben eine Import-Schnittstelle. Dokumente in Excel-Form oder in Word werden von dieser Schnittstelle verarbeitet, und von daher können wir manuell, extern ausserhalb des Tools bestimmte QSen durchführen, und so weiter, aber wenn diese Arbeit abgeschlossen ist, dann wird das Gesamte in das Tool importiert, und von da an wird nur im Tool gearbeitet. Bis zum Berichtswesen. 00:06:37-3

I: Und wenn sich zum Beispiel.. Im Fachkonzept kommt es immer wieder vor, dass es das Änderungen gibt, oder dass zum Beispiel der Kunde sagt da gehört noch eine Änderung gemacht. 00:06:45-0

B: Genau, dann wird das auch direkt im Tool angepasst. Die nachträglichen Anpassungen werden im Tool durchgeführt. Es ist grundsätzlich so, dass für die Fachkonzeptabteilung, oder die Fachkonzeption und die Entwicklung alle Veränderungen die in (unv. Markern) durchgeführt werden, werden in JIRA dokumentiert. Und so kriegen wir auch die Information, und können auch nachträglich diese Anpassungen im Tool durchziehen. 00:07:20-9

I: Und nutzen sie da zum Beispiel - in JIRA gibt es ja so automatische Email-Benachrichtigungen und so. Verwenden sie da eher diese Funktion, oder läuft das eher so, dass man rübergeht zur Fachkonzeption, und dort nachfragt "gibt es was neues" oder kommen die Fachkonzeptler herüber, oder schreiben die ein Email oder rufen an? 00:07:37-2

B: Ich bevorzuge grundsätzlich den direkten Austausch mit den Kollegen. Natürlich kriegen wir die Emails über das JIRA-Tool. Wenn irgendetwas neu zugewiesen worden ist, oder verändert worden ist. Oder bestimmte Sachen geändert worden ist, oder eine neue Version eines Dokumentes erstellt worden ist, dann werden wir automatisch vom Tool informiert. Das ist schon sehr nützlich. Aber nichts desto trotz: der Weg zu dem direkten Kollegen wird bevorzugt, wenn irgendetwas anders sein sollte, oder eine neue, nachträglich dokumentierte Anforderung im Tool erfasst worden ist, dann ist das für mich immer so ein Zeichen ein Gespräch zu suchen. Das reicht mir nicht im Tool zu lesen "Ok, Version drei ist erstellt worden", mir ist es wichtig zu wissen die Hintergründe, können wir das im Prozess abfangen, ist das notwendig gewesen, kann man das nicht anders priorisieren dass das vielleicht der Entwicklungsprozess dadurch nicht gestört wird, und, und, und. Weil das Problem dass wir grundsätzlich haben ist, dass die Liefertermine fest definiert sind. Und die können wir nicht verschieben. Von daher: alles was mittendrin im Prozess anders sein soll ist immer ein Riesenproblem welches wir abfangen müssen. Und deshalb immer der direkte Kontakt. 00:09:19-9

I: Und wie passiert das zum Beispiel mit dem Kunden? Also es kommen ja immer, von **Kunde_1**, kommen ja öfter Wünsche herein. Wer ist jetzt von Seiten von **Kunde_1**, kommen die dann direkt zu ihnen, beziehungsweise zum Testteam? Oder gehen die direkt ans Fachkonzept? Ich glaube das Testteam ist ja der erste Ansprechpartner. 00:09:42-5

B: Ich bin eigentlich die grosse Schnittstelle zum Kunden hin. Wenn irgendwas nicht funktioniert, oder wenn die einen Wunsch haben, wenn die bestimmte Sachen anders haben wollen bin ich der direkte Ansprechpartner dafür. Sehr oft in der Vergangenheit war das so, dass auch Teile der Entwicklung direkt angesprochen worden ist, oder Teile der Fachkonzeption oder des Produktmanagements angesprochen worden ist. Wir sind dabei diese Wege sauber zu definieren, für den Kunden direkte Ansprechpartner zu benennen. Und deshalb habe ich am Eingang gesagt dass wir dabei sind vieles zu optimieren. Prozessbezogene Sachen, die Abläufe, das muss nochmal neu definiert werden, neu abgestimmt werden, und kommuniziert werden. Weil wie es ist momentan ist es nicht optimal. 00:10:40-5

I: Und zum Beispiel die Tickets die von den **Kunde_1** erstellt werden. Die haben ja dort ein eigenes Ticketsystem, das wird dann von der **TST_4** glaube ich ... 00:10:51-5

B: Genau, erfasst... 00:10:51-3

I: ... bei uns reingestellt. 00:10:58-6

B: Genau. Gut, die arbeiten mit dem System in **Süddeutschland**, und die Tickets werden da erst mal "ge-QSt" von **TST_4**, und dann erfasst im System, und dann die werden in das **VersicherungsGmbH**-Verwaltungssystem integriert. Dann kriegen die automatisch eine ganz andere Nummer, eine interne Nummer, und damit arbeiten wir. 00:11:22-9

I: Und wo sitzt zum Beispiel **TST_4**? 00:11:26-1

B: Die sitzt direkt beim Kunden. Und die hat auch eine Dispatching-Rolle. Das heisst, es geht nicht nur um Ticketerfassung, es geht auch darum erst mal zu kucken "sind diese Tickets berechtigt", sind die so beschrieben dass man die versteht? Weil in der Vergangenheit hatten wir oft das Problem, dass der Kunde ein Ticket erfasst hat, und gesagt hat "funktioniert nicht". Was, in welchem Zusammenhang, mit welchen Testdaten, in welchem Geschäftsprozess, das wurde nicht beschrieben. Dann hat die Entwicklung teilweise Tage gebraucht um das Problem zu verstehen, zumindest um zu verstehen wo ein Problem entstehen könnte, und das ist sehr mühselig. Von daher ist die Arbeit die sie bei **KUNDE_1** durchführt extrem wichtig für die weitere, für die Abarbeitung der Tickets bei **VersicherungsGmbH**. 00:12:29-6

I: Wie intensiv ist da der Kontakt zwischen ihnen, beziehungsweise **TST_4** oder dem Testteam und dem

Testteam von **KUNDE_1**? 00:12:42-4

B: Sie hat direkten Kontakt zum Qualitätsmanagement bei **Kunde_1**, und das Qualitätsmanagement bei **Kunde_1** hat direkten Kontakt zu mir. Das heisst, die Vorgehensweise, die Abläufe die ich definiere, erstelle, meistens gehe ich den Weg nach **Süddeutschland** und präsentiere die. Wie wir intern arbeiten, welchen Einfluss das auf die Arbeit von **Kunde_1** haben kann, oder haben muss, und dann wird gegebenenfalls auch Anpassungen vorgenommen bei **Kunde_1**. Das heisst, die Zusammenarbeit ist extrem eng. 00:13:18-8

I: Und wie oft fahren sie da hin zum Beispiel? 00:13:18-8

B: Im Schnitt pro Quartal ein bis zwei Mal. 00:13:26-5

I: Dann noch eine Frage zu ihrem Team: wie verschaffen sie sich einen Überblick, zum Beispiel über die Auslastung ihrer Mitarbeiter? Oder über eventuelle Probleme die ein Mitarbeiter hat? 00:13:41-6

B: Ja gut, eine Möglichkeit, eine grundsätzliche Möglichkeit ist die zugewiesenen Tasks in JIRA zu überwachen und zu kucken wo er steht, wer hat wieviele Tasks, wie umfangreich sind die, uns so weiter. Aber unabhängig davon, in dem Bereich den ich betreue, haben wir viele andere Aufgaben die nicht in JIRA abgebildet werden, oder abgebildet werden können. Und dann habe ich eigene Mittel dafür aufgebaut aus meiner langjährigen Berufserfahrung bei Prozess- und Qualitätsmanagement. CMMI-Themen habe ich bei grossen Unternehmen eingeführt, und von daher irgendwann im Laufe des Berufslebens hat man seine eigene Methodik entwickelt, und das ist immer hilfreich. Also im Excel habe ich auch eine Art Auslastungs-Matrix für meine Mitarbeiter, und im Rahmen von Wochenmeetings kann ich immer überwachen wo wir stehen, wer was macht, wo die Probleme sind, die werden da besprochen. Und meine Datei auch entsprechend angepasst. 00:14:57-5

I: Was sind das zum Beispiel für Probleme die man nicht im JIRA erfassen kann oder will? 00:15:07-1

B: Das sind praktisch Tätigkeiten, also zum Beispiel dass man einen Systemtest plant, oder einen Lasttest plant, oder Integrationstest oder Auslieferungstest. Oder dass man ein Fachkonzept dabei ist zu erstellen oder zu reviewen, das sind Tasks die nicht in JIRA erscheinen. Die Tasks die wir in JIRA abwickeln sind praktisch die Tasks zu den beiden Schnittstellen die ich am Anfang erklärt habe: Schnittstelle zur Fachkonzeption und zur Entwicklung. Das sind die Sachen die wir in JIRA bearbeiten. Aber alles andere ist ausserhalb des Tools. 00:15:48-7

I: Und das JIRA selber sagen sie verwenden sie selten? 00:16:04-7

B: Ja genau, selten. Also ich würde sagen für mich persönlich, dass ich praktisch nur Experten bei mir beschäftige, also das heisst Leute die eine langjährige Berufserfahrung haben und selbstständig arbeiten können und so weiter. Ich mache Vorgaben, ich gebe die Strategie vor, und die können schon selbstständig die Arbeit überwachen, und deshalb kucke ich ein bis zwei Mal die Woche in JIRA wie der Status ist. Und sehr oft ist das eigentlich unnötig. Weil die Sachen werden so schnell erledigt, und so gut umgesetzt, dass ich nur im Rahmen von vorgesehenen Meetings das noch einmal abfrage, und dann entsteht normalerweise ein Austausch oder eine Diskussion zu den einzelnen Themen. So für mich persönlich ist keine so grosse Notwendigkeit da. Ausser mir wird in JIRA irgendetwas zugewiesen, dann muss ich das auch bearbeiten, weil die Betreuung des TTS-System - also unser Fehlerverfolgungssystem - Probleme die da entstehen werden auch in JIRA erfasst, und die werden mir in JIRA zugewiesen. Weil ich bin für die Tools im Unternehmen auch zuständig. Und das heisst, wenn irgendwo Probleme entstehen mit (unv. Stelich), mit JIRA selbst und so weiter, das ist auch meine Hoheit, ich muss da eine Entscheidung gegebenenfalls treffen. 00:17:49-6

I: Genau, das wäre auch noch eine Frage die ich noch hätte: wie sehen sie das JIRA im Vergleich zu dem Ticketsystem, zu dem Haus-internen das es in Lotus Notes gibt? 00:17:57-5

B: Dieses Haus-interne System ist eine eigene Entwicklung. Ich habe sehr viele Systeme kennengelernt in meinem beruflichen Leben. Und das System hat mit Sicherheit viele Nachteile. Viele Sachen können wir da nicht so optimal abwickeln oder nachverfolgen. Aber für die Arbeit die wir hier verrichten ist das ausreichend. Ich weiss nicht, ob sie... es gibt Trackersysteme, Trackingtools und so weiter und so fort. Die haben bestimmte Sachen die einfach angenehm sind in der Benutzerführung. JIRA könnte man unter Umständen auch als Fehlerfassungssystem anbinden, aber dann ist das eine politische Sache. Die Schnittstelle und die Möglichkeit ist gegeben auf JIRA zu **Testtool xxx** - welches wir einsetzen für das Testmanagement - es gibt eine Schnittstelle zu JIRA. Aber dadurch, dass das andere System eine eigene Entwicklung ist, ist das im Einsatz, und das ist ein Thema was bis heute nicht grundlegend diskutiert worden ist. Von daher das wird einfach... benutzt. Testmanagement geht eigentlich davon aus, dass die Lebensdauer dieses Systems sehr begrenzt ist... 00:19:39-6

I: (unterbricht) von JIRA? 00:19:43-2

B: Nein, vom Ticketsystem. Das heisst, in absehbarer Zeit werden wir bestimmt das ablösen. Durch ein anderes System. 00:19:52-9

I: Ok. Aber das Ticketsystem ist ja eine ziemlich zentrale Sache. 00:19:58-0

B: Richtig. Weil das Problem bei uns ist, dass, eine Sache die nicht sofort abgestellt werden kann, weil Lotus Notes wurde in den letzten 18 Jahren intensiv genutzt, das heisst, wir haben sehr viel eigene Anwendungen eingebaut, und die Abhängigkeiten sind extrem hoch. Auch unser Zeiterfassungssystem ist auch im Lotus Notes abgebildet, und von daher ist das strategisch schon ein grosses Thema wenn man das angehen möchte. 00:20:40-3

I: Ja dann danke ich mal 00:20:56-2

B: Gerne. 00:20:46-2

I: Hätten sie vielleicht noch Anmerkungen, oder Fragen? 00:20:53-1

B: Nicht direkt Fragen, vielleicht eher.. also ich bin da auch ein bisschen neugierig. Das Thema der Arbeit beschränkt sich nur auf JIRA, oder vergleichbare Systeme? 00:21:11-3

I: Von den Tools her wollte ich es eigentlich begrenzen auf JIRA, und halt ein bisschen vergleichen, wie ist es zum Beispiel... Also das Oberthema ist ja Awareness. Also dass man weiss was machen die Kollegen, was passiert mit Dokumenten oder mit Sourcecode. Und da möchte ich halt auch ein bisschen den Vergleich von den Präferenzen, wie, ich weiss nicht, wie verwendet man es jetzt im Vergleich zum Sametime oder so. Dass man zum Beispiel sagt im Sametime hat man in der Kontaktliste hundert Leute, und im JIRA sieht man aber zum Beispiel diese Leute nicht. Oder man sieht nur.. Oder haben die Leute das JIRA so konfiguriert, dass sie da dieselben Leute sehen wie im Sametime. 00:22:02-9

B: Ja, das ist natürlich.. Das ist ein Problem das wir grundsätzlich haben: dass die Firma sehr schnell gross geworden ist, also dass sich die Firma sehr schnell weiterentwickelt, und ein Konsens hat nicht stattgefunden in dem Sinn dass man sagt: " Ok, was brauchen wir konkret für unsere Weiterentwicklung, und was nicht. Und so dass man sich auch zeitnah von den Sachen die man nicht braucht, sich trennt. Das hat nicht stattgefunden, und das ist was wir momentan im Unternehmen dabei sind zu optimieren. Weil es ist so, dass jede Abteilung andere Tools verwendet, jeder Teamleiter andere Abläufe hat, und, und, und. Mag sein, dass das für bestimmte Konstellationen die beste Lösung ist, aber Firmenübergreifend sollten bestimmte Tools einheitlich eingesetzt werden, damit man auch den Überblick hat. Und das ist was ein klein bisschen fehlt. Aber das ist bekannt und daran wird momentan gearbeitet. 00:23:11-9

I: Darf ich noch fragen welche Kontakte sie beim Sametime drinhaben? 00:23:24-2

B: Ich bin in der Gruppe der Abteilungsleiter drin, in bin in der Gruppe der Teamleiter drin, ich bin in der Gruppe der Javaentwickler drin, ich bin in der Projektgruppe mit dem Vorstand drin, natürlich auch meine eigene Gruppe, meine eigene Abteilung... Ich glaube es sind fünf Gruppen wo ich drin bin, ja. 00:23:50-3

I: Ok. Und ich nehme an, wenn sie in der Früh ins Büro gehen dann ist wahrscheinlich das Lotus Notes das erste was sie aufdrehen, oder? 00:23:55-3

B: Ja. Das ist immer, ich bin immer online praktisch. Und ich kann sehen wer sich schon angemeldet hat, wer schon da ist, und so weiter. Das ist eine sinnvolle Hilfe, weil bevor man aufsteht und zum nächsten Büro hinget, dann kucke ich erst mal "ist der überhaupt am Platz" und von daher, das ist schon ein sehr gutes Feature. 00:24:25-9

I: Ja, dann danke ich sehr herzlich. 00:24:30-3

B: Danke auch.

Interview 05_ENTW_DE_TL_STV_1

I: Was ist deine Aufgabe hier im Team und wie sieht dein täglicher Arbeitsablauf aus, also welche Aufgaben fallen da an? 00:00:15-6

B: Ich bin zum einen im Team als ganz normaler Softwareentwickler mit dabei, mache da Entwicklungsaufgaben im jeweiligen Produktionsabschnitt die anfallen, bin aber auch hauptsächlich dafür da den Produktivbetrieb zu betreuen, und produktive, eingestellte Tickets eben zu bearbeiten und die Fehler dann zu lösen nach Absprache mit dem Kunden. Dann mache ich als stellvertretender Teamleiter auch Arbeiten hinsichtlich der Projektplanung mit, übernehme auch die Aufgabenüberwachung und führe die Codereviews und Qualitätssicherungsmassnahmen bei uns im Team durch. 00:01:11-3

I: Mit welchen Leuten arbeitest du jetzt dabei zusammen immer? 00:01:15-9

B: Im Team meinst du? Also ich arbeite mit **ENTW_DE_TL** zusammen, dem Teamleiter, aber da ich ja durch die Qualitätssicherungsmassnahmen auch mit jedem Entwickler reden muss arbeite ich eigentlich mit jedem dann, zu gewissen Zeitpunkten des Produktionsabschnittes. 00:01:40-8

I: Und mit ausserhalb vom Team? 00:01:42-0

B: Mit anderen Teams habe ich jetzt wenig Schnittpunkte. Mit Abteilungsleitung bin ich natürlich auch häufig im Gespräch.. 00:01:54-8

I: mit **ENTW_DE_AL**? 00:01:54-8

B: Ja genau. Mit dem Produktmanagement wenn wir so Änderungsanforderungen, oder Änderungen während des Produktionsabschnittes eben reinkommen dass wie die einplanen und besprechen , und mit der Konzeption bei fachlichen Fragen bin ich halt noch im Gespräch. 00:02:20-9

I: Wie unterscheidet für dich die Zusammenarbeit zum Beispiel mit den Kollegen die du hier im Büro hast, oder hier am Gang hast, wie unterscheidet sie sich mit Leuten die einen Stock darüber sind, und wie unterscheidet sie sich mit den **Österreichern** zum Beispiel? Also jetzt zum Beispiel in der Wahl der Kommunikationsmittel, ob du jetzt hinübergehst, oder anrufst, oder ein Email schreibst? 00:02:41-8

B: Also bei den Leuten im Team ist natürlich immer die erste Wahl ein persönliches Gespräch zu suchen, dass wir dann im direkten Gespräch die Themen besprechen. Das klappt auch bei der Fachkonzeption am besten wenn man das so macht. Wenn man sich zum Beispiel ein Fachkonzept nimmt und das dann im Gespräch durchgeht und die einzelnen Themen dann untereinander bespricht und erklärt funktioniert so am einfachsten. Bei Leuten oder Kollegen die jetzt nicht direkt im Zugriff sind, wie zum Beispiel den **Österreichern**, funktioniert das schon eher indirekt, also dass du Emails schreibst, oder Tasks zuweist, oder Tickets zuweisen hauptsächlich. Dass man da dann in die Tickets entsprechende Kommentare reinschreibt damit die Leute dann da schon was mit anfangen können. Der persönliche Kontakt ist da natürlich deutlich weniger als jetzt mit den Kollegen die jetzt hier von Ort sind. Ich bin jetzt zwar auch ab und zu in **Österreich** dann vor Ort, aber das ist halt, das ist jetzt wenn man das auf die letzte Zeit zurückrechnet, einmal im Quartal ungefähr dass ich dann da hinfahre. 00:03:59-2

I: Und wenn du zum Beispiel jemandem ein Task oder Ticket zuweist, rufst du dann noch zusätzlich an und sagst "du, pass auf, ich habe dir etwas zugewiesen"? 00:04:09-2

B: Das ist unterschiedlich. Bei JIRA-Tasks machen wir das schon in der Regel, weil es dann auch gleich immer ein paar mehr JIRA-Tasks sind die da zugewiesen werden. Und die in der Regel auch ein fixes Fälligkeitsdatum dann haben. Und bei normalen Fehlertickets ist das eher nicht der Fall. Die werden dann zugewiesen, und dann ist der Kollegen dann selbst verantwortlich die zeitnah zu bearbeiten, da gibt es ja keine fixe Terminierung. Höchstens eben indirekt durch bestimmte Liefertermine die in gewissen Abständen immer da sind, aber die sind immer gleich, von daher stellen sich die Rahmenbedingungen dann sowieso von alleine ein. Und bei den JIRA-Tasks muss man halt dann noch sagen "ja ich geb dir jetzt meinetwegen noch die Summe X dieser Tasks", die Tasks haben einen Termin in, was weiss ich, drei Wochen oder so, fang damit bitte erst in der nächsten Woche an weil du diese Woche noch etwas anderes machen musst. Soetwas bespricht man dann schon. 00:05:14-3

I: Und wie ist das bei dir, du hast ja dann auch mit Releasemanagement öfter zu tun, oder? Eben gerade bei diesen Ticketsachen, da bist ja du immer für die QS verantwortlich? 00:05:25-0

B: Ja (gedehnt), aber das ist weniger jetzt in der Funktion des Releasemanagements, sondern wir sprechen dann eher, wann wird das ausgeliefert, oder was wurde ausgeliefert, gab es hier ein Problem mit den Patchleveln oder sowas. Was die mit den Tickets eben machen, die stellen einfach nur den Status um. Wenn man denen dann sagt

"hier, wurde analysiert" dann stellen die den Status entsprechend ein, da hat man jetzt nicht so den wirklichen Kontakt. 00:05:50-7

I: Aber wird quasi nicht jedes Ticket dann einzeln noch abgesprochen, sondern es wird einfach nur diesem fixen Prozess gefolgt, Status beim Ticketumstellen und tschüss. 00:06:02-5

B: Genau. Da hat sich das dann damit erledigt, ja. 00:06:06-1

I: Nachdem du ja auch stellvertretender Abteilungsleiter bist musst du ja wahrscheinlich auch am Laufenden bleiben, was die Mitarbeiter so machen, also die anderen Entwickler im Komposit-Team. Wie machst du das, oder wie kriegst du das mit? 00:06:26-7

B: Da versuchen wir natürlich sehr viel über JIRA eben abzudecken. Ist nicht ganz einfach. Wir müssen ja in der Iterationsplanung die wir immer im zweiwöchentlichen Abschnitt machen, wird halt geschaut "ja was habe ich jetzt für die Iteration angemeldet gehabt, wie ist der Stand dabei, und was ist da fertig". Das erfolgt auswertungstechnisch über JIRA natürlich, die Tasks die ich dafür angemeldet habe, die werden in JIRA abgefragt ob die eben schon geschlossen sind, oder ob die warten auf irgendwas, so dass man dann den Status da abgeben kann. Nichts desto trotz müssen wir aber immer noch dann in unseren Teammeetings da auch den Zwischenstand abfragen. Denn alles, jeden Schritt kannst du nicht so in JIRA dokumentieren, und da ist es halt besser, wenn man direkt vom Entwickler das Feedback kriegt "ja ich arbeite zwar daran, aber ich kann nicht weitermachen weil irgendwas anderes mich jetzt davon abhält". Und deshalb ist es halt so eine Kombination aus Statusabfragen aus JIRA, und die Mitarbeiter interviewen wie es da mit den Arbeiten aussieht. 00:07:36-5

I: Diese Iterationsplanung, die ist zweiwöchentlich? 00:07:37-5

B: Ja genau, zweiwöchentlich. 00:07:40-9

I: Und wie läuft das dann ab, die JIRA-Tasks gibt es ja glaube ich meistens länger, oder? Die werden am Anfang vom Meilenstein oder so meistens dann eingestellt, also nach der Meilensteinschätzung... 00:07:51-5

B: Genau, wir machen hier eine Schätzung, und dann wissen wir... 00:07:55-1

I: ...(unv.) zehn Untertasks oder so... 00:07:55-1

B: Genau, die werden dann anhand der Project-Planung terminiert. Wir kippen ja nicht direkt alles so in JIRA rein, wir nehmen ja erst mal unsere ganzen Tasks die sich aus den neuen Konzepten ergeben und packen die in MS-Project, das macht **ENTW_DE_TL** eigentlich hauptsächlich, kann da die Tasks schon Mitarbeitern zuordnen die die machen können, daraus ergibt sich dann eigentlich die Planung für den Meilenstein, wann welcher Task in welche Lücke reinpasst, dass er dann da gemacht werden kann. Und aus diesen Terminen die dann halt durch Project rauspurzeln am Ende werden die Fälligkeiten in den JIRA-Tasks dann ermittelt. 00:08:32-0

I: Ok. Und diese zweiwöchige Iterationsplanung...? 00:08:35-7

B: Die ergibt sich dann indirekt quasi daraus. In dieser Iterationsplanung ist dann eben drin, ich habe ein Summe X an Mitarbeitern, was machen die die nächsten zwei Wochen. Da sagt man "ok, drei von denen machen jetzt Tickets, zwei arbeiten an dem Haupttask, zwei arbeiten an dem Task, der macht das". Für die zweiwöchige Planung ist das Ziel, die und die Tasks daraus abzuschliessen, oder meinetwegen auch die und die Fehlertickets zu beheben. Und nach der Hälfte der Iteration, also nach einer Woche, wird dann hingekuckt wie hoch ist die Wahrscheinlichkeit dass diese Ziele erreicht werden können - wird der erste Status eingetragen - und am Ende der Iteration wir dann eine Abnahme gemacht was wurde davon erreicht und was nicht. Und diese Iterationsplanung ist halt hauptsächlich das Überwachungsinstrument von der Abteilungsleitung, von **ENTW_DE_AL**. Der mach das mit allen Entwicklungsteams - Komposit, Finanzen, Provision - um eben den Fortschritt innerhalb des Meilensteins auch zu überwachen. Damit er weiss was da passiert. 00:09:42-8

I: Das wird dann aber eben.. Da schreibt **ENTW_DE_TL** immer diese Status-Emails, oder? Es gibt zum Beispiel bei "Breiteneinsatz" oder so... 00:09:55-7

B: Das hat damit nichts zu tun. Das war jetzt nur Status-Emails rein spezifisch für die ein bestimmtes Projekt bei KFZ. Die Iterationsplanung wird eigentlich nur zwischen dem Teamleiter und dem Abteilungsleiter hin- und hergeschoben. 00:10:11-8

I: Ok. Also eben um das zu bewerten wie weit das ist schaut ihr dann auch nochmal ins JIRA rein. 00:10:22-1

B: Genau. 00:10:22-1

I: Also in dieser wöchentlichen, also nach einer Woche dann.. [00:10:23-0](#)

B: Für diesen Status nehmen wir dann einfach die Tasks in JIRA in eine Liste, kucken ob die geschlossen sind, ob die warten oder was auch immer, und (unv. sie sagen) "ok, da musst du jetzt für die zwei Wochen".. muss er zehn Tasks machen. Jetzt sind zur Hälfte der Zeit fünf schon geschlossen, die Wahrscheinlichkeit dass er die anderen schafft ist dann auch entsprechend bei 70 Prozent oder so. [00:10:47-8](#)

I: Und Ding.. musst du dann wahrscheinlich auch mitberücksichtigen - Ticketsystem. [00:10:48-8](#)

B: Das Ticketsystem ist auch mitberücksichtigt. In dieser Iterationsplanung werden auch Ziele angegeben, wir müssen jetzt zum Beispiel innerhalb der nächsten zwei Wochen die Ticketzahlen unter ein bestimmtes Niveau bringen zum Beispiel. Oder wenn da jetzt Tickets sind die einen grösseren Aufwand haben, dann werden die da auch mal explizit genannt, dass man eben in den zwei Wochen dann vier Tage für dieses Fehlerticket dann da kalkuliert. [00:11:17-0](#)

I: Und wie wird das festgehalten? [00:11:23-3](#)

B: Das ist eine Excel-Liste mal wieder, und die wird dann.. eigentlich wird die von **ENTW_DE_AL** so verwaltet. Der hat ja für die Koordination seiner ganzen Teams eigentlich auch Excel-Listen. Und ja, das JIRA wird halt von den Entwicklungsteams so direkt benutzt. Das hat dann eben so diese teamübergreifende Funktion. [@Kategorie JIRA-Bedürfniss: Obere Ebene benutzt JIRA nicht mehr, verlässt sich auf Excel, untere Ebene verwendet es schon]. Das benutzt aber **ENTW_DE_AL** dann aber eigentlich nicht dafür für seine Auswertung. Das heisst, du musst alle, du musst eigentlich die ganzen Statusinformationen die du von JIRA eben abfragen kannst nochmal in die jeweiligen zu Verfügung stehenden Excel-Listen übertragen. Das ist noch das grosse Problem einfach. [00:12:06-3](#)

I: Mit wem sitzt du eigentlich im Büro? [00:12:16-2](#)

B: Ich sitze mit **ENTW_DE_6** im Büro, und mit **ENTW_DE_7**. [00:12:19-4](#)

I: Und die sind auch bei Projekten dabei wo du selber mitarbeitest? [00:12:22-8](#)

B: Genau. Die sind auch in diesem ganz normalen LFS-Projekt halt mit dabei, machen auch Komposit-Sparten. Also eigentlich machen wir im Komposit-Team ja.. arbeiten an den Komposit-Sparten - Wohngebäude, Hausrat, Haftpflicht, Unfall und so weiter - dann gibt es natürlich einige Mitarbeiter die sind jetzt für Hausrat, die sind jetzt für Wohngebäude, die machen eher Unfall, Haftpflicht, und bei mir im Büro sind halt **ENTW_DE_6** und **ENTW_DE_7**, die auch am Hausrat und Wohngebäude-Teil mitarbeiten. [00:12:58-6](#)

I: Bist du wahrscheinlich Verantwortlicher für Hausrat und so, oder? [00:13:03-2](#)

B: Die Hausrat-Entwicklung die habe ich mehr oder weniger überwacht, so wie in dem Maße wie **ENTW_DE_TL_STV_2** das dann eben für den KFZ-Teil macht. [00:13:12-3](#)

I: Und wie ist das zum Beispiel wenn du.., du sitzt mit denen im Büro, und du hörst zum Beispiel wie **ENTW_DE_6** telefoniert mit jemandem. Schaltest du dich dann ein, wenn du hörst da gibt es ein Problem, oder ist das... [00:13:23-2](#)

B: Meistens schon. Also wenn ich merke.. Das Problem ist dass ja **ENTW_DE_6** auch erst sein ungefähr einem dreiviertel Jahr ungefähr da ist, und wenn man dann hört "na" - sie macht sehr viel natürlich selbstständig und klärt Sachen auch in Eigenregie, da muss ich nichts dazwischen machen - aber wenn man mal hört "ok", mit einem Ohr zugehört, und das klang jetzt vielleicht doch ein bisschen komisch, dann müsste man vielleicht doch eine andere Lösung anstreben, da rede ich dann gleich mit ihr worum es da ging, und ob das jetzt der richtige Weg war den sie gewählt hat. [00:13:55-3](#)

I: Und machst du das zum Beispiel mit anderen auch, mit **ENTW_DE_TL_STV_2** zum Beispiel? Oder redest du mit **ENTW_DE_TL_STV_2** zum Beispiel über Probleme und so Sachen? [00:14:03-9](#)

B: Da reden wir schon. Also wenn wir ein Problem haben wo wir sagen "ok, ich habe eine Lösung, aber ich bin mir nicht sicher ob das jetzt so das richtige ist" dann tauschen wir uns generell auch immer alle aus. Das passiert entweder bei Bedarf, dass jemand zueinander (unv. rüber) geht, und das dann nochmal mit dem bespricht. Oder das passiert dann auch in den Teammeetings die wir eben dreimal die Woche machen. Wenn dann jemand da einen Fall gelöst hat der ein bisschen umfangreicher oder ein bisschen schwieriger war, dann schildert er das dann eben. Und wenn dann jemand meint, dass er vielleicht nochmal eine andere Meinung dazu hätte, dann wird das dann da auch ausgetauscht in den Meetings. [00:14:38-7](#)

I: Frage: Wie musst du jetzt über verschiedene Dokumente informiert sein dass sich da was ändert, bei Fachkonzepten, Sourcecode, CruiseControl, solche Sachen? [00:14:55-2](#)

B: Was mache ich mit denen? [00:14:57-4](#)

I: Wie erfährts du zum Beispiel wie... dass es ein neues Fachkonzept gibt? [00:15:02-1](#)

B: Ok. Bei Änderungen die jetzt von anderen Teams bei uns einlaufen ist es eigentlich so, dass wir dann auch entsprechend den JIRA-Task zugewiesen kriegen. Und dann siehst du halt in deiner Taskübersicht was du jetzt da offen hast. Es gab halt mal die Zeit dass.. zu Beginn des neuen JIRA-Projektes was wir halt aufgesetzt hatten war das so konfiguriert, dass man keine Email bekommen hat wenn ein neuer Task einem zugewiesen wurde. Dann musste man halt immer pro-aktiv in seine Übersicht schauen, und sehen was hat sich geändert, was habe ich jetzt für Tasks. Das hat sich im weiteren Verlauf dann nicht so glücklich herausgestellt, es gingen dann doch Sachen unter die man dann nicht sofort auf einen Blick gesehen hatte, so dass es entweder eingeschalten wurde, dass du parallel auch noch eine Email halt bekommst wenn du eine neue Task bekommst. Und das ist im Regelfall so, wenn sich am Konzept was ändert, dann gibt es eine neue Konzeptversion die eben von uns QS benötigt, und dann kriegt man eben so eine QS-Task zugewiesen, und dann kriegt man das eben mit. Alle anderen Sachen muss man auf Eigeninitiative nachschauen sage ich so. Also man kriegt zwar oft durch CruiseControl Emails, allerdings sind das, wird man da ja zubombardiert eigentlich aus diesen Emails, dass man die eigentlich ignorieren MUSS. Man muss selbstständig in den CruiseControl reinschauen, einmal am Tag... [00:16:35-7](#)

I: Wie oft machst du das zum Beispiel? [00:16:34-1](#)

B: Ja wenn ich selbst Sourcen geändert habe und eingecheckt habe kucke ich natürlich mehrmals am Tag rein - ob das so funktioniert wie ich das gemacht habe. Ansonsten schaue ich ja.. einmal am Tag, kann auch sein dass ich einmal nicht reinschaue, und wenigstens dann am Ende der Woche um nochmal zu checken "wie ist der Stand auf dem CruiseControl" bevor jetzt das Wochenrelease gebaut wird. Haben wir jetzt noch Module die fehlschlagen wo man nochmal einschreiten muss, das mache ich dann in unregelmässigen Abständen, aber so dass man das halt noch überwachen kann. [00:17:09-6](#)

I: Ok. Und wie ist jetzt nochmal dieser Weg vom Fachkonzept? Angenommen da gibt es eine Änderung. Dieser QS-Task - an wen geht der, geht der an **ENTW_DE_TL** oder...? [00:17:22-6](#)

B: Im Regelfall gehen die Tasks.. Wenn JIRA-Tasks teamübergreifend zugewiesen werden dann gehen die in der Regel immer an den Teamleiter. Und der ist dann weiter dafür verantwortlich dass die dann verteilt werden. [00:17:44-3](#)

I: Dann eine Frage dazu wie du dein Sametime eingerichtet hast: und zwar, wieviele, ode welche Gruppen und welche Benutzer hast du im Sametime in deiner Kontaktliste? [00:17:58-3](#)

B: In meiner Kontaktliste habe ich das so eingestellt, dass die Email-Verteiler - also wir haben für jedes Team auch einen Email-Verteiler, zum Beispiel CT_KMP, CT_KMP das ist ja der Email-Verteiler für uns - den habe ich als Kontaktgruppe auch so eingerichtet. Und so habe ich das für jedes Team auch angelegt. Dann sieht man wer in welchem Team gerade online ist, oder wer da ist und wen man bei Problemen direkt fragen kann. Und das habe ich dann auch für die anderen Teams, für Support, oder für Produktmanagement, Auftragsmanagement, die habe ich halt alle so eingruppiert, dass man dann sieht wer von denen da ist. Der Vorteil ist so, wenn sich da Kollegen.. wechseln manchmal auch die Teams, oder es kommen neue Kollegen hinzu, dann hast du die eben dann gleich mit in der richtigen Liste einsortiert. [00:18:52-0](#)

I: Achso. Das kann man direkt im Sametime angeben? Da brauche ich nur einen neuen Kontakt hinzufügen.. [00:18:57-4](#)

B: Du sagst "neuen Kontakt hinzufügen, neue Gruppe hinzufügen" und dann nimmst du "bestehende Gruppe auswählen" und kannst dann diesen Verteiler direkt auswählen. [00:19:10-9](#)

I: Ok. Und sagen wir zum Beispiel: wenn du ins JIRA reinsiehst, da siehst du ja auch in deiner Übersicht gewisse Leute. Sind das dann die selben Leute auch.. Oder welche Leute sind das, die du zum Beispiel in JIRA siehst? [00:19:28-2](#)

B: In JIRA... [00:19:35-3](#)

I: Also da hast du ja vom Support wahrscheinlich niemanden drin, oder? [00:19:37-9](#)

B: Nein, gar nicht. Also ich habe jetzt auch noch nie direkt in JIRA irgendeine Kontaktliste oder soetwas gepflegt.

00:19:44-8

I: Man sieht ja zum Beispiel ab und zu wenn irgendjemand einen Task aufmacht, dann steht drin "Autor soundso, Bearbeiter soundso".. 00:19:54-4

B: Die werden ja vorgegeben vom JIRA-Admin, da habe ich jetzt noch nicht geschaut ob man das irgendwie beeinflussen kann, aber da habe ich nichts geändert. Also die die vom jeweiligen System dann eingetragen sind die sind da in der Liste dann auch drin. 00:20:16-1

I: Wo siehst du die Vorteile und Nachteile im Vergleich zwischen JIRA und Ticketsystem? 00:20:34-4

B: Wenn ich jetzt das JIRA auch als Ticketsystem nutzen würde, nicht? 00:20:39-6

I: Ja. Oder jetzt nicht, sagen wir einfach in Bezug dass du dir eine Übersicht machst - wieviele Sachen sind offen, wer hat wieviele Sachen noch zu erledigen, wo gibt es Probleme oder so. 00:20:57-0

B: Ich kann dir nur sagen was wir jetzt mit den jeweiligen Tickets machen. Bei unserem Ticketsystem ist es ja so, dass du eine Gesamtliste hast. Du kannst da zwar verschiedene Ansichten auch auswählen - so eine Ansicht wäre, wenn du das jetzt nach JIRA überträgst ein entsprechender Filter den du in JIRA definierst, und dann halt nur eine begrenzte Menge der Daten dann anzeigt. Ich arbeite bei uns in der Ticketübersicht eigentlich mit der Gesamtliste. 00:21:28-1

I: Wo du von allen Teams die Tickets siehst. 00:21:30-7

B: Genau. Von allen Teams sehe ich dann alle Tickets (...) damit lässt sich eigentlich noch ganz gut arbeiten. Das sind jetzt nicht so viele Einträge dass man das jetzt nicht mehr verwalten könnte. Es geht für mich am schnellsten, weil du eben einfach dann da durchblättern kannst, wenn ich auch mal bei anderen Leuten in die Tickets kucken muss. Da ich ja bei uns die Tickets auch verteile muss ich halt auch recht häufig einen Überblick haben "welche Tickets gibt es schon". Wenn jetzt neue Tickets reinkommen, oder wer hat in welchem Ticket gearbeitet, wer bekommt das jetzt vielleicht am ehesten - und da ist es eben einfach wenn man eine Gesamtübersicht hat. In JIRA muss man von Haus aus eher viel mit Filtern arbeiten. Wenn du da keinen Filter hast, dann hast du einfach nur einen grossen Sack wo die ganzen Tasks drin sind. Da hast du dann keine Übersicht. Und so bist du halt dann immer davon abhängig die Filter so zu setzten. Das ist wahrscheinlich einfach mal eine Konfigurationssache, dass man das einmal einstellt. 00:22:27-4

I: Hast du das zum Beispiel gemacht bei dir? 00:22:27-4

B: Wir haben jetzt für die jeweiligen Produktionsabschnitte entsprechende Filter dabei. 00:22:40-7

I: Können wir uns die dann anschauen? 00:22:40-3

B: Die können wir uns dann anschauen, ja. Also ich weiss auch von den Kollegen bei uns im Team, dass die das eben nicht haben, obwohl wir das halt sehr oft auch sagen. Wir haben halt bei **ENTW_DE_TL** in seinem JIRA ganz viele Filter definiert, die man dann einfach abonnieren kann, oder einfach kopiert, an die man, die halt veröffentlicht sind, da drankommt, und dann kann man die ja nutzen. Das machen aber viele Mitarbeiter bei uns im Team auch nicht. Die sehen halt einfach nur ihre Task, und haben halt nicht so die Gesamtübersicht zu den jeweiligen Abschnitten die es dann gibt. Müssen sie wahrscheinlich auch nicht unbedingt, weil sie auch nicht zwingend überall immer informiert sein brauchen, aber für uns - diejenigen die die Planung machen - müssen wir das halt so machen. Da haben wir eben die jeweiligen Tasks eingestellt. 00:23:34-4

(JIRA-Login) 00:24:07-1

I: Ok, also das ist jetzt dein Standarddashboard.. 00:24:06-6

B: Genau. Dieses Dashboard ist eigentlich ein KMP-spezifisches. Das hat **ENTW_DE_TL** eingerichtet für alle Mitarbeiter bei uns im Team. Man hat eben hier so eine Statistik über alle dir aktuell zugewiesenen Tasks. Bei mir ist es so, ich habe jetzt sechs Tasks für die ich eine Fachkonzept-QS machen soll, oder die ich eben weiterverteilen soll, je nachdem wer die Themen dann jetzt bearbeitet. Dann gibt es hier eine Liste mit allen Tasks die eben noch so offen sind.. 00:24:41-8

I: Das sind auch deine Tasks? 00:24:40-7

B: Genau, das sind jetzt aber Tasks die mir zugewiesen sind - projektübergreifend. Wir hatten ja früher dieses KMP-JIRA-Projekt. Daraus resultieren jetzt noch ein paar alte Tasks. Wir arbeiten ja jetzt eigentlich mit dem **LFS**-JIRA-Projekt, das sind jetzt die aktuellen Tasks daraus. Aber das ist eben eine komplette Liste, alles mir

zugewiesene Unteraufgaben. [00:25:06-8](#)

B: Dann gibt es hier zwei Kalender, in denen werden halt dann die jeweiligen Tasks dargestellt die dir jetzt zugewiesen sind, wann die fällig geworden sind. Ich sehe jetzt hier, am 24. August sind mir jetzt diese Tasks alle zugewiesen worden. [00:25:23-4](#)

I: Ach, die sind neu? [00:25:26-9](#)

B: Die sind jetzt alle neu dazugekommen. Diese Ansicht zeigt mir wann wurden mir welche Tasks zugewiesen. [00:25:34-2](#)

I: Und das sind die Fälligen? [00:25:34-2](#)

B: Das sind die Fälligen. Alle Fälligen. Ich weiss jetzt nicht wann diese anderen fällig sind. Wenn wir jetzt mal beispielhaft so einen auswählen (klick).. Das ist jetzt ein Fachkonzept-QS, die wäre jetzt.. "Fälligkeit der Teilaufgabe 26.8." [00:26:02-0](#)

I: Ach, ok. Und das darüber das ist vom Haupttask.. [00:26:02-4](#)

B: Das ist der Haupttask. Das ist halt ein Konzept was schon diverse Produktionsabschnitte mitgelaufen ist. Deswegen gibt es jetzt so einen alten Fälligkeitstermin (12.4.2011). (...) Da wir heute den.. heute ist der 24., nicht? Ist jetzt ein bisschen merkwürdig, warum die auch heute gerade alle fällig sind, da hat wahrscheinlich derjenige der es zugewiesen hat einen etwas optimistischen Eintrag reingestellt, oder er hat es einfach vergessen und es wurde vorbelegt. Das kann natürlich auch sein, das müsste man überprüfen. Das ist aber so die Hauptansicht, an der orientiert man sich was habe ich zu tun. [00:26:43-4](#)

I: Ok, und wenn du jetzt zum Beispiel etwas tun musst, wo gehst du dann weiter? [00:26:48-8](#)

B: Ich würde mir dann erst mal schauen "was habe ich jetzt hier für Tasks?". Wenn du da auf den Link klickst und die Übersicht, was kommt da? Ok, dann sehe ich... das sind alles QS-Tasks, da sind auch Produkt-QSen drin, die würde ich jetzt an die jeweiligen Produkt-Entwickler weiterleiten. [00:27:09-5](#)

I: Achso, die machst du jetzt nicht selbst in dem Fall. [00:27:09-8](#)

B: Nein, da mache ich jetzt keine Entwicklungs-QS, das macht jetzt der Produktentwickler dann. [00:27:16-1](#)

I: Das kriegt dann **ENTW_DE_2** oder.. [00:27:18-5](#)

B: Die wurden mir jetzt zugewiesen, weil **ENTW_DE_TL** eben noch im Urlaub ist, damit ich die weiterverteilen kann. [00:27:24-2](#)

I: Die wurden dir direkt jetzt von **FK_TL** zugewiesen? [00:27:27-0](#)

B: Ähm (gedehnt), nicht zwingend (klickt). Die wurden manchmal auch zu.. von.. (sucht).. zugewiesen von.. (...) hm... kann natürlich sein dass mir **FK_TL** zugewiesen hat. [00:27:46-6](#)

I: Oder weiss zum Beispiel **FK_TL**, weiss der dass **ENTW_DE_TL** in Urlaub ist, und dass du sein Stellvertreter bist? [00:27:52-1](#)

B: Also normalerweise wissen die Teamleiter wenn sie im Urlaub sind, und wissen dann auch wer Stellvertreter ist. Wahrscheinlich, kann natürlich sein dass der Task jetzt ganze paar Tage bei **ENTW_DE_TL** gelegen hat, und deswegen der jetzt schon fällig ist am 26.8. und **FK_TL** ihn mir jetzt direkt gegeben hat. Das weiss ich jetzt nicht, das kann natürlich sein. [00:28:19-7](#)

B: So, und ansonsten haben wir jetzt hier diese Filter. Diese **LFS**-Filter die sind eben von **ENTW_DE_TL** erstellt, und wir benutzen die eben mit. Wenn ich mir jetzt eine Übersicht geben lassen möchte über alle Haupttasks zu 11-3, also dem aktuellen Produktionsabschnitt, dann nehme ich den eben, und dann habe ich hier eine Liste - alle Aufgaben die wir hier jetzt für diesen Produktionsabschnitt machen müssen - alle Haupttasks. [00:28:53-6](#)

I: Das ist jetzt für das ganze Team? [00:28:52-9](#)

B: Für das ganze Team, genau. Das sind Produkttasks und Dialogtasks, und alles was dazugehört. Dann kann ich eben die jeweiligen Tasks durchgehen, und dann schaue ich da eben rein. Sind die Impl-Aufgaben da jetzt geschlossen, oder welchen Status haben die. Es gibt auch bei **ENTW_DE_TL** noch einen Filter der eben die jeweiligen Unteraufgaben auch schon selektieren kann zu einem bestimmten Termin. Den nutze ich jetzt weniger

oft, denn das ist eigentlich eher **ENTW_DE_TLs** Aufgabe die Unteraufgaben zu kontrollieren und so den Fortschritt zu überwachen. Ich mache das jetzt.. also wenn ich was wissen will gehe ich direkt über den Haupttask rein, und schaue mir an wie ist der Status zu dem ganzen Thema. Denn wir haben ja auch nicht nur das JIRA, sondern auch diese Excel-Listen-Planung, sage ich mal jetzt, die **ENTW_DE_AL** vorantreibt, und da ist eigentlich die Liste die wir jetzt haben - auch nochmal eine Excel-Liste drin, und in diese Excel-Liste müssen dann auch diverse Termine eingetragen werden immer. [00:30:01-6](#)

I: Also diese JIRA-Liste kommt in die Excel-Liste rein, oder wie? [00:30:06-2](#)

B: Ja, so ungefähr. Also die werden schon auf unterschiedlichen Wegen erzeugt, die sind innerlich dann aber deckungsgleich schon eher. Denn in dieser Excel-Liste "Status Softwareproduktion" stehen dann immer die Themen drin, die wir wirklich auch machen für den jeweiligen Meilenstein. Hier kann es aber schon mal sein, dass da auch Sachen drin sind die jetzt.. die haben eben die Zielversion 11-3, aber die haben dann - auch wenn wir die jetzt nicht in 11-3 machen - kriegen die da meinetwegen die Zielversion 12-1 ebenso. Wenn ich jetzt mal zum Beispiel hier 'Wohngebäude' reinschaue, in den Haupttask, der hat dann jetzt erst mal nur 11-3 als Lösungsversion, wenn wir den aber nicht abschliessen in 11-3, dann bekommt der natürlich 12-1 noch als zusätzliche Lösungsversion dazu. Das führt aber auch dazu, dass der weiterhin in dieser Liste drin ist. Das ist immer davon abhängig wie definiere ich das jetzt. Definiere ich die Lösungsversion nur für einen konkreten Produktionsabschnitt, oder lasse ich das über mehrere Laufen. Ich glaube hier zum Beispiel, "Hausrat Neuanlage" der hat jetzt mehrere.. der hat auch nicht mehrere drin, ok. [00:31:13-1](#)

I: Eine kurze Zwischenfrage: dieses Thema, das ist quasi immer ein Fachkonzept das umgesetzt wird [00:31:17-3](#)

B: Hauptaufgabe, genau. [00:31:22-6](#)

I: Und da ist dann drin Fachkonzeption selbst erstellen, QS für das Fachkonzept, dann Implementierung.. [00:31:24-8](#)

B: Diese ganzen Untertasks, genau. [00:31:28-5](#)

I: Ok. Und bei Implementierung habe ich dann zum Beispiel wiederum zehn Unterschritte wo ich sage, keine Ahnung, "Stammdaten anlegen" [00:31:32-5](#)

B: Nein... genau. An einem Thema gibt es X Unteraufgaben. Das ist ja so in JIRA, dass du nur diese zwei Ebenen verwalten kannst. Und wenn wir mehrere Implementierungsaufgaben haben, dann gibt es eben mehrere Implementierungsaufgaben da drin, direkt. [00:31:51-3](#)

I: Achso, da sind dann schon mehrere da drin? [00:31:51-3](#)

B: Genau. (...). Das sind jetzt Teilaufgaben zur Implementierung. Dialog XY anpassen, alles Implementierungsaufgaben. Und genauso gibt es dann eben hier Fachkonzept-QS-Aufgabe, oder Fachkonzept... Und wenn man dann da reingeht ist da immer das aktuelle Fachkonzept dazu drin. So hat man alles zu dem Thema auf einer.. auf einem Task zusammengefasst. [00:32:26-3](#)

I: Und schaust du selber zum Beispiel dann nach, wenn du siehst, ok, **FK_1**, ich weiss nicht, hat da dieses Ding gemacht, schaust du dann zum Beispiel ab und zu nach was **FK_1** gerade macht oder so? Oder ist dir das... [00:32:35-4](#)

B: Nein. [00:32:38-2](#)

I: Das spielt für dich keine Rolle. [00:32:37-3](#)

B: Spielt für mich keine Rolle. Also ich habe auch nie in JIRA gekuckt wer, oder was andere Leute zu tun haben. Habe ich eigentlich weniger gekuckt. [00:32:47-2](#)

I: Von deinen Mitarbeitern jetzt auch nicht? Von **ENTW_DE_6** zum Beispiel? [00:32:53-7](#)

B: Hm.. Indirekt. Also du siehst ja.. mich interessiert nur welche Tasks sind überhaupt offen. Im zweiten Schritt ist dann erst interessant "wer bearbeitet die Tasks". Und das sehe ich dann eben über die offenen Tasks, aber ich habe noch nicht gekuckt was ist, was macht jetzt zum Beispiel **ENTW_DE_6** nächste Woche oder sowas. [00:33:14-5](#)

I: Aber wird das zum Beispiel immer eingehalten von den Leuten, dieses Status-Setzen - "in Arbeit" und so? Weil ich habe vorher ein bisschen in JIRA herumgesurft, und habe gesehen, dass es schon manchmal Leute gibt die haben drei Sachen gleichzeitig "in Arbeit". [00:33:31-9](#)

B: Mit dem "in Arbeit" bin ich nicht immer so sicher ob das dann immer so eingehalten wird. Das ist unterschiedlich. Es gibt Kollegen die machen das konsequent, aber es gibt auch Kollegen die haben eine andere Arbeitsorganisation und die machen es dann eben nicht. Also "in Arbeit", ok, deshalb brauchst du eben auch noch diese Meetings unter der Woche um wirklich zu wissen woran arbeitet der jetzt. Das soll dann eben diese Lücke abdecken. [00:33:57-0](#)

I: Dann wäre noch eine Frage zu diesen Aufwandsschätzungen und Fälligkeiten die in JIRA eingetragen sind: wie akkurat sind die, und wie läuft das ab wenn das nicht eingehalten wird? [00:34:12-8](#)

B: Hm.. (...) [00:34:12-9](#)

I: Also ich weiss dass wenn sie nicht eingehalten werden, dann muss man dem **ENTW_DE_TL** eine Email schreiben und der verschiebt sie dann eben um eine Woche.. [00:34:24-0](#)

B: Genau.. Also man muss schon kucken, dass man die Planung - wie die Termine gesetzt sind - hinbekommt in der Entwicklung. Es kann natürlich immer Themen geben wo man sich entweder verschätzt hat, oder wo andere Themen dazwischenkommen - das muss man dann im Einzelfall abwägen. Werden die dann neu terminiert, oder ob man dann sagt "arbeite daran weiter, und wenn du fertig bist bist du fertig". Je nachdem. Das ist immer unterschiedlich. Wenn man jetzt sagt ich kann die nächsten zwei Wochen da sowieso nicht dran arbeiten, dann kriegen die mit Sicherheit gleich erst mal einen neuen Termin der realistisch ist. Aber ansonsten sind die Fälligkeiten eigentlich so geplant, dass die auch so hinkommen sollten. Die ergeben sich ja eben aus diesem MS-Project, wo alle Tätigkeiten hinterlegt sind und je nach Verfügbarkeit der Mitarbeiter und nach Aufwandsschätzung ergeben sich halt dann diese Implementierungsraster - wann welcher fertig ist - und daraus sind dann die Fälligkeitstermine schon relativ gebunden. Denn du sollst dann erst mit der nächsten Aufgabe fertig machen wenn das andere Thema erledigt ist. [00:35:25-1](#)

I: Ja, aber das ist ja wahrscheinlich trotzdem problematisch mit dem Ticketsystem, weil das ja da eben nicht abgebildet ist, oder? Und wenn ich dann zwar irgendwie meine Tasks in JIRA habe, und dann kommen aber auf einmal zehn Tickets rein die natürlich sofort gemacht werden müssen, oder meistens halt.. [00:35:41-5](#)

B: Genau, also im Regelfall haben wir bei der Planung - wenn wir dieses MS-Project da eben aufziehen - einen Ticketpuffer sowieso da mit einkalkuliert. Das heisst, wenn das jetzt Mitarbeiter X, das sagst du "ok, der kann 50 Prozent Neuentwicklung machen, 50 Prozent Tickets", dann ist der einfach nur für 50 Prozent in diesem Projekt eingeplant. Und aus dieser Verfügbarkeit ergeben sich ja dann auch Verschiebungen in den Fälligkeitsterminen. Also die sind schon soweit nach hinten geplant wie es nur irgendwie geht mit Puffern von Tickets und Urlaub, und anderen Aufgaben die man hat. [00:36:24-5](#)

I: Und wird da zum Beispiel das auch dann schon in Betracht gezogen, dass man, ich weiss nicht, den Terminkalender von Testteam oder so anschaut? .. Damit man weiss, ok die haben da jetzt eine Testoffensive geplant, oder auch vom Kunden... [00:36:35-1](#)

B: Wir haben natürlich Rahmenbedingungen in den Terminen drin - das ist ganz klar - das sind zum Beispiel "Feature Complete Version" oder "Testoffensive" oder wann die erste Auslieferung ist. An diesen Terminen.. die sind natürlich fix. Und wenn du dann so und so bis da fertig sein musst, hast du halt nicht so viel Spielraum da jetzt die Termine hin- und her zu schieben. Das muss man da natürlich abstimmen. Da komme ich dann wieder zu dieser Produktionsplanung von **ENTW_DE_AL**. Bei der hängt hinten auch immer so eine Risikoliste mit drin. Wenn man jetzt sagt "oh, ich kann jetzt diesen Termin für dieses Thema nicht einhalten" dann muss man da ein Risiko anmelden. Dann sagt man "Risiko: Ticket-Eingang Produktion verhindert Fertigstellung von Thema XY. Die Wahrscheinlichkeit dass das Risiko eintritt ist so-und-so hoch, wenn das passiert, dann verschiebt sich der Termin um zwei Wochen" meinetwegen. Das meldet man dann an, und wenn der Fall dann eingetreten ist, dann ist man wenigstens darauf vorbereitet dass sowas passiert. Dann wird der Termin diesen (unv. Führer) in den Excel-Listen sage ich mal entsprechend angepasst. Der Termin der JIRA-Task wird nicht immer angepasst. Es kann auch sein, dass die einfach mal da liegen. Und dann hast du eben diese Diskrepanz zwischen Excel-Liste, JIRA, .. ja, "was stimmt jetzt?,"... Solange es um Termine geht ist es ok, aber sobald du irgendwelche Statusabweichungen dann dazwischen hast, das ist dann schon ein bisschen komisch. Da wird es schwierig dann das synchron zu halten, wenn du jetzt im JIRA sagst "ich habe die Aufgabe schon abgeschlossen", in der Excel-Liste ziehst du den Status dann nach, sagt "ah, ja, hier da, fertig", irgendwann fällt dem Entwickler ein "ach nein, ich habe was vergessen", macht sie wieder auf, dann ist die da wieder offen, und der Status, dann hat das wieder keiner mitgekriegt... Da gibt es schon diverse Organisationsschwierigkeiten manchmal. Das ist schon echt grausam teilweise. [00:39:06-9](#)

B: Nein. [00:39:06-9](#)

I: Warum nicht? [00:39:06-9](#)

B: Also bei mir ist es so, dass ich in erster Linie an Tickets, also an echten Fehlertickets arbeite. Und da wir die eben nicht in JIRA abgebildet haben, hat es dafür keine grosse Relevanz. Die wenigen Sachen die ich dann in der Neuentwicklung doch mal mitmache, hm.. dafür habe ich mich damit noch nicht tiefgründig genug beschäftigt um das (unv...) 00:39:33-8

I: Also du hast das auch nicht geöffnet. 00:39:36-0

B: Nein. Ich hatte mal die ganzen Eclipse-Sachen eingestellt dafür, aber das ist dann wieder weggelaufen, weil ich das nicht benutzt hatte. Da ich Ewigkeiten dann da keine Sachen mit aufgezeichnet habe. 00:39:53-0

I: Ok. Ja, dann danke ich mal sehr herzlich. 00:40:04-7

B: Gerne. Wenn ich dir nochmal was erläutern soll dann sag mir bescheid. 00:40:18-2

I: Eine Frage habe ich noch kurz: wenn du für dich selbst navigieren musst in JIRA, hast du dann Bookmarks gesetzt im Browser? 00:40:21-6

B: Ja, genau, ich habe für die einzelnen JIRA-Projekte Bookmarks, und ansonsten, wenn ich innerhalb der Projekte navigieren muss, dann habe ich da eben die Filter mit denen ich dann arbeite. 00:40:40-1

I: Die einzelnen JIRA-Projekte sind dann eben diese Themen? Also dass du sagst, was weiss ich, Hausrat... 00:40:41-4

B: Nein, mein JIRA-Projekt ist jetzt hier "**LFS**". Und das alte JIRA-Projekt, zum Beispiel dieses KMP-Projekt was du auch kennst. Das ist das andere JIRA-Projekt. Weil, wir ja manchmal noch Aufgaben haben die wir noch nicht in das neue Projekt umgezogen haben, dann musst du halt in das andere Projekt noch reinkucken. Dafür arbeite ich mit Bookmarks, um die Projekte anzusteuern einfach. 00:41:04-2

I: Und die Fachkonzepte, hast du da dann bei dir lokal auch noch eine Version liegen? 00:41:05-0

B: Nein. Ich hole nur die aus dem JIRA. 00:41:11-3

I: Die öffnest du dann gleich direkt aus dem JIRA heraus und schautst..? 00:41:14-7

B: Ja, genau. Man ärgert sich manchmal ein bisschen weil man dann manchmal auch ein bisschen suchen muss, oder gerade wenn man eines öfter braucht, dann ärgert man sich "verdammt, habe ich gerade zugemacht". Aber ich halte mir... ich halte auch nichts davon diese ganzen Dokumente immer irgendwie doppelt abzulegen. Weil irgendwann hast du das Problem, dass du irgendeine Änderung nicht bei dir hast, die du verpasst hast irgendwie, und.. schon hast du den Salat. 00:41:46-3

I: Ok. Dann schönen Dank. 00:41:49-4

B: Ja. Dafür nicht.

Interview 06_FK_2

I: Also als erste Frage: was ist deine Aufgabe bei **VersicherungsGmbH**, und wie sieht so dein Arbeitsalltag aus? 00:00:12-1

B: Also ich bin in der Fachkonzeption tätig, und schreibe so quasi die Fachkonzepte entsprechend. Für den Bereich Produkt und auch Bestand. Mein Arbeitsalltag sieht eigentlich so aus dass ich.. ja, Fachkonzepte schreibe. Also.. (unv.) Tickets manchmal bearbeite, oder jetzt momentan grösstenteils Tickets bearbeite, fachliche Sachen kläre, und so weiter. Also alles im Bereich was so fachliche Sachen angeht, da bin ich zuständig. 00:00:46-2

I: Mit welchen Leuten arbeitest du zusammen? 00:00:49-4

B: Also hauptsächlich mit den Entwicklern auf Seite von **VersicherungsGmbH** und auf Seite von der **Kunde_1**, mit den entsprechenden Fachabteilungen, beziehungsweise mit der Versicherungstechnik dort. 00:01:01-0

I: Wie oft bist du mit denen in Kontakt, zum Beispiel mit den **Süddeutschen**? 00:01:07-8

B: Also eigentlich täglich. Kontakt zählt wahrscheinlich Email, Telefon? 00:01:12-9

I: Ja. 00:01:15-5

B: Ok, dann täglich. 00:01:16-5

I: Und eben, das hast du eh gesagt, Telefon und Email. Chat und so auch? 00:01:21-2

B: Nein. Wir sind nicht per Chat miteinander verbunden, das ist nicht so wie intern dass wir diese Chatmöglichkeit über Notes haben. Das haben wir mit den Kollegen von **Kunde_1** nicht, nein. 00:01:37-9

I: Und mit den Entwicklern bist du wahrscheinlich auch über Telefon, Email, Chat... 00:01:42-5

B: Nein, da meistens persönlich. Die sitzen ja nun gleich.. - also ausser jetzt ihr in **Österreich!** Mit den **Österreichern** entweder per Chat oder per Telefon dann genau. Oder per Email dann auch. 00:02:00-7

I: Bist du mit der Testabteilung dann auch irgendwie.. 00:02:06-8

B: Ja. Da bin ich auch.. also grösstenteils da, aber jetzt momentan im Bereich KFZ bin ich da mit einem Kollegen immer.. Also wir tauschen uns relativ.. 00:02:18-2

I: Mit **TST_1**? 00:02:18-2

B: Ja, genau. Also wir tauschen uns relativ häufig aus. 00:02:23-3

I: Was ist häufig? 00:02:24-4

B: Also auch mehrmals die Woche. Nicht täglich unbedingt, aber mehrmals die Woche auf jeden Fall. Es kommt immer darauf an, wie so der Arbeitsaufwand ist, aber.. 00:02:32-5

I: Und trifft euch ihr dann direkt im Büro, oder.. 00:02:35-0

B: Telefon, Chat, oder auch persönlich, ja. 00:02:39-9

I: Ok. Ja, welche Programme eben verwendest du für deine Arbeit? Also zur Kommunikation und auch so zum Arbeiten. 00:02:48-3

B: Also zur Kommunikation Lotus Notes, und ansonsten verwende ich hauptsächlich noch Word - wo auch die Fachkonzepte darin geschrieben werden - sowie auch Excel, wo die Testlisten drin geschrieben werden. Was verwende ich noch? Und dann entsprechend das JIRA, ja. 00:03:09-3

I: Die Testlisten schreibst auch du? 00:03:10-0

B: Ja. 00:03:11-9

I: Ok. Wieviele Leute arbeiten zum Beispiel gemeinsam an einem Fachkonzept? Machst du das immer alleine, oder seid ihr da zu mehr, oder... 00:03:22-4

B: Das kommt darauf an. Also grundsätzlich, bei der Ersterstellung, bei der Initiierung des Fachkonzeptes da wird.. das schreibt dann einer meistens. Ganz selten dass da wirklich mal mehrere Leute daran arbeiten. Aber so im Laufe des Fortschrittes des Fachkonzeptes, oder auch wenn sich das über mehrere Meilensteine hinzieht, dann wechselt das natürlich, dann schreibt auch manchmal ein anderer da in dem Fachkonzept (unv. noch mehr). Aber meistens ist das wirklich so, dass einer das geschrieben hat. Und, ja. Also dass da nicht mehrere daran arbeiten. 00:03:52-5

I: Ok. Falls jetzt jemand anderes etwas daran ändert, wie erfährst du da darüber, oder.. 00:04:02-1

B: Falls jemand anders was ändert? Jetzt am Fachkonzept? 00:04:01-9

I: Am Fachkonzept, ja. Oder kommt das vor? 00:04:05-9

B: Also.. sicherlich kommt das vor. Aber das erfahre ich dann meist nicht, ausser ich schaue mal selbst ins Fachkonzept rein. Ansonsten ist das sehr selten dass man es mitbekommt dann. 00:04:17-6

I: Ja, weisst du zum Beispiel was deine Kollegen, also die anderen Fachkonzepter woran die arbeiten, oder spielt das für dich eine Rolle? 00:04:30-8

B: Teilweise schon, also gerade der Produkt- und der Bestandsbereich arbeiten relativ eng zusammen. Und da ist es natürlich schon wichtig was die entsprechend machen. Oder halt gegenseitig zu wissen was der andere macht. Provisionsbereich, mit denen arbeitet man auch zusammen, ist aber auch eher selten dass man sich da austauscht. Also das ist meistens wenn es zu einem Problem kommt. Und genau das gleiche kann man auch zum Bereich Finanzen sagen, also das redet man eigentlich auch.. Also Kommunikation ist da, aber eingeschränkt, sagen wir es mal so. Es ist halt immer nur dann, wenn es zu Problemen kommt, dann redet man miteinander. 00:05:13-0

I: Ok. Ja wie erfährst du zum Beispiel darüber was die anderen machen, ist das einfach.. durch das tägliche Gespräch, oder.. Eben, da wäre jetzt zum Beispiel dann die Frage ob du im JIRA irgendwie nachsiehst, woran die gerade arbeiten, oder welchen Task die gerade in Arbeit haben, oder so. 00:05:29-0

B: Ok, also das mache ich nicht. Also ich habe.. JIRA nutze ich tatsächlich nur für mich persönlich. Also ich kucke nicht nach wer wo daran arbeitet. Da muss ich auch sagen das interessiert mich dann in dem Moment nicht wirklich. Wenn ich das wissen will, dann kann ich ja auch hingehen zu demjenigen und frage ihn einfach. Also, dafür nutze ich das JIRA definitiv nicht. Also meistens wirklich die persönliche Kommunikation, und ja, dass man darüber im Endeffekt erfährt wo.. wo dran der Kollege arbeitet. 00:05:57-2

I: Sitzt du mit deinen, mit den Leuten im selben Büro, oder.. 00:06:02-5

B: Nein. Also im Moment sitze ich alleine im Büro. 00:06:03-8

I: Aber ihr seid alle im selben Stock, oder? 00:06:06-1

B: Ja. 00:06:08-0

I: Und weisst du zum Beispiel wie stark deine Kollegen immer ausgelastet sind? Es gibt ja irgendwie Zeiten wo man sehr viel zu tun hat, Zeiten wo es ein bisschen ruhiger ist.. 00:06:23-4

B: Kann ich nicht sagen, nein. 00:06:22-0

I: Kannst du nicht sagen. Spielt aber auch keine Rolle dann für dich? 00:06:25-1

B: Grundsätzlich nicht, nein. Also.. es ist halt manchmal so, dass der eine mehr zu tun hat, der andere weniger, das bleibt dann nicht aus. Also bei Projektarbeit ist das halt so. Aber, weiss ich nicht wann genau das so ist.. 00:06:36-2

I: Und bist du zum Beispiel, weisst du was jetzt zum Beispiel die Entwicklerkollegen, woran die gerade arbeiten? 00:06:43-8

B: Also konkret nicht. So ganz grob weiss ich das natürlich, aber ganz konkret woran sie dann arbeiten nicht, nein. 00:06:52-3

I: Ok. Und wie erfährst du darüber zum Beispiel? 00:06:54-5

B: (lacht) Auch nur über, im Endeffekt über das Gespräch, und.. 00:06:59-7

I: Telefon. 00:06:59-7

B: Es ist halt auch sehr unterschiedlich. Das Problem ist halt, ich arbeite halt an sehr vielen Themen gleichzeitig, das heisst ich habe auch mit sehr vielen Leuten dann zu tun, und meistens ist es wirklich erst im Gespräch dass man halt, keine Ahnung, wenn QS für ein Fachkonzept gemacht werden soll oder so, dann weiss man halt wo dran der Kollege jetzt arbeitet. Oder aufgrund eines Tickets oder sonstiges. Dann, nur dann, glaube ich.. oder ist das meist so dass ich dann mitbekomme woran die wirklich konkret arbeiten. Ansonsten eigentlich auch nicht, nein. 00:07:35-3

I: Ok. Wieviele JIRA-Tasks hast du irgendwie ungefähr in (unv. Aufwand), bei wievielen Projekten bist du da drinnen? 00:07:54-0

B: Also momentan (...), wieviele sind denn das? Ganz grob würde ich vielleicht mal auf 10, 15 schätzen.. 00:08:06-3

I: Ok. Und bist du täglich da im JIRA unterwegs? 00:08:10-8

B: Ja. 00:08:13-5

I: Dann eine kurze Frage zu Sametime: wieviele Leute hast du da drin, oder... 00:08:19-1

B: Alle. 00:08:20-4

I: Alle? 00:08:20-4

B: Also ich habe mir keine Gruppen oder so angelegt. Ich habe mir alle die online sind.. kann ich sehen dann. Ich habe das nicht so gruppiert. 00:08:32-1

I: Und, ich weiss nicht, wenn du jemanden anrufen willst - schaust du dann vorher ins Sametime rein ob der jetzt online ist? Oder wie verwendest du das? 00:08:38-0

B: Ja, eigentlich schon, ja. 00:08:44-6

I: Das Ticketsystem wirst du eher wenig verwenden, oder? 00:08:48-9

B: Nein, eigentlich nicht. Ich verwende es eigentlich relativ häufig. 00:08:52-4

I: Schon? 00:08:52-4

B: Ja ja. Das Ticketsystem.. 00:08:55-4

I: Und von wo kriegst du deine Tickets, oder.. die laufen dann von Test und Entwicklern zurück? 00:09:01-1

B: Genau. Entweder von Test oder Entwicklung zurück. Und jetzt bekomme ich die halt, oder nehme ich sie mir einfach manchmal auch, gerade im Bereich KFZ, weil ich ja da momentan mehr oder weniger die fachliche Verantwortung habe. Da nehme ich mir die Tickets dann und bearbeite die dann auch. Also von daher.. ich habe da auch relativ viel mit zu tun. 00:09:24-3

I: Ok. Wie findest du da die Unterschiede - für deine Arbeitsorganisation - Unterschied zwischen Ticketsystem und JIRA dann? Weil im JIRA hast du ja auch deine Tasks - was findest du da einfacher oder besser um die Übersicht zu behalten zum Beispiel? 00:09:43-2

B: Kann ich jetzt gar nicht so sagen, also.. Ich finde beide Systeme nicht so optimal, würde ich sagen. Sowohl das eine als auch das andere ist ja.. eher suboptimal meiner Meinung nach, aber.. Also JIRA hat für mich den Nachteil man sucht häufig, oder manchmal muss man halt seinen Task suchen. Das ist, wenn der nicht zugewiesen ist, dann... Und die Suche ist halt... sicherlich kann man wohl ganz gut da drin suchen, aber da habe ich mich einfach überhaupt nicht mit beschäftigt muss ich sagen. Es ist für mich einfach nur ein Arbeitswerkzeug was funktionieren muss, und sicherlich kann ich es auch so konfigurieren, aber ich hatte bisher ehrlich gesagt nicht die Lust dazu mich da weiter damit auseinander zu setzen. 00:10:30-1

(JIRA-Login) 00:11:48-0

I: Ok, also das ist dein Standard-Dashboard wenn du reinkommst.. Das hast du selbst konfiguriert, oder..

00:12:05-3

B: Nein, das ist so eine Standardvorlage die man hier über diese Gadgets findet. Oder, nein, ich weiss gar nicht wo man die her findet (klick). Ich glaube hier wo findet man die wo, bei "Dashboard neu einrichten" oder sowas. 00:12:20-2

I: Und das ist alles was dir zugewiesen ist? 00:12:22-6

B: Genau, alles was mir zugewiesen ist. 00:12:23-8

I: Und das ist auch die einzige Ansicht die du verwendest? 00:12:25-3

B: Das ist die einzige Ansicht die ich verwende, genau. 00:12:30-9

I: Ok. Und welchen Task hast du jetzt zum Beispiel momentan in Arbeit? 00:12:39-4

B: Also eigentlich alle (schmunzelt). Das Problem ist, ich arbeite wahrscheinlich nicht so wie ein Entwickler damit muss man dazu sagen. Das Problem ist, ich habe sehr viele Fachkonzepte die einfach immer nur angepasst werden müssen. Also die werden pro Meilenstein, wenn da irgendwelche Fehler auftreten - das weisst du ja noch von KFZ sicherlich - dann werden die einfach nur nochmal angepasst, oder wird halt geschaut was falsch ist oder so. Und dadurch ist jetzt auch so eine Masse an Tasks geworden. Und ich nehme mir den Task dann meistens - muss ich dann auch gestehen - und nehme den "in Arbeit" wenn ich eigentlich schon fertig bin damit. Nehme den dann also quasi kurz in Arbeit, schliesse den dann sofort wieder, und gebe dann das ganze entsprechend zur QS. 00:13:24-2

B: Also ich arbeite tatsächlich - oder wahrscheinlich ist das allgemein in der Fachkonzeption - wir arbeiten dann damit doch nicht soo (gedehnt) intensiv wie die Entwickler. Also ich glaube die Entwickler sind da sehr, die haben das sehr viel mehr, also arbeiten dann sehr viel mehr mit. Und auch anders als wir. Soweit ich das weiss. Also von daher.. ich nutze es tatsächlich wirklich nur um einen Überblick zu behalten was ich schon erledigt habe an Aufgaben und was jetzt noch offen ist. Also, ich kann jetzt ja sehen, ok, für dieses Fachkonzept oder für dieses Thema hier, da ist der Status gerade auf 'wartet', das heisst (klick) - da kann ich ja hier kurz reinkucken - ok, wartet gerade auf die Entwickler-QS. Also, jetzt weiss ich dass sie in der Entwickler-QS ist. 00:14:09-0

I: Und wann wird das jetzt zum Beispiel geschlossen? Jetzt, das heisst, 'Anbindung VWB', das wird dann irgendwann.. oder bleibt das immer geöffnet das Projekt? 00:14:19-1

B: Nein nein, das wird irgendwann geschlossen. Also sobald, die sind ja dann.. Es gibt ja ein - sage ich mal - den Haupttask und dann noch diese kleinen Untertasks, und die werden halt geschlossen wenn alles durch ist. Also jetzt gerade das Fachkonzept, der Task 'Fachkonzept' wird geschlossen, sobald die QS durch die Entwickler abgenommen worden ist, und auch der Kunde das entsprechend abgenommen hat. Dann werden die drei im Endeffekt auch geschlossen. 00:14:52-5

I: Ok. (...). Wie findest du.. Kurz zu den Bedienelementen und so: also welche von diesen Buttons da oben verwendest du jetzt um dich zu.. zu navigieren? 00:15:29-1

B: Von denen da oben? Keinen. 00:15:27-6

I: Gar keinen. Verwendest du zum Beispiel diese 'Überwachen'-Funktion, damit du... 00:15:31-6

B: Nein. 00:15:31-6

I: Auch nicht. Liest du die Kommentare oder irgendwas? 00:15:34-3

B: Ja, Kommentare füge ich tatsächlich ab und an mal ein. Aber auch eher selten. 00:15:39-5

I: Ok. Wenn jetzt zum Beispiel nur so kleine Änderungen zurückkommen. Wird da dann dafür immer.. - also es gibt da ja diesen Ablauf dass man die Sachen 'in Arbeit' nimmt, und danach muss eine QS gemacht werden oder so - wird das immer.. dieser Weg durchlaufen, oder gibt es auch Sachen, ich weiss nicht, wenn nur kleine Änderungen sind, also wird dieser Weg immer dann durchlaufen wenn eine neue Fachkonzeptversion herauskommt, oder gibt es auch Änderungen die gemacht werden, ohne dass jetzt der ganze Weg wieder durchlaufen wird? 00:16:16-8

B: Also grundsätzlich wird immer dieser Weg durchlaufen dann nochmal, ja. 00:16:21-4

I: Schon. Ok. Wenn du jetzt zum Beispiel eine neue Version machst - weisst du welcher Entwickler dafür

zuständig ist? 00:16:29-6

B: Nein. Das wird, also der QS-Task wird zentral meistens an den Teamleiter zugewiesen, und der verteilt das dann entsprechend auf. 00:16:39-7

I: Ok. Also das ist bei dir dann egal. 00:16:41-1

B: Also meistens weiss ich wer es bekommt, aber so ist halt hier der Workflow der so vorgeschrieben ist und.. 00:16:53-5

I: Ok. Ja gut. Jetzt hast du eigentlich eh schon alles gesagt was wichtig ist. Also eben andere Leute, hast du auch gesagt, schaust du dir nicht an. Was deine Kollegen zum Beispiel fabrizieren. Also JIRA hast du wirklich nur für deine eigenen Sachen (unv. taziert) 00:17:20-5

B: Also wie gesagt, man muss da wahrscheinlich tatsächlich unterscheiden zwischen Entwicklern und der Fachkonzeption. Ich wüsste nicht, dass das ein Kollege von mir - jetzt also aus der Fachkonzeption - dass er das anderweitig verwendet als ich.. also so grossartig. Also ich glaube, alle verwenden es tatsächlich wirklich nur um ihre eigenen Tasks und Aufgaben zu verwalten, und sich aber jetzt nicht weitergehend mit Aufgaben anderer Mitarbeiter und Kollegen zu beschäftigen. 00:17:43-6

I: Wie ist das bei der Teamleitung zum Beispiel bei euch, im Team? 00:17:46-0

B: Keine Ahnung.. also.. 00:17:47-5

I: Schaut dein Chef da drauf? 00:17:47-5

B: Ich nehme es mal an. Also ich hoffe mal dass er ab und zu mal drauf schaut, aber genau weiss ich das auch nicht. 00:17:57-2

I: Ok. Dann würde ich sagen.. Ich weiss nicht. Machst du dir zum Beispiel Bookmarks von bestimmten Sachen? 00:18:11-2

B: Nein. 00:18:11-7

I: Gar nichts. 00:18:11-7

B: (lacht) Ich bin da wahrscheinlich.. wir sind da wahrscheinlich so ein bisschen vielleicht keine gute Auskunftquelle was das angeht. 00:18:25-6

I: Nein, das ist eh ok, das ist eh interessant. Ich will eh alle Seiten irgendwie beleuchten. 00:18:30-0

B: Also ich glaube das Problem ist tatsächlich dass das noch nicht so richtig.. also dass das noch nicht so richtig bei uns (Anm.: BATTERIE des Aufnahmegerätes leer!!) 00:18:36-4

(Fortsetzung aus Gedankenprotokoll: "... das Problem ist tatsächlich, dass es bei uns noch nicht so richtig angekommen ist." Seine restliche kurze Anmerkung konnte nach Ende des Interviews nicht mehr sinngemäss rekonstruiert werden.)

Interview 08_PMM_1

I: Also, du hast ja früher für KFZ gearbeitet, von daher hast ja ein bisschen Ahnung.. 00:00:11-3

B: Ja, das stimmt. Aber die haben ja ein neues JIRA-Projekt jetzt, ein Neues, was sie aufgesetzt haben.. Das habens sie ja gewechselt.. 00:00:17-6

I: Das hat es damals noch nicht gegeben.. 00:00:16-0

B: ... das ist jetzt deutlich umfangreicher, und vor allen Dingen auch team-übergreifender als vorher. Vorher waren es zwei verschiedene JIRA-Projekte die nebeneinander waren, und kommuniziert haben dadurch, dass Aufgaben an den Teamleiter des anderen zugewiesen wurde, und der hat dann eine eigene Task in seinem JIRA wieder gemacht. Das ist jetzt geändert und umfangreicher. 00:00:38-5

I: Jetzt gibt es dieses **LFS**-Projekt. 00:00:40-8

B: Genau, dieses **LFS**-Projekt. 00:00:39-7

I: Seid ihr da auch drin, von Produktmanagement? 00:00:44-6

B: Sind wir auch mit involviert, aber leider nicht von Anfang an. Sondern erst später wenn die Fachkonzepte fertig sind. Also wir quittieren im Prinzip dass die Fachkonzepte abgenommen sind. Weil Produktmanagement ist ja vom Prozess her Auftraggeber für die Entwicklungs.. 00:01:01-1

I: ... Fachkonzeption. 00:01:03-2

B: UND für die Entwicklung. Und die fertigen Fachkonzepte werden im Prinzip von uns quittiert. Das ist aber alles. Es ist nicht so, dass die Auftragsvergabe auch über dieses JIRA erfolgt. 00:01:19-9

I: Was ist die Auftragsvergabe? 00:01:18-6

B: Also, dass wir zu den Konzeptern sagen "erstelle jetzt für dieses Thema ein Konzept". Das erfolgt noch über einen anderen Weg. Initialbefüllung.. also das fängt.. den Auftrag erteilen wir mehr oder weniger "erstelle ein Konzept". Und dann legt der Teamleiter in der Konzeption, sprich **FK_TL** oder **FK_TL_2**, die legen dann den entsprechenden Task an. Mit allem was im Prinzip darunter hängt an Aufgaben. 00:01:46-2

I: Wie ist das zum Beispiel bei dir, mit welchen Leuten bist du da in Kontakt in deiner täglichen Arbeit? 00:01:57-2

B: Aus meiner Rolle jetzt als Produktmanager heraus? 00:01:58-5

I: Genau. Als Produktmanager - mit deinen eigenen Leuten jetzt, und... Also jetzt einmal unabhängig von JIRA. 00:02:06-2

B: Unabhängig von JIRA natürlich mit den anderen Produktmanagern aus den anderen Bereichen. Und dann bin ich für den Bereich Komposit zuständig, und so für.. mit **FK_TL** als Teamleiter für die Konzeption, mit **ENTW_DE_TL** als Teamleiter für die Entwicklung. 00:02:20-5

I: Ok. Und welche Tools verwendest du für die Kommunikation mit diesen Leuten? 00:02:25-8

B: Telefon und Email. Beziehungsweise die persönliche Kommunikation. Also ansonsten benutzen wir da keine Toolunterstützung. 00:02:37-7

I: Ok. Ja weil wir vorhin auch das KFZ-Projekt mit **Österreich** angesprochen haben: wie wichtig findest du es, dass man da weiss was die machen, oder was da abgeht? 00:02:50-9

B: Du meinst was jetzt in **Österreich**.. oder das.. 00:02:56-5

I: Also in der Zusammenarbeit im Projekt, dass da verschiedene Abteilungen sind.. 00:03:01-4

B: Ja, also prinzipiell ist es schon immer ein Problem wenn in gewissen.. oder wenn in Projekten unterschiedliche Standorte da sind. Denn je weiter das weg ist, desto grösser wird das Problem. Das wächst also im Prinzip mit den Kilometern. Weil einfach die Kommunikation immer weniger wird. Und von daher ist es schon immer extrem wichtig zu wissen, wer arbeitet wo dran und welchen Stand hat derjenige gerade. Das ist sowohl für die anderen

Teammitglieder wichtig, als auch natürlich für den entsprechenden Verantwortlichen für die Aufgabe extrem wichtig. [00:03:54-0](#)

I: Aber wie ist das.. Schaust du zum Beispiel, ich weiss nicht, was **FK_1** gerade macht oder so? Wenn du.. [00:04:01-5](#)

B: Nein, weil im Moment ist es so, dass wir als Produktmanager erst einmal nur definieren: wie stellen wir uns dieses Produkt vor dass da gebaut wird. [00:04:11-6](#)

I: Das macht ihr mit dem Kunden gemeinsam? [00:04:10-5](#)

B: Die Konzeption ist erst einmal nur für das Customizing zuständig. Und wir geben dann natürlich - wenn es dann heisst "ist das ein Standard oder nicht" - unseren Kommentar dazu. Beziehungsweise wir sind dafür zuständig, dass wir sagen wir möchten dass in diesem Meilenstein diese Aufgaben erledigt werden. Was weiss ich, wir möchten dass in Meilenstein XY irgendeine Stufungsgeschichte für KFZ umgesetzt wird. Dann geben wir den entsprechenden Auftrag da rein, und kontrollieren, ob das dann entsprechend das Ergebnis irgendwann vorliegt. Mit dem Konzepter selber haben wir da erst einmal nur in Ausnahmefällen zu tun. Der gibt uns natürlich schon zwischendurch mal Feedback, aber das läuft im Normalfall über den Teamleiter, der dafür zuständig ist dass er in seinem Produktionsabschnitt entsprechend die Planung aufsetzt. Also er kriegt im Prinzip am Anfang eines Meilensteines einen Sack voll Themen.. die priorisiert sind, und sagt dann "ich habe so-und-so viele Ressourcen, und ich habe einen so-und-so langen Zeitraum, daher ergibt sich nach der Prio diese und jene Machbarkeit für diese und jene Themen daraus. Und daraus setzt er dann die Leute ein - das ist aber die Aufgabe des Teamleiters - entsprechend der Konzeption. Oder auch danach in der Entwicklung. [00:05:27-6](#)

I: Weissst du zum Beispiel welche Leute jetzt die Fachkonzepte umsetzen, oder die Fachkonzepte schreiben, wo du das Produkt machst? [00:05:35-8](#)

B: Nochmal? [00:05:36-8](#)

I: Weiss du zum Beispiel selber jetzt, für, angenommen du machst jetzt ein neues Produkt - zum Beispiel Glas oder so - weisst du wer dann bei der Fachkonzeption zuständig ist für das Schreiben von dem? [00:05:44-7](#)

B: Nicht unbedingt. [00:05:48-7](#)

I: Das spielt aber auch keine Rolle für dich. [00:05:51-4](#)

B: Der Verantwortliche dafür ist nach unserem derzeitigen Prozess der entsprechende Teamleiter dafür. [00:05:57-6](#)

I: Ok. [00:05:57-4](#)

B: Das wäre anders, wenn man es jetzt als Projekt aufsetzen würde, und sagen würde ich (unv. nehme jetzt den Hut) auf dafür, dann wäre das eine ganz andere Geschichte. Aber derzeit ist es eben so, dass dafür verantwortlich ist der Teamleiter der gesagt hat am Anfang "ich mache das". Und der ist auch dafür zuständig dass die Planung im Prinzip bei ihm so funktioniert. Er gibt ja am Anfang eine Machbarkeit ab, und muss auch entsprechende Risiken melden und so weiter wenn das nicht klappt. Oder wir müssen dann entsprechend sagen "ok, wir sind in Zeitverzug, der Aufwand hier erhöht sich, dafür kippt hinten das andere Thema runter" - das muss dann natürlich schon mit uns abstimmen. Aber es ist nicht so, dass es der einzelne Konzepter machen muss. Das läuft immer eigentlich über die Teamleiter. [00:06:31-2](#)

I: Ok. Und hast du zum Beispiel den Teamleiter in deiner Sametime-Kontaktliste? **FK_TL?** [00:06:38-2](#)

B: Den habe ich da mit drin, ja, klar. Aber ich persönlich bevorzuge da so mehr das persönliche Gespräch. Dass ich also schon entweder nachfrage "wie ist der Stand der Dinge, seid ihr noch d'accord?", oder aber wenn er merkt dass es nicht hinhaut, dass ich schon erwarte dass er mich unverzüglich benachrichtigt. Weil wir dem Kunden gegenüber natürlich auch in der Pflicht sind und sagen müssen "pass auf, es gibt da eine Verschiebung". Oder wir kommen hinten raus, oder wie auch immer. Das ist dann.. [00:07:07-9](#)

I: Und hast du zum Beispiel, mit **FK_TL** jetzt, fixe Treffen oder so? [00:07:13-9](#)

B: Nein, keine fixen Treffen. Es gibt im Rahmen des Meilensteinprozesses natürlich fixe Geschichten, aber während der Meilenstein läuft gibt es keine fixen Geschichten. Sondern da.. [00:07:23-9](#)

I: Ok. Aber wie oft triffst du ihn dann zum Beispiel? [00:07:24-0](#)

B: Relativ häufig. Man kommt auch mit anderen Sachen durchaus ins Gespräch. Also alle zwei, drei Tage auf jeden Fall. 00:07:35-3

I: Und würdest du sagen das sind dann quasi eher informelle Sachen, also dass du.. oder.. Also du machst nicht extra einen Termin mit ihm aus, sondern du gehst runter zu ihm.. 00:07:47-5

B: Genau. Im Prinzip gibt es natürlich.. sind die Teamleiter sowieso verpflichtet, Risiken zu melden. Und zwar unverzüglich. Das heisst also, im Prinzip liegt die Aufgabe schon bei ihm, aber man trifft sich natürlich trotzdem auch über allgemeine Sachen, beziehungsweise sind dann auch fachliche Fragen eventuell noch zu klären wo wir dann natürlich schon mit im Boot sind. Oder wenn wir sagen "ok, wir wissen es kommt jetzt irgendwann dieses und jenes Thema, bereite dich schon mal darauf vor". Oder "es gibt zusätzliche Änderungsanforderungen die noch bewältigt werden müssen, schaffst du das noch zusätzlich, und wenn du das machst was gibt.." - also es gibt X Möglichkeiten wo man sowieso in der Kommunikation ist. Und von da her, im Normalfall, spätestens alle zwei, drei Tage, haben wir sowieso miteinander zu tun. 00:08:29-1

I: Ok. Und mit deinen eigenen Leuten hier, also mit den anderen Produktmanagern? 00:08:36-6

B: Mit den anderen Produktmanagern - es gibt auch noch **PMM_2**, die macht auch den Bereich Komposit mit - wir sitzen im selben Büro, haben also täglich miteinander zu tun. 00:08:45-1

B: Und ansonsten gibt es zu übergreifende Themen regelmässige Meetings einmal die Woche, wo im Prinzip jeder dann entsprechend informiert wird, oder wo auch Sachen die jetzt alle Bereiche betreffen entschieden werden. Das gibt es wöchentlich. 00:09:02-4

I: Was zum Beispiel in der Entwicklungsabteilung passiert, spielt das für dich jetzt noch irgendwie eine Rolle? 00:09:09-2

B: Ja klar. Weil wir sind auch Auftraggeber für die Entwicklung. Spielt also die gleiche Rolle wie die Konzeption. Ist genau das gleiche in Grün. 00:09:20-9

I: Und mit Test habt ihr da auch irgendwie Kontakt? 00:09:24-0

B: Auch Test sind wir natürlich mit dran, auch da haben wir natürlich Kontakt zu, weil für uns natürlich schon auch interessant ist: wann sind welche Liefertermine, weil das entscheidet sich natürlich alles.. hängt nun beim Testbereich. Wo wir.. also kriegen wir alles.. wir kriegen alle Entwicklungsschritte mit. 00:09:41-2

I: Ok. Und wahrscheinlich dann auch über die Teamleiter? 00:09:46-1

B: Im Normalfall ja. Bei den Testern, da gibt es dann Tester.. da gibt es meistens nur einen Tester der für einen Bereich zuständig ist, also da kann man dann auch zu dem entsprechenden Tester hingehen. Bei den Konzeptern und den Entwicklern ist das natürlich ein bisschen anders. Weil da sind es halt ein ganzer Bereich, da gehen wir zu **ENTW_DE_TL** oder zu **FK_TL**. 00:10:03-6

I: Schaut ihr zum Beispiel öfter ins Ticketsystem oder so rein? 00:10:07-2

B: Auch, ja. Wir kucken also schon was ist da an Tickets, beziehungsweise es laufen ja natürlich auch Änderungsanforderungen über das Ticketsystem, auch da werden wir natürlich schon darüber informiert. 00:10:18-5

I: Wie sieht deine Sametime-Kontaktliste aus, wieviele Leute hast du da drin stehen, und wie hast du sie organisiert? 00:10:32-0

B: Wieviele Leute kann ich dir jetzt gar nicht sagen, aber ich habe so ziemlich genau die rechte Bildschirmhälfte voll. Also die ist eingeblendet als schmaler Balken am Rand.. 00:10:42-1

I: Aber sind das eher die Leute von deinem Team, oder die anderen Teamleiter, oder... 00:10:45-4

B: Da sind auch die Leute von meinem Team, die sind.. allerdings spielen die hier in diesem Bereich - weil es ist ja nur dieser abgeschlossene Bereich hier oben - eher eine untergeordnete Rolle. Ich habe durchaus ausgewählte Entwickler im Bereich Komposit mit drin - also zum Beispiel **ENTW_DE_TL_STV_2**, **ENTW_DE_TL_STV_1**, **ENTW_DE_TL** ist da mit drin - also schon die wo ich meine durchaus Kontakt zu haben, wo ich einfach sehen muss ob sie da sind oder nicht. Das gleiche betrifft den Bereich Konzept, und hier sind dann noch.. im Verwaltungsbereich habe ich noch einige mit drin, also die Vorstände sind für uns durchaus noch interessant, und auch die sind in der Kontaktliste mit drin. 00:11:25-2

(JIRA-Login) [00:11:59-6](#)

I: Also das ist dein Standard-Dashboard, oder wie? [00:11:58-0](#)

B: Das ist mein eigenes Dashboard. Da sieht man auch im Prinzip, dass es für uns für das Produktmanagement relativ uninteressant ist was hier passiert. Ich habe nämlich keine einzige Task die mir zugewiesen ist. Weil der Schwerpunkt findet natürlich schon in den Bereichen der Konzeption und der Entwicklung statt. Wir haben hier zwar auch eigene Dashboards, aber die sind ersteinmal relativ uninteressant, weil JIRA da nicht unbedingt die Aussagefähigkeit hat die wir hier bräuchten. Sage ich mal so. Also es gibt dann.. Eine Kollegin hier gegenüber hat mal versucht solche zweidimensionalen Geschichten hier mit aufzubauen die schon.. um zu sehen finden wir da irgendetwas das etwas für uns etwas über den Zustand der Arbeit aussagt. Also es ist im Prinzip immer interessant natürlich Informationen zu kriegen - was weiss ich: [00:12:55-2](#)

I: "ich habe so-und-so viele Fachkonzepte", [00:12:57-7](#)

B: "wir kommen gegen das Ende der Fachkonzeptphase, wie ist da der Stand der Dinge, wieviele sind noch offen?". Weil offen würde ja erst mal heissen ich habe doch damit noch gar nicht angefangen und habe aber nur noch eine Woche Zeit, und wieviele Abnahmen gibt es [00:13:08-8](#)

B: Aber man sieht hier auch schon - wenn ich mal die Anzahlen hier sehe - das unterscheidet sich hier. Merkwürdigerweise. Und damit ist die Aussagekraft relativ gering. Also im Prinzip taugt sie nichts mehr. [00:13:22-2](#)

B: Auch hier, wenn man hier reingeht sieht man da nur: es sind diverse Aufträge noch nicht zugeordnet - aber warum steht dann eben nicht mehr. Und damit ist die JIRA-Geschichte für UNS relativ uninteressant hier. Wir haben sie zwar drin weil wir da auch irgendwelche Abnahmen machen. Wie hier, wo wir also im Prinzip sagen... Die Konzepte sagen "wir haben alles fertig, wir haben das Konzept fertig, wir haben die Testliste fertig, wir haben die Kundenabnahme und die Entwickler-QS alles fertig", und wir dann im Prinzip sagen "ok, wir haben hier keine Auflagen mehr dran" weil alle Termine sind gehalten, oder es ist irgendwann fertig, dann nicken wir das wirklich auch ab. aber mehr haben wir damit hier GAR nicht zu tun. [00:14:02-1](#)

B: Es gibt dann noch einen übergreifenden hier, ich glaube der ist noch ein bisschen... oder sind das auch nur diese Dinger? Also hier sind dann zum Beispiel im Grunde... Also es ist nicht so richtig.. nicht so richtig mit (unv. wo wir) hier im JIRA involviert sind. Das liegt ein bisschen auch da mit dran, dass die Aufträge aus einer anderen Ecke generiert werden. Das war vorher schon so, und das Produktmanagement also nicht als Startpunkt im JIRA mit drin ist. Obwohl es vom Prozess her der Startpunkt ist. Es ist also im Prinzip der Auftraggeber für die einzelnen Bereiche, die Aufträge werden aber nicht hier drüber... von uns aus generiert, sondern erst einen Schritt später. Warum das so ist - frag mich nicht. Keine Ahnung. Da bin ich wirklich überfragt. Also wenn an es so verwenden wollte, müsste man im Prinzip den Prozess hier aufsetzen. Dann kann man auch hier wirklich sagen "ok,..." [00:14:59-0](#)

I: Ihr macht eure Geschichten und gebt es weiter an Fachkonzept. [00:15:03-3](#)

B: Genau. Weil wir sind hier die erste Stufe und müssen es dann wieder abnehmen, und dann auch sagen "ok, und jetzt ist der nächste dran" wenn wir die Abnahme gemacht haben. Und wir müssten es hier einrichten, das wäre der sinnvollere Weg. Da gibt es noch so ein paar Schwierigkeiten wahrscheinlich mit dem wie die Aufträge momentan... [00:15:21-0](#)

I: Wie wird das momentan verwaltet? [00:15:19-9](#)

B: Da gibt es eine Notes-Datenbank. Das war vorher auch schon so, und (...) im einzelnen kann ich dir das gar nicht beantworten. Ich weiss, dass es da eine sogenannte FU-DB gibt, eine Funktionsumfangs-Datenbank, und es gibt „**AUFTRAGS-TOOL**“. Das ist so eine Auftragsdatenbank, und da werden nachher auch die Arbeitszeiten - die gebucht worden sind von den einzelnen Leuten - gegengerechnet. Was hier natürlich alles nicht der Fall ist. Und das Zusammenspiel hat man bei der Erstellung von dieser **LFS**-JIRA-Datenbank nicht berücksichtigt. [00:15:51-0](#)

B: Das heisst also, diese **LFS**-Geschichte ist erst an der Konzeption aufgesetzt worden. Mit dem Ziel, die beiden vorhergehenden Datenbanken die da waren - sprich für Komposit waren es dann die KKZ und die KMP, und für Leben gab es auch noch eine LBN oder irgend sowas - diesen Medienbruch wegzukriegen und zu sagen "wir machen jetzt nur noch eine und es muss nicht jeder irgendwie irgendwelche Tasks wieder anlegen wenn ich irgendetwas zugewiesen kriege", das war auch durchaus sinnvoll. In meinen Augen hätte man einen Schritt weitergehen müssen und sagen "ok, wir fangen jetzt vorne an". Erst dann hätte es wirklich einen Sinn gemacht. Weil so könne wir jetzt hier zwar reinkucken und sagen "ja puh, wir sollen hier irgendetwas abnehmen", aber richtig helfen tut uns das Ding eigentlich nicht. Weil wir eben nicht mehr dafür verantwortlich sind was da jetzt

wirklich passiert. Dadurch dass es erst später aufgesetzt wird, wissen wir auch nicht mehr was drinsteht. Und richtig auswerten kann man es auch nicht. [00:16:49-0](#)

B: Also persönlich finde ich JIRA schon.. ist schon ein gutes Tool - ist ja im Prinzip nur ein Fehler-Tool erst einmal, auch um Aufgaben zuzuweisen - aber es wird hier teilweise für Sachen benutzt wofür es nicht unbedingt gut geeignet ist. Also es benutzen dann die Teamleiter.. ich weiss nicht, **ENTW_DE_TL** benutzt es um ganz viele kleine Aufgaben zu verteilen. Es gibt Teamleiter im Bereich 'Kranken' - die sind jetzt zwar bei Komposit nicht mit drin - die machen ihre Planung damit, das halte ich für völlig ungeeignet. Das sind aber Geschmackssachen natürlich, und von daher.. [00:17:27-3](#)

B: Ich hätte mir gewünscht dass man das macht.. dass man A) eine Hausübergreifende Geschichte macht, und im Prinzip alle Teile die irgendwie mit dran sind mitberücksichtigt. Das wäre sicherlich schöner gewesen. Und so gibt es jetzt viele Sachen die schwierig sind. [00:17:42-1](#)

B: So ist es zum Beispiel.. es gibt hier ja Auftragsnummern. Und die Auftragsnummern sind da allerdings nicht - man sieht das wenn man da hier reinkuckt, das sind immer die hier vorne: 'PD_KO_FU_2703.3' - die stehen in den neueren Aufträgen vorne nicht mehr mit drin. Die stehen nur noch in dem Ding wo man sich das selber ankuckt.. vorne mit drin. [00:18:05-8](#)

I: Ja, eben. Da steht da nur noch dieses '**LFS**'. [00:18:07-8](#)

B: Ja. Und dadurch dass das vorne nicht mehr mit drin steht werde ich relativ schwach, zum Beispiel was die Filterfunktion angeht. Dann suche ich einen Punkt 'Hausrat_TÄ/nTÄ' und kriege nachher irgendwann eine Liste von 40 Einträgen, weiss aber gar nicht mehr welcher für mich wichtig ist. Das entscheidende Kriterium ist eigentlich immer diese Geschichte. [00:18:31-2](#)

B: Das zweite ist, dass wir im Prinzip jetzt mit fortlaufenden Nummern hier im Produktmanagement handhaben müssen. Das heisst also ein Meilenstein zu Ende, Auftrag wird zugemacht, und wenn das Thema aber noch nicht zu Ende ist kriegen wir die nächste laufende Nummer. Und demzufolge kriegst du auch in den Suchergebnissen in JIRA immer weitere, immer mehr Masse, wo du im Prinzip nachher nur noch, immer nur noch übereinander siehst 'Hausrat TÄ/nTÄ' und musst dann irgendwie dir das zusammensuchen. Und das ist sehr unkomfortabel. Also da hat man sich hier relativ, also nach meinen Augen, ins eigene Knie geschossen. Weil man diese Filterfunktion eben nicht benutzen kann. Und so hätte man sagen müssen: "ich habe einen Auftrag 'PD_KO_FU_2703' - das ist 'Hausrat technische Änderungen / nicht technische Änderungen' ohne Unternummer - und der läuft eben solange bis das Ding erledigt ist. Und wenn ich eine zusätzliche Anforderung kriege, dann wird der eben wieder aufgemacht. Weil so kann ich damit wenig anfangen. Das ist dummerweise jetzt so gelaufen. Ob das beabsichtigt ist oder nicht kann ich nicht beurteilen, aber für mich ist es so, wenn ich jetzt ein Thema suche, und ich kucke nach "ich habe einen Auftrag" und will wissen "welche Konzeptversion habe ich und wie steht der Auftrag gerade" muss ich relativ lange suchen bis das hier finde. Ich weiss nicht wie es bei **FK_TL** und bei **ENTW_DE_TL** ist, aber hier ist es sehr schwer zu finden, weil man.. Also in meinen Augen, dadurch dass man die Auftragsnummern herausgenommen hat, oder verwässert hat, das war unschlau. [00:19:59-0](#)

B: Auch wenn ich jetzt anders suche, man kann da ja auch mit dieser Verschlagwortung suchen, über diese Komponenten die man hier so vergibt. Dann finde ich auch X Einträge nachher dafür. Also ich komm immer wieder auf eine Liste endend die ich dann irgendwo finde. [00:20:10-1](#)

B: Also was weiss ich, wenn ich jetzt 'Hausrat' kucke... (sucht und klickt) wobei... Gibt es da irgendwo etwas wo es schon nach der neuen (klick) Nummer wie es jetzt im Moment läuft, weil das ist jetzt alles... Hier läuft das zum Beispiel. Da ist hier schon so.. da sind auch überall diese Nummern drin... das ist 2011-3 (...) naja, finde ich jetzt nicht. Und von daher nutzen wir es hier relativ wenig. Das heisst wenn wir die jetzt abnehmen müssen, ja, puh, und müssen wir dann eben machen, aber ansonsten nutzen wir hier JIRA fast gar nicht. [00:20:50-4](#)

I: Ok. Wie kriegst du dann Bescheid über eine Abnahme die du zu machen hast? [00:20:54-7](#)

B: Im Prinzip, sie.. kann man sich natürlich schon hier mal reinkucken. Wir haben es als Punkt, wenn wir uns wöchentlich vom Produktmanagement treffen, dass wir in diese JIRA-Datenbank dann reinkucken, und dann sieht man ... [00:21:08-0](#)

I: Ihr seht dann alle gemeinsam rein, oder wie? [00:21:09-0](#)

B: Wir kucken alle einmal gemeinsam rein, und dann sieht man "ok, ich muss jetzt Abnahme machen". Dann gehts du eben rüber und machst das dann eben. Fertig. Aber es läuft nicht so, dass ich jetzt - so wie es ja sein sollte - ich kriege hier irgendwie eine Aufgabe zugewiesen und mache es dann, sondern das läuft hier anders. [00:21:27-0](#)

I: Also die Abnahme selber ist dann.. [00:21:29-4](#)

B: Ja also wir nehmen es dann hier im Prinzip ab. Ansonsten gibt es dann auch noch Abnahmen.. Abnahmesitzungen, wo dann im Prinzip wirklich die entsprechenden Teamleiter - je nachdem ob es jetzt ein Entwicklungszyklus ist oder ein Konzeptzyklus - wo sie uns die.. ihre Aufträge wirklich zurückgeben müssen. Wie weit sind sie? Erledigt? Dann können wir es abhaken, oder sie kriegen noch Auflagen. Also das findet immer am Ende des Meilensteines zu einem bestimmten Termin statt. Aber nicht mehr hierrüber (JIRA) dann. Die werden dann im Prinzip abgehakt, "ja, der ist fertig..." [00:21:54-6](#)

I: Und da habt ihr dann eine Liste ausgedruckt und sagt... [00:21:58-8](#)

B: Mehr oder weniger. Aber wie gesagt, dadurch dass es vorne nicht bei uns reinläuft können wir das auch hinten nicht verwenden. Weil wir eben gar nicht wissen was da im Detail drinsteht, und wer was warum da reinstellt. Ob es nun aus dem Pflegeprozess zum Beispiel ist - eine kleine Änderungsanforderung die vielleicht an uns vorbeiläuft weil sie für uns hier relativ irrelevant ist - da werden da auch Aufträge für angelegt, das kriegen wir natürlich gar nicht mit. Und von daher gehen wir diese Summe so gar nicht durch. [00:22:28-7](#)

I: Na gut. Und wie oft siehst du dann selbst ins JIRA rein? [00:22:35-1](#)

B: Puh, alle zwei Wochen einmal? Höchstens.. eigentlich.. wenn wir hier alle gemeinsam reinkucken. Also einmal die Woche. Aber selber suchen.. "ich gehe jetzt auf JIRA und kucke da rein" - so gut wie nie. Wie gesagt. Auch die Auftragsverfolgung nicht darüber läuft. Und auch wenn wir jetzt anfangen umzustellen, Planungstechnisch umzustellen, würde ich JIRA nicht als Mittel der Wahl nehmen. Sondern ich würde immer sagen ich habe eine Project(Anm.: sic! Englisch ausgesprochen)-Planung, die ist etwas gröber, und der entsprechende Teamleiter oder wie auch immer kann dann granular runterbrechen diese ganzen Geschichten, und dann dafür meinetwegen Arbeitspakete im JIRA definieren - das ist mir relativ egal. Aber für die übergeordnete Planung ist das Tool in meinen Augen nicht so zu gebrauchen. Und für den Arbeitsprozess schon, aber dummerweise ist diese **LFS**-Geschichte nicht auf den kompletten Arbeitsprozess umgesetzt, sondern nur auf einen Teil davon. Und damit ist es für den Teil der ausgegrenzt ist relativ wertlos. [00:23:45-6](#)

I: Das war jetzt sehr interessant, danke. Sehr aufschlussreich. [00:23:49-7](#)

B: Ja, bis jetzt hast du wenn du mit anderen Leuten gesprochen hast wahrscheinlich andere Meinungen gehört, von **FK_TL**, oder ich weiss nicht wen du schon gesprochen hast. [00:23:57-9](#)

I: Hm.. verschieden. Ja **FK_TL** war eigentlich schon recht.. findet es schon.. Es sehen schon auch irgendwie alle eine Menge Schwachpunkte die auch alle darin sehen... [00:24:06-7](#)

B: Also was ich zusätzlich noch als Schwachpunkt empfinde ist, dass es demjenigen der es dazu nutzt um Aufgaben zuzuweisen und Aufgaben wieder zurückzunehmen wenn sie wieder erledigt sind, dass es dazu führt dass die Leute nicht miteinander reden. Und in der Informatik ist es nach meinen Erfahrungen ist es.. der wichtigste Punkt ist dass eigentlich die Kommunikation.. die miteinander stattfinden muss. Und wenn die jetzt hier ersetzt wird durch Tool-Verwendung, dann hebelt man sich da ganz schnell mit aus. [00:24:40-9](#)

B: Das ist, auch solche Sachen wie dieses 'Neunzig-Prozent-Phänomen': ich melde immer 90 Prozent fertig, habe aber noch 80 Prozent der zeitlichen Aufwände offen.. oder irgendwie so. Also das Thema ist zu 90 Prozent erledigt, aber die letzten 10 Prozent kosten mich 90 Prozent der Zeit. Auch das kann ich damit fast nicht berücksichtigen. [00:25:00-3](#)

B: Weil ich kriege kaum mit - wenn ich da zum Beispiel auch die Zeiterfassung mit mache - wenn sich die Zeit erhöht. Weil für den Projektleiter ist immer wichtig - wie ist... Ich habe die Aufgabe irgendwo mal geschätzt, mit welcher Methode auch immer, aber ich habe im Prinzip eine Aufwandszahl, eine geschätzte. Mit der plane ich, und wenn ich merke dass eine Realisierung oder eine Umsetzung dass das nicht hinhaut, dann brauche ich entsprechend sofort ein Feedback dazu. Und auch da ist es gefährlich das so zu verwenden, weil das möglicherweise missverständlich, oder gar nicht, oder zu spät ankommt. Von daher ersetzt es niemals nicht das Reden miteinander. [00:25:41-0](#)

B: Und solche Verwendung von solchen Tools... Also ich bin da immer sehr skeptisch was diesen Einsatz angeht. Es fördert mehr dass jeder so sein Ding macht. [00:25:51-7](#)

B: Es hat echte Vorteile, weil jeder natürlich sehen kann "was mache ich", auch wenn derjenige - oder auch "was macht der" - wenn derjenige nicht da ist, WO ist irgendeine Aufgabe ist für jeden transparent, aber man muss immer genau aufpassen, und darf nicht das Ding für zu viele Sachen benutzen. Weil es ist auch immer noch nur ein Ticketsystem, und das darf man nicht vergessen. Also man muss da ein bisschen vorsichtig sein. Das ist mal meine persönliche Meinung dazu. Es ersetzt nicht, oder es ist.. es kann nicht dazu verwendet werden

ausschließlich irgendwelche Projektgeschichten zu steuern. Es kann es unterstützen, ja, aber es kann nicht dazu verwendet werden das ausschließlich zu machen. [00:26:27-0](#)

B: Wo es wirklich gut ist, sind jetzt solche Geschichten wie Teilaufgaben teamübergreifend zuzuordnen, und dann da auch zu sagen "ok, du kriegst jetzt hier einen Termin", "wird mir vorgesetzt".. Also ich kann auch nachweisen "ich habe es da reingesetzt", und dafür ist es wirklich gut. [00:26:42-2](#)

Aber in der Projektaufgabe ist es wieder schlecht. Beziehungsweise mit Vorsicht zu genießen. Man muss immer genau wissen wo man das Ding wirklich einsetzt. Also es ist nicht das Allheilmittel. [00:26:54-7](#)

I: Ja.. es sagen eh die meisten Leute dass es halt irgendwie noch nicht ganz optimal funktioniert. Also schon für gewisse Sachen, ich weiss nicht, **ENTW_DE_TL_STV_1** sagt schon dass es für diese ganzen QS-Geschichten nicht schlecht ist... [00:27:09-4](#)

B: Genau, da hat er auch recht.. [00:27:11-8](#)

I: ...weil dadurch halt wirklich die Wege eingehalten werden und man weiss wo das aktuelle Fachkonzept liegt zum Beispiel. [00:27:19-9](#)

B: Weil das dann auch die Email-Weitwurfgeschichte ablöst, und wenn man (unv.) man schickt das an irgendeinen Teamleiter, aber dann guckt da wieder keiner rein, weil der ist dann jetzt gerade mal krank, und dafür ist diese Geschichte wirklich gut. Das ist eben genau das was ich meinte: Aufgaben teamübergreifend irgendwo mit Termin zuzuweisen, dafür ist es wirklich gut, weil es auch sofort transparent ist. Aber es gibt eben auch Geschichten wo es.. wie gesagt, Produktmanagement ist nicht richtig mit beteiligt. Also wir könnten ja im Prinzip auch Aufgaben zuweisen damit, und es gibt mit **Prozess xxx** auch natürlich genaue Termine die irgendwo einzuhalten sind.. könnte man sicherlich auch bombig darüber abbilden, (unv. da bin ich involviert worden), und... Es ist nicht so einfach den ganzen Prozess damit abzubilden. Das ist schon relativ komplex, weil man muss sich auch vorher schon genau überlegen was will ich nachher damit machen. Sprich, also die Einrichtung des JIRA an sich, die Initialeinrichtung, die muss passen, denn ansonsten kann ich nachher damit nicht das auswerten was ich wirklich auswerten will. Und das ist sehr komplex wie die Erfahrung zeigt. Also ich habe es jetzt einmal bei KKZ habe ich es mit **FK_TL** zusammen gemacht, und da haben wir es nur innerhalb eines Teams gemacht - das war schon nicht ohne. Man muss sich über die Schlagworte Gedanken machen, man muss dafür sorgen dass bei irgendwelchen Tasks die nicht mit Schlagworten zugeordnet waren weil sonst finde ich nachher nichts mehr wieder. Das ist... [00:28:46-8](#)

I: Ja vor allem, da müsste man ja auch für jeden Subtask dann auch wieder die selben Schlagworte und alles... [00:28:54-1](#)

B: Genau. du musst also wirklich kucken wie verschlagworte ich die ganze Geschichte. Die Schlagworte am besten auch schon vorher geben, weil sonst hast du mit den Such- und Filterfunktionen echte Probleme. Beziehungsweise wertlos in dem Augenblick. Und es muss jedem klar sein, auch das zeigt die Erfahrung aus der Vergangenheit, es gibt ganz schnell unterschiedliche Wahrnehmungen wie mit was zu arbeiten ist. Wann ich zum Beispiel wo einen Anhang dranhängen muss, wann ich wo was.. auch da ist es, wenn es nicht ganz klar geregelt ist arbeiten die Leute unterschiedlich, und SOFORT ist das Tool wertlos. Weil ich eben nicht mehr reingehen kann und sagen:" wie ist jetzt der Stand der Dinge?" Weil wenn zwei Leute dran arbeiten, und der eine hängt mal da was dran, und der andere hängt es an einen anderen Task mit dran, dann habe ich verloren. Das ist einfach so. Das muss also ganz klar geregelt sein was wo mit umzugehen ist. Weil sonst klappt die ganze Geschichte nicht. Das hat man jetzt schon versucht, aber ich glaube man ist immer noch nicht so ganz.. [00:29:53-6](#)

I: Bei den Entwicklern haben sie es auch zum Teil durch die Rechtevergabe.. also es dürfen neue Tasks.. dürfen wirklich nur mehr **ENTW_DE_TL**, **ENTW_DE_TL_STV_2** und **ENTW_DE_TL_STV_1** machen. Und, ich weiss nicht, ... [00:30:06-3](#)

B: Die Entwickler arbeiten auch mit viel mehr Unteraufgaben als zum Beispiel die Konzepter. Also auch da ist die Arbeitsweise schon eine andere. Und von daher.. es ist so dass wenn man sagt, ok, wir machen da zum Beispiel die Konzepte mit, dass man.. die Konzepte hängen da drin und die Kundenabnahmen, das ist.. Leicht missverständlich ist dass viele Leute sagen "ich finde da auch alles", also sowas und dann sagt, ok, es wird ein Dokumentenmanagementsystem da draus, das ist es eben nicht. Und also da muss man wirklich aufpassen, dass man da nicht sagt.. dass man das zum Allheilmittel hier macht. Also es ist wirklich ein Ticketsystem das im Moment dazu dient, irgendwelche Teilaufgaben irgendwelchen Leuten zuzuweisen. Mehr sollte es auch nicht werden. [00:30:49-5](#)

B: Nichts mit irgendwelchen Ablagen für Dokumente und so weiter, und... Am besten macht man tatsächlich einen übergreifenden Prozess, also den kompletten Prozess auch wirklich komplett abbilden, und nicht nur einen Teil. So wie es jetzt ist. Das wäre auch ein bisschen besser. Aber ansich, als Hilfsmittel ist das schon durchaus,

durch die Flexibilität die es bietet, echt gut. [00:31:21-9](#)

I: Ja. Na eben. Ich weiss nicht.. Es hat jeder irgendwie ein paar Probleme damit, oder viele Probleme damit, aber... [00:31:30-9](#)

B: Ja, aber das liegt glaube ich tatsächlich daran, dass nie so richtig.. Ja, niemals nicht alle Beteiligten alle irgendwo mit-involviert gewesen sind. Dieses **LFS**-Projekt haben jetzt im Prinzip **FK_TL** und **FK_TL_2** in ihrem Büro mehr oder weniger aufgesetzt... Die Tester zum Beispiel sind auch nicht mit drin. [00:31:53-5](#)

I: Ja, aber die haben ja glaube ich wieder ihr eigenes Tool für die Test-Sachen. Die nehmen die JIRA-Sachen und schmeissen die ihr Test-Ding rein. [00:32:00-7](#)

B: Ja, aber warum? Wenn ich einen Auftrag reinkriege:"mache, was weiss ich, einen neuen Tarif für KFZ", da gehört dazu ein Konzept, da gehört die Umsetzung zu, und da gehört der Test zu. Und irgendwann ist alles fertig, und dann kann ich sagen: "so, ich bin jetzt soweit, jetzt kann ich das an die Kunden-(unv. Box) geben" Und wenn ich so ein Tool schon nutze, dann sollte man auch wirklich den kompletten Entwicklungsprozess mit allen Teilen machen. Sofern man ihn so macht wie man ihn hier jetzt macht. Sagt man wieder man macht den in Projektform, dann ist das möglicherweise sofort unbrauchbar. Weil sich vielleicht Konzepter, Entwickler und Tester in einem Raum gegenüber sitzen. Weil da, ob ich dann damit noch was anfangen kann ist die zweite Frage. Weil ich dann vielleicht auch ein anderes Vorgehensmodell habe. Weil jetzt bin ich ja nur.. das Wasserfallmodell was wir hier jetzt haben - es wird das Konzept gemacht, das muss komplett fertig sein bevor es in die Entwicklung geht - wenn die alle in einem Raum zusammensitzen machen sie vielleicht ein agileres Vorgehen, und dann kann ich das Ding sofort in die Tonne kloppen, weil ich das dann nicht mehr gebrauchen kann. [00:33:00-3](#)

I: Ja, aber das ist ja auch durch die Firmenstruktur so vorgegeben, oder? Du hast da die einzelnen Abteilungen.. [00:33:04-5](#)

B: Ja. Man wechselt.. man überlegt ja sowieso zum Jahreswechsel eine neue Struktur aufzubauen, Richtung Matrixstruktur. Viele Formen in Projektform abzubilden, und dann kann ich mit dem ganzen Ding nichts mehr anfangen. Dann ist das Ding sofort weg, dann kann ich es nur noch gebrauchen wenn ich sage "ich habe eine Linientätigkeit die noch genauso abläuft wie bisher". Aber ansonsten ist die ganze Arbeit, und auch alles was da drin abliegt auch, sofort wertlos. [00:33:33-0](#)

I: Aber diese neue Form, das würde dann heissen dass sich wirklich dann der Fachkonzepter, oder die Fachkonzepterin, zwei Entwickler und ein Tester wirklich in einem Raum sitzen? [00:33:40-1](#)

B: Nein. Man wird eine typische Matrixstruktur machen die dann im Prinzip heisst: ok, es gibt disziplinarische Vorgesetzte, wer auch immer das dann sein man, die haben eine Anzahl X an Mitarbeiter, und es gibt Aufgaben die in Projektform abgewickelt werden. Dafür gibt es einen Projektleiter. Und aus diesem entsprechenden dann Mitarbeiterpool Leute. Sprich, also die disziplinarischen Vorgesetzten geben die Leute, ihre Leute, ab in die Projekte. Können sie auch nicht mehr einplanen in dem Augenblick. Sondern die unterstehen fachlich voll dem Projektleiter. Nur der Projektleiter sagt was sie in diesem Augenblick zu machen haben. Der disziplinarische Vorgesetzte ist nur noch zuständig für Urlaub, Gleitzeit, Überstunden, also diese Regelungen. Beurteilungsgespräche, Gehaltsgespräche. Das ist natürlich erst mal relativ wenig für den, aber der disziplinarische Vorgesetzte kann natürlich trotzdem auch Mitglied in einem Projekt sein, oder sein eigenes Projekt machen, oder wie auch immer. Das ist die typische Matrixorganisation die man dann so hat. Aber dann sitzen im Normalfall natürlich Konzepter und Entwickler wahrscheinlich - weil muss ja nicht so sein, abhängig vom Vorgehensmodell - in einem Raum sich gegenüber, oder arbeiten gleichzeitig an einem Thema. Vielleicht kann man sie.. also zum Beispiel.. mit irgendwelchen Pilotgeschichten zum Beispiel arbeiten. Und dann ist aber das Fachkonzept nicht fertig, sondern.. Es wird dann erst ein Teil entwickelt, und dann kommt es.. Also andere Vorgehensmodelle kann das Ding so in der jetzigen Form nicht abbilden. Weil da im Prinzip, durch diese Unteraufgaben, ja schon vorgegeben ist wie zu arbeiten ist: Erst muss das Konzept fertig sein, dann muss es abgenommen werden vom Kunden und vom Entwickler, und dann fängt der Entwickler erst an. Das ist schwer hier das anders abzubilden. Oder aber du sagts: ich fange mit allem gleichzeitig an, was natürlich relativ wenig bringt. Weil dann hab ich keinen Vorteil davon. Und von daher ist es wahrscheinlich eh zeitlich begrenzt bis das Ding nicht mehr verwendet wird. [00:35:47-0](#)

B: Hinzu kommt, dass im Bereich Komposit es so ist, dass momentan die Konzepte fast alle fertig sind, es kommt also kaum noch was dazu. Es gibt aber einen Stau bei der Entwicklung. Die Konzepte sind teilweise mit den alten **LFS**-Geschichten (Anm.: JIRA-Projekte KMP und KKZ) noch erstellt worden, die Umsetzung erfolgt jetzt mit dem neuen **LFS**. Auch da ist es in der Gesamtsicht, wenn ich die Filterfunktion nutze relativ wenig aussagekräftig. Da finde ich dann, was weiss ich, eine Umsetzung für Glas, der Konzeptteil ist vielleicht auch oben dran, und wenn der Konzepter nett war hat er die Konzepte vielleicht auch noch reingehängt, aber im Normalfall hat er die Task nur zugemacht. Das heisst also, wenn ich dann Glas suche finde ich das Konzept nicht, aber ich finde irgendeinen Umsetzungsstand. Also auch da sagen dann die Filterergebnisse etwas aus, was irreführend ist möglicherweise.

Von daher ist das Projekt **LFS** wenn man sich jetzt hier eine neue Struktur aufsetzt, und dadurch dass es auch noch nicht so alt ist, relativ wenig aussagekräftig. Aber um solche Geschichten wie QSen oder soetwas zuzuweisen, teamübergreifend, dafür ist es wirklich prima. [00:36:57-1](#)

I: (...) Ja. Das Allheilmittel ist es glaube ich eh wirklich nicht.. [00:37:52-5](#)

B: Eben. Man darf eben sich nicht... das ist hier so ein bisschen die Gefahr vielleicht gewesen.. Weil es kam irgendwann mal JIRA rein, dann haben es die ersten benutzt, die anderen haben immer nur gesagt "hm, JIRA". Dann waren alle ganz begeistert davon, haben es verwendet. Man darf aber nicht vergessen, dass das Tool nicht den Prozess steuern darf. Beziehungsweise den der es bedient. Sondern der der es bedient soll das Tool steuern. Und das ist hier... Das ist aber ein **VersicherungsGmbH**-Problem. Das ist kein generelles Problem, sondern wenn man es eben andersrum macht, wenn man sich vorher überlegt "wie will ich arbeiten" und dann sagt "ok, ich setze das auf". Und es war schon, als man das **LFS**-Projekt hier jetzt aufgesetzt hat, war klar dass der Prozess nach dem wir arbeiten noch nicht gut ist. Und da zu sagen "ich unterstütze das ganze noch mit einem Tool" ist relativ unschön. [00:37:52-5](#)

I: Ja, aber das ist halt auch vorgegeben durch diese Meilensteinstruktur, oder? Dass ich sage "ich habe hier die fixen Termine", erster, zweiter, dritter Meilenstein.. [00:38:01-2](#)

B: Der Prozess den man hier momentan hat und lebt - mit **Prozess xxx** und wie das ganze Ding heisst - wo man ja aber schon bei KFZ oder bei den Sachsparten merkt: "das haut eigentlich nicht hin", weil die Arbeitspakete sind eben nicht in einem Meilenstein durch. Beim nächsten Meilenstein kommt ein neues und macht vorne den ganzen (unv. Protz) wieder auf, und mir explodieren die ganzen Aufwände. [00:38:21-7](#)

I: Das geht ja schon mehrere Meilensteine.. [00:38:22-6](#)

B: Genau. Man hätte also wirklich sagen müssen ich mache es in irgendeiner Projektform, packe Konzepter und Entwickler und Tester da irgendwie zusammen, mache eine Projektplanung, und liefere zu Meilensteinterminen nur noch Softwarepakete erst einmal aus, die aber noch nicht fertig sind. Die können dann vielleicht schon mal getestet werden, aber das ist es dann aber auch. Das Vorgehensmodell was man hier hat ist ein reines Wasserfallmodell was eher darauf abzielt, dass man schon fertig ist mit der Softwareentwicklung, und eher eine reine Wartung und Pflege, beziehungsweise ein paar Weiterentwicklungsgeschichten macht. Das passt nicht dazu, so grosse Pakete wie Wohngebäude, Hausrat, oder gar KFZ ist ja (unv.) Das geht nach hinten los. [00:39:01-6](#)

I: Ja, dann danke ich mal herzlich, das war sehr informativ. [00:41:53-2](#)

B: Kein Problem. Wenn du noch Fragen hast einfach.. jederzeit kommen.

Interview 09_ENTW_DE_3

I: Also erste Frage ist einmal was ist deine Aufgabe hier im Team, und was sind so alltägliche Arbeitsschritte die du machst? 00:00:21-2

B: Also die Aufgabe ist eigentlich.. wir haben ja hier dieses Schadenmanagementsystem für **LFS**, und die Aufgabe ist quasi das Schadenmanagement weiterzuentwickeln, das heisst die ganzen Abläufe zu programmieren. Hauptsächlich Dialog und.. also so ein bisschen Datenhaltungsgeschichte. 00:00:43-2

I: Schadenmanagement ist 'Leist', oder? 00:00:46-2

B: Ja, genau. 'Leist' heisst das bei uns - Leistungsaufnahme. Ja, die Hauptaufgabe ist eigentlich Programmierung. Ab und zu ein paar SQL-Sachen, Migrationsskripte, sowas. Und alles was damit anfängt eigentlich. Dokumentation machen wir ja nicht so viel. Zum Beispiel Qualitätsanalyse von Konzepten, Fehlerbehebung.. ich denke das ist im Grossen und Ganzen das was ich hier mache. 00:01:33-4

I: Und wodurch sind deine Aufgaben vorgegeben? 00:01:42-7

B: Ja eigentlich, wir halten uns ans Konzept. Oder wenn wir Fehler beheben, dann entweder besprechen wir mit den Konzeptern, oder lesen im Konzept nochmal nach, und dann sehen wir schon dass da irgendwas falsch ist, oder dass irgendwie ein Fehler durch einen anderen Fehler entstanden ist... Beheben wir den meistens so nach eigenem Ermessen. Häufig ist ja klar was da falsch ist, oder man sieht schon was da richtig ist, oder wie es richtig geht. Wie wir das ganze umsetzen ist uns überlassen. 00:02:14-5

I: Und mit welchen Leuten bist du eigentlich im Team? 00:02:25-7

B: Mit **ENTW_DE_5**. Und ja.. bisher sind wir eigentlich nur zwei Leute. Der dritte ist ja gegangen. 00:02:35-8

I: Aber ihr gehört schon auch noch zu Komposit, oder? 00:02:36-3

B: Ja, genau. 00:02:38-9

I: Also euer Teamleiter ist **ENTW_DE_TL**. 00:02:39-3

B: Ja. Das ist **ENTW_DE_TL**. Ja und es gibt ja einzelne Sparten in Komposit, so KFZ, Sachsparten, Hausrat, Wohngebäude, Glas, und Unfall, Leben. Und diese Schadensabbildung ist quasi übergreifend über alle anderen Sparten. 00:02:59-6

I: Welche Sparten machst du selbst? 00:03:01-5

B: Zurzeit ein bisschen KFZ noch, also wir hatten gerade Nacharbeiten KFZ, und davor hatten wir Sachschaden. Und jetzt mache ich so ein bisschen Debugging, also Fehlerbehebung. 00:03:15-1

I: Und wer sind jetzt zum Beispiel die Leute mit denen du jeden Tag zu tun hast, also innerhalb vom Team und auch nach aussen. 00:03:25-0

B: Ausserhalb mit den Konzeptern, ja ab und zu mal, wenn ich was nachtesten muss und nicht ganz genau weiss wie es in anderen Geschäftsvorfällen läuft, wie zum Beispiel in 'Leben', dann frage ich da auch einen Konzepter oder Entwickler von Leben. Aber mit Leben habe ich eher weniger zu tun. Also es ist schon eher 'KFZ' und 'Sach'. 00:03:50-0

I: Ok. Aber was für.. ihr macht.. 'Leist' ist wirklich für alle Sparten, nicht nur Komposit, sonder auch.. 00:04:01-8

B: Genau. Also wenn eine Lebensversicherung zum Beispiel einen Todesfall hat, oder einen Invaliditätsfall, dann muss das ja irgendwie abgebildet werden. Das funktioniert dann auch über den 'Leist'. 00:04:11-5

I: Und wer macht Fachkonzept für 'Leist'? Sind das dann mehrere Leute, oder.. 00:04:20-1

B: Das sind zwei meine ich. **FK_3** und **FK_4**. 00:04:26-6

I: Ok.. die kenne ich nicht. Aber die kennen sich dann quasis auch fachübergreifend, oder spartenübergreifend aus? 00:04:30-2

B: Ja, ich glaube die sind von **Kunde_1**. Ich weiss nicht.. Die entwerfen halt so einen Geschäftsvorfall,

besprechen das mit **Kunde_1** wie das aussehen soll, dann geben die uns das zur Qualitätsanalyse. Dann sagen wir "ja, das kann man umsetzen, oder das muss so und so sein" und dann.. Wir sind immer so ein bisschen eingeschränkt durch unser Framework, also man kann nicht alle Ideen einfach so umsetzen. So Quasi ohne dabei unser Framework zu umgehen (unv. was so ...), ja und dann.. das ist eigentlich meiste (unv.). Qualitätsanalyse, dass wir da was ändern müssen oder da was ändern müssen. 00:05:10-1

I: Dass sie sagen, man muss eine bestimmte Ansicht, oder wenn man auf eine bestimmte Aktion drückt, dass dann irgendwas.. 00:05:19-1

B: Genau. Entweder sowas, oder zum Beispiel.. Da sind wir ja quasi mehr die Entscheidenden. Oder wenn wir zum Beispiel einen Fehler haben von irgendwelchen alten Konzepten und wir kennen den nicht so genau, dann kennen sich die schon eigentlich ganz gut mit den Versicherungsabläufen aus, und wir können die dann fragen "ja, wie ist da das Vorgehen, was wäre das richtig, oder wie ist der Fehler behoben?". Also wir hatten da jetzt gerade was mit einer 'schwebenden Fallgruppe' - also eine Fallgruppe bildet quasi immer einen Schaden ab - dass die für Sachbearbeiter nur angezeigt werden wenn sie den Schaden nur selber in Schwebe gesetzt haben. Ja, das war da ganz nützlich. 00:06:02-5

I: Und wie gut weisst du zum Beispiel Bescheid was die Fachkonzepter so machen? Also ob die zum Beispiel gerade anfangen ein neues Konzept zu schreiben, oder ob sie das Konzept gerade ändern an dem du gerade arbeitest? 00:06:17-1

B: Also ich weiss ungefähr wann die unser Konzept ändern, wann die das überarbeiten, aber ich weiss nicht wann die ein neues anfangen, und ich weiss auch nicht wirklich an welchem Thema die gerade arbeiten. Obwohl es gibt ja immer so ein paar Gerüchte was als nächstes kommen soll, also meistens wäre die Überraschung sehr gross wenn irgendetwas anderes kommt. 00:06:37-5

I: Die Gerüchte die werden wie besprochen? 00:06:39-6

B: Ja, es heisst "ah, wir kriegen eine neue Sparte" oder "das und das" 00:06:45-3

I: Das hört man so am Gang dann irgendwie, oder in der Zigarettenpause. 00:06:45-0

B: Ja, sozusagen. Oder es wird etwas umgeplant, und dann kommt doch etwas anderes.. 00:06:57-8

I: Wenn es dann tatsächlich mal ein neues Fachkonzept gibt, oder eine neue Version: wie kriegst du das dann? 00:07:05-3

B: Also wir sprechen natürlich schon darüber, aber letztendlich, wenn es zur QS kommt, also zu Analyse, dann ist natürlich im JIRA-Task. Dann sehe ich das bei mir auf dem Dashboard, (unv. 'unter' / 'und da').. ja, neue Aufgabe, neue Funktion, und dann kucke ich mir das an. (...) **PQM_1** hat ja die Prozesse ein bisschen umgebaut im JIRA, und dann setzte ich mich da quasi.. ja im 'bearbeiten'-Status ein, dann stelle ich die QS auch in JIRA, und dann gebe ich das zur Überarbeitung wieder zurück. Über eine Aktion. Und dann warte ich quasi bis das wieder überarbeitet worden ist, dann kann man das akzeptieren und dann.. Ja, das ist eigentlich ganz gut gelöst im JIRA. 00:07:48-6

I: Aber schaust du dann.. kriegst du dann eine Email von **ENTW_DE_TL** oder so dass es eine neue Version gibt und dass du da was machen sollst? Oder.. 00:07:58-4

B: Nein, ich schaue da eigentlich täglich rein. Also mit JIRA arbeite ich ja jetzt auch zum ersten Mal - ich bin ja auch Berufseinsteiger - ja, da musst ich mich ein bisschen dran gewöhnen. Erst fand ich das ein bisschen unübersichtlich, aber das lag daran, dass die Prozesse halt schlecht organisiert waren. Und ja, jetzt finde ich das eigentlich sehr handlich. Man loggt sich ein, dann sieht man sein Dashboard, und dann sieht man die Aufgaben, und dann.. ja, kann man die bearbeiten. Beziehungsweise auf 'wartend' oder auf 'schliessen' stellen. Das ist ganz gut. 00:08:29-2

I: Hast du dir dieses Dashboard selbst konfiguriert? 00:08:33-6

B: Nein, das hat **ENTW_DE_TL**, oder unser Teamleiter hat uns geschickt. Wir haben da so ein Standard-Dashboard, und je nach Benutzer sieht der dann.. werden die Aufgaben zugeteilt. 00:08:44-8

I: Und mit dem kommst du aus, also hast keine eigenen Filter mehr oder so? 00:08:51-7

B: Nein (gedehnt). Also ich hatte ab und zu mal, also am Anfang als die Prozesse noch nicht so gut waren, da habe ich mir eigene Filter gemacht. Dann so ein paar.. ja quasi, was sind das? Das sind ja keine SQL-Statements. So Suchstatements geschrieben. Über die Filter. 00:09:00-7

I: Dieses JQL oder wie das heisst. [00:09:03-2](#)

B: Ja, genau. Und dann haben ich die meistens benutzt, und wenn ich etwas Neues gesucht habe, dann habe ich die Schlagwörter quasi ersetzt dadurch. Und das gefunden. Oder einfach in der JIRA-Suche gesucht. [00:09:17-9](#)

I: Und schaust du zum Beispiel manchmal nach was **ENTW_DE_5** so macht irgendwie, in JIRA? [00:09:23-2](#)

B: Ja, ab und zu. Also insgesamt was KMP macht, und so ein bisschen, um den Überblick nicht zu verlieren. Aber zur Zeit, wenn viel zu tun ist, dann sehe ich eigentlich nur nach "was muss ich machen". Wie zum Beispiel jetzt, da ist **ENTW_DE_5** ja im Urlaub, und sein.. ja. Da ist ja auch noch die neue Auslieferung von der neuen Version, und da ist das ein bisschen Stressig.. und dann mache ich dann eigentlich nur so dass die Arbeit erledigt wird. "Nächsten Task, nächsten Task, nächsten Task", "Fehlerbehebung, Fehlerbehebung". Ja. [00:09:57-4](#)

I: Wenn du zum Beispiel das 'Leist' machst für eine bestimmte Sparte: wieviel musst du dann eigentlich mit den anderen Entwicklern von dieser Sparte sprechen? Also zum Beispiel du machst 'Leist' für KFZ, wieviele Berührungspunkte gibt es dann? [00:10:12-7](#)

B: Also wir haben ja jeweils unsere Tasks. Die stehen eigentlich ja festgeschrieben in JIRA, beziehungsweise wir teilen das Konzept in Tasks auf, und daraus werden dann in JIRA Aufgaben erstellt. Und dann ist meistens schon besprochen dass einer das macht oder ein das, **ENTW_DE_5** das macht und ich das. Wenn es irgendwie grössere Probleme gibt oder Schwierigkeiten, oder wo man sagt "ja, das ist eine heikle Ecke", da arbeitet man lieber zu zweit dran, dann arbeitet man quasi zusammen an einem Computer und überlegen beide wie man das lösen kann. Aber ist nie so wirklich dass man sich da in die Quere kommt. Es ist eher so dass, wenn wir zum Beispiel mit Produkt arbeiten, weil die uns, oder weil wir von Produkt Information kriegen, dann arbeitet man quasi auch parallel daran. Also man kommt sich nicht wirklich in die Quere. [00:11:14-1](#)

I: Gibt es für Produkt, also für 'Leist' meine ich, ein eigenes Produkt? [00:11:17-3](#)

B: Nein.. [00:11:22-9](#)

I: Oder ist das quasi das Produkt was zum Beispiel **ENTW_OE_3** und **ENTW_OE_2..** [00:11:26-0](#)

B: Also es gibt bei Produkt verschiedene Bereiche.. [00:11:27-3](#)

I: (unterbricht) ich kenne mich eben selbst nicht so gut aus mit Produkt.. [00:11:27-0](#)

B: Ich auch nicht (lachen). Aber ich glaube für uns ist **ENTW_DE_1** verantwortlich, und der stellt uns die Information bereit.. Und wenn der ein Konzept kriegt zum Beispiel was nicht mit unserem Konzept zusammenpasst.. ja dann müssen wir das natürlich erst mal besprechen, und dann klären wir das auch noch mit dem Konzepter. [00:11:52-1](#)

I: Wie merkt man das, dass es nicht zusammenpasst? [00:11:51-2](#)

B: Das sind so ganz einfache Informationen. Wenn du zum Beispiel irgendwie so Wahrheitswerte hast, also Boolean-Werte, dass sie einmal zum Beispiel als Boolean abgebildet werden, oder "ja, nein" oder als "1, 0", oder als String mit "J" und "N".. Und das ist, wenn die das nicht konsequent durchziehen dass es ein immer ein und der gleiche Datentyp ist, ja... das ist dann ein bisschen ärgerlich so für die Umsetzung. Dann muss man das noch ein bisschen anpassen, und da wir ja häufig - um Speicher zu sparen - viel Information in ein und demselben Datenfeld speichern wollen in der Tabelle, da wollen wir nicht extra ein neues Feld dafür einfügen. Nur weil der sagt das soll ein String sein. Von da... Aber eigentlich ist das ganz gut geregelt so von den Aufgaben her. [00:12:53-5](#)

I: Wie ist das beim Programmieren: hast du mit **ENTW_DE_5** zum Beispiel viele Klassen wo ihr beide in derselben Klasse arbeitet? [00:13:00-7](#)

B: Ja. [00:13:07-6](#)

I: Sprecht ihr euch da ab, damit ihr euch nicht gegenseitig etwas überschreibt, oder wie funktioniert das? [00:13:09-1](#)

B: Ja, wir haben ja ein Repository, und dadurch, wenn einer eincheckt und der andere auch - wenn es denn da mal zu kommen sollte dass jeder auf der gleichen Methode arbeitet, und das ist schon.. ja unwahrscheinlich - dann sieht man ja schon beim Einchecken oder beim Updaten dass irgendwelche Konflikte entstehen. Dann kann man die schon beheben. Aber meistens ist es schon so, dass jeder an einer anderen Geschichte arbeitet, und

nach und nach an den selben Klassen arbeitet. [00:13:41-7](#)

I: Also du fragst dann nicht noch extra nach "du, **ENTW_DE_5**, du bist eh nicht gerade in dem und dem..."
[00:13:44-3](#)

B: Ja. Also ich weiss - wir sprechen ja auch so miteinander - und dann weiss ich was er macht und dann... Also man kommt sich eigentlich nicht in die Quere, vor allem nicht bei zwei Leuten. Ich denke mal, wenn das sieben Leute, acht Leute sind, dass es mehrere Konflikte geben würde. Aber so.. bisher ist noch nichts schwieriges entstanden. [00:14:05-8](#)

I: Ok. In deinem Sametime, in deiner Kontaktliste: Wie hast du die organisiert? [00:14:25-3](#)

B: Also einmal ist da dir Gruppe KMP da drin, dann habe noch alle Leute aus **VersicherungsGmbH**, und dann habe ich mir unsere Leist-Konzepter nochmal in eine extra Gruppe gesetzt, dann noch zwei Leute aus Architektur, **ARCH_1** und **ARCH_2**, und von Support noch zwei Leute, **SUPP_1** und **SUPP_2**. Und von Releasemanagement und Datenbanken die vier, **RMM_DB_1**, **RMM_DB_2**, **RMM_2** und **RMM_3**. Und dann habe ich noch eine kleine Gruppe mit Leuten mit denen ich so noch recht viel in der Firma zu tun habe, einfach nur so zum Chatten und zum Mittagessen-verabreden und sowas. [00:15:20-3](#)

I: Für was brauchst du die Architektur-Leute? [00:15:22-2](#)

B: Also unser Fachbereich - der 'Leist' - ist ja Spartenübergreifend, und vieles ist so konzipiert, dass es nur in einer eigenen Sparte funktioniert. Besonders was das Tabellenmodell betrifft. Wir haben ja noch kein Hibernate, wir haben ja unser eigenes System, ja.. das muss manchmal erweitert werden. Die ganzen Frameworks die wir benutzen ist ja alles quasi inhouse-Software, und ja.. manchmal wenn es Schwierigkeiten oder Fragen gibt, dann muss man mit denen reden. Und **ARCH_1** und **ARCH_2** sind eigentlich sehr gut was so das Know-How angeht. [00:16:13-7](#)

I: Hast du mit denen persönlich zu tun, oder chattest du sie an, oder schreibst du ihnen ein Email? [00:16:16-7](#)

B: Also manchmal chatte ich die an, oder rufe sie an, oder treffe sie zum Meeting. Oder manchmal kann man das auch einfach so kurz lösen, oder einer von denen kommt runter oder ich gehe hoch. Also alle möglichen Wege. [00:16:29-9](#)

I: Dann eine Frage: verwendest du MyLyn - dieses Eclipse-Plugin wo man sich die JIRA-Tasks ansehen kann?
[00:16:39-2](#)

B: Ich habe es mal verwendet, aber ich war zu faul um das immer zu nutzen. Das war ein bisschen umständlich. Und jetzt checken wir - wenn wir an etwas arbeiten - checken wir direkt mit den JIRA-Nummern ein, und dann wird das an dem Task quasi als Revisionsnummer mitgespeichert, plus dem Kommentar. Wenn man es mal ankucken will was da genau gemacht worden ist. Ich finde es so herum auch leichter als mit dem MyLyn-Plugin. Da musste man zuerst den Task aktivieren, dann kam es auch zu komischen... [00:17:20-3](#)

I: ...klappt es wieder zu.... [00:17:19-0](#)

B: Ja, und dann hat man vergessen das eine zu deaktivieren, dann hat man was anderes angefangen und dann gab es einen kuriosen Mix. Ja, so ist es einfacher wie es jetzt ist. [00:17:29-8](#)

I: Ok. Dann noch eine Frage zum Ticketsystem: wie findest du das? Um jetzt zum Beispiel die Übersicht zu behalten. [00:17:42-2](#)

B: Also wir haben ja noch das Lotus Notes-Ticketsystem.. mittlerweile habe ich mich daran gewöhnt. Ich finde es nicht besonders gut. Ich finde Lotus Notes nicht besonders gut. [00:17:52-1](#)

I: Wieso? [00:17:56-0](#)

B: Also das schlimmste an Lotus Notes ist, dass es häufig irgendwie hängt, manche kuriosen Fehler auftreten. Und dass man irgendwo daran arbeitet, und dann wenn man von Sametime oder irgendwo eine Benachrichtigung herbekommt, dass sich das Fenster in den Vordergrund setzt, aber dass sich der Fokus nicht auf dem Fenster ist, sondern dass der Fokus quasi noch in dem Programm ist in dem man gearbeitet hat. Das ist eigentlich blöd. (Anm.: Dieses Feature kann ich Lotus Notes deaktiviert werden, diese Einstellung ist aber sehr tief im Menü "vergraben") [00:18:20-8](#)

I: Und für die Ticketbearbeitung selbst? Ich weiss nicht wie bei euch der Status ist, ob ihr viele Tickets habt, oder ob das momentan ein grosses Thema ist? [00:18:32-3](#)

B: Normalerweise kriegen wir - im letzten Meilenstein zum Beispiel hatten wir recht wenige Tickets, und wenn dann waren das so kleinere Sachen, Flüchtigkeitsfehler... Aber das.. es ist übersichtlich. Ich weiss nicht, wenn man zum Beispiel so viele Tickets hat wie jemand anders, vielleicht zwanzig, dreissig Stück, dann ist das vielleicht unübersichtlicher. Aber wir haben ja die.. wir können ja die verschiedenen Statusse benutzen wie 'realisiert', 'angenommen', 'zugewiesen' und die ganzen Sachen, und.. das lässt sich eigentlich noch ganz übersichtlich bedienen. Aber ich glaube dass man das noch optimieren könnte das System. [00:19:21-0](#)

I: Und wenn du da zum Beispiel reinsiehst: schaust du dann deine eigenen Tickets an, oder die vom ganzen Team? [00:19:25-5](#)

B: Zur Zeit schaue ich meine und **ENTW_DE_5** an, weil da waren noch viele wichtige Sachen für die Auslieferung - weil die läuft ja diese Woche aus - und da habe ich beides quasi angekuckt. Und **ENTW_DE_TL** ist ja grade im Urlaub, da kucke ich da ob ich noch irgendetwas machen kann. [00:19:45-0](#)

I: Also ob da quasi irgendwelche 'Leist'-Tickets herumliegen. [00:19:50-4](#)

B: Genau. Und **ENTW_DE_TL_STV_2** weiss uns die Tickets ja auch recht zügig immer zu. Also heute war zum Beispiel ein Fehler aufgetreten, da hatte ein Konzepter ein Ticket eröffnet, und bevor ich mir das geholt hatte, hatte ich das eigentlich schon weil **ENTW_DE_TL_STV_2** mir das zugewiesen hat. (...) [00:20:15-8](#)

B: Aber allgemein finde ich, dass man irgendwie so eine zentralere Quelle benutzen sollte. Also wir benutzen jetzt JIRA und Lotus Notes für Aufgaben und für Tickets, und das sollte man schon irgendwie in eine Quelle setzen. Und bitte nicht Lotus Notes. (lacht). [00:20:30-7](#)

(JIRA-Login) [00:21:14-9](#)

I: Ok, also das ist dein Standard-Dashboard. Wie navigierst du da jetzt zum Beispiel herum wenn du da in der Früh den Computer einschaltest? [00:21:34-4](#)

B: Ja, ich logge mich ein, dann lande ich auf der Seite mit dem Dashboard, und dann habe ich hier meine Aufgaben. Ja hier liegt zum Beispiel eine QS von dem Nacharbeiten 'Leist'-Konzept - das ist schon ein bisschen älter. Ich kann das rauffohlen (klickt) (...) [00:21:55-2](#)

B: Hier haben wir quasi unsere QS, da können wir die bearbeiten. Da hab ich hier mal für die Version Fünf eine erstellt, dann.. Ja, ich habe es so gemacht, dass ich zur Version Fünf immer eine QS mache. Das heisst, ich nehme einmal so ein Panel, dann habe ich hier die Frage, dann habe ich das so ein bisschen umeditiert da bei der Frage. Da haben die immer, bei meinem Vorgänger haben die immer das Namenskürzel und das Datum und so in dicker Schrift geschrieben - ich habe das jetzt als Frage gemacht, das fand ich eigentlich ein bisschen netter. [00:22:31-5](#)

B: Und dann habe ich hier die Frage, und wie man sieht warte ich auch noch auf Antworten. Ja.. ich weiss nicht. Normalerweise sagt man immer das soll innerhalb von drei Tagen passieren, das ist jetzt aber schon ein paar Wochen alt. [00:22:43-8](#)

I: Das ist jetzt dieser 'Dokument wird nachgebessert', das heisst es liegt jetzt momentan beim Fachkonzepter? [00:22:48-2](#)

B: Genau. Da habe ich die Aktion 'nachbessern' gegeben, und dann kriegt er ..Task.. (unv..) jemand anders hingehen. Wir haben hier unsere Prozesse: hier habe ich zum Beispiel 'FQ' - das ist unsere Qualitätsanalyse vom Fachkonzept - das war mein Task, und da steht ja jetzt auch der Status 'Dokument wird nachgebessert'. Wenn das hier.. das Fachkonzept wenn da steht 'wartet', normalerweise sollte er es schon wieder haben weil ich es ihm zugewiesen habe. [00:23:24-1](#)

Ja, hier sind die ganzen Konzeptversionen drinnen... Wahrscheinlich hat er in einem anderen Konzept noch was zu tun, oder er hat es noch nicht umgestellt, oder.. ich habe irgendwas falsch gemacht (lacht). [00:23:32-0](#)

I: Also normalerweise, wenn du die QS auf 'nachbessern' stellst dann müsste er im Prinzip das Fachkonzept wieder 'in Arbeit' nehmen. [00:23:41-3](#)

B: Ja genau, dann müsste er das 'in Arbeit' nehmen. Da mir der Tast nicht zugewiesen ist habe ich jetzt weniger Aktionen. Der müsste das eigentlich haben. [00:23:51-1](#)

Wir haben in unserem Prozess immer verschieden Sachen, ich kann nicht alle erklären. Einmal dieses

Fachkonzept ist 'FK', und dahinter, die Hintergrundfarbe teilt die Aufgabe immer zu einem Fachbereich ein. Blau ist Konzeption, blau-gelb ist Entwickler und Konzeption, grün ist.. Ach ja, grün ist Test, das Testteam - PMQ heisst das jetzt, Prozess- und Qualitätsmanagement - ich glaube dann gibt es noch pink, und das ist Produktmanagement meine ich, ich bin nicht sicher. [00:24:36-2](#)

I: Was macht Produktmanagement (unv. in diesem Dings)? [00:24:37-4](#)

B: Ich denke mal dass sie irgendwie das Produkt beschreiben und irgendwie eine grobe Idee an die Versicherung schicken, solche Sachen. [00:24:48-1](#)

I: Ich habe nämlich heute mit **PMM_1** gesprochen, der ist auch von Produktmanagement, und der hat halt gemeint, dass sie selbst eigentlich total wenig mit JIRA machen, und zum Beispiel auch keine Tasks erstellen. Darum frage ich jetzt gerade wieso die auch da drin sind. [00:25:13-2](#)

B: Welche meinst du? [00:25:15-3](#)

I: Die rosaroten. Bei diesem 'SR'.. [00:25:21-3](#)

B: Ich kann ja mal draufklicken (klick). Ich weiss gar nicht was das ist, 'SR'. [00:25:21-5](#)

I: Aber das ist wird jetzt quasi zwischen Blau und Rosa verhandelt, dieses Ding, wie? [00:25:27-6](#)

B: Ja, also die Konzepter und das Produktmanagement. Das arbeitet schon häufiger zusammen wenn das erstellt wird. Ich weiss auch nicht was die im Einzelnen da machen. [00:25:38-2](#)

B: Hier kann man auch Kommentare schreiben. Wichtig sind für mich eigentlich während der Implementierung diese Tasks hier wo 'IMPL' steht. Erst ist es so, dass wenn mein Name, oder mein Benutzer hier bei dem Task steht, dann kann ich das als interne QS an **ENTW_DE_TL_STV_2** schicken, dann kuckt er sich das nochmal an. Ja, und dann schliesst er das. Dann ist der Task geschlossen, und wenn alle Tasks geschlossen sind, dann haben wir noch Testnachweise. Da sieht man ja auch dass quasi wir und das Testteam. (...) Dann teste ich die Funktion nach, und hefte das Testdokument dann hier an.. Soll ich mal eines aufmachen? [00:26:32-3](#)

I: Ja. [00:26:37-2](#)

B: (Klickt) (...) Ja, das sieht dann so aus (Anm.: ein Formular für Testnachweise auf Basis eines MS-Word-Dokumetes wurde geöffnet). Hier tragen wir die Version ein, das ist Release, ein paar andere Angaben hier sagen was wir testen wollen. Da beschreiben wir das möglichst abdeckend, sodass, wenn irgendwelche Schritte eingedeckt sind die nicht normal entstehen dann machen wir ein paar Screenshots ran.. [00:27:17-7](#)

I: Achso, das schreibt der Entwickler selbst, diesen Ablauf. [00:27:19-3](#)

B: Ja. Und dann testet der Tester das nach. [00:27:28-3](#)

I: Wie, der Tester testet das auch nochmal nach? [00:27:30-4](#)

B: Ja der testet.. Also wir erstellen diese Testnachweise, und der testet die nach. Und wenn das alles funktioniert so wie es beschrieben ist, dann schliesst er glaube ich den Task. [00:27:38-9](#)

I: Also das wird dann aber auch quasi doppelt getestet? Also du musst es selbst einmal durchlaufen, erstellst dazu die Screenshots.. [00:27:47-5](#)

B: Ja also... Manchmal ist es so dass man etwas testet und da sieht man schon einen Fehler. Und dann testet er es trotzdem noch nach, in der offiziellen Version, und öffnet dafür ein Ticket. Ja. Häufig ist es auch so, dass die Tester noch ein bisschen extra drum herum testen und noch ein bisschen mehr sehen, so dass sie einfach mal eine Ahnung haben was da überhaupt passieren soll. [00:28:21-7](#)

B: (...) Ja, was könnte ich dir noch zeigen (...) Das hier ist die Schätzung wie lange wir für die gesamte Aufgabe brauchen, 'DD' ist 'Dokupool Dokumentation'.. Technisches Konzept ist Beschreibung der implementierten Sachen.. Soll ich nochmal zurückklicken? [00:28:58-2](#)

I: Nein, eine kurze Frage nochmal, weil du vorhin gemeint hast dass du diese eine Task von der QS oder so, dass die jetzt eigentlich 'in Arbeit' sein sollte, oder schon überfällig ist. [00:29:08-9](#)

B: Diese hier, nicht? [00:29:07-9](#)

I: Ja, genau. [00:29:09-7](#)

B: Soll ich hier raufklicken? [00:29:10-5](#)

I: Ja, mach mal. (...) Oder nein, geh nochmal zurück bitte. (...) Das müsste jetzt eigentlich hier bei **FK_3** liegen, oder? [00:29:23-0](#)

B: Genau. [00:29:25-1](#)

I: Kennst du den, oder ich weiss nicht, hast du irgendeine Ahnung wieso das jetzt nicht ist, oder was das los sein könnte? [00:29:32-7](#)

B: Ich denke einfach dass der irgendetwas anderes macht und das nicht umgestellt hat. [00:29:36-5](#)

I: Ok. Aber triffst du den **FK_3** öfter, oder kennst du den persönlich? [00:29:38-9](#)

B: Ja ja, heute Morgen war ich bei ihm. [00:29:40-9](#)

I: Ok. Also der ist nicht völlig aus der Welt.. [00:29:47-3](#)

B: Ja... Also vor zwei Wochen hatte ich auch ein Konzept hier zur QS, und dann habe ich das hier am Dashboard gar nicht gesehen. Da hatte ich nur die.. kurz raufgekuckt, da waren nur die üblichen, aber da war noch eines mehr. Ich weiss nicht ob ich das übersehen habe oder... Ja weiss ich auch nicht. Da lag das dann wohl schon eine Woche oder anderthalb Wochen herum, und eigentlich sagt man immer so drei Tage soll man zur QS benötigen und dann sollte es fertig sein. Aber naja, manchmal dauert das etwas länger. Wenn man gerade irgendetwas anderes hat. [00:30:22-5](#)

I: Und wie ist das dann mit den Terminen, wenn da irgendetwas nicht eingehalten wird? Angenommen du brauchst jetzt für die Implementierung.. sagen wir die ist mit einer Woche angesetzt und dann stellt sich heraus, dass das viel länger dauert? [00:30:36-3](#)

B: Naja, das ist eigentlich nicht so problematisch. Wir haben ja, zum Beispiel wenn man jetzt so eine Implementations-Task hat - (klickt) ich kann mal hier da drauf kucken - zum Beispiel 'Allgemeine Abzüge': da sind wir mit (unv. **ENTW_DE_TL_STV_1**) wie ich mit angefangen habe, die Felder eingearbeitet, und es gab eine erste Konzeptversion. Und da waren Sachen unklar oder nicht beschrieben, und dann.. ja dann dauert das schon mal länger. Aber das ist dann auch nicht so wild. Wir sind ja immer dabei dass die, dass Prozesse vorangetrieben werden, dass die Qualität gut werden soll. Ja, Hauptsache zum Release ist es dann fertig, so wie jetzt. [00:31:24-4](#)

I: Ok. Aber wenn, dann machst du das quasi, (unv. meldest) du das dem **ENTW_DE_TL** oder so? [00:31:31-9](#)

B: Nein (gedehnt). **ENTW_DE_TL** hat damit recht wenig eigentlich zu tun. Eigentlich.. die Implementations-Tasks die werden eigentlich immer erst zum Schluss geschlossen.. Da gibt es immer ein paar Abhängigkeiten dazwischen die man so nicht sieht, weil das ja recht getrennt ist.. Ja und dann werden Sachen jetzt so jetzt dann fertig, und dann schliessen wir die. Also meistens öffne ich die auch nicht wieder wenn die mal geschlossen sind. Wenn da noch Nacharbeiten sind, ja.. dann arbeite ich da einfach an dem Feld dran. Ich weiss ja letztendlich was zu tun ist, aber... [00:32:07-4](#)

I: Also du machst dir nicht einen eigenen Task dafür auf.. keine Ahnung, "das und das gehört noch gemacht".. [00:32:13-6](#)

B: Ja, also wichtig ist einfach so dass man so einen kleine Liste hat in der man sieht "ach ja, 'das muss man noch machen', 'das muss man noch machen', 'das muss man noch machen'", und dann weiss man einfach noch vom Vortag, oder vielleicht ein paar Tage früher dass das noch ein Problem war. Und dann setzt man sich da wieder ran. Häufig ist es auch so, dass man, oder dass ich mir Abends kleine Notizen auf Papier und Zettel, und dass ich dann Morgens weiss "ach ja, da war ich dran", und dann kann ich da weitermachen. Aber mittlerweile habe ich mich eigentlich daran gewöhnt, dass ich JIRA öffne, kucke ob es irgendetwas neues gibt und.. ja. (...) [00:32:53-8](#)

B: Achso, du meinst ja dass sie, das Produktmanagement nicht so viel damit arbeitet. [00:32:59-4](#)

I: Ja, genau. Und weil ich aber vor allem selbst gesehen habe, dass sie eigentlich schon trotzdem diese farbliche Markierung gibt für Produktmanagement. [00:33:11-7](#)

B: Ich kann da nochmal hin (klick) (...) so (...). Hier unten, nicht? (...) Ich weiss nicht genau wie die Prozesse bei

denen sind, in deren Fachbereich. Aber ich denke mal dass die auch ihre Tasks benutzen. Also ich glaube wir wollen ja, ich glaube zur Zeit ist es so... [00:33:42-0](#)

I: ...ich glaube sie haben irgendetwas gesagt dass sie für die Abnahme, oder so, dass quasi das Fachkonzept zurückgeht wieder ans Produktmanagement und dort abgenommen wird. [00:33:50-2](#)

B: Also wir wollen ja wirklich dorthin dass wir mehr JIRA benutzen als Lotus Notes, so wie ich das verstanden habe. Aber das finde ich auch ganz ok. Es ist recht schnell, übersichtlich, ... Man kann es eigentlich recht gut konfigurieren.. [00:34:11-6](#)

I: Verwendest du zum Beispiel solche Funktionen wie dieses 'Überwachen' hier? Dass du dann eine Email-Benachrichtigung bekommst für... [00:34:21-8](#)

B: Also ich bin froh wenn ich nicht so viele Emails kriege. Und wenn was neu ist, dann sieht man das ja hier. Oder wenn man da mal reingeht dann funktioniert das. [00:34:31-7](#)

I: Von ein paar Leuten habe ich gehört, dass die sich überhaupt im Lotus Notes einen Filter gemacht haben der ihnen sämtliche JIRA-Nachrichten einfach mal... Hast du das auch? [00:34:41-7](#)

B: Nein. Ich kriege auch recht wenig. Also ich kriege manchmal welche wenn mir ein Task zugewiesen worden ist. Auch nicht immer, ich weiss nicht genau womit das zusammenhängt. Manchmal, zum Beispiel bei dem letzten Konzept für die QS hätte ich gewünscht dass ich da eine Email gekriegt hätte (lacht). Also ich habe auch gehört die für jeden.. bei allen möglichen Aktualisierungen da eine Email kriegen - ich weiss nicht ob das bei Aktualisierungen auftritt - aber dass sie halt quasi diverse Emails für ein- und dieselbe Aufgabe kriegen. Das ist natürlich Konfigurationssache. [00:35:24-1](#)

I: Ich glaube **ENTW_DE_1** hat das gesagt, dass er sich einen Filter gemacht hat, eben weil er mit diesen automatischen Nachrichten zugemüllt wird. Der sieht sich die gar nicht mehr an. [00:35:38-4](#)

B: Ja, das Problem habe ich nicht. Wenn ich mal eine JIRA-Email bekomme bin ich überrascht. Ich habe gestern oder vorgestern hatte ich mal eine. Davor... [00:35:54-6](#)

I: Wie ist das, wenn du.. angenommen... du bist im Ticketsystem und da muss irgendein... Also angenommen du bist im Ticketsystem und bekommst ein neues Ticket herein, und musst dazu nochmal in das betreffende Fachkonzept schauen. Wie findest du dann dieses Fachkonzept? Hast du das lokal bei dir abgespeichert, oder machst du das über JIRA? [00:36:30-2](#)

B: Häufig ist das so, dass... Ja wenn da ein Fehler ist, dann weiss ich ja ungefähr worum es geht. Und dann ist da in den Annotations im Programmcode ja schon eine Nummer für ein Konzept. Und dann kann man hierher, über die Suche (Klick)... Wir haben hier immer diese Bezeichnungen für die Konzepte.. Sagen wir ich suche etwas für Sachschaden (tippt).. Da kommt man hier hin... Das ist noch die alte Organisation von JIRA, die fand ich nicht so gut, da muss ich immer ein bisschen länger kucken... [00:37:35-4](#)

I: Achso, das ist ein Konzept von einem früheren Meilenstein, oder wie? [00:37:38-7](#)

B: Ja, und da waren die Prozesse noch nicht so gut organisiert. Drum kann das jetzt ein bisschen... (klickt und sucht) (...) Früher waren die immer so verknüpft... so ein bisschen dezentral... Das ist alles durcheinandergewirbelt worden das jetzt... [00:38:13-4](#)

I: Das heisst, quasi durch die Änderung von dem **LFS**... also von der Projektstruktur im JIRA. [00:38:25-2](#)

B: Ja, dass es besser geordnet worden ist. (...) [00:38:31-6](#)

I: Aber damit musst jetzt quasi noch leben, dass du da herumsuchen musst.. [00:38:34-4](#)

B: Ja, das war schrecklich. Denn ich hatte.. irgendwo hatte ich mir glaube ich mir mal einen Filter geschrieben, und wenn ich das gefunden habe, dann habe ich mir die Seite als Bookmark gespeichert. Aber wie gesagt... jetzt finde ich das eigentlich ganz gut strukturiert mit dem Dashboard und so. Ja, das ist nur eine Frage der Konfiguration, denke ich, wie gerne man damit arbeitet. [00:39:05-6](#)

I: Ja ok. Ich glaube das waren dann alle meine Fragen... [00:39:30-2](#)

B: Ja also wie gesagt. Alte Konzepte sind noch recht schwer zu finden, weil die... na weil quasi.. ich weiss nicht wie lange **VersicherungsGmbH** jetzt schon JIRA benutzt.. [00:39:39-8](#)

I: Noch nicht so lang glaube ich, ein Jahr oder so? 00:39:41-2

B: Noch recht neu glaube ich. Und das wird halt immer weiter optimiert. Vom Gefühl her habe ich das Gefühl, dass es jetzt wesentlich besser läuft als vor drei, vier Monaten. Ich weiss ja nicht genau wann der Wechsel zu diesem neuen Prozess.. zu dieser neuen Prozessabbildung durchgeführt worden ist.. aber es ist eigentlich ganz übersichtlich so. Man findet alles schneller, und man weiss auch was man machen muss quasi (lacht). 00:40:14-9

I: Ok, dann danke ich ganz herzlich. 00:40:16-2

B: Gerne.

Interview 10_RMM_1

I: Ja, die erste Frage ist einmal zu deiner Aufgabe hier bei **VersicherungsGmbH**, und was du täglich so machst. 00:00:11-8

B: Also ich betreue das Buildsystem und Tools im Bereich CSM die wir entweder selbst einsetzen, oder die anderen Entwicklern zu Verfügung gestellt werden. 00:00:23-8

I: Welche Tools sind das? 00:00:22-4

B: Zum Beispiel ist das Jobzilla, die CSM-Gui - damit hatte ich auch zu tun. Das Buildsystem ist ja im Grunde genommen auch so ein Tool, auch Subversion wird von uns betreut. Der Virtual Server und die Clients die die Leute installiert haben auf ihren Rechnern. Da müssen wir uns auch drum kümmern. Macht der Interne Support, aber da müssen wir uns auch darum kümmern. Solche Sachen 00:00:50-1

I: Mit welchen Leuten arbeitest du dann täglich zusammen? 00:00:58-9

B: Also täglich mit den Kollegen von CSM, eigentlich mit allen. CruiseControl ist ja auch eines von den Tools die wir betreuen, und von daher auch mit den DB-Admins, weil da ja die Datenbanken dahinterstehen. Und dann kommt natürlich dazu, dass Entwickler sich melden und dann darum bitten irgendwelche Änderungen vorzunehmen, neue Module in CruiseControl einzusetzen zum Beispiel fällt mir jetzt ein. Oder Probleme mit dem Buildsystem haben. Eclipse betreuen wir ja auch, da bekommen wir gelegentlich mal Probleme mit Eclipse. Und das passiert dann meistens in der Form, dass einfach jemand anruft, oder anchattet, oder eine Email an Releasemanagement. Oder wenn du das schon ein bisschen raus hast an wen er sich melden muss eine Email gleich an mich schreibt. Oder eben an jemand anders. Und ansonsten haben wir natürlich auch gelegentlich mal dass wir uns gegenseitig absprechen, dass **RMM_TL** sagt "entwickle mal dies, entwickle mal das". Sowas in der Richtung. Bau mal dies ein, bau mal das ein. 00:02:05-8

I: Ok, also bei euch intern im Büro - ihr macht das meiste glaube ich mündlich, habe ich gehört. 00:02:11-0

B: Genau. 00:02:13-8

I: Und triffst du dann zum Beispiel andere Entwickler auch öfter? Oder wieviel Kontakt besteht da jetzt, ausser dass die Email schreiben? 00:02:27-5

B: Es gibt ja einige Leute die auch ein bisschen mit dem Buildsystem zu tun haben, zum Beispiel **ARCH_1**. 00:02:33-3

I: Der ist von Architektur, oder? 00:02:32-2

B: Der ist von Architektur, genau. Ich weiss gar nicht genau in welchem Masse der jetzt projektmässig da ausgekoppelt ist, aber der hat ziemlich viele Umbauten da am Buildsystem gemacht für die OSGI-Einführung. Und in dem Fall arbeitet man dann mit dem natürlich relativ direkt zusammen. Oder es kommt auch mal, manchmal gerade von der Architektur so eine Anforderung wie "wir möchten gerne dass im Buildsystem" und dann kommen völlig neue Prozesse die da eingebaut werden sollen. Also zum Beispiel diese **xxx-Datenbank** die da seit einiger Zeit befüllt wird seit Anfang des Jahres und ausgeliefert wird. Also da wird eine **xxx-Datenbank** angelegt, befüllt, geprüft das Ganze - hoffentlich - und dann mit ausgeliefert. Und solche.. Und dann hat man natürlich, in dem Fall zum Beispiel **ARCH_3**, aus der Abteilung Arch.. aus unserer Abteilung zu tun. 00:03:26-3

I: Und wie wird das dann eben geplant.. 00:03:26-7

B: Und auch das wird in der Regel dann so.. Also in dem Fall zum Beispiel, bei der **xxx-Datenbank**-Sache, da wurde das wohl erst mit **RMM_TL** besprochen, damit das auch vorher geklärt ist.. sagen musste ob das tatsächlich in unserem Team passiert, ob CSM sich darum kümmert, oder ob das jemand anders einbaut. Weil grundsätzlich könnte es ja auch theoretisch jeder machen. Und nachdem das sozusagen organisatorisch abgesprochen war dass wir dafür.. dass auch unsere Zeit dafür dann verwendet wird, kam dann zum Beispiel die Aussage von **RMM_TL** an mich: "da wendet sich jemand von Architektur an dich", und dann kam da auch eine mündliche, oder per Email "ich schicke dir nochmals die Details per Email" oder so lief das dann. Also das.. Beispielsweise, in dem Fall mir **ARCH_3** eine Email geschrieben hat mit den Anforderungen was er das benötigt. Zum Teil auch schon ein bisschen vorbereitete Ant-Skripte die er hatte hat er mir da mit geschickt, und die habe ich dann weiter verwendet. Und ansonsten ist es so, dass wenn zum Beispiel ein neues Modul (unv. angeliefert).. Wir haben ja jetzt diese OSGI-Einstellung, da war ja ziemlich viel Aufsplittung in Test- und Compile-Module - ich weiss nicht in welchem Maße du das mitgekriegt hast - da war dann also.. da gibt es ja immer das Modul "sowieso" und das Modul "sowieso-Test". Und da mussten die alle angelegt werden, in den CruiseControl

eingefügt werden und Schemen in den entsprechenden Datenbanken dafür angelegt werden. Und auch noch einige Umbauten entsprechend an den.. am Buildsystem. Das geht dann aber so, dass wenn ein Modul angelegt werden muss, dann möchten wir eigentlich ganz gerne eine Email immer haben, damit wir hinterher kucken können wer hat das Modul eigentlich haben wollen. Und.. [00:05:17-8](#)

I: Das kommt dann von den Entwicklern, von den einzelnen? [00:05:16-3](#)

B: Also im Prinzip kommt.. [00:05:18-6](#)

I: (unterbricht) Was weiss ich, Komposit hat irgendwie Sparte KFZ, und braucht Modul soundso. [00:05:21-5](#)

B: Ich weiss gar nicht von wem das, von wem die ursprüngliche Anforderung kommt. Ich nehme an es kriegt immer der Teamleiter von dem betroffenen Team sagt "wir brauchen ein neues Modul - Entwickler XY, kümmer dich mal da drum". Der wendet sich dann an uns, aber wir sollen das auch nicht einfach so anlegen, sondern die Regel ist, dass **ARCH_TL** als Teamleiter von Architektur das eigentlich immer bestätigen soll. Das heisst, entweder schicken die Leute direkt eine Email an **ARCH_TL** und der leitet das an uns weiter, oder sie schicken eine an Releasemanagement, oder persönlich an einen von uns, und wir fragen dann bei **ARCH_TL** nach. [00:05:57-2](#)

I: Ok. [00:05:58-5](#)

B: Und dann gibt der normalerweise sein ok, weil er eigentlich nie etwas dagegen einzuwenden hat. Es ist aber so als Mechanismus vorgesehen, dass er da grundsätzlich sagen könnte "nein, das machen wir anders" oder sowas. Und es ist auch so, dass wir prüfen, ob die Module nicht zu lange sind. Weil es da gewisse Längenbegrenzungen gibt, weil wir ja an den CruiseControls entsprechende Schemen für die Module haben.. [00:06:21-6](#)

I: Bei der Namensgebung für die Module? [00:06:27-2](#)

B: Genau. Also das ist ja so, dass wir das Modul zu dem passenden Schema für das CruiseControl testen. Das heisst dann ein Prefix wie "STTRK" für Trunk und Unterstrich, dann der Modulname. Oder "ST29018", Unterstrich, und dann irgendwas. Und weil Oracle-Schemanamen nur dreissig Zeichen lang sein dürfen - ja, auch 2011 noch - ist dann.. müssen wir dann.. das wissen nicht alle, und dann frage ich manchmal nach "ist das richtig dass das Modul so lange ist".. und bei den Testmodulen da ist es ok.. und naja. Und dann frage ich noch manchmal nach. Ich habe auch so ein Skript zum Anlegen der Module, und das prüft das automatisch. Und manchmal darf man diese Prüfung ausschalten und solche Sachen. [00:07:16-1](#)

I: Und da fragst du dann direkt beim Entwickler nach, also da weisst du dann von wem diese Anforderung gekommen ist? [00:07:21-7](#)

B: Genau ist es so, dass ich dann bei **ARCH_TL** meistens nachfrage "ist das ok dass die Länge überschritten werden darf?". Oder ich schicke an beide gleich die Email zurück, in Kopie an beide, dass ich sage "das ist zu lang, seid ihr euch sicher, ist das gewünscht?". Weil meistens ist es dann so, dass das nicht bedacht war, und dann weiss der Entwickler aber auch nicht... Zum Beispiel bei 'Zahlungsverkehr', da kann der Entwickler auch nicht einfach so entscheiden "na ok, dann kürze ich das einfach ab", sondern dann geht er nochmal zu seinem Teamleiter oder so. [00:07:54-1](#)

B: Achso, und es ist auch so, dass wir jetzt natürlich um es einzuchecken immer ein Ticket brauchen. Weil wir ja diese Eincheck.. Und da können wir auch eine JIRA-ID nehmen... diese Eincheck-Prüfungen haben. Die sagen... [00:08:08-2](#)

I: Für den SVN-Commit. [00:08:07-9](#)

B: Genau. Es muss eckige Klammer auf, eckige Klammer zu.. und dann... Das gilt natürlich auch im Prinzip für das Anlegen von einem Modul. Weil das ist ja eigentlich nur ein (unkl.Make-DMS)-Subversion. Und weil ich das unter meinem normalen Benutzernamen anlege gilt das dann auch für mich. Das ist eigentlich irgendwie so eine automatisierte Sache. Und das vergessen die Leute auch meistens, oder wissen sie nicht, und da muss man dann auch wieder nachfragen. Und da mach ich es manchmal auch der Einfachheit halber dass ich per Chat nachfrage. Bevor ich da wieder eine Mail, und das dauert bis der die gelesen hat, und dann kommt die zurück und das dauert bis ich die gelesen habe. Da frag ich dann eben per Chat nach, weil die wissen ja.. die haben ja für ihre Commits dann auf diesem Modul auch schon irgendwas. Entweder ein Fachkonzept, oder jetzt in letzter Zeit meistens JIRA-IDs. [00:08:53-4](#)

I: Achso, dann lässt du dir quasi von ihnen die JIRA-ID geben, oder wie? [00:08:57-9](#)

B: Genau. [00:08:58-0](#)

I: Oder sie erstellen einen eigenen Task dann für dich.. [00:09:00-0](#)

B: Ja, das könnte auch vorkommen.. Ich glaube das ist auch schon vorgekommen dass sie dann extra dafür schnell mal ein JIRA-Ticket 'Neues Modul XY' anlegen. Aber meistens haben sie auch schon... [00:09:09-4](#)

I: (unterbricht) Das weisen sie dann direkt dir zu? [00:09:12-6](#)

B: Nein, das geben sie mir einfach als.. [00:09:15-3](#)

I: Ok, sie weisen es sich selbst zu und schicken dir dann per Chat die ID. [00:09:19-0](#)

B: Ich weiss nicht wem das zugewiesen ist. Wahrscheinlich.. also ich glaube in vielen Fällen ist es auch schon ein bestehendes Ticket was sie für ihre ganze Arbeit an diesem neuen Modul, oder zumindest für diese anfängliche Arbeit an dem Modul haben. Wie die das immer organisiert haben das weiss ich ja nicht, aber irgendwie was müssen sie ja auch schon haben, weil sie ja auch demnächst da einchecken wollen. Und die geben sie mir dann. Damit kann ich dann einchecken, weil es wird ja beim Einchecken nicht geprüft, ob das mir zugewiesen ist. Und es soll ja eigentlich glaube ich.. die Prüfung soll ja nicht so sein. Also wenn das existiert, dann soll jeder damit einchecken können, mehr oder weniger. Also so verstehe ich das. Also jedenfalls wird es nicht geprüft. Und.. ich brauche das nur um einchecken zu können, und das ist dann auch zuzuordnen. Das ist ja der Sinn der Sache.. dieser Anforderung. So läuft das. [00:10:05-9](#)

Also, wie man sieht, alles ziemlich an JIRA vorbei eigentlich. Und es könnte.. sie könnten auch wenn sie keine JIRA-ID haben, können sie mir im Prinzip auch eine Fachkonzept-ID zum Beispiel sagen. Alles womit ich einchecken kann. [00:10:17-7](#)

I: Gut. Wie ist das dann zum Beispiel bei den **REA**-Anträgen, den Nachlieferungsanträgen. Da ist ja meistens so, dass irgendwie Freitag Abend glaube ich, oder Mittwoch oder so ist immer Eincheckschluss. Und dann ist es häufig so, dass trotzdem noch irgendwer irgendwie etwas noch fertig macht, oder noch nicht fertig ist bis zur Eincheckdeadline. Wo dann die Entwickler öfter auch mal sagen: "Du, ich brauche noch eine halbe Stunde oder so", oder? Kommt das vor? [00:10:51-8](#)

B: Also da habe ich jetzt nicht so viel tun damit in letzter Zeit. Vielleicht kann dir **RMM_2**, wenn du ihn auch noch mal befragst, mehr dazu sagen. Oder **RMM_3**. Oder **RMM_TL**. [00:11:03-1](#)

I: Also du musst mehr diese.. [00:11:04-9](#)

B: Also ich habe das auch schon gemacht. Also das läuft dann.. wenn es um diese Ausweitung der Deadline geht.. ich weiss gar nicht. Also zu unserer Zeit haben wir gesagt: "eigentlich gilt die ja, die Deadline", aber das war dann ja eher meistens schon von den Leuten mit **ENTW_DE_AL** abgesprochen. Und der darf das ja.. von dem kommt ja die Deadline. Und der darf entsprechend natürlich auch sagen: "ok, ihr dürft später einchecken". Dann hatten sie es also mit dem schon abgesprochen, und dann kam an uns eine Email oder so. [00:11:30-9](#)

I: Von **ENTW_DE_AL** der sagt.. [00:11:32-3](#)

B: Nein, von den Leuten. Ich weiss nicht... Also sie kamen auch immer... Jetzt grade war ja diese Woche, gestern, war glaube ich auch für 3-0, für eine von den 3-0-4er.. 10 glaube ich.. 3.0.4.10 Releases ein Eincheckschluss, und da kamen zum Beispiel von Provision ein, zwei Meldungen. Und auch kurz vor vier nochmal.. kamen Leute mit einem Chat.. haben mich angechattet und gefragt "ist das ok?". Aber ich habe das dann an **RMM_2** weiter... Also wenn es ok ist - und es ist in der Regel ja dann meistens doch ok - dann genehmigen wir einfach den REA. Weil da gibt es ja keinen tatsächlichen technischen Mechanismus der das unterbindet. [00:12:20-1](#)

I: Ok. Aber habt ihr dann zum Beispiel, eben wenn ihr wisst "ok, jetzt ist Eincheckdeadline", habt ihr dann... beobachtet ihr... und wie es der Fall war bei Provision, dass die noch schreiben, euch anchatten "es dauert noch". Habt ihr die dann, habt ihr dann die Provisionsleute zum Beispiel im Sametime auf dem Radar ob die jetzt noch online sind, oder wie bei denen der Status ist, wie lange das noch dauern wird? Oder sitzt ihr dann einfach da und denkst dir "ok, jetzt warte ich halt noch bis..." [00:12:50-8](#)

B: Da müssten wir **RMM_2** fragen wer das jetzt gemacht hat. Ich glaube ich habe das... Ja man ist.. Also ich habe das zuletzt gemacht, da waren wir noch im alten Haus. Und da waren Provision gleich nebenan. Und die waren das meistens. (...) Nein, aber... ich weiss nicht. Man kuckt natürlich schon, ob die noch daran arbeiten, ja klar. Aber es heisst dann ja auch meistens "ja, wir schaffen das bis dann und dann", oder "wir schaffen das gleich". Also das ist... man kuckt natürlich manchmal wer noch online ist, aber diejenigen melden sich dann wie

gesagt öfter auch mal im Chat. [00:13:28-4](#)

I: Du tust selber auch Entwickeln, mit Java programmieren, oder? [00:13:50-2](#)

B: Ja. [00:13:50-2](#)

I: Verwendest du das MyLyn? [00:13:53-9](#)

B: Nein, das benutze ich überhaupt nicht. Als das mal eingeführt wurde habe ich das mal anfänglich benutzt, aber mache ich nicht mehr. Das war mir ein bisschen zu umständlich. Da ist man dann ja... In Eclipse dieses Plugin, nicht? Ich sehe das noch gelegentlich dass manche Leute damit einchecken, dann gibt es ja immer diese typischen Eincheck-Commit-Kommentare, aber sonst... [00:14:25-4](#)

(JIRA-Login) [00:16:29-3](#)

I: Ja, würdest du mir vielleicht kurz ein bisschen zeigen... dieses Dashboard, hast du das selbst konfiguriert? [00:16:36-8](#)

B: Ja, da habe ich mir ein bisschen zusammengesetzt, dass meine Projekte die ich öfter bearbeitet habe da angezeigt werden. Wobei das kann sein, dass das schon nicht mehr so ganz aktuell ist, denn die 'Jobzilla-GUI' zum Beispiel.. ja gut, hier steht es jetzt auch gar nicht... Ach nein, das sieht ganz gut aus. Also 'Buildmanager'... achso das ist auch so ein Ding das wir intern benutzen, was in Java geschrieben ist, was wir machen um Branches abzuspalten, um Releaseumgebungen aufzusetzen, und so Kram. Auch für den CruiseControl, um den einzurichten. 'Buildsystem', da kommt eigentlich nie etwas rein, aber da könnte man theoretisch... 'Bearbeiten Jobzilla', das ist ja bekannt. 'SÄM', achja richtig, das machen wir natürlich auch noch. 'CruiseControl' und 'SVN', das sind Projekte an denen ich öfter arbeite. Und hier sind die mir zugewiesenen Tickets. Und Vorgänge 'in Arbeit', aber das.. weiss ich gar nicht. Da ist eigentlich nie etwas drin, weil ich nie etwas auf 'in Arbeit' setze, muss ich ehrlich zugeben. [00:17:39-9](#)

I: Wie oft bist du zum Beispiel drin in JIRA? [00:17:44-3](#)

B: Ich bin eigentlich dauernd eingeloggt. Also ich habe dieses 'Login speichern' angemacht. Aber ich gehe eigentlich selten auf mein Dashboard, sondern es ist eher so, dass ich... Man kann ja Firefox machen, dass man hier 'Controll + L' und dann 'jira' (tippt)... und dann sehe ich hier schon immer irgendwie ein Ticket - also in diesem Fall jetzt gerade nicht (Anm.: weil fremder Rechner, deshalb keine History) - aber da sieht man irgendwie eine Ticketnummer.. eine URL von wo man ein Ticket aufgerufen hat. Da gehe ich da hin, und gebe dann da die korrekte Ticketnummer ein, mache 'Return' und dann lande ich bei dem Ticket. [00:18:16-7](#)

I: Also du gehst dann direkt über die JIRA-ID rein? [00:18:19-0](#)

B: Genau. Ich gehe über die JIRA-ID eigentlich rein. Und dann, oder manchmal mache ich es so, dass ich... man kann ja statt der Ticket-ID kann man ja auch das.. vom Bindestrich einschließlich an wegnehmen. Und dann hat man das Projekt. Das funktioniert ja auch. Also du kannst ja... da ist die URL ja so ähnlich. [00:18:36-4](#)

B: Und da kucke ich mir das Projekt an. Und da kucke ich, das ist immer ganz interessant, also wenn man hier sich mal ein Projekt ankuckt hat man diesen tollen Aktivitätsstrom hier. Da sieht man immer, gerade wenn ich mit jemand anders zusammenarbeite... Zum Beispiel hier was ich machen würde.. Das klappt natürlich immer besonders gut wenn man die Tastatur gut kennt (tippt)... So. Da kann man dann einfach so... kuckt man da rein, und dann sehe ich hier '**RMM_4** hat hier gerade irgendetwas gemacht'. [00:19:08-5](#)

I: Ok. **RMM_4** ist auch bei euch im Team, oder? [00:19:10-8](#)

B: Genau. **RMM_4** ist auch bei uns im Team, und diese CSM-Gui mit dem man Statements abschicken kann wo jetzt die Jobzilla-Gui drin integriert ist, die hat er hauptsächlich geschrieben, aber ich arbeite da auch daran. Und da funktioniert das auch so ein bisschen mit JIRA wie man sieht, weil gelegentlich mal ein Benutzer aus dem Bereich 'Test' ein Ticket eröffnet.... Während es bei Sachen... bei "SÄM" und beim Jobzilla und allen diesen Dingen wo ich hauptsächlich interne Anforderungen kriege, eigentlich nie passiert, dass mir jemand ein Ticket in JIRA eröffnet. Sondern, wie gesagt, ich werde angerufen, ich bekomme Emails, jemand kommt zu mir. [00:19:47-3](#)

I: Der **ARCH_2** hat das dann **SONSTIGE_4**... Der hat das dann gleich euch zugewiesen, oder wie? Der weiss dass ihr.. [00:19:57-1](#)

B: Wir können ja mal kucken. Ich nehme an, das hat er gleich... [00:20:03-0](#)

I: Also der weiss, dass du und **RMM_4** daran arbeitest.. 00:20:10-3

B: Nein, ich glaube es gibt auch dieses... man kann auch sagen 'zuweisen automatisch' oder so. 00:20:16-7

I: Wie funktioniert das? 00:20:15-9

B: Also wenn ich einen Vorgang erstelle, dann kann man hier doch sagen 'Bearbeiter - automatisch'. 00:20:30-6

I: Ok, und an wen geht das dann? 00:20:30-1

B: Dann landet es wahrscheinlich automatisch bei **RMM_4**, weil er wahrscheinlich... das kann man ja konfigurieren wer dann... 00:20:35-3

I: Achso, das geht dann an den Leiter von dem Projekt wahrscheinlich. 00:20:41-3

B: Also ich weiss nicht ob er das wissen muss wem er das zugeordn... Hier steht ja "**RMM_4**" - **RMM_4**. Aber es kann auch sein, dass er das gewusst hat. Weiss ich nicht. Es könnte ja auch mal die Konstellation geben dass ich sage "mach ein Ticket dafür auf bitte, und weise es gleich mir zu", oder sowas. Wenn ich auch als Entwickler hier eingetragen bin. Weiss ich gar nicht ob man das irgendwo sieht.. Das ist natürlich auch eines von diesen Dingen: ich kann jetzt hier gar nicht sehen, wer Entwickler ist. Aber ich konnte es sehen, wenn ich sage (klickt) 'Vorgänge', 'Vorgang erstellen', dann sieht man: mich kann man da auch auswählen. 00:21:27-7

I: Ok. Und ihr kriegt dann eure Tickets, also wenn ihr jetzt zum Beispiel so eine Entwicklung macht wie 'CSM-Gui': plant ihr das vorher mit JIRA, oder wie läuft da die Planung ab? 00:21:40-2

B: Ich mache das nicht, **RMM_4** macht das glaube ich öfter. Weil wenn wir uns die Tickets ankucken die es hier gibt, dann hat er die oft auch eröffnet. Also... gut das ist jetzt hier gerade in den letzten Tagen ein schlechtes Beispiel, aber... '**RMM_4** erstellt - Suchfunktion auch in Result- und Statementbrowser' (Anm.: liest die Zeile aus dem Aktivitätsstrom vor, ohne die JIRA-ID zu nennen!). Zum Beispiel. Und was da passiert ist, gehe ich mal davon aus - ich weiss nicht ob du ihn auch noch befragen willst, er kann dir da sicher noch was erzählen - aber in diesem Fall ist wahrscheinlich **RMM_TL** zu **RMM_4** gegangen und hat gesagt "wir hätten das gerne noch, dass es eine Suchfunktion geben würde". Und dann hat er, um sich das nicht auf einen gelben Zettel irgendwo schreiben zu müssen, das in JIRA eingetragen. Und man kann dann ja manchmal Kommentare noch dazu schreiben, und dann ist ja auch ganz praktisch, dass das dann jeder sehen kann. Kommt hier selten, aber wie man sieht gelegentlich vor. In diesem Fall war es ein Kommentar der sich darauf bezieht wann er das Ticket, beziehungsweise dass das Ticket nicht umgesetzt wird bis die die Serverstruktur umbauen. Und solche Sachen. 00:22:51-6

B: Und das mache ich auch manchmal, das habe ich zum Beispiel gemacht.. (sucht) mal kucken.. es gibt ja.. bei 'SÄM' müsste das sein... genau. 'SÄM-518', das war so eine Geschichte von den DB-Administratoren die sie haben wollten für SÄM. Und da habe ich, da ich das nicht gleich bearbeitet habe, habe ich - weil viele da gesagt haben 'ok, das ist ein etwas komplizierteres Thema, da müssen wir uns ein bisschen Gedanken darüber machen' - habe ich dafür ein Ticket aufgemacht. Das aber auch gleich mir zugewiesen ist, gehe ich mal davon aus, genau. 00:23:32-1

Und das schöne daran ist natürlich auch, wenn ich 'Meldung durch' eintrage, weiss ich, dass **RMM_DB_2** dann eine Email kriegt wenn ich es schliesse. Oder wenn ich etwas kommentiere. So dass sie dann Bescheid weiss. Wobei ihr das eh immer dann nochmal sagen würde. Da sie ja praktisch nur einen Raum weiter ist, ist das ja auch immer ziemlich unkompliziert. Und so mache ich das natürlich bei.. auch wenn **RMM_TL** irgendetwas haben möchte, dann.. 00:24:01-0

I: Ok... Diesen Aktivitätsstrom da jetzt noch mal zum Beispiel. Also du siehst da schon immer rein, und... ich weiss nicht... 00:24:06-8

B: Ich kucke mir das schon an was passiert, das ist auch immer ganz interessant. Sind natürlich in diesem Fall selten Sachen von denen ich NICHT weiss. Aber bei der CSM-Gui, da macht **RMM_4** manchmal einfach so Sachen, und da kucke ich, und dann sehe ich das. Das ist dann natürlich ganz interessant. Es gibt natürlich auch Sachen die er da nicht einträgt, und davon weiss ich dann natürlich nicht. Also.. es ist.. man weiss nicht.. Es ist kein vollständiges Abbild der Änderungen die es gibt. Darüber muss man sich dann eben auch bewusst sein. Aber das ist eh so mehr... Na sagen wir mal es wäre nicht so weiter dramatisch, wenn ich das nicht sehen würde, und nicht wissen würde. Das ist mehr so 'kucken aus Interesse'. 00:24:45-4

I: Bist du, ich weiss nicht,.. wie lange ist **Daniel** schon hier in der Firma? 00:24:48-3

B: Also länger als ich. 00:24:51-1

I: Länger als du? Achso. Ich habe nur zuletzt geschaut im Organigramm, und da steht ' **RMM_4** - Student'.
00:24:56-9

B: Ja, er hat glaube ich angefangen zu Studieren während er noch bei der **VersicherungsGmbH** war.
00:25:02-4

I: Achso, ok. Also das heisst, du bist nicht für ihn verantwortlich 00:25:08-2

B: Nein. 00:25:08-1

I: ...und siehst deshalb nach was er macht, sondern einfach nur.. weil du auch an dem Projekt arbeitest.
00:25:16-7

B: Nein, das ist eigentlich eher so das... Also das interessiert mich nur, so in dem Sinn. 00:25:24-5

I: Siehst du dir von anderen Leuten auch an was die so im JIRA treiben? Zum Beispiel von **RMM_DB_2** und **RM_DB_1**? 00:25:33-9

B: Nein, das eigentlich nicht. 00:25:36-9

I: Also rein aus Interesse oder so? 00:25:36-2

B: Nein... Also nicht bezogen auf die Personen, sondern bezogen auf das Projekt mehr. Und da jetzt aber auch nicht auf irgendwelche Projekte die sie gemacht haben, also das.... Also ich interessiere mich eigentlich nur für Projekte mit denen ich auch zu tun habe. Also insgesamt sind solche Tools natürlich dann manchmal ganz interessant wenn man mal so kucken will 'was passiert eigentlich bei dem-und-dem Projekt', auch wenn man damit nichts zu tun hat. Aber kann man dann ja auch meistens nichts mit anfangen. 00:26:10-3

I: Und wenn du jetzt wirklich wissen willst wie es in einem Projekt steht: verlässt du dich dann zum Beispiel auf das JIRA, oder gehst du dann einfach rüber zu **RMM_4** und fragst... 00:26:18-9

B: Wenn ich wirklich etwas wissen will frage ich ihn eher, oder chatte ihn an. Also da er ja direkt neben mir ist, kann ich ja einfach vorbeigehen. (...) Das ist mehr eine Sache die so neben her läuft. Also das ist mehr so eine Sache die zusätzlich das nochmal dokumentiert zum Nachlesen, so ungefähr. 00:26:38-9

B: (...) Also in diesem Fall, in der CSM-Gui gerade in letzter Zeit, wo ich auch relativ viel gearbeitet habe, da läuft das auch so dass wir uns relativ eng abstimmen und immer miteinander reden "ich habe jetzt gerade das-und-das gemacht" und so... Da benutzen wir nicht JIRA für. Aber das hat natürlich auch damit zu tun, dass wir nur zu zweit sind. Da ist es natürlich ziemlich einfach das zu koordinieren, weil ich muss ja nicht mehrere Leute fragen was sie gemacht haben, und im Kopf behalten wer was gemacht hat. Und dass wir halt im gleichen Team sind, und das relativ problemlos alles läuft. 00:27:09-2

I: Wie ist das dann zum Beispiel beim Programmieren? Wenn du deinen Code aus dem SVN updatest - schaust du dir dann an diese Kommentare? 00:27:19-3

B: Ja, genau. Da kuckt man rauf und... Natürlich manchmal auch direkt auf die Änderungen, das ist ja eigentlich dann immer am aufschlussreichsten. Welche Dateien geändert worden sind, und was geändert worden ist. Und da ist es natürlich auch so dass man sich manchmal so ein bisschen abstimmt, damit es nicht zu bösen Konflikten kommt. Und da ist es auch so, dass ich natürlich auch weiss womit ich rechnen kann, weil wir natürlich auch vorher schon darüber gesprochen haben. Also dass ich weiss was kommt. Das ist nicht so "Oh! Da!".. also manchmal vielleicht schon, aber meistens nicht. 00:28:02-4

B: Genau. Und so ist es.. ja. Bei Dingen wie SÄM auch zum Beispiel. Bei SÄM arbeitet ja noch **ARCH_2** mit dran, und da ist die Aufteilung sogar ziemlich klar. **ARCH_2** hat so ein Modul das heisst '**SÄM-Core**' und ich habe ein Modul - und **RMM_4** auch - das heisst '**SÄM-Ant-Tasks**'. Und ich arbeite immer in dem einen, der an dem anderen. Und der Fluss ist immer: SÄM-Core muss zuerst die Änderungen haben, dann kannst du mal die benutzen. Weil umgekehrt gibt es die Benutzung nicht, die Abhängigkeit ist so, dass meines von seinem so... sinngemäss. Und daher sind bestimmte Dinge vorher mit ihm abgesprochen. Und er benutzt das ziemlich konsequent, **ARCH_2**. Er macht sich auch für alles was reinkommt einen Task auf, sodass man sich bei ihm auch darauf verlassen kann dass man sieht was er gemacht hat. Manchmal kommen auch Sachen bei ihm rein die er uns nicht erzählt, weil das nur die wie gesagt... 00:28:58-7

B: (holt etwas weit aus) Das sind zum Beispiel, da gehört dazu diese Prüfung von dem SÄM-Anträgen dass man da bestimmte Datenbankbedingungen einhält, und da ist es ein typischer Fall der vorkommt ist, dass die zu strikt

gefasst sind. Das hat sich mal jemand überlegt, so ein bisschen am grünen Tisch, und in der Praxis stellt sich heraus es gibt massenweise Altfälle die dem nicht entsprechen, die jetzt aber auch nicht geändert werden können. Gerade im Bereich **Kunde_2**, weil da die Änderungen häufig dann auch noch Altsysteme betreffen. Da kann man nicht einfach so sagen "ok, dann machen wir die doch nullable oder machen sie nicht nullable". Das kann man nicht einfach so machen. Und dann baut er da Ausnahmebedingungen ein und solche Dinge. [00:29:38-0](#)

B: (kommt wieder zum eigentlichen Thema zurück) Und das kriege ich gar nicht mit. Und da ist es dann auch schon zu Überraschungen gekommen was plötzlich sich geändert hatte, und wo ich so ein bisschen überrascht war. Wobei dann natürlich.. DANN muss ich zugeben auch das erste was ich mache - ich frage ihn. Weil er sitzt auch direkt gegenüber, und dann gehe ich auch hin und sage so "hast du da und da was dran gemacht?". Das mache ich sogar.. manchmal tendiere ich dazu, erst... Also mein Haupttool für solche Fälle ist eigentlich SVN. Ich kucke in SVN "was ist das passiert". Und dann das nächste, der nächste Schritt ist ich gehe hin und frage "was hast du gemacht". Oder manchmal gehe ich auch so direkt hin und sage "was hast du (unv...)" ohne vorher nachzucken. Dann ärgere ich mich aber meistens, weil da hätte ich eigentlich auch gleich nachkucken können. [00:30:18-6](#)

B: Und das ist auch, wenn es um andere Sachen geht, wenn zum Beispiel irgendein Build-Fehler plötzlich irgendwo auftritt. Und dann.. dann sind das häufig manchmal.. Also gerade bei diesen 'Export-Daten-Modulen' die auch grösstenteils von uns betreut werden. All das sind ja dann so Module, wo JEDES Team drin rumbastelt, und kein Team sich um die Build-Skripte kümmert. Sondern die liegen bei uns. [00:30:40-7](#)

B: Und dann kommen Buildfehler, und die Leute können es nicht wirklich zuordnen und fragen dann bei uns nach. Und da ist auch immer der erste Schritt "kuck mal nach was sich da geändert hat (in SVN, Anm.)". "Wer hat das geändert?" Und man kennt natürlich, also aus Erfahrung, und weiss, was für Änderungen zu was für Problemen führen können. Da könnte ich gar nicht... wüsste ich gar nicht, wo ich in JIRA nachkucken kann. Das sind dann.. Beziehungsweise da kann ich natürlich in die Commit-Geschichte kucken und kann kucken welche JIRA-IDs da angegeben werden. Aber da wird ja inzwischen ziemlich viel wohl mit JIRA organisiert und... Ja, es ist auch schon vorgekommen dass ich da mal gekuckt habe "oha, das ist eine JIRA-Id, was ist das gewesen, was haben die da gemacht?". Aber auch das ist dann ja eigentlich schon fast eher aus Interesse, denn aus der JIRA-ID geht ja nicht hervor, was das Problem mit dem Commit war letztlich. Das ist eigentlich auch mehr so aus Interesse was da abgeht. Und für die eigentliche Fehlersuche muss man in die Commit-Geschichte kucken und in die geänderten Dateien, Änderungen in den Dateien, Diff's sich anucken. [00:31:47-1](#)

B: Wie gesagt. oder einfach mal.. Wenn es Sachen sind.. da kann ich ja dann fachlich auch nicht zu sagen normalerweise. Das sind ja dann eher formale Geschichten, wie irgendwelche Führungszeichen oder so. Bei solchen Geschichten wo ich fachlich auch etwas dazu weiss frage ich, wie gesagt, gerne mal nach. "Was habt ihr das gemacht", und "wie soll ich jetzt darauf reagieren" oder sowas. [00:32:09-6](#)

I: Und eben wenn jetzt zum Beispiel so Buildfehler auftreten beim CruiseControl: fragst du dann zum Beispiel nach bei den Entwicklern? [00:32:19-4](#)

B: Also ich muss sagen, ich habe ja wenig mit zu tun mit Projekten die tatsächlich in CruiseControl gebaut sind. Das 'SÄM-Ant-Tasks' ist eigentlich das einzige was ich mache was in CruiseControl gebaut wird, ausser das BuildSystem natürlich, wo ich dann natürlich weiss auch was betroffen sein kann. Und ich muss... Also wenn... Ja, wenn es Buildfehler gibt, merke ich das meistens dann entweder weil ich weiss, ich habe das etwas riskantes gemacht, ich muss überwachen ob es einen Buildfehler gibt. Oder weil ich weiss - also inzwischen haben wir ja ziemlich viele generische Buildskripte, das heisst ich ändere gar nicht modulspezifisch etwas an irgendeinem Modul - das heisst ich kann das vorher gar nicht so genau wissen, was schiefgehen kann. Wenn ich tatsächlich etwas Modulspezifisches mache, kucke ich natürlich nach ob das Modul durchbaut. Da kriegt man dann eine Email. Wenn ich etwas beim generischen Buildskript mache, habe ich meistens das Gefühl "das wird schon nicht zu Fehlern führen", oder ich weiss wo es zu Fehlern führen kann - zum Beispiel ich mache etwas an dem generischen Buildskript für Persil-Module. Dann baue ich, dann klicke ich testweise mal - bei einem Persilmodul wo ich weiss das geht schnell - auf 'Build', kucke was passiert, und denke mir, so nach dem Motto "wenn eines von diesen Modulen durchbaut, dann bauen die Übrigen auch durch". Oder wenn nicht, dann liegt es jedenfalls eher an mir. Und es gibt aber natürlich auch Fälle, da merke ich das gar nicht. Weil ich nicht damit rechne, dass eine Änderung die ich gemacht habe zum Buildfehler führt. [00:33:52-4](#)

B: Oder es sind dann ja auch gerne mal Sachen, die fallen schon in meinen Bereich, aber es sind gar nicht unbedingt Änderungen von mir die zu Buildfehlern führen. Und deswegen kommt es manchmal vor, dass sich Leute an mich wenden - und das passiert eigentlich AUSSCHLIESSLICH per Email oder per Chat, da kommt nie ein Ticket oder sowas - "sag mal, dies und das baut nicht..". Die wissen dann ja auch nicht, ob das wirklich meine Schuld ist. Die wissen ja das ja genausowenig. Die denken nur "das könnte etwas mit dem Buildsystem sein, wisst ihr da was, habt ihr das was gemacht?". Dann kommt so eine Frage, und dann kuckt man sich das an. Also zum Beispiel 'Exportdaten-Build'. Das.. kriege ich nicht mit, dass der nicht baut, weil ich nicht regelmässig drauf

kucke, beziehungsweise ich kucke zufällig doch mal drauf und dann sehe ich es - ich kucke natürlich schon ab und zu mal auf den CruiseControl. Aber dann ist es halt Zufall. Und deswegen kommt es gelegentlich mal vor, dass ich es eben noch nicht mitgekriegt habe, und jemand anders sich aber nun ärgert, weil er da was eing_checked hat und nicht weiss, ob das durchbaut. Und dann fragt er nach. Und da kann es dann auch sein, dass es nicht seine Schuld war, sondern es war jemand der vor ihm etwas eing_checked hat. [00:34:53-3](#)

B: Und das ist.. da gibt es zum Beispiel Fälle, wo man syntaktisch korrekte SQL-Skripte eincheckt, aber wir haben da so eine Prüfung die die SQL-Skripte prüft, und die... hat etwas strengere Anforderungen als SQLPlus. Da müssen bestimmte Dinge dann in Anführungszeichen stehen. Oder wo SQLPlus das nicht stört wenn sie es nicht tun. Und da kriegen sie einen Buildfehler und können das schwer zuordnen, weil die Fehlermeldung dann auch komplett irreführend ist und so. Und dann, wie gesagt, kommt eine Email oder wird man angechattet oder sowas. [00:35:29-3](#)

I: Ok. Dann noch eine kurze Frage zum JIRA: Verwendet ihr die Kommentar-Funktionen unten ebne überhaupt zum Kommunizieren? So statt einer Chatnachricht oder so, dass man irgendwie unten in den Kommentar etwas reinschreibt und sich dann darauf verlässt, dass das ankommt? [00:35:50-3](#)

B: Es kommt schon vor, dass... Also wenn man mal ein Ticket hat das plötzlich von jemand gekommen ist, wie zum Beispiel von einem Benutzer aus dem Testbereich mit dem jetzt nicht unbedingt... also mit denen haben wir auch jeden Tag zu tun, aber... Dann schreib... also es kann passieren... Du meinst so Sachen wie, man schliesst ein Ticket mit 'das haben wir doch schon hundertmal besprochen' oder sowas. Oder 'das ist ein Duplikat von Ticket sowieso' und hoffe dass er das dann liest. [00:36:22-0](#)

I: Zum Beispiel, ja. [00:36:22-6](#)

B: Ja, das kommt vor. Ich weiss gar nicht.. [00:36:27-3](#)

I: Ich habe zum Beispiel, bei **PMM_1** habe ich geschaut, da habe ich ein Ticket (Anm.: Fehler: gemeint ist eigentlich Task!!) gesehen, da haben sie unten wirklich viele Kommentare. Und das war halt wirklich so wie ein Gespräch, also so wie man sonst Email hin- und herschickt. [00:36:37-0](#)

B: Ja, also zum wirklichen Kommunizieren in dem Sinne nicht. Aber naja, das geht ja alles ineinander über. Wo haben wir denn.. (sucht und klickt) SÄM nochmal. [00:36:49-2](#)

B: Manchmal schreibt man etwas rein, was nicht bloss als Selbstdokumentation gemeint ist. [00:36:56-7](#)

B: (sucht)... da müsste man natürlich mal ein Beispiel für finden.. Hier kann man leider nicht erkennen, ob es kommentiert ist, ob es da Kommentar zu gibt. Das wäre natürlich auch mal eine nette Sache. [00:37:16-2](#)

B: Aber es gibt ja noch... Ok, **ARCH_2** hat mal was kommentiert... **ARCH_2** kommentiert öfter mal was. Aber ich habe das bestimmt auch mal gemacht. Hier hatte ich eines selbst aufgemacht, **ARCH_2** zugewiesen, und habe es dann aber selbst gelöst. Das war einer der wenigen Fälle, wo ich in seinem Modul rumgebaut habe. Und habe das dann komm.. ich habe es geschlossen, und dabei den Schliesskommentar dazu "ich habe mir erlaubt, das selbst zu machen". [00:37:45-2](#)

I: Ok, aber das hast du ihm sonst nicht mehr Bescheid gegeben, oder? [00:37:50-2](#)

B: Genau. Da habe ich ihm sonst nichts Bescheid gesagt. Das kann er natürlich auch in Subversion nachvollziehen was ich da gemacht habe, aber ich habe auch das Ticket geschlossen, damit er weiss, dass er das nicht mehr machen muss. Und in dem Fall gehe ich dann davon aus, dass er es dann gelesen hat. [00:38:07-1](#)

I: Der bekommt dann wahrscheinlich eine Email-Benachrichtigung, oder? [00:38:11-5](#)

B: Ja, das Ticket war ihm ja zugewiesen, das hatte ich ihm ja zugewiesen. Wenn mir eines zugewiesen ist und jemand schliesst es, kriege ich - glaube ich - eine Email-Benachrichtigung. [00:38:19-0](#)

B: Oder aber ich sehe es nicht mehr, und (unv. finde es nicht und) wenn ich mich frage: "Moment, ich hatte doch da ein Ticket, wo ist denn das geblieben?" und kucke es mir nochmal an, dann sehe ich ja den Kommentar. [00:38:32-5](#)

B: Hier hatte ich mal eine Ergänzung dazugeschrieben, aber das ist jetzt mehr so... [00:38:33-3](#)

I: Dokumentation eher.. [00:38:38-6](#)

B: Also nicht so wirklich zur Kommunikation jetzt in dem Sinne. Nicht so wie wenn man mailen würde oder sowas.

Mehr so dass es jemand später noch sieht dass bearbeitet.. [00:38:52-4](#)

I: Wie findest du das JIRA zum Beispiel im Vergleich jetzt zum Ticketsystem, also rein von der Übersicht? [00:39:07-3](#)

B: Zu dem Alten, zu dem NTS, oder... [00:39:08-2](#)

I: Zum Notes-Ticketsystem. Dieses "TTS". [00:39:11-5](#)

B: Ja es gab ja auch mal, der Vorgänger von JIRA war ja so etwas ähnliches wie das TTS. Das kennst du nicht mehr, nicht? [00:39:18-2](#)

I: Nein. [00:39:17-6](#)

B: Das hiess "NTS". Das war glaube ich schon das neue Ticketsystem. Im Vergleich dazu ist es ganz gut. [00:39:36-2](#)

I: Das TTS jetzt? [00:39:36-0](#)

B: Also "ITS", JIRA, ist ganz gut im Vergleich zu dem Alten. Ich benutze es eigentlich gar nicht intensiv genug um die Vorteile wirklich zu haben. Also solche... Wie gesagt, was ich ganz hübsch finde ist dieser Aktivitätsstrom. Das ist eine nette Sache. Das gab es anfänglich bei JIRA auch nicht, aber das ist irgendwann mal dazugekommen. Und... ja, ansonsten ist Notes halt schwierig immer.. Da ist JIRA eigentlich schon attraktiver, so allgemein vom Layout, von der Bedienung her. Wobei ich das gar nicht mehr so genau in Erinnerung habe. Das war auch relativ schnell nachdem ich zur **VersicherungsGmbH** gekommen bin ist das abgelöst worden. Mit (unv. ITTS? I-TTS?) arbeite ich momentan nicht so viel. Nur dass ich da, wenn ich Builds mache, dass ich mal Tickets erstelle. Aber es gibt natürlich auch hier die Möglichkeit, wobei das alte System das auch irgendwie hatte, mit Versionen zu arbeiten. Das hatte ich ursprünglich gemacht, zum Beispiel für die Build-Manager, dass ich... [00:40:57-5](#)

I: Das ist dieses Ding hier unten, oder? (sucht, klickt) (...) "Versionen" (unv...) [siehe Video 25:33] [00:41:05-5](#)

B: Genau. Dass ich diese, also dass ich das zusammenfasse. Dass ich sage: "Ich bringe jetzt eine neue Version heraus, und da packe ich das-und-das rein", und dann konnte man ja auch die Tickets immer zuordnen zu einer Version, und dann hat er einem auch automatisch Change-Notes generiert. Und das kann das Ding ja grundsätzlich. Das habe ich anfänglich auch genutzt, aber es ist enorm viel Arbeit, weil man wirklich für alles was man macht ein Ticket dann einstellen muss - wenn man will dass es in den Change-Notes auftaucht. Und man muss auch die Versionen immer generieren. Und ich habe das auch noch in Erinnerung, dass das dann wenn man etwas falsch gemacht hat, und man die Versionen umstellen wollte, dann war das immer... also das war dann... ich habe das als ziemlich umständlich in Erinnerung. Und... und was ist, wenn ich das jetzt doch nicht in dieser Version gelöst habe, und solche Geschichten. Und man muss wie gesagt jede Version die man... und es ging relativ schnell, also ich wollte das ja auch immer, wenn mich einer das.. Das war ja auch immer.. Da kam einer wirklich mündlich bei mir an, und hat gesagt "wir müssten (unv...) da ist ein Fehler passiert, das Ding ist abgestürzt" - dann wollte ich das natürlich schnell "ups, da stürzt was ab, die wollen, brauchen das, die müssen das jetzt benutzen", also kucke ich schnell mal rein "oja, da habe ich eine Null-Test vergessen", dann repariere ich das, erzeuge die JAR, schicke sie rüber. Das geht ja schnell, das ist ja in einer viertel Stunde kann ich den Fehler möglicherweise behoben haben und die können weiterarbeiten. Wenn ich da jetzt aber zusätzlich immer noch diese Versionsverwaltung für machen muss, dann ist es ein kompletter Overkill für ein Tool das drei Benutzer hat. Und weil das ja auch noch.. das ist das interne Tool für CSM, der Build-Manager. Was wirklich.. Wobei das auch zum Teil schon gemacht bevor ich bei CSM war, als ich bei **ARCH_5** war. Und deswegen habe ich das mal irgendwann gelassen. Wobei ich eigentlich sagen muss, falls es irgendwie interessant ist: ich pflege da ziemlich konsequent die Change-Notes. Also das ist eine Text-Datei, ganz simpel, wo drin steht "Version XY, gemacht dies, gemacht das". Da kann ich das eben reinschreiben, und ich pflege auch die Versionsnummern. Also ich mach das schon richtig, ich schicke nicht eine JAR raus mit der gleichen Version wo etwas anderes drin ist. (...) Naja, kommt natürlich alles mal vor, aber wie gesagt, in so einem komplett informellen... Aber ich mache das schon, aber ich mache es nicht mit JIRA. Und es ist dann natürlich auch so, dass ich darüber nachdenken muss. Und dass ich mich auch vertun kann. [00:43:29-7](#)

I: Also das heisst, diese Versionsnummer die hier sind.. sind dann nicht unbedingt... [00:43:36-4](#)

B: Ja ich weiss gar nicht. Da steht glaube ich beim 'Build-Manager' noch irgendetwas drin, aber... Schön wäre natürlich wenn... genau, hier gibt es nur irgendwie... pff... [00:43:47-2](#)

I: Also das sind nicht mehr die aktuellen Quasi. [00:43:50-3](#)

B: Das ist nicht mehr aktuell, das wird nicht mehr genutzt. [00:43:52-7](#)

B: Und diese Fälligkeiten und so, die benutze ich sowieso nicht. Und diese ganzen Zeitschätzungen... Da gab es bei SÄM mal so eine... Am Anfang dass wir dafür.. dass wir auf alle Tickets Zeitschätzungen reingestellt haben. Aber ich muss... ich habe mich in JIRA nie damit auseinandergesetzt. Ich gehe mal davon aus... also wenn jemand das benutzen würde, wäre es ganz praktisch. Wenn man das konsequent.. und das ist natürlich die Frage: wie konsequent organisiert man seine Arbeit nach (unv. Time) so einer Zeitschätzung. In dem Fall, in dem Projekt wo ich das mache, ist das eigentlich nicht so. Sondern es gibt eigentlich mehr so Anforderungen wie "das muss noch in die Version XY von **xxx** mit rein". Oder "wir brauchen das, bevor wir in **xxx** das-und-das einführen". Oder "das muss in **xxx** mit rein". Und dann weiss man das ja. Und das kann man hier drin dann glaube ich auch nicht so günstig abbilden. Also dieses "dafür brauche ich wahrscheinlich vier Stunden" oder so, das... [00:44:52-5](#)

I: Dann eine abschliessende Frage noch: welche Leute hast du in deiner Sametime-Kontaktliste? [00:44:59-4](#)

B: Ich habe, wir haben ja ganz **VersicherungsGmbH** einmal drin - es gibt ja dieses " **VersicherungsGmbH**-als-Gruppe" - das habe ich sowieso immer, das ist nur meistens zugeklappt. Und dann habe ich mir das nach Teams eingeteilt. Und die Teams mit denen ich zu tun habe, inzwischen eigentlich alle. Und dann habe ich so, nach-und-nach ab und zu mal ein paar Leute von denen ich wusste, dass sie im Team sind. Und ab und zu halt wenn ich mit jemandem zu tun hatte. Oder wenn ich wissen wollte ob jemand online war, zum Beispiel weil der noch was von mir wollte, und ich wollte mich zurückmelden, aber das kann ich ja natürlich erst wenn er da ist. Und da habe ich jetzt, zum Beispiel ist da 'Interner Support', 'Komposit', PRP, 'Leben', 'Architektur' und so weiter. Und mein eigenes Team. Und dann eigentlich so eine persönliche Kontaktliste für mich selber habe ich dann nicht. Ich klicke dann die Leute nach Bedarf an, so ungefähr. Und... das ist meistens, was bei mir in Notes so offen ist. Also nicht der Kalender, sondern die Kontaktliste. Weil man.. ja in Notes... ja ich weiss nicht, man könnte die ja auch völlig anders arrangieren... [00:46:05-8](#)

I: Ja es gibt ja diesen Feed-Reader, die Kontakte, und darunter den Kalender.. [00:46:06-7](#)

B: Ja genau. Den Feed-Reader benutze ich gar nicht, und Kontakte habe ich meistens offen, und den Kalender... habe ich selten offen. [00:46:17-8](#)

I: Ok. Na gut, dann wäre es das von meiner Seite. Herzlichen Dank, das war super ergiebig. [00:46:27-4](#)

B: Kein Problem.

Interview 11_PQM_1

I: Ok, kannst du mir kurz sagen was du in der Firma machst, wofür du zuständig bist? 00:00:08-2

B: Ich bin im Qualitäts-, beziehungsweise im Prozessmanagement tätig, und habe da so einen ganzheitlichen Blick über die Prozesse im Unternehmen, und versuche wo es geht halt die Prozessabläufe zu unterstützen. Und so ist auch der Kontakt eigentlich zu JIRA - wie es heute benutzt wird - entstanden. 00:00:28-8

I: Und kannst du mir etwas erzählen über die Einführung von JIRA hier bei der **VersicherungsGmbH**? Wann ist das losgegangen und ... 00:00:37-5

B: Ja.. das war mehr so ein schleichender Prozess. JIRA wurde eingeführt eigentlich als reines Ticketsystem. Um für die ganzen hier intern von uns selbst entwickelten Tools auch ein Issuetracking durchführen zu können. Und wir hatten da vorher so eine eigen.. auch dafür eine eigene Entwicklung auf Notes-Basis die wir ablösen wollten... 00:00:57-6

I: Das TTS. 00:00:58-4

B: Nein, das "NTS" damals. Also das TTS ist das Ticketsystem für **LFS**, und wir hatten aber für unsere internen Tools die wir hier zur **LFS**-Entwicklung verwenden hatten wir ein eigenes Ticketsystem. Und das sollte abgelöst werden, und da habe ich damals die Evaluierung für durchgeführt. Und da kam halt JIRA im Endeffekt bei raus. Allerdings auch mit der Vision das vielleicht irgendwann mal als allgemeines **LFS**-Ticketsystem auch verwenden zu können. 00:01:39-3

I: Wieso wurde gerade zum Beispiel JIRA verwendet und nicht ein anderes System? Oder gab es da Alternativen überhaupt? 00:01:45-5

B: Also wir haben einen umfangreichen Anforderungskatalog damals erstellt und haben uns.. ich habe mir dann in Summe glaube ich im ersten Schritt vierzig verschiedene Systeme angekuckt, die dann erst mal grob aussortiert, und zum Schluss blieben noch vier, fünf über die wir dann hier testweise installiert haben und verschiedene Anwendungsszenarien durchgegangen sind. Und ja, JIRA blieb dann über. Also wenn du da Interesse an den Ergebnissen oder dem Evaluierungskatalog oder so hast, das könnte ich dir noch rausholen. 00:02:12-6

I: Wenn du das hast, das wäre super, ja. Dankeschön. Ja, und wann hat dieser Prozess begonnen? 00:02:36-8

B: Vor fast jetzt genau drei Jahren. 00:02:38-5

I: Drei Jahre schon? 00:02:42-7

B: Genau. Und wir haben es dann die ersten.. na ich würde mal so sagen das erste Jahr auch wirklich nur als reines Ticketsystem für Fehler, etcetera verwendet. Wir im Qualitätsmanagement als kleines Team damals haben uns überlegt wir wollen aber... wir brauchen noch eine Möglichkeit um unsere Planung für das Team effektiver gestalten zu können, und haben uns überlegt, dafür auch JIRA zu verwenden, und waren dann so quasi die Vorreiter die dann angefangen haben ihre Aufgaben komplett darüber zu verplanen, zu buchen, etcetera. Ja, und das haben wir dann auch ein gutes Jahr intensivst betrieben, und dann haben die anderen Abteilungen und Teams so langsam mitgekriegt "ah, da kann man was machen", und "JIRA ist vielleicht doch mehr als ein reines Ticketsystem", und haben dann für sich auch selbst herumprobiert. Auch nicht koordiniert oder irgendwie so, sondern jeder für sich im stillen Kämmerlein. Bis man irgendwann einmal spitzgekriegt hat, dass ja die einzelnen Teams eigentlich mittlerweile alle damit arbeiten, aber alle unterschiedlich, jeder angepasst für seine eigene Welt. Und dann sind wir vor einem dreiviertel Jahr, Jahr hingegangen und haben uns überlegt das zu vereinheitlichen. Und dabei ist halt dieses JIRA-**LFS**-Projekt entstanden, was heute so für die Gesamtplanung, oder für das Issuetracking, Aufgabentracking verwendet wird. 00:04:27-2

I: Seit wann gibt es jetzt dieses **LFS**-Projekt? 00:04:28-6

B: Seit Anfang des Jahres. 00:04:32-3

I: Und davor war das ja irgendwie in dieses KKZ und das Komposit-Projekt getrennt... 00:04:40-4

B: Genau, jede Sparte, oder jedes Team hatte quasi sein eigenes Projekt. Also die Fachkonzeption hatte für den Bereich 'Leben' ein eigenes Projekt, und für den Bereich Komposit ein eigenes Projekt. Und auch die Entwicklungsteams hatten jeweils ihre eigenen Projekte. Und die haben sich dann so projektübergreifend versucht sich so die Tasks zuzuschieben, und das hat zu einer Menge Verwirrung oder Unstimmigkeiten geführt.

Und so kam man dann auf die Idee, das zu vereinheitlichen, oder in einem Projekt einzugliedern. [00:05:14-5](#)

I: Wie wurde das dann.. oder welche Abteilungen waren dann da eben beteiligt an dieser Vereinheitlichung? [00:05:22-5](#)

B: Also getrieben wurde das von uns, also vom Qualitätsmanagement. Und beteiligt waren die Entwicklungsabteilung, die Fachkonzeption, und - mit den jeweiligen Teams, also komplett die Teamleiter - im ersten Schritt, und auch das Testteam. Wir haben dann in mehreren Workshops uns zusammengesetzt und überlegt wie müsste das aussehen, welche verschiedenen Aufgaben gibt es überhaupt, was für Workflows sollte man da hinterlegen, was für Informationen brauchen wir. Und das haben wir dann so Schritt für Schritt verfeinert. [00:05:57-1](#)

I: Das wurden dann diess QS-Sachen und so zum Beispiel. Diese Prozesse die jetzt im JIRA abgebildet sind. [00:06:03-1](#)

B: Genau. [00:06:06-3](#)

I: Und wurde das jetzt schon evaluiert wie das hinsieht, oder ist das geplant für die Zukunft? [00:06:24-4](#)

B: Ja. Also wir haben pro Produktionsabschnitt oder pro Meilenstein - wie man immer es auch jetzt nennt bei **VersicherungsGmbH** - immer einen Reviewtermin in dem wir durchgehen was ist bisher so.. was hat sich bewährt, was ist weniger gut. Wir haben uns überlegt wir wollen jetzt für den nächsten Produktionsabschnitt nicht zu viele Änderungen immer einführen, sondern dann testweise eine Änderung durchführen, kucken "hat sich die bewährt oder nicht?", wenn sie sich bewährt hat dann wird sie komplett für den Rest übernommen, ansonsten wieder zurückgestellt. Das haben wir bisher jetzt einmal im grossen Rahmen durchgeführt. Die nächste Reviewrunde wäre jetzt nach Ende dieses Produktionsabschnittes, also irgendwann innerhalb glaube ich der nächsten anderthalb Monate, um zu kucken was hat sich bewährt, was wollen wir anpassen, was müssen wir ändern. [00:07:22-3](#)

I: Und hat es da, als das JIRA eingeführt wurde an breiterer Front, hat es da Schulungen gegeben für die Leute, oder wie ist das abgelaufen - die Einführung von JIRA selbst? [00:07:33-7](#)

B: Ja, es hat dazu Schulungen gegeben, die hat mein Kollege durchgeführt der im Moment gerade im Urlaub ist. Zum Einen sind dafür Handbücher erstellt worden, ich weiss nicht ob du die kennst? [00:07:47-0](#)

I: Habe ich gesehen zum Teil, die sind ja gleich beim Einloggen. [00:07:49-2](#)

B: Ja, genau. Und auf Basis dieser Handbücher wurde dann mit jedem Team eine Schulung durchgeführt. Allerdings danach, also eine auffrischende Schulung hat es nicht gegeben. Nur den Hinweis, dass wenn es Probleme gibt sich die Leute direkt an mich wenden können und ich dann zu Verfügung stehe. Wobei ich auch nicht weiss, inwieweit das tatsächlich dann noch im Bewusstsein der Kollegen ist. [00:08:17-3](#)

I: Und ist zum Beispiel geplant, dass das JIRA auch mal das TTS ablösen soll oder so? Für das Bugtracking? [00:08:24-0](#)

B: Ich weiss nicht wie die momentanen Planungen sind. Das war früher mal im Gespräch. Das Problem ist halt, dass wir an das TTS, das wir im Moment benutzen, sehr viele andere Systeme noch mit angebunden haben. Und deswegen ist es nicht so einfach abzulösen. Aber vom Grundsatz her ja. [00:08:45-6](#)

I: Die anderen Systeme das sind jetzt REA, und diese ganzen Sachen. [00:08:49-1](#)

B: Genau, und was da sonst noch alles dranhängt. Ich habe da selbst keinen Überblick da drüber. [00:08:58-7](#)

I: Wie ist das jetzt - wenn man sich die Firmenprozesse anschaut - es ist ja glaube ich das JIRA selbst noch nicht.. quasi über alle Abteilungen jetzt drinnen. Also es ist... eben der Grossteil von dem Workflow ist ja glaube ich trotzdem zwischen eben Fachkonzeption und Entwicklung. [00:09:18-1](#)

B: Genau. Plus Test. [00:09:18-9](#)

I: Ja, aber Test zum Beispiel, die haben ja glaube ich wieder ein eigenes Testsystem, wo sie irgendwie ihre JIRA-Tasks in das Testsystem reinstellen, und das dann wieder innerhalb von dem Testsystem abwickeln, oder? [00:09:31-0](#)

B: Sie führen ihre Tests innerhalb des Testsystems durch. Also es gibt kein extra Aufgabentracking ausserhalb von JIRA. Und das einzige was sie machen ist halt, sie kriegen geliefert: "hier, da ist etwas fertig implementiert,

das muss getestet werden". Dann nehmen sie diese Information und starten damit ihre Testfälle. Sie protokollieren, oder dokumentieren trotzdem die Ergebnisse in JIRA. Also es ist jetzt nicht so, dass sie dann in ein anderes Tool wandern und da ihre Aufgabenplanung machen oder so, sondern sie nutzen halt unser Testmanagementtool um den Test durchzuführen. Weil dafür ist JIRA nicht gedacht. [00:10:08-6](#)

I: Ok. Und am anderen Ende jetzt von der Produktionskette, beim Produktmanagement... [00:10:18-4](#)

B: Genau. Auch die stecken mittlerweile im Prozess mit im Prozess drin. Also.. die haben ja diese Liste von.. immer als Hauptpunkt so ein Thema. Da darunter hängen ja sehr viele Unteraufgaben, und - geht glaube ich im ersten Schritt los mit Auftrag und SchätZRaster. Beim SchätZRaster zum Beispiel, da hängt am Anfang das Produktmanagement drin als Auftraggeber was auch die SchätZRaster jeweils abnimmt, beziehungsweise wieder zurückgibt an die Entwicklung. Genauso gibt es auch einen Task der sich "Abnahme von Produktmanagement" oder so nennt, auch da ist das Produktmanagement mit drin. [00:10:56-7](#)

I: Und diese Tasks, die erstellt aber glaube ich die Fachkonzeption, soweit ich das mitgekriegt habe. Also diesen SchätZRaster-Task. Also rein jetzt im JIRA selbst. [00:11:05-0](#)

B: Genau. Sofern einmal festgelegt wurde "was wollen wir uns für diesen Produktionsabschnitt alles ankucken" geht die Fachkonzeption los und erstellt halt quasi für diese Themen jeweils die Standardtasks die sowieso immer bearbeitet werden müssen. (...) Was aus der Historie einfach so entstanden ist, weil das Produktmanagement am Anfang noch nicht mit eingebunden war in diesen Prozess. Weil die davon noch nichts wissen wollten (lacht). Also rein theoretisch wäre es auch möglich dass das Produktmanagement auch direkt hingeht, und das Importtool was wir dafür haben anschmeisst, und das dann nicht die Konzeption macht, sondern das Produktmanagement. [00:11:51-1](#)

I: Dieses Importtool.. Oder wie werden diese Tasks im vorhinein geplant? Wird das mit Project oder Excel oder.. irgendwie gemacht? Oder was importiert dieses Importtool? [00:12:05-7](#)

B: Das Importtool importiert eine Excelliste. Die dann.. [00:12:14-1](#)

I: Die wird dann direkt in das JIRA 'reingeschossen'. [00:12:14-6](#)

B: Genau. Da gibt es.. hat eine ehemalige Auszubildende bei uns hat als Abschlussarbeit das Importtool entwickelt, und auf Excel-Basis werden die entsprechenden Tasks dann angelegt. [00:12:33-3](#)

I: Und wie ist das aus deiner Sicht, wie hat sich das bisher bewährt, das ganze JIRA da. Also eben... [00:12:43-8](#)

B: Also ich bin der Meinung.. also es ist.. wie sagt man so schön: "Es ist nicht alles Gold was glänzt", aber es ist trotzdem ein grosser Fortschritt zur Situation vorher. Und sicherlich gibt es viele Stellen noch wo man etwas verbessern, etwas anpassen kann, und was vielleicht auch sich einfach noch nicht so im Arbeitsalltag widerspiegelt. Weil es ist halt eine Veränderung und nicht jeder nimmt Veränderungen gerne an. Und manche sind da eher aufgeschlossener und manche weniger aufgeschlossen. Das sind halt so die üblichen Probleme mit denen man dann kämpfen muss, und wo man... [00:13:22-8](#)

I: Und was waren zum Beispiel die Probleme die es vorher gegeben hat? [00:13:24-8](#)

B: Also das grösste Problem war, wenn die Fachkonzeption im Fachkonzept-Projekt gearbeitet hat, und dort auch seine Fachkonzepte, etcetera, also die Dokumente wirklich an den Tasks hinterlegt hat, und dann gesagt hat "hier das ist fertig" oder "das muss jetzt zur QS an die Entwicklung". Dann hatte die Entwicklung zum Beispiel keinen Zugriff auf dieses Projekt. Das heisst, die haben dann ein Email gekriegt mit dem Dokument darin, haben das in ihr Projekt importiert, haben das hinterlegt. Haben dann ihren QS-Task bearbeitet.. [00:13:56-8](#)

I: (unterbricht) Das war als es diese zwei.. als es das Komposit und das KKZ-Projekt gegeben hat. [00:14:00-5](#)

B: Genau. Und so ging das dann hin- und her, und inzwischen hatten sich da die Versionen, etcetera schon verändert. Und dadurch dass wir das jetzt wirklich an einer Stelle haben, und das wirklich festgelegt ist "ein Fachkonzept liegt immer.." oder "das einzig gültige Fachkonzept liegt auch an diesem Fachkonzept-Task" hat sich an der Stelle auf jeden Fall eine Menge verbessert. [00:14:22-4](#)

I: Ok. Und ja, gibt es Pläne eben das... diese Produktionskette - weil ich vorhin gesagt habe dass Produktmanagement zuerst nicht eingebunden war - dass man das nochmal revidiert oder so? Oder nicht revidiert, aber, ich weiss nicht, den Prozess noch weiter anpasst in diese Richtung. Oder zum Beispiel, ich weiss nicht, Releasemanagement die sind ja mehr so über dieser ganzen Produktionskette. Dass man die zum Beispiel auch einbindet da. [00:15:02-7](#)

B: Also Releasemanagement ist an der Stelle noch nicht dafür geplant, weil deren Einsatz eigentlich erst - jetzt zumindest was die reine Umsetzung dieser Themen angeht - erst.. ja, wenn das abgeschlossen ist aktiv werden. Von daher ist das noch nicht geplant, weil es für den momentanen Einsatzzweck noch keinen Mehrwert bringt. Sondern eigentlich das Projekt eher noch unübersichtlicher macht. Da wir aber im Moment dabei sind, überhaupt unsere Entwicklungsprozesse und das Vorgehen an sich zu überarbeiten, kann es durchaus sein, dass es da noch ganz andere Anpassungen gibt. Also wir sind auch im Moment dabei, als Ergänzung zu JIRA noch 'Confluence' implementieren zu wollen, und damit quasi dann auch die Anforderungen etwas zentraler dokumentieren zu können, und auch die Verknüpfungen zu den jeweiligen Tasks besser hinkriegen zu können. [00:16:06-7](#)

I: 'Confluence' ist auch von Atlassian, oder? [00:16:08-9](#)

B: Genau. Das Wiki. [00:16:13-5](#)

I: Ach, das ist das was **PQM_2** macht. Der sieht sich doch das Confluence an. [00:16:19-1](#)

B: Ja, genau. Allerdings in einem anderen Zusammenhang, aber genau das ist.. [00:16:24-1](#)

I: Im JIRA gibt es ja zum Beispiel auch die Möglichkeit, dass man das mit CruiseControl und so Entwicklungstools verknüpft. Das passiert ja jetzt momentan noch nicht. Also momentan... [00:16:36-0](#)

B: Also die einzige Verknüpfung die wir im Moment haben ist mit Subversion. Dass beim Einchecken die JIRA-Id mit angegeben wird, und man sich dann zu jedem Task dann anschauen kann was wurde dazu eingchecked. Es gibt meines Wissens auch eine Verknüpfung zu 'Fisheye'.. [00:16:55-0](#)

I: Was macht das nochmal? [00:16:55-6](#)

B: Keine Ahnung (lacht). Ich weiss auch nicht, wer das im Unternehmen tatsächlich benutzt. Das ist glaube ich auch so eine Art Analysetool für Repositories mit dem man sich anschauen kann.. was weiss ich.. wer hat wieviel eingchecked, und was hat sich im Repository verändert. Auch ein Tool von Atlassian. [00:17:18-2](#)

I: Aber habt ihr jetzt schon Lizenzen für diese ganzen Atlassian-Sachen? [00:17:22-3](#)

B: Ja. [00:17:23-2](#)

I: Also nicht nur JIRA, sondern die ganze... [00:17:26-3](#)

B: Also wir haben für.. nein, nicht für alle. Wir haben für Confluence, allerdings nicht eine unternehmensweite Lizenz, sondern nur so eine kleine. Weil wir es im Moment noch am Testen sind. Wir haben Fisheye-Lizenzen, und ich glaube, dann gibt es ja noch Clover oder so heisst das... da bin ich aber auch überfragt. [00:17:48-5](#)

I: Clover ist ja glaube ich so ein Test-Tool, oder? [00:17:50-1](#)

B: Ich, wie gesagt... [00:17:52-5](#)

I: Ich kenne das auch nur irgendwie... [00:17:54-4](#)

B: Es ist halt auf der Internetseite immer mit drauf, und ich habe irgendwann gesehen dass wir das für auch Lizenzen haben. Aber ich weiss auch nicht, wer das im Unternehmen tatsächlich benutzt. Oder ob es noch jemand benutzt, und einfach nur.. [00:18:07-6](#)

I: Ok. [00:18:18-2](#)

B: Also ich glaube jedenfalls dass da noch eine Menge Potential und Unterstützungsmöglichkeiten drin steckt. (...) Die Frage ist halt nur, ob das so die grössten Probleme sind die wir im Moment haben, oder ob wir nicht andere Baustellen haben die man erst mal angehen muss bevor man in der Richtung weiter optimiert. [00:18:37-8](#)

I: Was wären das leicht für andere Baustellen? Oder... jetzt so im Projektfluss.. [00:18:48-7](#)

B: Ja ich glaube, dass die Baustellen eher um die Projekte herum liegen im organisatorischen Bereich. Und dann ist.. macht das wenig Sinn, sich über irgendwelches Tooling Gedanken zu machen. Sondern da muss man auf der Ebene erstmal aktiv werden um die Rahmenbedingungen zu schaffen. Weil egal welches Tool ich verwende, es soll ja immer meinen Prozess bestmöglich unterstützen. Und ich versuche nach Möglichkeit nicht mich.. mich nicht nach einem Tool zu richten, sondern das Tool meinem Prozess anzupassen. Und solange der noch nicht ausgereift ist, oder wenn die Vision dafür noch nicht vorhanden ist, macht es dann wenig Sinn da noch weiter.. darauf aufzubauen. [00:19:35-1](#)

I: Ok. Dann dankeschön mal. [00:19:42-3](#)

B: Gerne. [00:19:44-7](#)

I: Vielleicht kannst du mir noch sagen wie du das JIRA für deine eigene Arbeit jetzt verwendest. Oder welchen Nutzen du daraus ziehst jetzt. [00:19:56-7](#)

B: Also für mich persönlich ist es eine grosse Hilfe zur Fokussierung. Also ich habe auf meinem Dashboard dann so die verschiedenen Projekte an denen ich arbeite, und habe die einzelnen Aufgaben da drin. Und sehe halt auf den ersten Blick was habe ich denn noch zu tun. Und was ist wirklich erst mal wichtig oder nicht. Weil häufig hat man es ja doch, dass man so bestimmte.. ich sage mal Lieblingsthemen hat um die man sich eher kümmert, aber wenn ich halt jeden Tag direkt vor der Nase habe "ich muss mich trotzdem noch um etwas anderes kümmern", dann hilft es einem doch sehr sich dazu zu motivieren das endlich vom Tisch zu kriegen. [00:20:35-4](#)

I: Und hast du auch Projekte die du gemeinsam mit anderen Kollegen von dir im Team bearbeitest, und... [00:20:39-7](#)

B: Genau. [00:20:42-9](#)

I: ... und wie nützt du das dann aus, das JIRA, dass du... ich weiss nicht.. Schaust du ins JIRA zum Beispiel diese Kommentarfunktionen und so rein, oder bestimmte Filter oder so, damit du siehst was deine Kollegen machen? [00:20:51-0](#)

B: Was meine Kollegen machen interessiert mich.. also eher sekundär. Solange es nicht meine Aufgaben betrifft. [00:21:00-9](#)

B: Aber ich habe ein Dashboard das auf meine verschiedenen Projekte ausgerichtet ist. Und das mir auch anzeigt wann hat denn an den Aufgaben an denen ich beteiligt bin, oder die mich interessieren - also die ich beobachte oder sonstwie - was es da für Veränderungen gegeben hat. [00:21:20-8](#)

B: Und genauso auch Filter die mir anzeigen was... was ist in den letzten... ich glaube in den letzten vier Tagen an Aufgaben für mich erstellt worden, so dass ich immer sehe was kam jetzt noch dazu. [00:21:34-2](#)

I: Wer erstellt für dich die Aufgaben? Machst du das selbst oder bekommst du die von aussen? [00:21:37-4](#)

B: Das ist unterschiedlich. Je nach Projekt. Und auch je nach Vorgehensweise. Häufig ist es so, dass du.. die Haupttasks von jemandem kommen, und dann die Teilaufgaben von mir selbst erstellt werden. Aber das ist halt auch unterschiedlich. [00:21:57-9](#)

I: Ok. Ja na gut. Dann dankeschön mal [00:22:06-7](#)

B: Gerne gerne.

(zweite Aufnahme mit **PQM_1** - Nachtrag zu Confluence)

I: Also Confluence. [00:00:04-2](#)

B: Confluence als.. als quasi gemeinsame Projektschnittstelle zum Kunden. Über die der Kunde seine Anforderungen einreichen kann, die auch dort mit ihm quasi diskutiert werden, abgestimmt werden. Und im nächsten Schritt dann aber auch die Basis bilden für den hinterher ablaufenden Entwicklungsprozess. Das heisst, man hat dort seine gemeinsame Projektstruktur in Confluence, die dann auch die zugehörigen JIRA-Tasks widerspiegelt und den Bearbeitungsstatus, etcetera. So dass man eine zentrale Stelle hat die immer.. ja.. tagesaktuell anzeigt wie ist der Stand im Projekt. . Und welche Aufgaben befinden sich gerade in welchem Status. Welche Anforderungen sind gerade getestet, welche sind fertig. Wann sind die nächsten Releasetermine. Und so weiter. Und Meetingprotokolle, etcetera. [00:00:59-5](#)

I: Ok. Und momentan gibt es da wahrscheinlich eine Menge Word-Dokumente wo das alles drinsteht. [00:01:02-8](#)

B: Genau. Im Moment liegt das alles im Filesystem und wird dann auch per Email hin- und hergeschickt. Also wir haben kein gemeinsames Filesystem für den jeweiligen Kunden und **VersicherungsGmbH**, sondern es geht dann "wir haben hier unser Filesystem und der Kunde hat seine Seite", und da gehen dann die Email hin- und her. [00:01:24-4](#)

I: Aber es wäre dann so geplant dass quasi ein zentraler Server dann steht wo das alles drüberläuft. [00:01:29-0](#)

B: Genau. Wo dann die einzelnen Kunden ihre eigenen Bereiche haben und die einzelnen Projektabschnitte und so weiter. Genau. 00:01:36-8

I: Und das ist jetzt aber auch noch in der Erprobung quasi. 00:01:42-5

B: Genau. Das ist in der Planung. Wir wissen auch noch nicht so genau ob wir Confluence an dieser Stelle wirklich als Mittel der Wahl sehen, oder ob es nicht vielleicht noch andere Plattformen gibt die dafür genutzt werden können. 00:01:57-0

I: Aber wie ist das dann zum Beispiel, irgendwie... wenn jetzt sind ja (unv. Kunden).. das ist ja quasi ein Vertrag, oder? Also so ein Word-Dokument wo drinsteht.. der Anforderungskatalog zum Beispiel. Das wird ja von beiden Seiten unterschrieben. Wie funktioniert das dann mit einem Wikisystem, das ja immer geändert werden kann? 00:02:17-0

B: Auch da kann ich ja hingehen und sagen... 00:02:19-0

I: ... Das ist so eine Art Versionierung, oder so.. 00:02:20-6

B: Genau. Ich habe mein Konzept, das ist jetzt abgestimmt mit den entsprechenden Beteiligten, und dann kann ich dieses Konzept ablegen. Und ich kann natürlich auch die entsprechende Abnahmeerklärung dazu.. kann ich in Confluence ablegen. Ob ich es nun.. es muss ja nicht als Wiki-Seite sein, sondern auch als PDF-Dokument zum Beispiel. Und dann habe ich halt den Status der abgestimmt wurde habe ich vorliegen, und dann kann aber zusätzlich noch die gesamte Diskussion und Kommunikation die dazu geführt hat, dass es im Endeffekt dieses Dokument geworden ist... habe ich ja trotzdem noch mit Vorhalten und hinterher darauf verweisen und nachvollziehen warum hat man sich an der Stelle für dieses oder jenes entschieden. Das ist der grosse Vorteil. 00:03:15-6

B: Gerne.

Interview 12_ENTW_DE_TL_STV_2

I: Was ist deine Aufgabe im Team und welche Arbeiten erledigst du jeden Tag? 00:00:15-1

B: Ja ich bin eigentlich Hauptentwickler im Team Komposit, entwickle momentan halt die KFZ-Sparte mit, hauptsächlich, nebenbei auch noch die Sparte Hausrat. Nebenbei.. dann mache ich auch noch stellvertretend die Teamleitung, und mache sehr viel Kundenkommunikation, Absprache mit den Kunden, Fehlerverbesserungs.. fachlich. 00:00:37-3

I: Mit welchem Kunden bist du da unterwegs? 00:00:40-7

B: **Kunde_1**. 00:00:43-3

I: Nein, aber mit **Mitarbeiter_Kunde_1_1** jetzt, oder.. 00:00:44-6

B: Ja, **Mitarbeiter_Kunde_1_1**, **Mitarbeiter_Kunde_1_2**, **Mitarbeiter_Kunde_1_3**, **Mitarbeiter_Kunde_1_3**.. 00:00:49-3

I: Die sind von der fachlichen Seite, oder.. 00:00:52-3

B: Die sind von der fachlichen Seite, ja. Im Bereich KFZ-Sparte und.. 00:00:59-6

I: Und wie kommunizierst du mit denen? 00:01:00-6

B: Per Telefon und vor Ort. 00:01:03-0

I: Vor Ort? 00:01:02-7

B: Ziemlich viel vor Ort. Wir machen auch sehr viel telefonisch, aber wir treffen uns jetzt mittlerweile alle zwei bis drei Wochen in **Süddeutschland**, um dieses KFZ-Projekt jetzt endlich mal zu einem Ende zu bringen. 00:01:14-3

I: Da werden jetzt eben so offene Fragen besprochen, oder wie? 00:01:18-2

B: Ja, Fragen, Probleme, "wie gehen wir da weiter vor", und so. Unser aktuelles Problem ist halt momentan **xxx**. Das ist momentan ein grosses Thema was Probleme macht. Und wie wir das zu lösen haben. 00:01:35-1

I: Bei diesen Treffen in **Süddeutschland** da ist dann auch immer **FK_1** glaube ich mit, oder? Oder irgendjemand vom Fachkonzept? 00:01:42-8

B: Also momentan sind da kaum welche mit dabei. Das mache alles ich. 00:01:47-8

I: Alles du, ok. Und wie geht das dann von dir aus weiter an das Fachkonzept? 00:01:52-4

B: Wenn wir die Sachen besprochen haben, die fachliche Sache wie das zu laufen hat oder wie wir es ändern sollen, dann setze ich mich halt hier in **Norddeutschland** mit den entsprechenden Fachkonzeptern zusammen und schreiben das dann auf. Schreiben das halt zusammen auf, und lassen das dann vom Kunden nochmal abnehmen. 00:02:07-8

I: Über JIRA läuft das wahrscheinlich nicht, oder? 00:02:14-3

B: Nein, über JIRA läuft DAS nicht momentan. Also wir hatten das über JIRA zu Anfang gemacht, da waren ein paar (unv. Proble..) Sachen, aber das ist halt... momentan ist das leider so viel, dass wir das gar nicht erfassen können. 00:02:27-8

I: Ok, da wird dann ein Word-Dokument zusammengeschrieben, oder wie läuft das dann? 00:02:30-3

B: Wir machen entweder Word-Dokumente, oder halt über Tickets, über das Ticketsystem. Die sind aber.. die werden leider nicht nach JIRA übertragen momentan. Also hauptsächlich sind das die Tickets die dann wieder aufgemacht werden, wo die Probleme beschrieben werden. Die gehen aber momentan nicht nach JIRA rüber. 00:02:46-9

I: Die werden direkt mit dem Fachkonzept.. da wird dann quasi ein Ticket eröffnet, oder wie? 00:02:50-9

B: Nein, das wird vom Kunden eröffnet. Wir haben das dann meistens fachlich und technisch sowieso schon besprochen wie wir es haben wollen, und dann wird das dann dementsprechend bei uns in den Konzepten nachgearbeitet. Und die Konzepte kommen dann halt den Weg über JIRA wieder natürlich zur Abnahme und sowas dann. Aber wir haben ja unser.. wir haben halt unser anderes Ticket ja auch noch, in der Notes-Datenbank, und da bleiben sie auch noch vorerst. Die werden ja vorerst noch nicht nach JIRA übertragen. Weil in JIRA haben wir ja momentan nur die Konzepte, Aufgabenplanung und so, aber keine Bugs.. also da verwalten wir unsere Bugs halt noch nicht drin. Daher arbeite ich momentan recht wenig in JIRA. 00:03:32-3

I: Das haben schon ein paar gesagt. Und mit welchen Leuten arbeitest du jetzt innerhalb vom Team jeden Tag zusammen, oder am meisten? 00:03:42-9

B: **ENW_DE_8**, **ENW_DE_TL_STV_1**, mit den **österreichischen** Kollegen dann halt noch, und das war es dann schon auch momentan. Alle anderen.. die sind momentan nicht in der KFZ-Sparte involviert. Und fachlich sind das halt noch Hauptansprechpartner momentan noch **FK_2**. 00:04:04-5

I: Für das Fachkonzept. 00:04:06-4

B: Ja. Und **ARCH_4** noch vom Architekturteam. 00:04:09-7

I: Wie oft triffst du zum Beispiel **ARCH_4** oder so? Oder wie oft redest du mit denen? 00:04:16-4

B: Zwei-, dreimal die Woche. 00:04:17-4

I: Und was wird da besprochen jetzt mit Architektur? 00:04:19-9

B: Wie jetzt beispielsweise diese Point-of-Sale-Geschichten - also das sind all die Anträge rein die vom Vermittler kommen. Da gibt es Probleme mit **xxx**, und da haben wir jetzt uns auch schon zusammengesetzt und schon mal überlegt wie wir das alles besser machen können. Die Suchen.. Aber das wird demnächst alles weiter ausgearbeitet. Also der ist halt der technische Ansprechpartner für diese Schnittstellen Point-of-Sale und EVB. 00:04:41-7

I: Das wird aber auch alles quasi mündlich in Besprechungen ausgearbeitet? 00:04:45-5

B: Das wird alles mündlich ausgearbeitet, ja. Aufgrund von Tickets natürlich. Und dann halt entsprechend nachdokumentiert. 00:04:53-4

I: Was würdest du sagen, weil du eh gesagt hast du arbeitest mit den **Österreichern** zusammen, wie unterscheidet sich da die Arbeit am meisten jetzt. Im Vergleich.. also wenn es da jetzt irgendwie Probleme gibt oder so, zwischen **Österreich** und -wenn du da etwas besprechen musst - und **Norddeutschland**? 00:05:18-4

B: Naja, das ist ziemlich schwierig über Telefon. Also wir können halt momentan nur telefonisch kommunizieren - also von meiner Seite aus, weil ich in **Österreich** ja momentan gar nicht bin weil ich soviel in **Süddeutschland** bin - und daher versuchen wir eigentlich, Probleme die wir da jetzt haben nur noch telefonisch zu lösen. Beziehungsweise über die Tickets die sie dann natürlich auch bekommen, über die Analysen, die kontrollieren wir halt zwei-, dreimal nach, auch den Code technisch nochmal, und versuchen das dann dementsprechend dann alles vernünftig über die Bühne zu kriegen. 00:05:47-0

B: **ENW_DE_TL_STV_1** ist ja auch häufiger in Österreich noch. In Österreich haben wir ja die Hauptthemen nur noch **Thema-XY** liegen. Den Produktbereich von KFZ, der ist aber sehr sauber. Und **Thema-XY** liegt halt hauptsächlich noch da, und das macht eigentlich **ENW_DE_TL_STV_1** ja momentan irgendwie alles. 00:06:07-1

B: Klar habe ich auch meine Probleme mit **Thema-XY** und versuche.. aber das geht halt alles nur telefonisch. Ist aber allerdings sehr schwer, weil man das Verständnis dafür immer sehr.. schwierig rüberbringen kann wenn man nicht vor Ort ist. 00:06:22-0

I: Und welche Rolle spielt da eben dann das Ticketsystem oder JIRA dass man jetzt am Laufenden bleibt wie der Stand ist in **Österreich**? Dass man zum Beispiel merkt, ok, die haben bei irgendeinem Ticket Probleme oder so und kommen da nicht weiter? Oder dass man merkt es verzögert sich das geplante... 00:06:45-1

B: Naja, das ist halt das Problem bei dem Ticketsystem ist halt.. ist ja kein.. ist ja fast.. für mich wie JIRA.. Aber man sieht ja die Tickets, wir ziehen natürlich auch dem Ticketsystem jeden Tag, oder alle zwei Tage Statistiken raus und kucken halt wie das Wachstum ist von den Tickets, welchen Status die bekommen. Und wenn halt Tickets dann auch zu lange herumliegen, dann fragen wir auch entsprechend nach "was ist da los?", "wo sind die Probleme?" dann. Anders kommen wir da nicht ran. 00:07:08-2

I: Die Statistik aus dem Ticketsystem: wie macht ihr die? 00:07:10-5

B: Das weiss ich nicht. Das machen immer **ENW_DE_TL_STV_1** und **ENW_DE_TL**. Für unser Team machen sie die Statistiken, und **ENW_DE_AL** kann da noch mehr Auswertungen drüberfahren über das Ticketsystem. Das weiss ich aber nicht wie das funktioniert. Ich bin mir jetzt nicht sicher. Ich glaube das wird... die exportieren das irgendwie in Excel oder sowas, und dann kommen halt ein paar schöne Diagramme da raus. Aber genau weiss ich das gar nicht wie die das machen. Habe ich auch selber nicht gemacht. 00:07:37-7

I: Und wie oft schaut da dann zum Beispiel ins JIRA rein? Also schaut du zum Beispiel... verwendest du das JIRA, damit du nachschaust woran meinetwegen **ENTW_OE_1** gerade arbeitet oder so? 00:07:51-7

B: Jetzt bei neuen Themen mache ich das schon. Also ich kucke natürlich auch wegen meiner Rolle als stellvertretender Teamleiter dauernd mal rein. Was an QS noch reinkommt, Implementierungsaufgaben und hast-du-nicht-gesehen, da muss ich dann auch immer mal reinkucken. Das war es dann aber auch schon was ich momentan damit mache. Für mich selber verwalte ich da jetzt momentan wie gesagt nichts drin, weil es momentan für mich da drin nichts zu verwalten gibt. Ausser im Moment als stellvertretender Teamleiter die entsprechenden Tasks zu verteilen und im Auge zu behalten. Und dementsprechend immer mal nachzufragen warum das noch nicht fertig ist. 00:08:27-6

I: Schaust du zum Beispiel nach, woran die Fachkonzepter gerade arbeiten? 00:08:36-9

B: Nein. 00:08:39-6

I: Spielt das für dich eine Rolle was die gerade machen, die Fachkonzepter? 00:08:42-8

B: Eigentlich spielt das für mich jetzt nicht grossartig eine Rolle momentan. Also wie gesagt, ich bin seit zweieinhalb Jahren in diesem KFZ-Projekt drin, und ich mache überhaupt keine anderen Sparten mehr. Ich mache zwar halt noch, wenn jetzt QSen kommen jetzt für neue Sparten oder so, neue Sparten entwickelt werden, neue Konzepte die jetzt irgendwie in den Dialog-Bereich reingehen, die QSen wir dann schon mit. Ich QSe die auch mit, das machen wir zu zweit. Aber da kommt momentan halt nichts rein. Aber da können wir nachher sonst gerne noch durchgehen, vielleicht liegt da noch irgendwo eine rum. Aber woran jetzt die Fachkonzepter eigentlich genau arbeiten oder, oder was die in Planung haben, interessiert jetzt mich persönlich nicht wirklich. 00:09:20-1

I: Und mit der Testabteilung, wie sieht es da aus? Bist du da irgendwie am Laufenden was die so vorhaben? 00:09:26-2

B: Im Bereich JIRA jetzt? Oder.. 00:09:29-3

I: Puh, generell. Ob die jetzt zum Beispiel, ich weiss nicht, Testoffensive haben, oder was auch immer. 00:09:35-4

B: Nein. Bin ich überhaupt nicht auf dem Laufenden was die machen. Ich weiss nur dass die ziemlich viel Stress momentan haben, weil die ziemlich unterbesetzt sind noch, aber.. 00:09:46-3

I: Das weisst du weil.. die gleich nebenan sitzen? 00:09:48-6

B: Genau (lacht). Aber ich weiss jetzt nicht was die irgendwie in der Aufgabenverwaltung oder so machen, oder woran die genau arbeiten weiss ich nicht. 00:09:59-4

I: Dann eine Frage zu deiner Sametime-Kontaktliste: wie hast du die organisiert? Welche Gruppen... 00:10:11-4

B: Ich habe die Gruppen eigentlich nach Team sortiert. Eben die entsprechenden Teamverteiler von jedem, und die sind auch.. die habe ich alle als Gruppen in Sametime drinnen. Also jetzt nicht sortiert nach "Best Friends" oder nach Stufen, Kategorien so für mich selber, sondern einfach nach Team sortiert. Also die öffentlichen Teamadressen, und dadurch habe ich die. 00:10:30-5

I: Wie oft siehst du hinein, in Sametime? 00:10:34-8

B: Also ich melde mich morgens an, aber sonst kucke ich eigentlich nicht rein. 00:10:40-1

I: Aber kuckst du zum Beispiel nach ob deine ganzen Kollegen schon am Arbeitsplatz sind, oder irgendwo weg? 00:10:45-5

B: Nein. Also ich kucke da erst rein, wenn ich jemanden speziell haben will. Also entweder kucke ich da rein und

rufe ihn dann an, und wenn er nicht online ist und nicht erreichbar ist, dann kucke ich halt nochmal in den Betriebskalender rein ob er wirklich abwesend ist, krank, in **Süddeutschland**, oder was auch immer wo er sein könnte. Aber so den Sametime nutze ich persönlich eigentlich nicht so.. nicht so häufig. Da greife ich lieber zum Telefon. [00:11:12-3](#)

I: Frage: Verwendest du das MyLyn? [00:11:16-9](#)

B: Jein (zögerlich, lacht). Wir haben das eigentlich.. benutzen das zumindest eigentlich für Neuentwicklungen.. sollten wir das eigentlich nutzen. Es ist aber noch nicht so durchgesetzt überall. Und daher.. wie gesagt, ich bin seit einem halben Jahr nicht mehr in der Neuentwicklung, also seit einem halben Jahr benutze ich das nicht mehr. Aber wenn ich denn mal dazu kommen sollte wieder Neuentwicklung zu machen, dann werde ich es auch wieder nutzen. Also es ist echt.. es ist klasse. Also für Nachvollziehbarkeit, und codetechnisch was man geändert hat. Ist manchmal ein bisschen schwierig zu handeln, ich hoffe mal da kommen noch mal ein paar Updates oder so, aber ansonsten werde ich es wieder nutzen nachher. [00:11:51-4](#)

I: Und welche - wenn du es früher schon einmal gehabt hast - welche Tasks hast du dir dann anzeigen lassen? Da kann man ja auch so verschiedene Filter erstellen. Hast du nur deine eigenen Sachen dann, oder auch die von anderen Leuten gehabt? [00:12:03-7](#)

B: Ich habe.. hauptsächlich habe ich meine eigenen Filter gehabt - für mich meine eigenen Aufgaben. Und den Rest habe ich dann über die JIRA-Oberfläche gemacht. Also alles was so Verwaltung, Teamsachen angeht habe ich im Internet Explorer gemacht, oder Firefox, oder wie auch immer. Aber ansonsten, im Eclipse habe ich halt nur meine eigenen Tasks gehabt. Die auch mir zugewiesen sind. [00:12:23-7](#)

I: Hast du dann auch private Tasks oder so, oder Todos oder soetwas angelegt? In MyLyn oder in JIRA? [00:12:28-9](#)

B: Nein. Damals schon, in alten JIRA-Projekten konnten wir das ja noch machen, konnten die selber verwalten, aber bei den neuen Richtlinien können wir ja halt keine eigenen Tasks mehr aufmachen. Da gibt es ja nur noch Implementierungstasks die man einmal anlegt. Klar kann man, wenn man neue Aufwände hat oder so sich da eine Implementierungstask machen, aber so eine Todos-Liste, so ich sage mal so kleine Merker oder so, das gibt es bei uns im JIRA-Projekt ja nicht mehr. [00:12:57-0](#)

I: Dann eine Frage noch zum Ticketsystem: Wie nützlich findest du das, um Übersicht zu behalten über das Team oder über den Projektablauf? [00:13:07-7](#)

B: Ziemlich unübersichtlich. Also ich wäre schon froh, wenn wir das in JIRA hätten. Weil es wesentlich leichter zu verwalten ist in JIRA. Man kann über JIRA mehr.. also (unv. jeder) Statistiken ausdrucken, die Filter entsprechend sitzen und so. Und das ist mit dem Notes-System was wir da haben ist das halt ziemlich schwierig. [00:13:25-5](#)

I: Und welche Tickets siehst du dir an im Ticketsystem? Nur deine eigenen, oder.. [00:13:30-4](#)

B: Nein, eigentlich kucke ich mir alle an. [00:13:32-4](#)

I: Alle? Also von allen Kompositentwicklern? [00:13:34-9](#)

B: So gut wie, ja. [00:13:37-1](#)

I: Fachkonzepter auch? Weil da gehen ja auch ab und zu dann Tickets rüber. [00:13:40-2](#)

B: Die.. da.. Fachkonzept da schaue ich so ein-, zweimal die Woche rüber, ob da überhaupt irgendetwas neues aufgetreten ist, weil wir kriegen ja nicht unbedingt immer alles gleich mit. Also normalerweise gehen die Tickets ja im Prinzip erstmal an die Entwicklerteams, und die geben das an die Fachkonzepter weiter. Aber ab und an bei den Fachkonzeptern schaue ich dann mal ein-, zweimal die Woche rüber ob da noch irgendwelche Tickets herumliegen die demnächst auf uns zukommen könnten. Aber ansonsten... [00:13:59-6](#)

I: Das sind Tickets die sie selbst aufgemacht haben dann, oder? [00:14:03-6](#)

B: Naja, wenn ein Ticket von uns intern, vielleicht vom Testteam oder sowas, wenn die das fachlich sagen dass es verkehrt ist oder sowas, dann gehen die eventuell nicht gleich zu uns. Oder vom Kunden kommen dann welche die sie dann vielleicht abgesprochen haben die sie dann gleich dahin zugewiesen werden. Sowas kann ja auch mal passieren. Also wir haben ja im Prinzip keine grosse, feste Anlaufstelle wo die Tickets als erstes gesammelt werden. Regelfall ist eigentlich, dass die Entwickler, dass die die Tickets zuerst bekommen, und werden dann teilweise da an die Fachkonzepter zurückgegeben. Zur Nachbesserung. Aber es kann halt immer mal sein, dass die Tickets da von irgendwo anders noch reinkommen. [00:14:34-7](#)

(JIRA-Login)

I: Ist das das KMP-Standard-Dashboard, oder ist das ein eigenes.. [00:15:30-3](#)

B: Nein, das sind die.. Wir.. Ich habe da Dashboards habe ich mal unsere Standard-Dashboards die von **ENW_DE_TL** da bereitgestellt worden sind. [00:15:39-1](#)

B: (...) (sieht seine Tasks und hält kurz inne) Genau. Ich habe gerade ein bisschen mehr gekriegt, weil **ENW_DE_TL_STV_1** in Urlaub gegangen ist, ok. (lacht). [00:15:51-5](#)

B: Also das ist im Prinzip für mich jetzt hier dieses Hauptaugenmerk mal, für mich selber. [00:15:56-0](#)

I: Diese zweidimensionale Filterstatistik. [00:16:00-2](#)

B: Genau. Das ist eigentlich für mich. Da kucke ich eigentlich immer kurz rein was bei mir jetzt herumliegt aktuell. (...) Das ist klar. (...) [00:16:07-7](#)

B: Wenn ich jetzt noch Teamleiter-Sachen übernehme kucke ich mir halt im Prinzip direkt die Vorgänge an. (...) Ich hab jetzt meinen Link hier nicht.. [00:16:26-0](#)

I: Hast du normal einen Firefox-Link, oder was? [00:16:30-5](#)

B: (unv ja). [00:16:28-1](#)

B: Das ist im Prinzip hier das allgemeine **LFS**-Projekt wo wir uns ja nur noch mit beschäftigen. Was ich jetzt nochmal öfters dann immer kontrolliere sind halt für die beiden wichtigen Leute in dem Team noch. **ENW_DE_TL**, was da bei ihm noch aktuell immer rumliegt, beziehungsweise bei **ENW_DE_TL_STV_1**, weil da können ja im Prinzip auch die ganzen QSen, Fachkonzepte die gehen ja im Prinzip erst mal an die Teamleitung, beziehungsweise stellvertretenden Teamleiter. Das schaue ich mir dann eben an was bei einem dann.. [00:16:54-8](#)

B: Hier ist eigentlich der Haupt.. Augenmerkmal. Hier kommen sie im Prinzip alle an. Hier schaue ich dann mal nach was hier jetzt aktuell noch so rumliegt. Ob hier irgendwelche Sachen sind um die sich irgendwelche Entwickler noch kümmern müssten, wie hier zum Beispiel Testlisten QSen. (...) Also hier interessiert mich im Prinzip alles was noch zu QSen reinkommt. Also Testlisten-QS, Fachkonzept-QSen, da schaue ich dann rüber was herumliegt. [00:17:22-1](#)

I: Und das sind Sachen die dann eben an andere Entwickler weitergeleitet werden müssen. [00:17:25-6](#)

B: Genau, die leite ich im Prinzip hier nur weiter. [00:17:28-2](#)

B: Und die ganzen Schätzungstasks und technische-Konzepte-Tasks und sowas, da kümmern wir uns im Prinzip nachher ja erst zusammen erst drum. Sobald die ganzen Aufwände geschätzt werden, die Arbeitspakete definiert sind. Dann setzen wir uns halt zu dritt, oder mit **ENW_DE_TL** oder mit **ENW_DE_TL_STV_1**, oder auch einzeln, und verteilen die ganzen Sachen dann entsprechend auch noch. [00:17:48-2](#)

I: Ah (zeigt) du sitzt aber mit **FK_TL** da zusammen, oder wie? [00:17:50-6](#)

B: Nein, das ist ja nur der Autor. Da sitzen wir ja im Prinzip nur Team-intern zusammen. [00:17:56-0](#)

I: Nein, weil du gesagt hast, dass ihr euch zu dritt irgendwie zusammensetzt oder so für die Schätzung? [00:18:00-0](#)

B: Ja genau, teamintern. Das wird ja alles nur noch teamintern gemacht. [00:18:03-4](#)

I: Wer ist da dann involviert? [00:18:04-5](#)

B: Alle vom Team. Also jeder der da seine QSen da macht - der schätzt sie dann ja auch - und dann erstellt der da auch die entsprechenden Arbeitspakete. [00:18:12-0](#)

I: Also wenn du da **ENTW_OE_1** etwas zuweist, eine QS oder so... [00:18:13-9](#)

B: Genau, zum Beispiel. [00:18:14-7](#)

I: Dann muss **ENW_OE_1** das selbst schätzen, oder.. 00:18:16-2

B: Ja genau. 00:18:17-2

I: Oder du gemeinsam mit ihm? 00:18:18-9

B: Ja, eigentlich machen wir das zu zweit, wenn wir das hinkriegen, aber meistens haben wir zeitlich.. macht das eigentlich nur einer. Aber die ganzen Arbeitspakete laufen dann ja bei **ENW_DE_TL** alle auf, und wir importieren die dann dementsprechend nach JIRA, und verteilen die dann auch dementsprechend. Weil deswegen liegen ja hier auch noch so viele Schätzungstasks herum und... Aber die bleiben erstmal hier liegen. 00:18:36-7

B: Im Prinzip das gleiche Vorgehen mache ich dann eigentlich mit **ENW_DE_TL_STV_1** nochmal, ob da irgendetwas wichtiges herumliegt. 00:18:44-3

I: Wo ist da zum Beispiel.. Da hast du irgendwie jetzt 'Schätzraster'. Das betrifft eigentlich nicht euch jetzt, oder? 00:18:55-7

B: Nein. Ich weiss nicht.. ich habe es nicht.. Die liegen bei uns, weil die irgendwie gemacht werden müssen zusammen mit der Fachkonzeption noch. Genau weiss ich das nicht, deswegen liegen die alle bei **ENW_DE_TL** noch herum. Deswegen liegen hier ja so viele Sachen noch herum, weil **ENW_DE_TL** noch.. (Anm.: **ENW_DE_TL** ist zum Zeitpunkt des Interviews in Urlaub) 00:19:13-2

B: Beziehungsweise hier liegen dann auch noch so Implementierungssachen, die dann erstmal schon verschoben worden sind, damit die von den Entwicklern weg sind, damit du da entsprechend die Auswertung auch fahren kannst. Wieviele Tasks haben sie noch, und sowas. Aber das sind alles im Prinzip ziemlich viele Sachen weil die erstmal alle von dem Meilenstein her verschoben worden sind. 00:19:33-3

B: Die ganzen Schätzungs- und Implementierungssachen haben wir zwar schon mal gemacht, aber die können wir halt nicht schliessen, weil die irgendwie verschoben worden sind, oder jetzt in 'nachbearbeiten' sind. Deswegen liegen hier auch so viele noch herum bei **ENW_DE_TL**. 00:19:43-0

B: Ja und mit den Schätzrastern, das weiss ich nicht. Da musst du ihn vielleicht nochmal fragen in den Interviews, was da jetzt genau dahintersteckte, warum die jetzt bei uns liegen. Weil normalerweise gehören die nicht uns. Aber.. eigentlich haben sie sich da wohl geeinigt, dass wir da noch zusammen bearbeiten mit denen. 00:19:55-4

I: Aber was (unv. sagt) da jetzt, zum Beispiel wenn du da siehst 'Faktorenanpassung Wohngebäude'... (klickt) 00:19:59-6

B: Ja, wie gesagt. Ich weiss es nicht was es mit diesen Schätzrastern auf sich hat. Das ist eigentlich für die Fachkonzeption. Diese Schätzgeschichten. Warum die jetzt hier liegen kann ich dir echt nicht beantworten. 00:20:10-9

I: Ok. du hast mit Wohngebäude ja auch nichts zu tun. 00:20:14-3

B: Ja, ich.. Mit Wohngebäude: QSen ja, aber ich weiss nicht wozu diese Schätzraster jetzt in JIRA drin sind. Müsste ich wieder mal nachlesen, aber eigentlich haben wir damit nichts zu tun. Aber wie gesagt, wenn du **ENW_DE_TL** vielleicht nochmal interviews kann dir der das genauer sagen. Oder **FK_TL**. 00:20:36-9

I: Gut. Ja dann.. Wie navigierst du.. also du hast gesagt du gehst immer über dieses **LFS**-Projekt rein.. 00:20:44-7

B: Genau. Also wenn das jetzt für mich ist - also wie gesagt das sind hier meine Aufgaben die ich noch (unv. liegen) habe was QSen angeht - da habe ich mir jetzt keine eigenen Filter für gebastelt, weil ich das einfach so dann wesentlich umfangreicher mitkriege was so nach Filter nach Versionsnummern und so.. Und so hast du im Prinzip einmal eine komplette Übersicht. Auch was vielleicht für zwei Meilensteine später anfällt, das siehst dann hier mit auch. Ohne den Filter wieder zu ändern. 00:21:06-6

B: Und für mich selber habe ich eigentlich nur Filter gebastelt nach allen.. nach kompletten Themen. Hier sind zum Beispiel alle Wohngebäude-Themen. 00:21:18-9

I: Ein Thema ist immer so 'Neuanlage Änderung'... 00:21:20-8

B: Hauptaufgaben. Hauptaufgaben. Also diese Haupttasks in JIRA. 00:21:25-1

I: Also quasi was.. das ist immer ein Fachkonzept eigentlich. [00:21:28-6](#)

B: Genau. Das sind im Prinzip immer die Fachkonzepte. Danach habe ich im Prinzip.. das sind so meine Hauptfelder die ich jetzt so benutze. Dann noch hier meine Tasks nochmal so aufgelistet dass ich schneller hinkomme [00:21:39-1](#)

B: Und ansonsten benutze ich für meine Übersicht halt das Dashboard. (...) Das ist momentan eigentlich so das einzige was ich so mit JIRA mache. [00:22:00-8](#)

I: Können wir vielleicht in irgendeinen Task reingehen? [00:22:05-7](#)

B: Klar [00:22:06-6](#)

I: Fachkonzept QS. (...). Verwendest du zum Beispiel so verschiedene.. diese Überwachungs.. diese Funktion, dass du da eine Email bekommst wenn sich da etwas tut? [00:22:22-0](#)

B: Selten. Also ich nutze sie schon, ich kenne sie, aber ich nutze sie momentan halt sehr selten. [00:22:32-0](#)

I: Und wie kriegts du generell.. oder zum Beispiel angenommen es gibt jetzt eine neue Version vom Fachkonzept oder so. Oder irgendetwas zum QSen. Kriegt du da dann auch eine Email-Benachrichtigung, oder.. [00:22:41-7](#)

B: Ich bekomme eine Email, aber die gehen.. sind bei mir durch den Filter gehen die gleich weg. Also... [00:22:48-1](#)

I: Ein Lotus-Notes-Filter? [00:22:49-5](#)

B: Genau. Also die werden gleich wegsortiert. Die gehen zwar in einen Ordner rein wo ich sie mir anschauen kann - also sie landen nicht gleich im Papierkorb - aber ich kucke halt, wie gesagt, zwei-, dreimal am Tag über das Dashboard drüber, und dann schaue ich mir die an. Also deswegen war ich auch gerade so verwundert, dass da auf einmal 15 QSen sind. Weil ich Freitag noch reingekuckt habe (Anm: Interview war Montag), da waren noch keine QSen, heute (unv. war ich noch nicht da).. Also so an den Zahlen siehst du halt dann auch schon.. Wobei wenn da jetzt nur einer.. ein, zwei vielleicht kommen, dann könnten die vielleicht untergehen, aber die kommen dann spätestens im Dashboard von der Fälligkeit nach oben. Das man sich das dann anschauen muss. [00:23:23-8](#)

I: Ok. Und diese Kommentarfunktionen da unten.. wie häufig verwendet ihr die zum Beispiel? [00:23:28-9](#)

B: Also ich benutze die Kommentarfunktion eigentlich bei jedem Vorgang den ich bearbeite. Mache ich ein entsprechendes Kommentar rein. [00:23:37-6](#)

I: Da schreibst du quasi mit was du gemacht hast. [00:23:39-5](#)

B: Genau. [00:23:40-4](#)

I: Aber dass du jetzt zum Beispiel, ich weiss nicht, wenn du... [00:23:44-6](#)

B: Wenn ich auch Anmerkungen oder soetwas habe nutze ich die auch, klar. Also wenn ich jetzt irgendwie eine Anmerkung zu einem Task habe den ich irgendwie mal so durch Zufall irgendwo finde, dann wird eben.. schreibe ich da auch eben ein entsprechendes Kommentar zu. [00:23:54-9](#)

I: Fragst du dann.. schreibst du dann rein, keine Ahnung, "was ist mit dem-und-dem los?". [00:23:57-7](#)

B: Genau. Oder "wenn diese Implementierung abgeschlossen wird bitte noch das gegenkontrollieren" oder irgendwie sowas. [00:24:04-2](#)

I: Und schreibst du da dann zum Beispiel noch ein Email dann zusätzlich, oder verlässt du dich darauf dass das.. [00:24:08-0](#)

B: Da verlasse ich mich dann darauf dass die Leute das auch lesen. [00:24:11-1](#)

I: Ok. Und das geht aber wahrscheinlich einher damit, dass du es irgendwem anders dann zuweist, oder? [00:24:16-6](#)

B: Ja, genau. Oder es ist irgendjemand anders zugewiesen, dann kriegt der ja auch noch eine Email-Benachrichtigung dass das geändert worden ist. Das kriegts du ja auch: wenn das jemand anders bearbeitet, kriegts du ja auch eine Email-Benachrichtigung. Und ansonsten gehe ich davon aus, wenn sich jemand um einen

Task kümmert, dass sich die auch die entsprechenden Kommentare dazu mit durchliest. (...) Das hatten sie ja auch extra wieder.. das wurde ja ganz zum Anfang haben sie das glaube ich ausgebaut bei einer Bearbeitung. Da hatten wir keine Email-Benachrichtigung. Und einige Leute verlassen sich halt auf diese Email-Benachrichtigungen und kucken sich halt dann auch die Task an. Aber ich denke mal, dass jeder der den Task sich in Arbeit nimmt, sollte sich auch immer die Kommentare dazu durchlesen, und nicht nur die Beschreibung. [00:24:57-1](#)

I: Ja aber eben den Email-Filter in Notes haben ja mehrere Leute. [00:25:00-8](#)

B: Ja genau. Es könnte.. wie gesagt, wenn es halt gerade um diese Planung geht oder die ganzen Arbeitspakete die da irgendwann einmal importiert werden, dann kommen halt da - Hausnummer - sechs- oder siebenhundert Emails aus JIRA raus. Und dann kannst du da nicht jede einzelne noch wegsortieren oder ankucken. [00:25:17-5](#)

I: Dann.. wie findest du zum Beispiel irgendwie Fachkonzepte wieder die du öfter brauchst. Hast du die gebookmarket, oder suchst du dann im JIRA direkt nach einem Fachkonzept? [00:25:34-6](#)

B: Also dafür habe ich halt meine Hauptaufgaben gemacht (klickt). Also da, wie gesagt, wenn ich jetzt hier (klick) da sollten eigentlich alle Hauptaufgaben drinnen sein. Auch schon sortiert nach Name. Darüber suche. Und wenn ich darüber nichts finde, suche ich halt nach der Fachkonzept-Nummer. Also wenn ich hier zum Beispiel (beginnt zu tippen) die Nummer im Kopf habe oder sowas, dann suche ich entweder daran (drückt 'Enter') oder nach irgendeiner Bezeichnung. Ja.. und dann versuche ich sie darüber zu finden. [00:26:05-3](#)

I: Ok. (unv. da ist ja auch viel altes Zeug dann drinnen, oder?) [00:26:10-6](#)

B: Gut, jetzt sucht er jetzt halt überall. Aber dann findet man die eigentlich.. also über diese Fachkonzeptnummern findet man die Konzepte eigentlich relativ schnell und gut. (...) [00:26:23-9](#)

I: "2201".. Ok, aber das war quasi in einem.. im vorigen Meilenstein. [00:26:30-8](#)

B: Das war eine alte.. Es gibt jetzt das neue.. das gibt es.. das wird jetzt auch noch angepasst das Konzept was ich jetzt beispielsweise gesucht habe. Dieses "2201". Aber das ist halt noch nicht in diesem **LFS**-Projekt drinnen. [00:26:44-1](#)

I: Na gut. Dann noch eine Frage zu den Fälligkeiten und so: wie wird das jetzt gehandelt, oder wie akkurat ist das.. [00:26:59-0](#)

B: Wir versuchen sie einzuhalten. Also.. Klar, wir haben eine Planung gemacht, da werden sie irgendwann importiert, und die werden halt auch jede Woche kontrolliert. Oder alle zwei, drei Tage werden sie halt kontrolliert. Und wenn irgendwo was ansteht.. Und wenn die Fälligkeiten überschritten werden, wird.. müssen auch dementsprechende Massnahmen definiert werden. Also das machen wir dann schon. Beziehungsweise wenn sich dann herausstellt dass es halt einfach nicht machbar ist, dann muss halt die Planung angepasst werden, und die entsprechenden anderen Tasks auch verschoben werden, angepasst werden von den Fälligkeiten. Also das ist sehr wartungsintensiv, JIRA. Je grösser das Projekt wird, desto mehr Wartung muss man da auch reinstecken. Und auch Kontrolle. [00:27:42-7](#)

I: Und das macht dann **ENW_DE_TL**, du, und **ENW_DE_TL_STV_1**? [00:27:45-9](#)

B: Ja. Haupsächlich macht das bei uns **ENW_DE_TL**, aber wenn.. wie gesagt, wenn er das auch dann irgendwann nicht mehr schafft, kucken wir uns das zu dritt an und verschieben die Sachen dann entsprechend. Und bei Tickets meistens (unv. in Bearbeitung) wird da.. meistens ist das so, wenn wir irgendwie.. sowieso ein Thema irgendwie sich hinten rausläuft, dann wird das meistens eigentlich komplett verschoben. Dann kannst du im Prinzip den ganzen Haupttask irgendwo dann verschieben. [00:28:09-3](#)

I: Ok. Dann eine Frage noch zu der Situation: Freitag Abend oder so, und es muss noch irgendetwas nachgeliefert werden. Das hat jetzt weniger mit JIRA zu tun. Das ist ja, ich weiss nicht, oft so dass dann eben, oder ich weiss nicht, ich kenne das halt von **Österreich**, dass die noch irgendwie daran arbeiten oder so, und dann ist schon die Deadline. Dann müssen sie aber noch irgendetwas nachliefern, ja. Dann müssen sie, ich weiss nicht, dann rufen sie **ENW_DE_TL_STV_1** oder so an, **ENW_DE_TL_STV_1** ruft dann **ENTW_DE_AL** an, und **ENTW_DE_AL** ruft **RMM_1** an. Wie ist das für euch in **Norddeutschland** als Hauptentwickler.. [00:28:53-3](#)

B: Nicht anders. [00:28:55-7](#)

I: (lacht). Aber wie bleibt ihr dann zum Beispiel, ich weiss nicht, **ENTW_OE_TL** kommt dann zum Beispiel oft irgendwie.. keine Ahnung. **ENTW_OE_1** arbeitet an einem Ticket, **ENTW_OE_TL** kommt dann ständig rüber und fragt wie der Status ist. Wie macht ihr das, dass ihr da am laufenden bleibt. Also genügt euch da der REA-

Antrag oder so dass ihr seht "das ist gerade noch in Arbeit", oder... [00:29:20-7](#)

B: Naja, der REA-Antrag.. erstmal nicht. Erstmal muss das Problem behoben werden bevor man einen REA anstellt. Wenn wir sagen "wir müssen da unbedingt etwas nachliefern".. klar müssen wir auch.. treten wir auch mit Releasemanagement "wie lange seid ihr heute noch da", "könnt ihr (unv. uns überhaupt leisten)".. Entweder sagen wir irgendeinen Zeitraum, was weiss ich ich, wir sagen da "wir brauchen noch zwei Stunden". Aber wenn das nach einer Stunde irgendwie feststellbar ist, dass wir es nicht mehr schaffen, dann muss das leider abgesagt werden das Thema. Also ansonsten ist das Vorgehen genau das gleiche. Und wenn wir sagen.. dann wird solange gearbeitet bis das entweder fertig ist, und wenn alle da sind das noch geliefert werden kann - inklusive natürlich auch Nachtests, deployen und hast-du-nicht-gesehen - ist nichts anderes. Also das Vorgehen ist bei uns hier genau das gleiche wie bei euch dann auch. [00:30:05-0](#)

I: Aber eben du rufst dann zum Beispiel **ENTW_OE_1** nochmal an, um zu wissen wie es aussieht. [00:30:10-5](#)

B: Genau. Entweder ruft **ENTW_OE_1**, oder ich, **ENTW_DE_TL_STV_1** oder **ENTW_DE_TL** dann irgendwie an, dann kucken wir das (unv. Ding), wenn die sagen das es klar ist - gut. Ich weiss jetzt nicht wie das bei euch ist, ob dann eure Vorgesetzten oder wer da bleibt, wenn wir jetzt sagen wir machen das fertig, dann braucht jetzt nicht unbedingt **ENTW_DE_TL** hier sitzen. Der sagt "mach das fertig" und dann ist gut. Aber klar, wenn sich das jetzt schon nachmittags irgendwann herausstellt, dann wird auch alle zwei Stunden, oder Stunde wird auch immer gefragt "wie siehts es aus, wie siehts es aus, wie siehts es aus?". Man muss halt einen Status liefern, und nachher stellt sich halt irgendwann heraus.. ja, zum Zeitpunkt X sagst du dann entweder "ich schaffe es" oder "ich schaffe es nicht". Und wenn du dann sagst du schaffst es, dann kann der Abend ein bisschen länger werden. [00:30:51-3](#)

I: Aber da ist dann eigentlich das Ticketsystem selbst.. reicht jetzt nicht aus, dass ich einschätzen kann wie lange der Entwickler jetzt noch brauchen wird, oder? [00:30:59-6](#)

B: Nein. Das können wir nicht machen. (...). Aber das ist auch bei JIRA aber genauso schlecht nachvollziehbar wie lange man noch dafür braucht. Also gerade bei so kurzfristigen Sachen ist das wirklich sehr schwierig. Weil wenn du morgens mit nur einem Problem... Morgens kommen die Probleme auf den Tisch, und wenn du sagst, du musst bis 16 Uhr fertig sein, ich glaube nicht.. das kannst du auch nicht in JIRA erfassen. Ich wüsste jetzt zumindest nicht wie. Von vornherein kann man sagen "dafür brauche ich ungefähr so-und-so lange", das kannst du auch in JIRA eintragen als Aufwandsschätzung, aber ob du das nachher einhältst, das ist ziemlich schwierig bei diesen Problemen. Bei Tickets. Ich glaube das kennst du aber selber auch, aus dem VWB-Bereich. Wenn du sagst "ja, das könnte eigentlich nur eine Kleinigkeit sein", nachher stellt sich aber heraus, das ist doch so gross. Das lässt sich halt sehr schwierig einschätzen. Gerade so Tickets oder so lassen sich immer sehr schwierig einschätzen. Also die meisten zumindest. Bei einigen sagst du "ok, das ist eine Kleinigkeit", aber eigentlich kann sich ja auch herauskristalisieren da brauchst du nachher zwei Wochen für. [00:32:06-5](#)

I: Ja, das kenne ich leider das Problem. Ok. Ich bin glaube ich durch mit meinen Fragen. Dann danke ich mal ganz herzlich. [00:32:36-4](#)

B: Na gut. Ich kann dir ja sonst noch mal zeigen.. wir hatten ja so Tickets auch in JIRA übertragen für ein Projekt in **LFS**. Falls dich das interessiert. [00:32:41-9](#)

I: Ja, bitte. [00:32:45-3](#)

B: (sucht) Jetzt muss ich aber selber suchen. Ob ich die wieder finde. (klickt). [00:32:56-4](#)

I: Das sind die **xxx**-Sachen? [00:33:00-4](#)

B: Genau. (...) (klickt). Das ist halt unser Problem. Wenn du die Tickets hier herüber überträgst, hast du halt diese doppelte Pflege. Du hast du in JIRA und in Notes. [00:33:22-1](#)

I: Ok. Aber da wird in.. also.. [00:33:25-5](#)

B: Das war im Prinzip so ein **xxx**-Geschichte hier. Und da haben wir dann halt auch die Tickets aus dem Ticketsystem hier rüber übertragen. Hier wurden dann auch die Aufwände frisch geschätzt, um das ein bisschen besser in den Griff zu bekommen, aber der Pflegeaufwand... Bei zwei Systemen ist der echt gigantisch. [00:33:45-8](#)

I: Aber eben wofür habt ihr das gemacht? Einfach um einmal zu sehen wie das laufen könnte? [00:33:50-7](#)

B: Genau. Um die Aufwände zu schätzen für die einzelnen Tickets. Es gab halt entsprechend viele Tickets. Wie man hier sieht an dem Haupttask, das waren so an die 60 Stück. Also das waren im Prinzip die wichtigsten. Und

die wurden dann auch.. ein bisschen geschätzt - das war nicht so toll wie man hier sieht - aber so.. das war im Prinzip mal so ein Versuch die Sachen vernünftig raus zu kriegen. [00:34:15-0](#)

I: Wieviele Tage sind das da? [00:34:16-5](#)

B: Nein, die sind gar nicht geschätzt worden. Hier sind die geschätzt worden. Nein, geschätzt wurden sie alle. [00:34:22-7](#)

I: Ok, das ist die Gesamtzeit quasi. [00:34:27-7](#)

B: Genau. [00:34:32-3](#)

I: 24 Wochen. [00:34:34-8](#)

B: Tja. Wie gesagt, man muss halt täglich die Sachen abgleichen. Wir haben jetzt nicht unbedingt so einen Abgleich-Job der die beiden Systeme abgleicht - das wäre natürlich das Ideale - haben wir leider nicht. Und ja, dadurch ist das halt ziemlich gross. [00:34:51-9](#)

I: Wird das jetzt noch weiter gepflegt, oder ist das jetzt abgeschlossen? [00:34:54-1](#)

B: Das ist erst mal abgeschlossen jetzt. Aber wie gesagt, man hat daran gearbeitet, und dann kamen halt immer noch mehr Tickets, noch mehr Tickets.. Die wurden hier auch immer reingepflegt, aber irgendwann haben wir das auch abgesagt. [00:35:08-6](#)

I: Und für die Nachverfolgung eben, habt ihr dann.. wie habt ihr das dann verwendet? Habt euch ihr dann im JIRA daraus eine Statistik gezogen, oder..? [00:35:16-2](#)

B: Daraus haben wir dann Statistiken gemacht, ja. Ich weiss nicht.. [00:35:17-3](#)

I: (unterbricht) Also mit diesen Filterfunktionen und was es da alles gibt. [00:35:19-9](#)

B: Ja genau. Da musst du **ENTW_DE_TL** mal fragen was er das jetzt rausgezogen hat aus diesen Tickets hier. Also wie gesagt: eigentlich ist es schöner, die Tickets, also die Bugs auch über JIRA zu verwalten und zu pflegen, aber da wir halt mehrere Kunden haben, und das eine Ticketsystem halt momentan noch das Führende ist - und man das ja auch nicht einfach so ablösen kann, du musst ja auch eine komplette.. die ganze Historie dann, eigentlich alles nach JIRA übertragen - ich weiss nicht, ich glaube da wird daran noch gearbeitet. Aber dann gibt es glaube ich.. da soll es vielleicht ein eigenes Projekt geben. Aber das steht noch nicht fest. [00:35:58-9](#)

I: Ok. Dann herzlichen Dank. [00:36:02-4](#)

B: Bitte. Ich hoffe es hat dir was gebracht.

Interview 13_FK_1

I: Also als erstes: Was sind deine Aufgaben, was ist deine Rolle hier in der Firma und deine alltäglichen Aufgaben? 00:00:12-7

B: Also ich bin Business-Analyst im Bereich Komposit, und erstelle dort Konzepte für Komposit - also das ist Unfall, Haftpflicht, KFZ, Hausrat, Wohngebäude und Glas - und schreibe da Konzepte und Testlisten, und habe zusätzlich noch die Teilprojektkoordination, das heisst ich bin Schnittstelle zwischen unserem Kunden **KUNDE_1** und **VersicherungsGmbH** für alle Fragen zu Komposit. 00:00:37-4

I: Ok. Was sind dann im kleinen, so feiner granuliert, die Sachen die du täglich erledigen musst? 00:00:50-5

B: Also angefangen im Meilenstein mit Schätzung erstellen für Konzepte, dann schließlich die Fachkonzepte schreiben, Testlisten schreiben, die Konzepte mit der Entwicklung abstimmen und mit dem Kunden, Testlisten mit dem Testteam abstimmen. 00:01:08-4

I: Und welche sind dann.. mit welchen Kollegen arbeitest du dann in deiner eigenen Abteilung zusammen, und mit welchen jetzt übergreifend? Also übergreifend hast du jetzt eh schon ein bisschen gesagt, so Test und Entwicklung und Kunde.. 00:01:21-7

B: Also im eigenen Team.. also ich bin für den Dialogbereich zuständig, da arbeite ich dann auch mit den Kollegen von Produkt zusammen. Ausserdem muss ich mich auch darum kümmern, dass die Schnittstellen geklärt werden zu Provision und Zahlungsverkehr, da haben wir bei uns auch Kollegen im Team. 00:01:38-8

I: Wie sieht da die Zusammenarbeit aus? Um jetzt irgendetwas abzustimmen macht ihr da Meetings, oder... 00:01:48-5

B: Genau. Also zu Beginn des Meilensteins bekommen wir ja immer die FUBen - die Funktionsumfangsbeschreibungen - vom Kunden, und dann schaut man schon zum Beginn des Meilensteins: sind da noch weitere Teilbereiche betroffen, und setzt sich dann möglichst frühzeitig mit den Kollegen zusammen, macht einen Termin und geht das gemeinsam durch. Und schaut, müssen die da auch was machen, oder gibt es die Funktionalität schon und es sind keine Anpassungen erforderlich. 00:02:13-2

I: Ok. Und das.. wie wird das dann.. die Arbeitsorganisation gemacht? Also wenn du jetzt sagst du machst mit deinen Kollegen irgendwie, keine Ahnung, die in Provision müssen etwas machen, du musst etwas machen.. wie wird das dann.. 00:02:25-3

B: (unterbricht) Genau. Das hängt davon ab. Manchmal reicht der Kunde schon eine eigene FUB für den jeweiligen Bereich ein, und manchmal schreiben wir dann gemeinsam das Konzept. Also jetzt bei den letzten Sparten war das so, dass Provision ein eigenes Konzept geschrieben hat, man sich mit denen nur abgestimmt hat, während Finanzen - da waren meist nur kleine Änderungen - die haben dann ein eigenes Kapitel in meinem Konzept gehabt. Das heisst, wenn ich mit meinem Bereich fertig war habe ich denen das Konzept gegeben, die haben ihren Teil ergänzt, und dann haben wir es gemeinsam zur QS gegeben, und dann wurde das in der Entwicklung auch einmal durchgereicht. Dann wurde das einmal von den Dialog-Entwicklern geQSt, und einmal von den Finanzen-Entwicklern, und anschliessend haben wir es dann gemeinsam zurückgekiert. 00:03:05-5

I: Ok. Das wird dann das Worddokument herumgeschickt. 00:03:09-9

B: Genau. 00:03:08-6

I: Und wie verwendet ihr da JIRA jetzt? 00:03:11-2

B: Genau, da verwenden wir JIRA. Da habe ich einen Task für Fachkonzepterstellung und für die QS, und an den Fachkonzepttask hänge ich dann das Konzept, und an den QS-Task den setzt ich dann.. weise ich dann entsprechend der Entwicklung zu. Gebe denen drei Tage Zeit, standardmässig, und dann bekomme ich das auch zurück wenn sie es dann QSt haben. 00:03:33-5

I: Das geht dann direkt an **ENTW_DE_TL** wahrscheinlich, oder? 00:03:35-8

B: Genau. Bei den Konzepten geht das immer an den Teamleiter, und der reicht das dann bei sich weiter an den Entwickler wo er meint, dass der das wahrscheinlich später auch machen wird. 00:03:46-6

I: Und.. ich weiss nicht, wie wichtig ist das zum Beispiel für dich zu wissen, wie weit jetzt die Entwickler sind bei der... oder zum Beispiel wer.. wer jetzt dein Fachkonzept umsetzt, und wie weit der jetzt damit gekommen ist?

Oder... [00:04:01-9](#)

B: Also wenn ich die Rückmeldung wirklich innerhalb von drei Tagen bekomme, ist es für mich nicht so wichtig wie weit sie da jetzt sind. Meist brauchen sie dann aber doch länger, und dann ist es schon interessant nachzuhaken, und zu fragen... [00:04:15-5](#)

I: Aber weisst du zum Beispiel konkret wer deine Sachen QSt oder so? [00:04:19-8](#)

B: Ja meist haben ja die Entwickler dann eine Rückfrage zu dem Konzept. Das heisst, die rufen dann an, und in der Regel setzen wir uns dann auch wieder zusammen und schicken denen eine Einladung (Anm. zu einem Besprechungstermin), und dann schauen wir uns gemeinsam die Fragen an. Weil erstmal lesen sie es sich in Ruhe durch, und meist haben sie dann doch irgendwelche Fragen, und dann setzt man sich zusammen und spricht das gemeinsam durch. [00:04:37-6](#)

I: Wie funktioniert das dann zum Beispiel mit den **Österreichern**? Wo man sich ja nicht so zusammensetzen kann? [00:04:41-9](#)

B: Die schreiben die Fragen dann mehr in JIRA rein, und im Zweifel telefoniert man dann, wenn man das schlecht beschreiben kann. [00:04:49-3](#)

I: Ok. Wie genau weisst du darüber Bescheid was deine Kollegen in der Fachkonzeption so machen, und wie sehr ist das für dich wichtig? [00:05:06-7](#)

B: Also sehr wichtig ist für mich was die Kollegen von Produkt machen, weil ich von Produkt ja abhängig bin. Das heisst, es muss erst das Produktkonzept weitestgehend fertiggestellt sein, damit ich weiss wie ich dann die Dialogkonzepte schreiben muss. Und von daher arbeite ich da sehr eng mit den Kollegen zusammen, und frage da auch regelmässig nach.. [00:05:26-2](#)

I: Sitzt du mit dem Kollegen in einem Zimmer? Oder.. [00:05:30-0](#)

B: Ich saß mal zusammen mit **PMM_1**, und jetzt ist aber nur noch **FK_2** über von Produkt, und der sitzt direkt im Nebenbüro. Von daher sind das kurze Wege. [00:05:40-4](#)

I: Also da gehst du dann direkt rüber, oder ihr ruft euch zusammen oder so. [00:05:42-3](#)

B: Genau. [00:05:45-0](#)

I: Wie erfährst du zum Beispiel über.. (murmelt) obwohl das ist wahrscheinlich eh dasselbe.. eben über irgendwelche Probleme die es jetzt zum Beispiel gibt im Projekt oder so, oder in einem Meilenstein. (...) Ich weiss nicht, wenn es zum Beispiel Themen gibt, wo immer wieder Tickets zum selben Thema oder so auftauchen, wie... [00:06:19-9](#)

B: Also wenn jetzt viele Tickets auftauchen, dann meldet sich ja meist irgendwer aus der Entwicklung und sagt: "hier, da gibt es noch Klärungsbedarf", "da ist das Konzept vielleicht ungenau", "da gibt es noch Anpassungsbedarf" oder "der Kunde hat seine Meinung geändert". Und dann stimmen wir uns.. entweder wenn wir es fachlich direkt bei **VersicherungsGmbH** sagen können, dann passe ich das Konzept an, oder ich kontaktiere dann erstmal den Kunden und kläre mit dem was der genau für Anforderungen hat - wenn wir da zum Beispiel eine Konzeptlücke haben - und passe dann das Konzept an, und gebe es dann wieder an die Entwicklung, damit die dann entsprechend die Tickets gemäss Konzept wieder beheben können. [00:06:55-8](#)

I: Und wie sehr muss man da jetzt - sage ich mal - 'antizipieren'? Dass man, ich weiss nicht, irgendwie versucht zu wissen was jetzt, zum Beispiel, der Kunde.. womit der Kunde als nächstes kommen wird oder so. Oder lässt man sich da... [00:07:17-2](#)

B: Ja man weiss ja nie mit was der Kunde als nächstes kommt (lacht). [00:07:22-6](#)

I: Ja, wie soll ich das sagen? (...) Ich weiss nicht.. Wenn du das Fachkonzept schreibst oder so.. Ich weiss nicht.. überlegst du dann zum Beispiel "ok, bei dem-und-dem Absatz wird sich jetzt sicher Entwickler XY melden oder der Kunde wegen dem-und-dem wieder irgendetwas anmerken"? [00:07:57-1](#)

B: Ja manchmal weiss man schon dass da Fragen kommen werden oder das anders gemeint ist oder so. Aber das versucht man ja dann frühzeitig, schon gleich während man das Konzept schreibt zu klären. Das heisst, wenn ich zum Beispiel sehe in der FUB sind Sachen nicht beschrieben, ich weiss aber das war in einer anderen Stelle wichtig dass dann auch die-und-die Punkte passieren sollen, dann rufe ich dann schon mal den Kunden an, und frage ob sie das nicht bei der neuen Sparte auch haben möchten was sie bei einer anderen Sparte schon haben,

und nehme das dann zusätzlich noch ins Konzept rein. Oder auch die Entwickler, wenn denen dann bei der QS auffällt "hm, bei einer anderen Sparte hatten wir an der oder der Ecke ein Problem" dann schreiben die das schon gleich in die QS rein und sagen "hm, willst du nicht den Punkt auch noch ergänzen im Konzept, dass wir da schon gleich sicherstellen, dass wir bei der neuen Sparte nicht wieder Probleme bekommen". [00:08:41-9](#)

I: Und dieses.. wenn du ein Fachkonzept hin- und herschickst, das war ja früher eben.. früher wurde das mit Email geschickt. [00:08:49-6](#)

B: Genau. [00:08:50-3](#)

I: Das ist mittlerweile nur noch komplett über JIRA, oder? [00:08:53-8](#)

B: Genau. Inzwischen machen wir das alles über JIRA. Selbst für die alten Sachen wurden JIRA-Tasks nachträglich noch angelegt, wenn das Konzepte auch wieder mal aufgemacht werden müssen. Das läuft jetzt alles über JIRA. [00:09:07-2](#)

I: Wenn du mit anderen Fachkonzeptern ein Worddokument schreibst oder so. Da gibt es ja diese Markup-Funktionen. Verwendest du die? [00:09:23-4](#)

B: Dieses "Änderungen nachverfolgen"? [00:09:25-0](#)

I: Genau. [00:09:25-0](#)

B: Das machen wir hauptsächlich bei Konzeptänderungen. Oder wenn man für jemanden etwas QSt, dann macht das auch Sinn. Also manchmal schicken wir uns gegenseitig Konzepte, und sagen "kannst du mal drüberschauen", und wenn man eben dann Anmerkungen zu (unv. kommen) ist, dann macht man das auch häufig mit Änderungsverfolgung. Dann kann der andere dann entscheiden, ob er die Änderung übernimmt, oder noch anpassen will. [00:09:46-1](#)

I: Wie funktioniert das wenn ihr eben zum Beispiel - weil du gesagt hast dass du mit Finanz zum Beispiel gleichzeitig Konzept schreibst, oder in einem Konzept - [00:09:53-8](#)

B: (gleichzeitig) Ja, das machen wir dann immer nach und nach [00:09:52-3](#)

I: (gleichzeitig) ... ist es dann auch so.. du sagst dann "ok, ich bin jetzt fertig" und dann tschüss, und dann weisst du aber auch "ok, jetzt rühre ich das Fachkonzept nicht mehr an solange es der Kollege hat"? [00:10:04-1](#)

B: Genau, das machen wir dann immer. Bei Finanzen die sind immer sehr schnell, die haben immer nur ein kleines Kapitel, und das geht dann meist recht schnell, weil sie es.. manchmal schreiben die auch parallel schon in einem anderen Dokument dann vor und kopieren sich das dann nur noch rein wenn sie wissen sie haben eine neue Version. [00:10:20-3](#)

I: Dann Frage: wie hast du dein Sametime organisiert. Welche Gruppen hast du da drinnen? [00:10:29-6](#)

B: Da habe ich einmal mein eigenes Team drin mit allen Teammitgliedern. Dann CT-Komposit, also die Entwickler als Team komplett, dann Testteam und Produktmanagement. Und internen Support habe ich noch. [00:10:51-0](#)

I: Und wie oft kuckst du da rein, in die Kontaktliste? [00:10:54-9](#)

B: Das habe ich parallel in meinem Lotus Notes auf. Also ich habe das Fenster so aufgeteilt, dass ich einmal Emails, und rechts dann Sametime immer offen habe. Und von daher kann ich oben an dann immer sehen, wer aus meinem Team und aus der Entwicklung drin ist. Und weiter runterscrollen komme ich dann auch zu den anderen Sachen. Und man sieht ja auch bei den Emails vorne an immer, wenn man gerade von jemandem eine Email bekommen hat, ob derjenige anwesend ist oder nicht. [00:11:17-4](#)

I: Wieviel hast du mit den Testern dann zu tun? [00:11:31-9](#)

B: Ja mit den Testlisten.. Mit den Testern schicke ich ja die Testlisten dann zur QS, und... [00:11:38-2](#)

I: Das geht auch über JIRA. [00:11:40-5](#)

B: Genau, das geht über JIRA. Auch erst an den Teamleiter **TST_TL**, und da ist es aber so, wenn man die erste QS zurückbekommen hat, dann darf man es direkt an den Tester das nächste mal zuweisen. Bei den Fachkonzepten muss man die Konzepte immer an den Teamleiter zuweisen. Weil das passieren kann, dass er die bei der nächsten QS einem anderen Entwickler gibt. Und bei den Testlisten ist es aber so, dass die nur beim

ersten Mal an den Teamleiter gehen, und danach an den von dem man dann die erst QS bekommen hat. 00:12:08-3

I: Das ist ja quasi auch so mit der Testabteilung abgesprochen, dass ihr.. 00:12:12-9

B: Genau. 00:12:12-1

I: Und du weisst dann eben auch, wer die QS gemacht hat. Das steht dann im QS-Dokument drin. 00:12:17-8

B: Ja. Das sieht man dann ja im Dokument. Und wie gesagt, da kriegt man dann da die QSen zurück, und auch da ist es so, wenn ich dann nicht genau verstehe was sie geändert haben wollen, oder meine das ist nicht nötig, dann setzen wir uns auch zusammen und sprechen dann die Punkte durch. 00:12:33-7

I: Und QS geht jetzt auch mittlerweile komplett über JIRA habe ich gesehen. 00:12:37-4

B: Genau. 00:12:40-0

I: Da sind ja dann diese rosaroten Felder da drin. Wie ist das zum Arbeiten? 00:12:42-7

B: Ja, in der Testliste ist hinten ein Reiter wo die vom Testteam die offenen Fragen reintragen, die tragen es nicht direkt in JIRA ein. 00:12:51-5

I: Achso, das ist nur für die.. 00:12:52-6

B: Das ist nur bei den Konzepten. 00:12:56-6

I: Ok. Und wie ist das so zum... Also früher hat es ja dieses eigene Dokument gegeben, QS-Dokument, ... 00:13:05-0

B: Das mit dem eigenen Dokument, das fand ich ein bisschen umständlich, und jetzt ist es halt einfach ein zusätzlicher Reiter in der Testliste. Und das ist eigentlich sehr gut, weil man dann mit der Änderungshistorie auf die offenen Punkte verweisen kann. Sagen kann "ok, die habe ich angepasst", und was man dann konkret angepasst hat schreibt man dann als Antwort hinter die Frage vom Testteam. 00:13:22-8

I: Aber ist das dann schon tatsächlich übersichtlicher, also wenn du das dann zum Beispiel einarbeiten musst... Ich weiss nicht, ist da nicht ein Word-Dokument dann schon kompakter oder so? Weil man es auch ausdrucken kann oder so... (murmelt) im Vergleich zu JIRA? 00:13:37-9

B: Naja, meist haben die ja nicht so viele Punkte, und wenn dann da hinten drei oder fünf Fragen sind, dann kann man das meist schnell machen. Und das ist halt eigentlich praktisch, weil man ja dann einfach nur in den letzten Reiter geht, sich die erste Frage durchliest, dann in die entsprechende Testgruppe geht, das kurz anpasst, reinschreibt "habe ich so-und-so angepasst", und dann macht man den nächsten Punkt. Das geht eigentlich sehr gut, weil man genau ein Dokument hat und nur zwischen den Reitern wechseln muss. 00:14:02-6

I: Nein, und aber eben jetzt auch beim Fachkonzept, bei der QS.. wo eben die QS selbst dann nicht im Fachkonzept drin ist, sondern... 00:14:14-0

B: Ja, beim Fachkonzept da muss man sich daran gewöhnen. Das ist in JIRA.. also ich finde die Aufbereitung in JIRA nicht so schön, weil das da ja so komisch mit HTML oder was das ist verschlüsselt ist. Da muss man immer aufpassen dass man sich das Layout nicht zerschiesst, aber.. gewöhnt man sich auch dran. 00:14:30-9

(JIRA-Login) 00:15:23-9

I: Ok, würdest du mir das bitte zeigen, dein Dashboard? 00:15:22-1

B: Ja, das ist das Standard-Dashboard was unser Teamleiter **FK_TL** uns eingerichtet hat. 00:15:28-8

B: Da habe ich hier oben meine offenen Aufgaben die ich noch erledigen muss. Da sind leider immer ein bisschen zu viel drin, weil bei vielen Sachen muss ich gar nichts machen, aber erzeugt die immer so alle auf einmal und dann hat man da manchmal welche die man gar nicht machen muss. 00:15:47-6

B: Dann habe ich meine "in Arbeit" befindlichen Konzepte, das sind jetzt alles Fachkonzepte von diesem Meilenstein die aktuell zur QS in der Entwicklung liegen. Da warte ich jetzt auch die Rückmeldung - das sieht man dann auch hier - das sind alles Fachkonzepte in QS. 00:16:04-1

I: Also die liegen jetzt gerade bei der Entwicklung. [00:16:06-0](#)

B: Genau. Die warten darauf dass die QSt werden.. die liegen auch schon ein bisschen länger. Die haben im Moment leider sehr viel zu tun, von daher dauert das dann länger als die drei Tage bis man da mal eine Rückmeldung kriegt. [00:16:15-1](#)

I: Wieso weisst du, dass die gerade viel zu tun haben? [00:16:18-5](#)

B: (verwundert) Weil sie mir das erzählen wenn ich nachfrage (lacht) warum mein Konzept noch nicht QSt ist. [00:16:28-5](#)

B: Und ja, die Anzeige ist hier glaube ich noch nicht so ganz richtig, weil hier jetzt nur eines als überfällig angezeigt wird. Die sind aber eigentlich alle schon überfällig. [00:16:38-3](#)

B: Wenn sie die dann QSt haben, kommen die meist hier an: "meine nachzubessernden FK-QSen" wo ich dann sehe, was ich noch machen muss. [00:16:46-2](#)

B: Und das ist jetzt eines was schon abgenommen wurde, da warte ich jetzt noch darauf, dass noch weitere Konzepte abgenommen werden, dann schicke ich die gesammelt an **KUNDE_1**. Weil das ist nur eines, wo ich Kleinigkeiten nachgebessert habe. Das ist vom alten Meilenstein. Und das heisst, das wird dann nur Informationshalber und formell zur Abnahme nochmal an den Kunden geschickt. [00:17:09-3](#)

I: Abgenommen heisst, dass die QS vom Entwickler durchgegangen ist, oder wie? [00:17:14-1](#)

B: Nein.. Die Kunden müssen ja das Konzept immer abnehmen, und das ist aber schon eines was im alten Meilenstein schon abgenommen wurde, wo jetzt nur eine minimale Anpassungen gemacht wurde. Was wir dann formell nochmal zur Abnahme an den Kunden schicken müssen, damit der auch immer die aktuelle Version hat. [00:17:32-3](#)

B: Und genau.. Dann habe ich hier auch meine Testdokumente, also die Testliste und den Testnachweis. Die gehen zur QS ans Testteam, da warte ich dann auf Rückmeldung, und WENN die eine Rückmeldung haben, dann stehen die hier bei den "nachzubessernden Testdokumenten". Wenn die ok sind, dann verschwinden die dann irgendwann. Weil die die dann schliessen. [00:17:59-8](#)

B: Genau.. das sind noch alte Dokumentationsaufgaben. Dadurch dass wir im moment nicht so richtig dokumentieren können im Dokupool stehen die häufig dann einfach noch herum. [00:18:14-6](#)

B: Da sieht man die geschlossenen Aufgaben die ich dann irgendwann schon mal erledigt habe. Und wenn ich ein Konzept zur Kundenabnahme an **KUNDE_1** geschickt habe, dann steht das hier bis ich das dann schliessen kann wenn ich dann wirklich die Abnahme vom Kunden bekommen habe. [00:18:32-0](#)

I: Und wie oft am Tag schaust du ins JIRA? Oder... [00:18:38-0](#)

B: Ich versuche einmal täglich reinzuschauen. Eigentlich sollen wir auch Emails bekommen wenn sich etwas tut, das klappt leider nicht so wirklich. Das heisst, sobald ich ein Email bekomme, schaue ich auf jeden Fall in JIRA rein, und ansonsten versuche ich wenigsten einmal am Tag zusätzlich reinzuschauen ob sich etwas geändert hat. [00:18:55-3](#)

I: Und, eben wie planst du dann zum Beispiel deinen Arbeitstag? Also wie weisst du, was du als nächstes machen musst? Hast du das.. läuft das jetzt nur über JIRA? Oder... [00:19:09-0](#)

B: Nein, meist arb... Also ich fange ja mit den Konzepten an nach Priorität. Oder, je nachdem, wenn es zu einem Konzept mit der höchsten Priorität das Produktkonzept noch nicht fertig ist, schaue ich halt, ob es zum zweithöchsten Konzept schon fertig ist. Dann fange ich mit dem an, und arbeite meine Konzepte so der Reihe nach ab, schicke die zur QS, wann immer eine QS zurückkommt schiebe ich das dann wieder zwischen, arbeite erstmal die QS ab damit dann das Konzept möglichst schnell an **KUNDE_1** zur Kundenabnahme geschickt werden kann, und in der Zwischenzeit arbeite ich dann in dem nächsten Konzept weiter. Und so gehe ich die halt der Priorität nach durch. [00:19:46-8](#)

I: Hast du da irgendwie einen Zettel oder so, schriftliche Notizen was du... so eine Todo-Liste? [00:19:48-8](#)

B: Nein.. Dadurch dass ich in der Regel nur zwei bis drei Konzepte pro Meilenstein habe, habe ich das dann im Kopf, und weiss, welches gerade wo ist. [00:19:57-7](#)

I: Ok. Und siehst du dir zum Beispiel auch an, woran deine Kollegen jetzt zum Beispiel arbeiten, was die so

machen? Also im JIRA, oder... [00:20:08-9](#)

B: Nein... im JIRA gar nicht. Ich schaue regelmässig im Verzeichnis ob es neue Versionen vom Produktkonzept gibt, aber das mach ich über den Explorer, nicht über JIRA. [00:20:19-4](#)

I: Über welchen Explorer? [00:20:20-5](#)

B: Normal Windows. Über das Verzeichnis. [00:20:22-5](#)

I: Achso, die liegen in einem.. am Netzlaufwerk, oder wie? [00:20:26-4](#)

B: Genau. [00:20:28-9](#)

I: Und da schaust du immer regelmässig rein. [00:20:31-0](#)

B: Da schaue ich zwischendurch mal rein. Normalerweise sagt mir der Produktkollege bescheid wenn er da nochmal nachträglich etwas geändert hat. (...) Zwischendurch schaue ich da auch einfach mal rein. [00:20:42-7](#)

I: So eigene Filter oder soetwas hat du dir nicht gemacht? [00:20:52-1](#)

B: Nein. [00:20:54-4](#)

I: Und wenn du jetzt zum Beispiel.. ich weiss nicht.. es kommt irgendein Kundenwunsch oder so herein und du musst jetzt irgendein bestimmtes Fachkonzept ändern oder anpassen. Wie findest du dann das Fachkonzept? Oder... im JIRA? [00:21:12-3](#)

B: Wenn es gerade vom aktuellen Meilenstein etwas ist, dann habe ich es ja meist hier noch irgendwo liegen. Wenn es etwas älteres ist gehe ich normal über die Suche, und dann suche ich hier mit der Konzeptnummer. [00:21:28-4](#)

I: Verwendet ihr zum Beispiel diese Kommentarfunktionen? [00:21:37-2](#)

B: Ja, wenn man ein Konzept zur QS gibt, dann macht er das Kommentarfenster mit auf. Und manchmal schreibe ich da was rein. Zum Beispiel wenn jetzt ein Konzept - Finanzen zum Beispiel - gerade einen Urlaub hat, und ich weiss der muss das Kapitel noch beschreiben, ich will das aber schon mal zur QS schicken, dann schreibe ich als Kommentar rein "Finanzen-Kapitel ist noch leer, erfolgt später. Bitte schon mal QS auf den aktuellen Stand durchführen" damit die dann mir nicht in die QS böse reinschreiben "hier, da fehlt aber noch das Finanzen-Kapitel" [00:22:04-3](#)

I: Und verlässt du dich dann zum Beispiel auf den JIRA-Kommentar, oder.. [00:22:07-2](#)

B: (gleichzeitig) Ja, (unv. den sieht man ja auch selber). [00:22:09-2](#)

I: (gleichzeitig)... gibst du dem dann zusätzlich noch - ich weiss nicht - über Telefon bescheid, oder ein Email oder so dass... [00:22:13-3](#)

B: Nein, da verlasse ich mich dann auf JIRA. [00:22:14-6](#)

I: Ok. Ja gut. (...) Kann ich da mal reinkucken? (klickt) Das ist eines von dir, oder? [00:22:36-3](#)

B: Genau. Da müsste man die Releasenotes beschreiben, das ist jetzt aber zum Beispiel ein Thema.. da müsste eigentlich Produkt eine Anpassung machen. Und da war jetzt ein Puffer geplant wenn wir an den Dialogkonzepten auch etwas anpassen müssen. Die mussten wir jetzt auch minimal anpassen, das wird aber für die Releasenotes keine Auswirkungen haben. Das heisst, der wird ohne eine Änderung geschlossen werden können. Aber normalerweise müssen wir zu den Konzepten später Releasenotes schreiben, das heisst, da wird kurz beschrieben was das Thema des Konzeptes war, damit **KUNDE_1** weiss was sie mit der Lieferung bekommen. Und wenn es da Einschränkungen nachher aus der Entwicklerseite gibt, weil die nicht fertig geworden sind, dann ergänzen die die Releasenotes, und schreiben dann "das-und-das kommt erst mit einer späteren Auslieferung". [00:23:22-6](#)

I: Ok. Aber dieser Punkt der wird quasi mal default-mässig angelegt, oder.. [00:23:30-6](#)

B: Genau, die werden immer alle angelegt. [00:23:30-7](#)

I: Verwendest du zum Beispiel diese Überwachen-Funktion hier? [00:23:40-0](#)

B: Ja, weil vorne in diesem Dashboard was **FK_TL** uns angelegt hat, werden einige Tasks nur angezeigt wenn man sie gerade überwacht. Sonst sehe ich die gar nicht. Also wenn ich Testlisten zur QS gebe, dann sind sie ja nicht mehr mir zugewiesen, das heisst ich sehe sie da nicht mehr, und deswegen packe ich die dann immer auf "überwachen". Und das gleiche bei den Konzepten auch, die packe ich dann auch auf "überwachen", damit ich sie da am Dashboard noch sehen kann. [00:24:04-7](#)

I: Ok. Gut. Schaust du zum Beispiel öfter darauf was der **FK_TL** so herumliegen hat in seinem JIRA? [00:24:30-0](#)

B: Nein. Der hat so viele JIRA-Einträge (lacht), da schaue ich nicht drauf. [00:24:37-2](#)

I: Oder mit den Finanzen-Leuten oder so? [00:24:38-4](#)

B: Nein. Ich schaue wirklich nur meine JIRA-Aufgaben an. [00:24:46-1](#)

I: Na gut. Ja dann sage ich herzlichen Dank. [00:24:57-1](#)

B: Gerne. [00:24:59-9](#)

I: Eine Frage noch: hast du eben irgendwie so Bookmarks dann auch im Browser? Zu bestimmten JIRA-Tasks oder so? [00:25:03-6](#)

B: Nein, nur die Anfangsseite auf das Dashboard. Mehr nicht. [00:25:08-8](#)

I: Na gut. Dann sage ich herzlichen Dank. [00:25:16-1](#)

B: Gerne.

Interview 14_TST_3

I: Die erste Frage wäre mal eben was du genau machst, also Aufgabe im Team, und... [00:00:05-8](#)

B: Ok. Ja, ich bin Testdesigner für den Bereich Provision. Ja, was ich da mache: ich QSe Fachkonzepte, die Testlisten dazu, und bin auch da mit Ansprechpartner was die Fachlichkeit angeht. Und prüfe natürlich auch manuell als Tester die Testlisten ab, und decke somit halt die ganzen Testanforderungen die aus dem Konzept entstehen in Form von freien Tests ab. [00:00:38-1](#)

I: Mit welchen von deinen Kollegen bist du dafür in Kontakt? Oder musst du dafür in Kontakt sein? [00:00:44-2](#)

B: In der Regel mit der Konzeption, Fachkonzeption, und mit den Entwicklern, da es auch sehr häufig Missverständnisse gibt zwischen dem fachlich Richtigen und dem technisch Umgesetzten. Ja, dass es da Abweichungen gibt die halt im Test auftauchen, spätestens da, oder schon vorher in der Konzeption. Nach der Konzeptionsphase bekommen wir ja die Dokumente, also sprich Testliste, Testanforderung und das Konzept zur QS, also zur Qualitätssicherung, und da.. wenn es da auch Lücken gibt bespreche ich die halt mit dem Konzepter, um gegebenenfalls die Kundenanforderungen neu zu definieren. [00:01:22-1](#)

I: Ein Fachkonzept bekommt der Tester auch? [00:01:24-6](#)

B: Ganz genau. Das Fachkonzept.. [00:01:26-1](#)

I: Ok, ich habe gedacht das geht nur an die Entwickler zur QS. [00:01:30-2](#)

B: Nein, das geht an die Entwickler zur QS, aber wir QSen in erster Linie ja die Testanforderungen - die sich aus dem Konzept ergeben - und die Testliste - also die Testspezifikation - und diese haben ja einen Rückschluss auf das Konzept. Und wäre blöd wenn ich das Konzept nicht haben würde, könnte ich den Rückschluss nicht kriegen. [00:01:47-4](#)

I: Also du hast Testliste, Konzept, und das tust du dann halt gegengleichen. [00:01:50-8](#)

B: Ganz genau. Also ich habe die Testanforderung aus dem Konzept, die Testliste die die Änderungen im Konzept voll abdecken soll, und wenn ich diese matche und mit das Konzept dabei durchlese erkennt man gegebenenfalls Lücken, oder Redundanzen, also sprich wo einiges mehr fabriziert ist. [00:02:09-2](#)

I: Mit welchen Leuten... mit welchen anderen Testern arbeitest du zusammen? Oder wieviel.. wieviel Zusammenarbeit besteht da? [00:02:17-6](#)

B: Da ich ja für Provision zuständig bin ist das natürlich relativ am Ende. Das heisst, im fachlichen Bereich - für Komposit meinetwegen oder Leben - müsste ich dann auch schon mal die Fachtester für Komposit oder Leben sprechen um gewisse Szenarien nachstellen zu können. Also auch da gibt es regen Austausch. [00:02:39-4](#)

I: Ok. Und wie passiert dieser Austausch? [00:02:40-9](#)

B: Indem ich hingehe, eine Email schicke, oder.. der kurze Weg. [00:02:45-9](#)

I: Ok. **TST_TL** hat gemeint, ihr habt für das Testen so eine Testsuite, oder so ein eigenes Tool. Könntest du mir das kurz erklären bitte. [00:02:57-2](#)

B: Ganz genau. Wir haben "**Testtool xxx**". Da werden die Testlisten und die Testanforderungen hineinimportiert. Und aus einer Testliste erstellen wir eine Testsuite wo die einzelnen Spezifikationen aufgelistet sind, und das ist unser.. ja, Reporting- und Controllingtool womit wir halt sehen.. also dieses Fehlerverfolgen.. können.. Wo wir sehen, wenn wir jetzt eine Testspezifikation abgetestet haben - also einen Testfall der aus 30, 35 Testschritten besteht - da kann man genau sehen welchen Testschritt halt, und welcher Fehler vorhanden ist. Man kann eine Testsuite auch wieder aufleben lassen, wenn jetzt zum Beispiel Fehler behoben sind springt man da rein, und kann an der Stelle weitermachen. Also es ist nichts anderes als ein Controllingtool, Reportingtool, ja. Und für die Automatisierer, die nutzen das um automatisierte Tests zu dokumentieren. [00:04:10-6](#)

I: Das sind dann Integrationstests, oder? [00:04:12-4](#)

B: Integrationstests, Regressionstests, und natürlich auch die Systemtests bei einer Auslieferung. [00:04:19-2](#)

I: Ok. Wie schaut deine Sametime-Kontaktliste aus, welche Leute hast du da alle drinnen? [00:04:37-0](#)

B: Natürlich die Leute mit denen ich in erster Linie zu tun habe. Das heisst.. mein.. Also das Entwicklerteam Provision, dann die Fachkonzeption, und mein Team in erster Linie. Natürlich habe ich da noch ein paar weitere Kontakte mit denen man hin und wieder mal zu tun hat, wie zum Beispiel Architektur wenn es um irgendwelche Jobfragen geht, Produktmanagement wenn es um Kundenanfragen geht, beziehungsweise um Änderungsanforderungen die aus dem Konzept resultieren. Also quer durch. [00:05:09-4](#)

I: Ok. Und wie oft schaust du da rein ins Sametime? [00:05:13-6](#)

B: Ins Sametime? Ja das läuft ja parallel immer mit. Ist auch ganz hilfreich um zu sehen wer gerade am Platz sitzt wenn man jemanden direkt anrufen möchte. [00:05:27-3](#)

I: Weisst du zum Beispiel Bescheid was die Entwickler, woran die gerade dran sind? Oder spielt das für dich eine Rolle? [00:05:35-7](#)

B: Ja. Ja, in der Regel weiss ich das schon. Spielt auch eine ganz bedeutende Rolle, weil in dem Projekt "**Provisionsprojekt**" das wir jetzt gerade machen - also das Provisionsprojekt für die Aussendienstabrechnung - da ist ganz wichtig dass ich es weiss welche Features als nächstes fertiggestellt werden, weil dazu gibt es keine separaten Testlisten, sondern die Tests die definiere ich aufgrund von Testnachweisen die erstellt worden sind. Oder von erkannten Problemen aus den Konzepten - sofern welche vorhanden sind - und da ist es ganz wichtig dass ich das ganz eng dran bin an der Entwicklung, und genau weiss welche Features wann fertig sind, damit ich auch mein Zeitfenster zum Testen habe. [00:06:24-8](#)

B: Wir haben alle zwei Wochen ein Sprint-Review. Da stellen die Entwickler ihre Fortschritte - das was sie fertig entwickelt haben - stellen die vor, und das ist auch der massgebliche Punkt, wo.. Ja.. In einem Sprint sind halt gewisse Themen zugeordnet an den Entwicklern, die bis dahin zu schaffen sind. Da wird das einmal vorgestellt, und für mich heisst das, wenn es dann voll-umfänglich OK ist, aus Entwicklersicht, dass ich dann erst mit meinen Tests anfangen kann. Und das findet alle zwei Wochen.. [00:07:01-9](#)

I: .. in einem Besprechungsraum.. [00:07:02-3](#)

B: Ganz genau, in einem Besprechungsraum, wo dann alle Entwickler die dem Projekt zugeordnet sind dabei sind. Auch Produktmanagement, der Projektleiter ist mit dabei, und natürlich auch die Konzeption - oder der Fachkonzepter der da mitgewirkt hat. [00:07:22-7](#)

I: Ok. Ist das dann auch eine andere.. Also Sprint ist ja irgendwie so agile Softwareentwicklung, oder? [00:07:29-4](#)

B: Ja genau, die ist ein bisschen dynamisch.. [00:07:30-9](#)

I: Habt ihr das irgendwie anders? Weil ich kenne das nur von Komposit, da ist ein anderes Vorgehen. [00:07:34-2](#)

B: Ja, also das Projekt "**Provisionsprojekt**" das läuft ein bisschen anders als dieses **KUNDE_1**-Projekt wo am Anfang schon klar ist welche Themen umgesetzt werden, auch in welcher Reihenfolge, also mit welcher Priorität. Das hat man... also in "**Provisionsprojekt**" hat man dieses nicht, da erfolgt das erst auf Zuruf. Welches Feature als nächstes reinkommt. Beziehungsweise es wird auch einiges umgestellt. Das macht allerdings der Projektmanager, **PM_Provisionsprojekt**, in Zusammenarbeit mit dem Kunden. Die sagen uns was als nächstes fertig ist.. oder sein sollte. [00:08:11-2](#)

I: Ist das dann von der Organisation auch schon.. eben weil es einen Projektmanager gibt.. ist das dann auch anders gehandhabt als bei anderen Projekten? Dass ihr dann wirklich auch.. [00:08:23-9](#)

B: Ja, also es ist wesentlich dynamischer, weil man kann da.. Wenn ein Feature nicht so in der Zeit umgesetzt worden ist wie angedacht, wie geplant, hat der Projektmanager immer noch die Möglichkeit mit den Kunden noch ein neues Datum auszuhandeln, oder zu vereinbaren, und dafür andere Themen nach hinten zu schieben um halt mehr Kapazitäten frei zu haben für dieses eine Thema. Das ist ja bei **KUNDE_1**, oder in diesem **LFS**-Projekt - was das andere, das normale - da ist es ja nicht so. Da gibt es die Deadline, da gibt es fixe Termine, und.. Ja, es ist laut Lieferstatus ist es auch fix was alles rein muss. Und natürlich erwartet man auf beiden Seiten eine sehr hohe Qualität. Allerdings ist dort schon in der Planung einiges... Man plant zu viel, mit zu vielen Kapazitäten die einfach nicht vorhanden sind. Also man packt in einen Meilenstein zu viele Themen rein für die Realisierung, für die Umsetzung, die eigentlich gar nicht zu schaffen sind. Und deswegen kommt es da hin und wieder dazu, dass einige Teile nicht in der Qualität sind wie sie hätten sein sollen. Dazu kommt auch, dass - was bei "**Provisionsprojekt**" Gang und Gebe ist - dass man je nach Entwicklungsgrad mit dem Kunden nochmal Austausch hält, also spricht eine Preview schickt damit der Kunde sich da... ja, zurecht finden kann. Weil es werden keine konkreten Anforderungen im Vorfeld definiert. Sondern es wird auf Zuruf auch entwickelt. Das findet ja im **KUNDE_1**-Projekt gar nicht statt. Da gibt es eine Phase für die Konzeption, und wenn ein Konzept abgenommen ist geht es in die Entwicklung, und so zieht sich das dann halt durch. Projekt "**Provisionsprojekt**"

ist da ein bisschen aussen vor. Da ist es sehr stark auf Zuruf. [00:10:15-3](#)

I: Und bist du dann eigentlich noch deinem Projektleiter vom Team - PQM heisst das glaube ich... [00:10:25-8](#)

B: Nein.. Also PQM das sind wir ja. **TST_PM** ist ja mein Teamleiter, ja, dem bin ich in erster Linie unterstellt - das hört sich blöd an - aber ja, er ist mein direkter Vorgesetzter was die Meilensteinthemen angeht. Was das Projekt "**Provisionsprojekt**" angeht ist **PM_Provisionsprojekt** der Projektmanager mir direkt überstellt, und auch weisungsbefugt. Aber das müssen die beiden unter sich aushandeln. Weil ich kann mich nicht zerreißen, und ich bin auch nur eine Kapazität. [00:10:59-1](#)

I: Und wie werden dann so einzelne Arbeitsschritte dann geplant bei dem Projekt bei euch jetzt? [00:11:04-7](#)

B: In "**Provisionsprojekt**"? [00:11:06-2](#)

I: Geht das dann über JIRA, oder wie läuft das ab? [00:11:08-4](#)

B: Nein, in "**Provisionsprojekt**" läuft das nicht in JIRA, sondern da gibt es wie gesagt diese Sprint-Reviews, und für mich ist es da massgeblich dass ich Testnachweise bekomme, die werden in einem zentralen Ordner abgelegt. Dazu bekomme ich meistens ein Email mit einer kurzen Beschreibung, und sobald ich diesen Testnachweis vom Entwickler habe, kann ich basierend darauf eine Testliste erstellen. Beziehungsweise Testspezifikationen definieren, die die Funktionen abdecken. [00:11:36-3](#)

B: Im **Kunde_1**-Projekt ist das ja anders, da gibt es eine Testliste, und die ist ja schon im Vorfeld, bevor die Entwicklung abgeschlossen ist, ist die im Vorfeld schon QSt worden. [00:11:50-4](#)

(JIRA-Login)

B: Ja, so sieht das bei mir aus. (...) Oh, ich habe da zwei neue Dinge zum Meilenstein bekommen. [00:12:35-3](#)

I: Ok, dieses Dashboard: hast du das selber konfiguriert, oder.. [00:12:45-6](#)

B: Ja. In Zusammenarbeit natürlich mit **PQM_1**, beziehungsweise mit **ENTW_DE_TL_PROV**, **ENTW_DE_TL_PROV** ist Teamleiter für das Entwicklerteam Provision. Da sind ganz einfache Filter drinnen. [00:12:58-7](#)

Die wichtigsten Filter für Meilensteinthemen sind diese hier, für "mir zugewiesen". Das heisst, für den nächsten Meilenstein weiss ich jetzt, dass ich diese Beiden, oder dieses Thema bekomme. Das habe ich jetzt zur QS, ich kann mir das jetzt einmal anzeigen lassen, und hier sehe ich die beiden Themen sind bei mir. Um tatsächlich eine vollständige QS durchführen zu können bräuchte ich auch das Fachkonzept, und hier sehe ich auch schon.. da gibt es irgendein Problem, da ist etwas nicht konkret genug beschrieben, also bestimmt wartet man da auf eine Kundenrückmeldung (Anm.: FK-Task ist Kollegin aus FK zugewiesen mit Status "wartet"). Ich kann erst mit meiner QS starten, wenn das Fachkonzept vollumfänglich abgenommen ist. Weil wenn es hier drin im Konzept Änderungen gibt, wirken sich die sich natürlich auch auf die Testliste und die Testanforderungen aus die für mich halt von Relevanz sind. [00:13:57-2](#)

I: Und dieses Projekt das hat jetzt aber nichts mit "**Provisionsprojekt**" zu tun? [00:14:00-0](#)

B: Das hat überhaupt nichts mit "**Provisionsprojekt**" zu tun. [00:14:01-8](#)

I: Ok, das sind jetzt für den nächsten Meilenstein.. was kommen wird. [00:14:03-7](#)

B: Ganz genau. Im Bereich Provision ist es auch recht angenehm, da es kaum mehr was im Meilensteinbereich, sonder immer alles in dieses Projekt "**Provisionsprojekt**" reingenommen. Da wird die Provision halt komplett neu aufgebaut. Also die Vermittlerverwaltung wird neu aufgebaut, also alles mögliche. Und das findet keinen Anklang mehr hier in diesem Bereich. Das sind jetzt nur noch "Partner", weil das Team Provision, das heisst "PRP" - also Partner- und Provisionsberechnungen. Deswegen sind das jetzt die letzten Themen die im Bereich Partner mir zugeordnet sind. [00:14:46-3](#)

I: Wo sitzen eigentlich die Provisionsentwickler? Sind die auch hier im dritten Stock? [00:14:49-5](#)

B: Hier vorne im dritten Stock, direkt hier gegenüber sitzen drei, vier Leute.. Nein, drei Leute. Ein Büro weiter, also die sitzen hier so im mittleren Bereich des Ganges, auch hier im Dritten. [00:15:03-1](#)

I: Und wie erfährst du dann zum Beispiel wenn es irgendwie eine neue, ich weiss nicht, Codeversion oder so gibt? [00:15:09-2](#)

B: Ja, also mit dem Code.. Ich selber lese selten oder gar nicht im Code. Für mich sind die Funktionen das.. Element, oder ja, das Wichtige. Wie es jetzt im Code beschrieben ist, ist mir im Grunde genommen Schnuppe. Ich führe überwiegend Dialogtests aus, und prüfe, dass dann halt in den Datenbanken auch die Daten.. ob Werte richtig persistiert worden sind, also erst mal richtig ermittelt worden sind, und ob sie richtig persistiert worden sind. Der Code selber ist mir ehrlich gesagt vollkommen Schnuppe. [00:15:42-1](#)

I: Aber wenn eben.. dass es jetzt zum Beispiel eine neue Funktionalität gibt: kriegst du da dann eine Email, oder irgendetwas vom Entwickler, oder läuft das dann alles über das Releasemanagement. Dass die sagen, es gibt eine neue.. [00:15:57-0](#)

B: Hm, ja. Also in der Regel läuft das ja über das Releasemanagement wenn die irgendetwas neu.. wie sagt man.. einen Dialog neu bearbeitet haben, dann wird das halt eingecheckt. Aber über Dialogänderungen bekommen wir natürlich auch eine Info. Allerdings sind wir nicht Ansprechpartner Nummer eins, sondern wir bekommen einfach beiläufig die Info dass sich da etwas getan hat. [00:16:15-2](#)

I: Über welchen Kanal bekommt ihr diese Info? [00:16:16-6](#)

B: Als, ja, als Blindcopy, oder nein, nicht Blindcopy, als CC. Dass wir da mit angehängt sind. [00:16:24-3](#)

I: So ein automatisch generiert vom.. [00:16:26-8](#)

B: (gleichzeitig) Ja, sobald jemand eine Dialogänderung meldet.. [00:16:28-6](#)

I:(gleichzeitig) ..vom CruiseControl. [00:16:30-1](#)

B: Nein, nicht CruiseControl. [00:16:31-8](#)

I: Achso, die Dialogmeldungen im Dokupool? [00:16:36-4](#)

B: Ganz genau. Also Dokupool sind die ja auch mit drinnen, und wenn Dialogänderungen sind, bekommen wir eine Email wer diesen Dialog gerade geändert hat. Und da kann man halt über einen Link reinklicken, und sich das dann im Dokupool durchlesen was da geändert worden ist. [00:16:56-1](#)

I: Und wie.. auf welchen Versionen testet ihr dann? Sind das.. wenn der Entwickler etwas eincheckt, wie schnell geht das dann bis es bei euch ist? [00:17:03-5](#)

B: Also grundsätzlich testen wir immer in der aktuellsten Version die vom Releasemanagement freigegeben worden ist. Es kommt ja jede Woche ein neues Build, die Entwickler - sage ich jetzt mal - bei der Fehlerbehebung.. Die beheben einen Fehler, checken das ein, und mit dem nächsten Build ist es in der Regel dann auch.. ja, steht es zur Verfügung. Es gibt allerdings auch Hotfixes, oder Nachlieferungsanträge, da wird das dann noch in die aktuelle Version mit nachgeliefert. Also je nachdem, nach Grad des Fehlers, wie kritisch dieser Fehler ist, kann es auch sein, dass es nachgeliefert wird, also in eine aktuelle Version. Kommt aber eher selten vor, weil wie gesagt, wir bauen jede Woche ein neues Build. Und in der Regel reicht die Zeit auch vollkommen aus. [00:17:58-7](#)

I: Ok. Gut, Benachrichtigung über Email ist es dann (unv...) [00:18:07-8](#)

B: Genau, ganz genau. [00:18:13-5](#)

B: Wenn jetzt zum Beispiel, gerade bei der Fehlerverfolgung ist das auch ganz interessant. Wenn ich jetzt einen Fehler aufgegeben habe der.. den weise ich dann dem entsprechenden Teamleiter zu, der weist das in seinem Team weiter bis zur Realisierung, und wenn das Ticket dann auf "testbereit", also sprich "realisiert" eingecheckt ist, dann wird das automatisch.. wird der Status auf "testbereit" gestellt, und dann bekomme ich auch wieder eine Email. Das heisst, dass dann mein Fehler realisiert und testbereit ist. Da kann man halt auch über das TTS, also über das Troubleshoot-System, kann man auch einen Nachttester eingeben, so dass der Eröffner nicht unbedingt der Nachttester ist. Weil es kann ja auch sein, dass der Entwickler einen Fehler sieht, und mich dann als Nachttester mit rein setzt. So bekomme ich dann halt auch Infos was zu tun ist, was geändert worden ist, was testbereit ist. [00:19:05-2](#)

I: Ok. Und das JIRA verwendet ihr dann nur für die QS, oder wie? [00:19:10-5](#)

B: Ganz genau. Beziehungsweise, wenn da jetzt irgendwie was.. eine Anmerkung von uns zum Konzept ist, dann natürlich auch für die Fachkonzeption dass wir dann eine neue Version machen mit den offenen Punkten die wir haben, und der Konzeption wieder zurückschicken. [00:19:26-3](#)

I: Diese Anmerkungen, wie macht ihr das? Verwendet ihr diese Kommentarfunktionen dann, oder... [00:19:31-7](#)

B: Nein. (klickt). So, hier sind ja jetzt schon drei Versionen da. Ich würde mir die hier jetzt hier herunterladen, durchlesen, wenn ich eine Änderung habe würde ich eine Version vier daraus machen. Die dann hier mit anhängen, und dann hier den Kommentar mit dazu. Und dann natürlich auch wieder zuweisen an die entsprechende Person. Also in diesem Fall an **FK_5**. (Siehe Video 6:51 - 7:20: Task mit FK in drei Versionen und mehreren Kommentaren, zeigt Download-Möglichkeit und Kommentarmöglichkeit) [00:20:00-5](#)

I: Achso, du erstellst selber gleich die neue Fachkonzeptversion, oder was? [00:20:03-6](#)

B: Ganz genau. [00:20:04-6](#)

I: Das heisst, das ist dann auch wieder ein anderes Vorgehen als beim Komposit-Team zum Beispiel. Da gibt es ja so einen QS-Button, wo sie dann im JIRA unten da so einen Kommentar.. [00:20:17-4](#)

B: ..ja, reinschreiben, und dann im Konzept vielleicht etwas anmerken, genau. Bei uns ist es so, dass wir uns die Konzepte lokal abspeichern, also herunterladen. Wenn da irgendetwas unklar ist, oder fachlich nicht korrekt - was eigentlich NIE vorkommt - oder wirklich eine gravierende Abweichung, dann muss man schauen, ob es entweder am Fachkonzept liegt, an der Testliste, oder an den Testanforderungen. Und die schicken wir dann zur Nacharbeitung zurück an den entsprechenden Ersteller. [00:20:50-6](#)

I: Ok. Und das ist aber jetzt ein Fachkonzept, und keine Testliste.. (zeigt auf den Bildschirm wo der FK-Task offen ist) [00:20:53-2](#)

B: Das ist jetzt Fachkonzept, genau. Und die Testliste die habe ich mir selber noch gar nicht angeschaut.. (klickt) Da ist jetzt die Version eins, die würde ich jetzt QSen, und in der Testliste... (Siehe Video 7:21-8:52: Task mit Testliste in Excel-Format --> öffnen der Excel-Liste und zeigen des QS-Reiters) [00:21:07-8](#)

I: Da gibt es ja irgendwie einen eigenen Reiter dann in der Testliste.. [00:21:10-5](#)

B: Ganz genau. Da gibt es "QS", also so einen Report, genau.. (...) Und hier könnte ich dann meine offenen Punkte in der Version 001 die ich jetzt hier habe, meine Frage.. Und dann würde ich diese Version zwei hochstellen, wieder zurückschicken, und warten bis die Antworten da sind. Aber in der Regel passiert dieses per Telefon, dass man da halt anruft und nachfragt. [00:21:39-6](#)

I: Dass du gleich direkt sagst dass das da ist. [00:21:41-6](#)

B: Ganz genau. Weil der kurze Weg ist doch besser, und geht auch schneller. Ist effizienter, als erst wieder eine neue Version in JIRA, das dann wieder zuweisen, der zieht sich das runter, schaut sich an, bearbeitet dieses, setzt es wieder neu rein als nächste Version, schickt es mir dann wieder zu... Ist zu umständlich. [00:22:03-2](#)

I: Ok. Und bei dem Fachkonzept zum Beispiel, wo würdest du, wenn du die Punkte ins Konzept reinschreibst, wo würdest du das reinschreiben? [00:22:11-7](#)

B: Im Fachkonzept? Die Punkte würde ich... da gehe ich mal hier in das Konzept rein.. (wechselt in FK-Task in JIRA) Also ich würde das jetzt mir herunterladen, und ich würde dann markieren, gelb hinterlegen, und dann ein separates Dokument mit meinen Fragen auflisten. Also gelb markieren... meinetwegen wenn jetzt hier etwas unklar ist, dann würde ich das hier markieren, mit einer entsprechenden Farbe (klickt)... so in etwa, und dann auf einem separaten Blatt mit beifügen und hochschicken. Wo dann die Fragen und Antworten.. [00:23:03-1](#)

I: Habt ihr dann irgendwie so ein standardisiertes Formular dann für diese QS? [00:23:07-9](#)

B: Ein leeres Blatt mit.. durchnummeriert. "Frage 1", und halt immer bezogen auf das jeweilige Dokument, also sprich auf das jeweilige Kapitel im Dokument. (Siehe Video 9:40 - 10:17) [0:23:24-3](#)

I: Ok. Das weicht dann eben auch ab von Komposit. Da hat es ja früher so ein eigenes QS-Dokument gegeben. [00:23:30-5](#)

B: Genau, ein Template. Genau. Aber wie gesagt, Konzepte selber anpassen machen wir nur mit Absprache des Konzepters. Weil das ist immer eine Sache zwischen dem Konzepter und dem Kunden. Und wenn der Kunde wirklich etwas unsinniges haben will, dann soll er es auch kriegen. [00:23:52-9](#)

I: Schickst du das dann zum Beispiel dem Entwickler auch mit, dass der auch gleich weiss, dass sich etwas geändert hat? Oder... [00:23:57-8](#)

B: Ja. Wenn sich da gravierend etwas ändert, dann natürlich. Klar. [00:24:03-6](#)

I: Also das machst du dann gleich von dir aus, und wartest nicht bis die Fachkonzepter wieder dann den Entwicklern bescheid geben. [00:24:10-9](#)

B: Also da haben wie ja es in Provision recht einfach. Da haben wir einen Konzepter, und der sitzt auch bei den Entwicklern, wegen dem "**Provisionsprojekt**"-Projekt. Da.. einfach hingehen, mit dem das besprechen, und dann gibt es von ihm gleich die Ansage "ja, das kommt dann so und so". Also da haben wir echt schon.. sind wir sehr konzentriert. [00:24:30-0](#)

I: Wieviele Leute seid ihr da in dem "**Provisionsprojekt**"-Projekt? [00:24:32-4](#)

B: Wieviele Leute? Wieviele Entwickler hat PRP... Also ich kann dir jetzt alle aufzählen (zählt). Zwölf Entwickler, ein Konzepter, ein Testdesigner, dann ist da der **PM_ Provisionsprojekt** noch mit dabei als Projektleiter, ja. 15 Leute. Und dann haben wir noch einen externen Tester für das Projekt. Also 16 Leute für dieses Projekt [00:25:17-4](#)

I: Und wie lange habt ihr schon dieses Vorgehen da, so mehr dieses agile Ding? [00:25:19-9](#)

B: Das agile... also dieses dynamische wie ich es so schön nenne, (...) das war analog zu M2010.2 fing es an. Genau. Zu Meilenstein M2010.2. Also quasi gut ein Jahr jetzt. [00:25:45-5](#)

I: Und wieso wurde das eingeführt dieses Vorgehen? Oder war das zuerst auch so wie bei Komposit, so ein Wasserfallmodell? [00:25:53-4](#)

B: Nein. (es folgen projektspezifische Ausführungen, die mit Awareness oder der Zusammenarbeit von KollegInnen nichts zu tun haben). [00:28:40-2](#)

I: Alles neu. [00:28:40-5](#)

B: Alles neu, ja. Provision ist ein bisschen anders als Komposit. [00:28:51-1](#)

I: Na gut. Schaust du zum Beispiel CruiseControl oder solche Sachen, kuckst du dir das an? [00:28:58-2](#)

B: Ja.. und zwar.. Nur indirekt. Natürlich haben wir auch in Provision ein paar Scripte laufen. Zum Beispiel unsere Testvermittler die wir für Tests nutzen, aber auch die Entwickler um ihre DB-Unit-Tests durchführen zu können, die nutzen einen Testvermittler den wir mit eingecheckt haben auf dem Trunk. Und wenn es dort irgendwelche Abweichungen gibt, weil bei "**Provisionsprojekt**" kommt es auch vor, dass das Datenmodell ein bisschen geändert wird - also ein paar Felder aus einer Datenbank fallen raus, oder werden ergänzt - dann kriegt man natürlich eine Meldung über CruiseControl dass das Script nicht mehr funktioniert. Eine Sache, aber das ist der Berührungspunkt den ich da habe. Dann haben wir noch ein "Vermittler-Konsistent-Check-Script" laufen, was zu allen Vermittlern die Peripherietabellen mit überprüft. Wenn es da Abweichungen gibt, gibt es auch eine Meldung von CruiseControl. Da weiss man, da hat sich wieder was geändert, da sollte man wieder neue Anpassungen vornehmen. [00:30:09-3](#)

I: Verwendest du so.. diese Funktionen "Überwachen" und so? [00:30:18-5](#)

B: Nein, gar nicht. [00:30:23-6](#)

I: Schaust du zum Beispiel manchmal rein was die **FK_5** macht zum Beispiel - in JIRA? [00:30:38-3](#)

B: Wenn ich mal Zeit hätte, dann eventuell. Aber dafür habe ich keine Zeit, um mich um die Belange der anderen... Wenn ich natürlich auf Sachen warte die mich da selber blockieren, dann schaue ich da schon mal rein, klar. Um gegebenenfalls Druck zu machen, weil man muss ja doch irgendwie vorankommen. [00:30:57-7](#)

I: Das ist eben wenn das da "wartet" oder so zum Beispiel, oder? [00:30:59-6](#)

B: Ganz genau. Oder wenn ich da etwas zurückgegeben habe, beziehungsweise **FK_5** könnte sich jetzt auch demnächst bei mir melden "was ist denn mit der QS, warum dauert das denn so lange", klar. Dafür wird das dann auch gerne mal genutzt. [00:31:17-3](#)

I: Weisst du zum Beispiel welche von den Entwicklern genau jetzt gerade die Funktionalität programmieren die du jetzt testen wirst? [00:31:24-6](#)

B: In der Regel schon, weil es gibt ja hier diese Implementierungstasks... (kreist mit der Maus)... Gibt es hier nicht... Doch. Hier. Aber ist noch nicht zugewiesen. Und sobald - das ist auch ein Thema für den nächsten Meilenstein - jeder Implementierungstask oder jede Implementierung wird einem Entwickler zugewiesen, und das ist dann auch mein direkter Ansprechpartner. [00:31:54-9](#)

I: Ok. Das weisst du dann also aus dem JIRA. [00:31:56-6](#)

B: Das weiss ich dann aus dem JIRA, genau. Beziehungsweise offen in die Runde gefragt "wer von euch macht das", und dann kriegt man da auch schon eine Antwort. [00:32:07-0](#)

I: Ok. Ja, dann sage ich mal herzlichen Dank. [00:32:10-4](#)

B: Ja, gerne.

Interview 16_ENTW_DE_AL

I: Ja, was ist deine Aufgabe im Team, und wie sieht dein täglicher Arbeitsablauf aus, so ungefähr? 00:00:12-7

B: Der ist sehr unterschiedlich. Also ich bin Leiter der Abteilung Softwareproduktion, das heisst, ich steuere die vier Entwicklerteams die so im Bereich Standardsoftware engagiert sind - das Team Leben, das Team Komposit, Zahlungsverkehr und DMS, und Provision und Partner. Mache das.. also disziplinarische Verantwortung dann für die vier Teamleiter und bin.. wir sind verantwortlich für den Hauptprozess.. oder für den Teilprozess Software entwickeln. 00:00:47-7

B: Alles was aus der Fach.. Input ist das, was die Fachkonzeption uns liefert, und Output generieren wir für den Testprozess, die übernehmen dann unsere Arbeitsergebnisse. Es ist also eine Prozessverantwortung, und dort Ergebnisse und Prozess zu steuern. 00:01:07-3

I: Und mit welchen Leuten kommunizierst du täglich, oder im Laufe deiner Arbeit? 00:01:13-6

B: Also die wichtigsten Kommunikationswege sind - für mich - die Kommunikationswege mit den Teamleitern, den vieren. Die das ja im Detail für ihre Teams ausgestalten, die dort auch viel Kompetenz und Verantwortung haben das für ihre Teams zu machen. Und wir führen das dann so auf der Abteilungsebene zusammen. Koordinieren das unter den vier Teams, verabreden uns, sorgen so ein bisschen für Einheitlichkeit. Das ist so ganz zentral für meine Arbeit. 00:01:53-1

B: Und klar gibts meinen Chef, den Ressortvorstand für die Produktion **VORSTAND_1**. Es gibt die Fachkonzeption, natürlich gibt es dahin auch immer wieder Fragen, Absprachen. Produktmanagement die so ein bisschen (unv...) auch gestalten sollen. Das spielt nicht unbedingt täglich eine Rolle. 00:02:19-2

B: Wichtig ist dann auch der Prozessnachfolger sozusagen - das Testteam. **TST_TL** und die Tester, wo es dann ja auch, wenn die testen, ein Zusammenspiel gibt. Die haben ja dann ihre Funde die wir wieder erledigen müssen. Und die müssen wissen wann wir Software übergeben können die sie testen können und so weiter. Und es gibt die Schnittstelle zum Kunden, dort bin ich Ansprechpartner, einmal für die.. für alles was sich um Liefertermine dreht, um den.. jetzt noch in der aktuellen Entwicklung die Vorbereitung.. also da begleite ich zusammen mit **TST_TL** - ich glaube das haben wir jetzt gar nicht so genau definiert, das hat sich so entwickelt - wir beide arbeiten im Grunde zusammen die Releasesetzung vorzubereiten, diesen Releasetermin vorzubereiten. 00:03:21-0

B: Und auch in der Zeit danach, wenn die Software in Produktion ist, dann steuere ich das Codefreeze-Verfahren und die Patchtermine. Also das ganze Verfahren wie gehen wir mit Fehlern um die aus der Produktion gemeldet werden. Also Patches planen, dann Fehleranalysen QSen, diesen Prozess behudeln - wie beauftragt der Kunde DAS, passt das mit dem was die Teams erledigen können, kennen die Teams die Termine, und so weiter. Das ist wie so ein eigener kleiner Prozess, den ich dann recht nahe an den Leuten auch steuere. 00:04:05-7

I: Und dann ist eben die Frage, wie informierst du dich über den Projektablauf, oder über den Projektfortschritt, oder über Probleme die es im Projekt gibt. 00:04:16-1

B: Wir haben.. also wenn man jetzt mal.. Fangen wir mal an mit dem.. Also ein Meilenstein beginnt immer mit der Lieferzusage. Das ist der Moment, wo wir von Produkt- und Auftragsmanagement eine Liste von Themen bekommen die jetzt abzuarbeiten sind. Die kennen wir auch, weil wir die Fachkonzepte.. die Leute kennen die Themen - ich persönlich weniger - weil die die Fachkonzept-QS machen. Und die Teamleiter bereiten einen Vorschlag vor welche Aufträge nehmen wir jetzt an, und auf welches Programm können wir uns committen für diesen Meilenstein. Im Rahmen der Lieferzusage wird dieses Programm festgezurt. 00:04:53-5

B: Dann beginnt der Kern unserer Tätigkeit, den wir.. wo die Teams anhand von JIRA ihre Tasks erledigen und steuern. 00:05:08-0

B: Auf Ebene der Abteilungen, mit den vier Teams, arbeiten wir in zwei-Wochen-Iterationen, dann OHNE JIRA - das wäre zu detailliert, das würde mich auch erschlagen wenn ich das so über vier und noch mehr Teams (unv. entscheiden) würde. Dieser Produktionsplanungsprozess, der dient dazu Ziele, die Termine im Auge zu behalten, und die Arbeiten der vier Teams untereinander abzustimmen. Das die eine Orientierung haben, welches Team beschäftigt sich gerade mit welchem Thema - es gibt auch ja immer wieder Themen wo es Zusammenhänge gibt, wo die vier Teams sich koordinieren müssen - und auch das Testteam was ausserhalb der Abteilung liegt, ist daran beteiligt an diesem Produktionsplanungsprozess. Die müssen planen können "wann kommen die Themen bei uns an". Das machen wir 14-tägig.. gibt es ein Produktionsplanungsmeeting was dann auch von den Teamleitern vorbereitet wird - "was planen wir in den nächsten zwei Wochen", man schaut auf das "wie läuft die aktuelle Iteration" - und es gibt eine Excel-Datei in der der Stand sozusagen kumuliert wird zu den Aufgaben, wo

man den Status abfragt zu Zwischenergebnissen, und eine Aussage macht zu den Ende-Terminen - wann wird man mit den Aufgaben fertig sein, was ist momentan der Plan. Und wir machen das Risikomanagement auch über diese Excel-Datei. All diese Sachen besprechen wir 14-tägig in diesem Meeting, und erstellen dann auch einen Bericht für den Produktionsvorstand, und für die die sich dafür interessieren - unseren internen Auftraggeber das Produktmanagement. [00:06:55-7](#)

I: Und eben so.. eben Probleme oder so, das kriegst du quasi über die Teamleiter mit, oder.. [00:07:15-0](#)

B: Ja. Über die Teamleiter, klar. Die rühren sich, wenn sie Probleme mit ihren Aufgaben haben, oder mit ihren Ressourcen, wie auch immer. Oder spät in der Phase kann es sein, dass auch der Kunde sich meldet. Das.. [00:07:38-7](#)

I: Und diese ganze Kommunikation, das ist aber dann eben alles auf mündlicher Basis? [00:07:46-4](#)

B: Ja. Primär ja, das ist auf jeden Fall das Initium, und findet dann seinen Niederschlag in den Daten zum Risikomanagement.. in unserem Softwareplanungsprozess. [00:08:05-9](#)

I: Ok. Wie wird dann dieses.. es gibt ja diese eine Excel-Liste eben wo diese ganze Softwareproduktion.. diese Planung eben drin ist. Wo eben auch die Risiken drinstehen. Wer bearbeitet die alles? Also wer hat darauf Zugriff, und wie läuft das da ab? [00:08:36-5](#)

B: Die erstelle ich, also ich fülle die Initial mit den Aufträgen, die Teamleiter machen QS drauf - "ist das tatsächlich alles", "fehlt da irgendetwas". Dazu nutze ich Daten die vom Auftragsmanagement kommen, aus der Lieferzusage. [00:08:52-8](#)

I: Die Aufträge sind dann die einzelnen Fachkonzepte, oder wie? [00:08:56-0](#)

B: Die.. ja, das ist eine Nummer und eine Bezeichnung für den Auftrag. Und unter dieser Kennung hängen Fachkonzept-Aufträge und Fachkonzept-Dokumente und dann eben auch Realsierungs-Aufträge, Implementierungs-Aufträge, und alles was daran machen hängt dann an dieser Kennung. Und so gibt es dann eben eine Zeile auf dieser Ebene.. führen wir das in dieser Datei. [00:09:28-1](#)

B: Dieses Laufwerk.. also lesen kann da jeder, da kann sich jeder darüber informieren. Schreibzugriff haben diejenigen die da Daten zuliefern müssen, das sind die Teamleiter und ihre Stellvertreter. [00:09:45-1](#)

I: Ok, aber nicht nur eben von der Entwicklung, sondern es schreiben glaube ich auch die Tester rein, oder? [00:09:49-2](#)

B: Die.. also hauptsächlich ist es dann eine Entwicklerangelegenheit. Es gibt auch zur.. zu einer Auslieferung haben wir Dokumentationsteile die von der Fachkonzeption erstellt werden, so dass auch die Fachkonzept-Teamleiter und ihre Stellvertreter da dokumentieren "wie ist der Status zur Onlinehilfe", "wie ist der Status zum Dokupool", also solche Sachen tragen sie da ein. [00:10:21-4](#)

B: Und das Testteam genau, die hat sozusagen die ganze.. also der ganze rechte Teil von der Datei, die ganzen Testlisten-Sachen, werden von dem STQ-Team (unv. eingetragen). [00:10:35-9](#)

I: Was würdest du sagen, wie gross ist bei dir der Anteil zwischen.. also an Informationen die du bekommst: mündliche Sachen und über, ich weiss nicht, solche Reports eben. Kannst du das irgendwie sagen, oder kann man das irgendwie vergleichen? [00:10:55-0](#)

B: Bezogen auf.. wie ich.. wie ich..? [00:11:00-7](#)

I: Auf das Gesamtprojekt halt eben.. Das du den Überblick behältst, wie der Status ist.. [00:11:06-4](#)

B: Ja, also alles was 08/15 ist, was sich im normalen Bereich beläuft, das läuft rein über die Datei. Das.. da spricht man ja kaum darüber, da hat man eher dann nochmal eine Kontrolle über "hat man nichts vergessen", und "ist das tatsächlich so" und... Also das läuft so über diese Datei. [00:11:27-7](#)

B: Da wo tatsächlich Probleme auftauchen, das merkt man.. das kommt darauf an. Also wir haben eben diesen zwei-Wochen-Rhythmus, wenn da in der Nähe dieses Meetings ein Problem aufpoppt, dann ist es recht wahrscheinlich, dass es dort in Vorbereitung auf das Meeting dann eingetragen wird, und wir in dem Meeting darüber sprechen, ich so das erste Mal darüber erfahre. Ist man gerade so mitten in der Iteration oder so, das Meeting hat gerade stattgefunden und es wird länger keines stattfinden, dann kommen die Teamleiter - die natürlich sowieso recht häufig bei mir vorbeilaufen, täglich eigentlich - dann würde ich erst mal mündlich darüber informiert. Und wir halten diese Datei auch nicht zwingend laufend aktuell, aber alle zwei Wochen muss sie

aktualisiert werden. [00:12:15-9](#)

I: Ok. Und was sind eben so die Risiken, oder wie fein ist das granuliert diese Risiken, also.. [00:12:21-7](#)

B: Risiken beziehen sich bei uns, diese.. wir nennen das dann.. Die Projektrisiken beziehen sich immer auf den vorgesehenen Termin: wann soll ein Auftrag oder ein wichtiges Zwischenergebnis - wir haben da jetzt dann zwei Qualitygates im Auge - ist der Termin der jetzt geplant ist, ist der in Gefahr? Und wenn er in Gefahr ist, um wieviele Tage mag es sich möglicherweise drehen. So auf dieser Ebene erfassen wir Risiken.[00:12:55-7](#)

I: Und Risiken sind dann einzelne Tickets, oder was kommt da alles rein? [00:13:01-7](#)

B: Ja, es können.. Alles was diesen Termin gefährdet. Das kann sein, dass jemand ungeplant ausfällt weil jemand krank ist - nicht nur für einen Tag sondern länger. Man stösst auf fachliche Probleme die man vorher in der Konzeptionsphase nicht gesehen hat - das ist ein ganz natürliches Phänomen. Natürlich genauso wie es technische Probleme geben kann. Es kann Abhängigkeiten geben, Zuarbeiten von denen man nicht weiss, ob die pünktlich erledigt werden. Das ist eigentlich so das klassische Repertoire. [00:13:44-9](#)

I: Wenn du jetzt zum Beispiel.. also man sieht es ja hier, das sind ja die Leute immer ziemlich eng immer zusammen. Also wenn hier ein Entwickler ein Problem hat, dann geht er eben rüber zum Fachkonzepter und fragt.. redet halt direkt mit dem. Wie siehst du das, wenn man sich jetzt zum Beispiel ansieht die Arbeit die mit den **Österreichern** passiert, wo liegen da dann die Probleme, oder wo gibt es da Sachen wo man sagt "da hakt es in der Kommunikation" oder so jetzt. Die dann vielleicht nicht auftreten würden, wenn die Leute hier vor Ort wären? [00:14:20-2](#)

B: Ich glaube, dass die heutige **LFS**-Landschaft recht.. eine recht komplexe Softwarelandschaft ist, die sehr eigen ist. Und bei der man sehr viel wissen muss, und bei der man sich lange einarbeiten muss bis man sich das zurechtfindet. Und in dem Sinne ist das kaum möglich auch einfach mal so Aufgaben nach Extern zu vergeben, oder.. Dadurch ergeben sich auch Schwierigkeiten wenn die Distanz zu.. zu Partnern die mir helfen können, wenn ich im Fachkonzept.. wenn ich ein fachliches Problem habe oder so. Wenn diese Distanz gross ist. Ich kann das jetzt gar nicht so spezifisch sagen. Also ich.. [00:15:21-8](#)

I: Aber wie ist es jetzt zum Beispiel für dich persönlich jetzt, dass du eben weisst ob es in **Österreich** Probleme gibt, oder so. [00:15:35-0](#)

I: Also dass du einfach über den Projektablauf.. ja da gibt es eben, ich weiss nicht.. dass man sagt "ok, jetzt läuft es gut", "jetzt läuft es mal weniger gut", "hier sind Tickets aufgepoppt", "da gibt es irgendwo einen Engpass", "hier ist dieses Problem".. wie weit bist du da jetzt zum Beispiel am Laufenden? Oder gibt es da einen Unterschied zum.. wenn du da jetzt - sagen wir mal - einen Kompositentwickler, wie du da informiert bist, und wie du zum Beispiel über Komposit in **Österreich** informiert bist. Oder spielt das eine Rolle für dich? [00:16:11-3](#)

B: Nicht so stark. Das spielt eine viel, viel grössere Rolle für den Teamleiter. Das ist die Ebene die noch dazwischengeschaltet ist sozusagen. Der muss das viel detaillierter im Auge behalten als ich, und der.. für den ist das natürlich was anderes ob er den im Büro nebenan sitzen hat, oder ob er das mal über die Telefonleitung erfahren muss oder so. Da ist die Ebene dazwischen, deswegen ist das für mich persönlich, und für meine Rolle jetzt.. da ist ja schon mal.. das ist eher mittelbar. [00:16:55-0](#)

I: Aber du tust ja zum Beispiel die REA-Anträge genehmigen, oder? Das liegt ja in deinem.. [00:16:58-4](#)

B: Das macht mehr das Releasemanagement. Ich unterstütze die mehr, oder springe da mal ein wenn ich sehe dass ich jetzt gerade etwas beschleunigen kann weil die nicht am Platz sind oder so. Also das unterstütze ich hier. [00:17:11-9](#)

I: Ok. Also da spielt es dann keine Rolle für dich.. Ich weiss nicht.. wenn du jetzt zum Beispiel weisst.. oder ich weiss nicht.. ist das dann planbar oder so irgendwie (murmelt: das ist eine schlechte Frage). [00:17:28-7](#)

B: Also für mich, also für die Rolle die ich da ausführe, ist das eigentlich.. weil ich viel über die Teamleiter steuere, ist das Problem für mich wenig existent. Ich kann das so ein bisschen... das spielt jetzt nicht so eine Rolle, ob jemand das.. Wir haben einen Entwickler in **Süddeutschland**, wir haben die Mitarbeiter in **Österreich**, das spielt für mich persönlich nicht so eine grosse Rolle, weil ich dann doch.. weil ich das doch über den Teamleiter mache oft, und nicht über 75 Entwickler in Einzelsprache. Ich mache das über.. ich mache vieles über vier Teamleiter, und nur da wo man direkt über einen Fehler, über eine Analyse oder so spricht, da ist der direkte Kontakt da. Aber sonst.. ist das ein Teamleiter-Problem, und ein Problem des virtuellen Teams. Und weniger für die Rolle des Produktionsleiters. [00:18:39-0](#)

I: Ok. Welche.. wie hast du deine Sametime-Kontaktliste organisiert? [00:18:44-7](#)

B: Da habe ich Teams eingebunden. Also man kann ja in Sametime.. Also Teams, da braucht man auch nicht einzelne Kontakte einbinden oder so. Ich habe glaube ich oben meine Teams stehen und die Fachkonzept-Teams, und das sind so die Gruppen die für mich dann so am wichtigsten sind. Die stehen da irgendwie so.. also das ist jetzt gar nicht mal streng die Reihenfolge, aber ich habe das Teams reingepackt mit denen ich häufiger zu tun habe. [00:19:19-9](#)

I: Wie oft schaust du da rein, in die Kontaktliste? [00:19:23-1](#)

B: Die ist eigentlich immer offen an der rechten Seite. Und.. wenn man mit Notes zu tun hat, ist die irgendwie immer da, die ist immer so im Blickwinkel irgendwo, ja. Also dass ich konkret reinschaue, ja dann wenn ich kucken will, ob jemand da ist oder nicht. Dazu muss man aber auch wissen ob.. was für eine Art Sametime-User er ist. Es gibt Leute die sind gar nicht da, es gibt Leute die sind nicht immer angemeldet, und es gibt Leute, da weiss man der ist.. wenn der jetzt da nicht sichtbar ist, dann ist er gerade mit seinem Laptop wo anders, oder er ist tatsächlich nicht da. [00:20:03-2](#)

I: Also JIRA.. hast du gar nichts? [00:20:13-4](#)

B: Nein. Also wir haben zusammen die Struktur aufgesetzt, aber ich sehe es.. habe es bisher gesehen als Tool der organisationellen Teamarbeit.. Auf wirklich sehr detaillierter Ebene. Und in der würde ich mich verlieren, wenn ich mich da täglich aktuell halten würde. Also ich kann da reinkucken, ich kann mich da auch zurechtfinden, aber ich nutze das ganz wenig, eigentlich gar nicht für meine Arbeit. Ich lasse mir dann eher wieder zuliefern im Rahmen der Produktionsplanung von den Teamleitern. [00:20:59-5](#)

I: Na gut. Ja, ich glaube das war es dann. Es ist halt schwierig das zu fragen. [00:21:17-3](#)

B: Ja, es ist... also meine Rolle, meine Arbeitsweise, die weicht dann eben auch ab von dem was dich im Kern sonst auch interessiert (Anm.: Interviewpartner wurde im Vorfeld über das Projekt informiert, insbesondere dass es dabei um JIRA geht. IP merkte schon vor Interview an, dass er JIRA nicht verwendet, wurde aber ob seiner Rolle trotzdem als IP ausgewählt), und wozu die anderen auch spezifisch Auskunft geben können. [00:21:32-3](#)

I: Ja, auf **ENTW_DE_TL** bin ich eh gespannt. (...) Wie oft schaust du ins Ticketsystem? [00:21:42-2](#)

B: Täglich. [00:21:41-8](#)

I: Das schon? [00:21:43-3](#)

B: Ja, das ja. Einfach auch, weil es eine Rolle spielt für alle möglichen Auslieferungen. Und ich erstelle ja auch wöchentlich Berichte zu diesen Tickets, um zu kucken wie liegen die Teams im Rennen, muss man da spezielle Aufmerksamkeit darauf lenken, und spezielle Aktionen planen, dass da nicht etwas aus dem Ruder läuft. Und ich kucke wirklich fast täglich rein glaube ich. [00:22:19-7](#)

B: Und vor allen Dingen auch, um die Produktion.. den Produktivbetrieb, den Codefreezebetrieb da im Auge zu behalten. [00:22:30-6](#)

I: Und wie merkt man das dann, ob es aus dem Ruder läuft? [00:22:33-0](#)

B: Wenn Ticket.. wenn Zahlen offener Tickets kontinuierlich steigen über mehrer Wochen. Wenn es heftige Ausschläge gibt. Also.. das sind so Trenddiagramme, die auch pro Woche ausweisen was dazugekommen ist, wurde gelöst irgendwie.. Und wenn da einmalig etwas anderes ist, das hat noch gar nichts zu sagen. Aber wenn es über ein Monat hinweg oder so eine Tendenz aufweist die dann... Spätestens dann muss man sich mal informieren woran das liegt - das kann auch gute Gründe haben die jetzt keine speziellen Aktionen erfordern, oder Probleme von denen man glaubt dass sie sich in den nächsten Wochen ohnehin erledigen. Oder man muss dann tatsächlich mal speziell etwas planen irgendwie. [00:23:25-8](#)

I: Die Auswertung die passiert dann aber mit.. Also du.. Oder du schaust jeden Tag rein, und weisst halt ungefähr wie der Stand ist? [00:23:41-3](#)

B: Bitte? [00:23:42-0](#)

I: Du schaut dann jeden Tag rein ins Ticketsystem, und weisst dann halt so wie der Stand ist. Also wenn da jetzt auf einmal zehn zusätzliche Tickets hereinkommen, dann siehst du einfach dass die Liste jetzt länger ist? [00:23:53-8](#)

B: Ja.. das.. man.. Das Ticketproblem ist ja ein sehr grosses momentan. So dass man dann.. dass dann auch so

bestimmte Entwicklungen auch.. gehen einem dann auch durch. Dann ist es schon gut, wenn man das dann auch in dieser kondensierten Berichtsform dann nochmal sieht wie sich da ein Graph entwickelt irgendwie. So hat man natürlich ein Gefühl dafür irgendwie.. man sieht so was.. wofür Ticketmeldungen reingekommen sind, dass viel zu einem bestimmten Thema kommt oder so, aber... 00:24:24-4

I: Also liest du das schon auch durch, die Beschreibungen, die Ticketbeschreibungen und so? 00:24:28-1

B: Ich sehe den Kurztext irgendwie. Wenn mir etwas auffällt, kucke ich auch mal rein. Ich kucke so ein bisschen, ob die.. wie die... verschiedene Attribute ob das so passt. Ich habe da glaube ich auch ein ganz gutes Gefühl dafür, ich sehe da auch wenn da mal etwas falsch übertragen wurde oder so. Und man hat dann auch immer mal so komische Vögel in solchen Systemen auch drin, die man dann gerade ziehen muss. 00:24:57-1

I: Ok. Dann sage ich herzlichen Dank. 00:25:03-0

B: Gerne.

Interview 17_ENTW_DE_TL

I: Also die erste Frage ist, was ist deine Rolle hier im Team, und was sind so die typischen Aufgaben die du täglich erledigst? 00:00:25-1

B: Also meine Aufgabe im Team ist Teamleitung, dazu gehören grob zwei Hauptblöcke. Das eine ist die Personalführung, das sind solche Sachen wie.. typische Personalmanagementgeschichten. Also Einstellungen, Zeugnisse, Beurteilungen und sowas. Aber auch Ansprechpartner sein für das Team und ähnliche Sachen, also alles was man unter Personalführung, Weiterentwicklung der Mitarbeiter versteht. Und das andere ist die Planung und Verteilung dann von Aufgaben. Das Controlling über diese Aufgaben, das Berichtswesen das dazugehört - nach Aussen berichten, innerhalb berichten, was auch da gefordert ist. Also die Prozesse in der Rolle Teamleiter dann (unv. im Alltag ausleben). Dass man da.. vorwiegend hängt das damit zusammen, dass man... oder die vorwiegende tägliche Aufgabe ist da eigentlich so eine Art Fortschrittskontrolle zu halten, dass man ein Controlling hat "wo stehen wir", "wie kommen wir aus Problemen raus", "wie manage ich Probleme" und sowas. Also das gehört da alles dazu. 00:01:45-9

I: Und wie merkst du zum Beispiel dass es Probleme gibt? 00:01:50-5

B: Naja, das können ganz unterschiedliche.. also Faktoren können das sein die Probleme aufzeigen.. Das ist ein.. 00:01:58-3

B: Zum einen Rückmeldungen von Aussenstehenden - ob das Kunden, Qualitätssicherung oder ähnliches ist. Also das sind dann mehr so.. wirklich technische Probleme, also dass man sagt wir haben ein Stück Software gebaut was nicht so funktioniert, oder nicht so aussieht wie es dann sein sollte. 00:02:15-0

B: Dann gibt es terminliche Probleme, das kann man... wir haben ein Meilensteinraster an dem wir uns ausrichten, und die Meilensteinraster.. muss man an bestimmten Punkte halt irgendwelche Teilaufgaben fertig haben. Ansonsten bekommt man da Probleme. Man hat eine Vorplanung gemacht für einen Meilenstein, und anhand der kann man sehen, wie weit man vorangeschritten ist. Das wird in Teilen mit JIRA gemacht, in Teilen aber auch mit 'MS Project', also das ist ein bisschen aufgesplittet leider. Es gibt sehr viele Tools die benutzt werden, und JIRA ist eines davon. Es gibt auch noch ein paar Excel-Listen und ähnliche Geschichten. Also es geht nich jeder in JIRA rein und kuckt konkret da rein. 00:03:04-1

I: Und was sind die Tools die du immer verwendest dazu? 00:03:06-5

B: Also als Tools verwende ich hauptsächlich halt Microsoft-Produkte, also die ganze Palette: Word, Excel und Project. Dann JIRA, dann das **VersicherungsGmbH** Ticketsystem (Anm.: TTS)... Das war es so grob. Lotus Notes, genau. 00:03:31-2

I: Welche Leute, mit welchen Leuten musst du jeden Tag sprechen, sprichst du jeden Tag? Oder kommunizierst du in irgendeiner Form? ..Oder hauptsächlich? 00:03:46-7

B: In Form von geregelten Berichten? Oder in Form von "mit wem sprichst du jeden Tag weil es irgendwie nötig ist? 00:03:53-1

I: Also am besten.. auf allen Ebenen. Mit deinen Entwicklern, mit den **Österreichern**, mit Testabteilung, mit Fachkonzeption, Abteilungsleitung.. 00:04:06-7

B: Also es gibt Leute mit denen ich jeden Tag spreche: das ist mein eigener Vorgesetzter - mein Abteilungsleiter, das ist.. da gibt es auch feste Termine, aber täglich auf jeden Fall ein loses Gespräch einmal Morgens was man für Probleme hat und wo man gerade steht - halt so einen allgemeinen Stand abgeben, ganz aktuell. Das ganze gibt es dann auch geregelterweise noch einmal wöchentlich und einmal zwei-wöchentlich, je nachdem ob das nun Personalthemen sind oder rein fachliche Themen. 00:04:41-0

B: Dann spreche ich täglich mit jedem Teammitglied, also mindestens einmal um einfach jeden Tag einen Stand abzuholen - was plagt denjenigen, wo ist er gerade dran - um da einen Überblick zu erhalten. Das ist auch ein Teil der Fortschrittskontrolle - das man sieht woran arbeitet derjenige, und wie kann man ihm weiterhelfen eventuell. Dann bei Bedarf, also das ist beileibe nicht täglich, auch mit der Qualitätssicherung. Also das ist.. da haben wir einmal pro Woche einen regelmässigen Termin, aber täglich halt nur wenn wirklich etwas ansteht. Und.. 00:05:24-1

I: Wer? Du und **TST_TL**? 00:05:25-8

B: **TST_TL**, oder sein Vertreter wenn er nicht da ist. 00:05:28-2

B: Und dann spreche ich mindestens einmal die Woche mit **ENTW_OE_TL** in **Österreich**, und bei Bedarf halt täglich mit.. letztendlich betrachte ich die Kollegen in **Österreich** ja wie eigene Kollegen. Man kann zwar nicht so gut vorbeigehen und jeden Tag mit jedem reden, aber wenn es Not tut wird da auf jeden Fall immer angerufen und da auch das Gespräch gesucht. [00:05:53-5](#)

I: Ok. Und wie bleibst du jetzt auf dem Laufenden über Tickets und solche Sachen? Oder wie ist da jetzt zum Beispiel, wenn man es vergleicht mit den Entwicklern die hier in **Norddeutschland** sind, eben da kannst du jeden Tag vorbeigehen. Bei den **Österreichern** hast du ja nicht so diese Möglichkeit. Wie weisst du da.. oder wie genau musst du das wissen, an welchem Ticket sie zum Beispiel arbeiten, oder an welchem JIRA-Task? [00:06:20-5](#)

B: Also das erste was man natürlich hat - für jeden Mitarbeiter, egal wo er sitzt - man kann sehen welche Tasks liegen noch bei ihm auf dem Tisch, und bis wann sind sie zu erledigen. [00:06:32-8](#)

B: Tickets werden ja nicht über Tasks abgehandelt, die werden ja über das TTS einfach verteilt, da gibt es dann nur Mengen. Also "bei Entwickler X in **Österreich** liegen fünf Tickets", oder sowas. Und den groben Überblick mache ich mir jeden Tag einmal mit einer Ticketstatistik: dabei ziehe ich die Tickets aus dem Ticketsystem ab die bei uns liegen, und schmeisse die in eine Excel-Tabelle rein und lasse mir über ein paar Pivot-Charts auswerten "wo liegen welche Tickets", "wie lange sind die nicht angefasst worden", "wie lange hat sich der Status nicht geändert" und so. Und da kriegt man den ersten Aufzeiger mit... wenn ein Kollege, was weiss ich, fünfzig Tickets liegen hat und die haben sich im Status seit einer Woche nicht geändert, dass man mal raufkuckt und sagt "was machst du, woran arbeitest du". WENN man das nicht sowieso weiss. [00:07:18-0](#)

B: Es kann natürlich auch sein, dass der Kollege die nur einfach sammelt und liegenlässt weil er gerade etwas anderes wichtiges zu tun hat. Also es kann durchaus sein, dass er eh an einer Neuentwicklung sitzt. Aber das ist immer der erste Hinweis, also wenn sich bei jemandem Ticket ansammeln, anhäufen, dass man da dann persönlich auch wieder nachfragt. [00:07:37-6](#)

I: Wie wird das... weil du eben gesagt hast.. Wenn jemand etwas liegen lässt, weil er etwas anderes macht, dass also die Prioritäten umgeändert werden - wie wird das kommuniziert? Melden sich da die Leute bei dir und sagen "du, ich mache jetzt etwas anderes", oder...? [00:07:55-1](#)

B: In erster Linie im Normalfall immer über den - also ich nenne es immer über den Teamleiter **Österreich**, über **ENTW_OE_TL** - kommuniziert. Erstmal gehen wir mal durch was die Leute machen, also das machen wir ja mindestens einmal in der Woche, teilweise auch häufiger. Dann priorisiere ich das durch mit ihm, also das geben wir dann vor und sagen "also bitte zuerst DEN Teil der Tickets, dann meinetwegen die Neuentwicklung, und dann (unv. irgendwas)".

B: Und Rückmeldungen laufen genauso. Also wenn irgendwelche Probleme gerade auftreten, dann ist auch da der Ansprechpartner **ENTW_OE_TL** der dann zurückruft und sagt "hier, an der und der Ecke kommen wir nicht weiter", oder, was weiss ich, "wir müssen da warten weil das Fachkonzept nicht ist..", oder der Entwickler ist krank, oder.. was auch immer da für Probleme auftreten können. Das läuft aber alles über diesen einen Kanal. Das ist auch deutlich hilfreicher, als das mit jedem Kollegen einzeln zu machen. Also das wäre sehr mühselig mit jedem einzeln das durchzusprechen. Dass... man dann braucht schon in dem virtuellen Team - oder wie man es immer nennen will - braucht man schon einen konzentrierten Ansprechpartner. Also weil man zu dem auch ein ganz anderes Vertrauensverhältnis aufbauen kann wie zu allen. Das ist ganz wichtig. Also man kann demjenigen viel einfacher Dinge verständlich und deutlich machen, wie es bei jedem einzelnen wäre. Weil bei jedem Einzelnen habe ich eine andere Art wie der angesprochen werden will. Muss.. wie auch immer. Also das ist dann schon deutlich simpler. (...) Nichts desto trotz muss man sich ab und zu auch da unten blicken lassen, um nicht ganz das Gesicht irgendwie zu verlieren. Und man muss ja präsent sein. [00:09:42-2](#)

I: Und wie genau musst du jetzt zum Beispiel Bescheid wissen über - also jetzt nicht nur über **Österreich**, sondern auch über die Entwickler hier - über.. eben deren Probleme oder woran sie arbeiten? [00:09:53-1](#)

B: Also sobald diese Probleme ein grösseres Ausmaß annehmen. Also sobald die Probleme dazu führen, dass sie nicht mehr weiterarbeiten können, dass sie nur noch mit doppeltem Zeitaufwand oder soetwas.. Also natürlich ist nicht jedes kleine Problem, soetwas wie "meine Umgebung baut gerade nicht" oder sonst irgendwas, das wäre ein bisschen banal. Aber da ist ein bisschen Augenmerk oder so auch gefragt. Das kann man dann immer wieder auch (unv. offen kommunizieren), sagen "ja dann lös das Problem einfach, nerv mit damit nicht" - in Anführungsstrichen - sondern, "da mach das einfach, frag nicht dauernd rum und lös es". Aber sobald die halt nicht mehr weiterkommen.. Das halte ich immer so, sobald ein 'Showstopper' in der Gegend herumsteht, muss man Bescheid wissen. Alles andere kriegt man auch in regelmässigen Meetings raus. Dass man so sagt dann.. "das kann auch zwei Tage später passieren", dass man sagt "hier, der Gang hat ein bisschen länger gebraucht, weil wir haben einen neuen Mitarbeiter", oder "der Rechner ist ausgefallen" oder ich weiss nicht was. Das ist aber

der... man kann da halt kein festes Maß nehmen. Man kann nicht sagen "immer wenn 'Schlimmigkeit sieben' eintritt oder so, das geht nicht. Das muss man irgendwie mit Augenmaß machen. [00:11:26-3](#)

B: Was die neuen Themen angeht, da ist man.. es kommt ein bisschen darauf an, was jetzt neue Themen sind. Es gibt natürlich auch neue Themen in Form von Kleinigkeiten - sage ich mal - so Änderungsanforderungen, kleine Changerequests und sowas, die kommen von heute auf morgen. Das ist halt so, die schneien rein, die kommen vom Kunden irgendwo, werden bei uns abgeladen, und wenn es ganz dringend ist kriege ich das Thema heute auf den Tisch und morgen muss es gemacht werden. [00:11:53-9](#)

B: Wenn es um die ganz normalen grossen Meilensteinthemen geht - oder grössere Meilensteinthemen - dann kriege ich zehn Wochen bevor wir mit der Implementierung ungefähr den Umfang... was da kommt, also das ist eine reine Themensammlung eigentlich. Die landen in den nächsten zehn Wochen in Fachkonzepten, da weiss man aber schon wovon man redet dann. Die Fachkonzepte kommen zur QS bei uns, einer der Teammitglieder QSt die. Ich lese fast jedes Fachkonzept einmal, aber ich QSe sie nicht. Ich lese meistens die erste Version, das ist manchmal schlecht, weil die erste Version häufig noch unvollständig ist. Aber ich mache das auch damit ich im Blick habe welche ich gelesen habe und welche nicht, deswegen mache ich es mir eigentlich zur Aufgabe immer die erste Version zu lesen um einfach einen Überblick über die Themen zu halten. Und selber eine Abschätzung treffen zu können "wie lange wird denn das ungefähr dauern, das Thema". Also ganz Abseits von irgendeiner Schätzung die dann auch von dem Entwickler noch kommt. [00:13:00-0](#)

I: Und über so.. überhaupt zukünftige Themen die jetzt vielleicht noch gar nicht in Fachkonzept sind, sondern.. ich weiss nicht, sei es dass eine neue Sparte oder so kommt. Wie.. oder spielt das für dich eine Rolle dass du das schon früher weisst, damit du das Team darauf einstellen kannst, oder.. [00:13:23-1](#)

B: Also es spielt schon eine Rolle, weil teilweise auch langfristige Planungen dabei sind, weil man... Gerade im Zusammenspiel **Norddeutschland** und **Österreich** hat man langfristige Planungen - so ein Vertrag geht ja nicht über zwei Monate die man mit der Firma da abschliesst, sondern vermutlich eher über ein halbes Jahr, oder ein Jahr, oder was auch immer. Und dafür spielt es natürlich schon eine Rolle. Und wir machen alle viertel Jahr gibt es dann.. oder nein, alle viertel Jahr nicht. Nach jedem Meilenstein gibt es eine Grobplanung in Form einer Roadmap, wo es mit den Themengebieten, in dem Bereich in dem das Team arbeitet, hingeht. Also das ist eine Veranstaltung von Produktmanagement, wo einfach nur vorgestellt wird "wie sieht das im nächsten Jahr aus", so mittelfristige Planung. Dass man sagen kann "ok, das wird knapp", oder "es wird nicht knapp", "wir haben zu viele / zu wenige Ressourcen", wie auch immer. Das ist allerdings eine ganz grobe Nummer. Also, wo man wirklich... [00:14:17-9](#)

I: Aber da wird von Produktmanagement quasi eine Besprechung angesetzt? [00:14:20-9](#)

B: Genau, genau. Und die führt dann dazu, dass wir ein erstes Gefühl dafür kriegen was kommt, und dass man auch ungefähr eine grobe Ressourcenplanung machen kann. Dass man sagen kann "ok, man braucht jetzt aber zehn Entwickler statt fünf" oder meinetwegen auch nur drei statt hundert oder so. Man sieht so ungefähr was kommt, kann das auf einen Entwickler genau nicht abschätzen, aber man kann schon eine Grössenordnung reinlegen. [00:14:46-4](#)

I: Musst du über, ich weiss nicht, bestimmte Artefakte, also CruiseControl oder irgendwelche Codeänderungen spielen für dich wahrscheinlich nicht so eine Rolle, oder? [00:15:05-6](#)

B: Also das ist unterschiedlich hier bei uns.. im ganzen Unternehmen. Also jedes Team hat natürlich seinen Code, seine Änderungen, die Buildfähigkeit, und was weiss ich nicht alles selbst in der Hand. Das ist bei uns im Team aber aufgeteilt. Bei uns kümmert sich um alles was Code und Codeänderungen und CruiseControl und ähnliches angeht, kümmern sich zwei andere Mitarbeiter darum. Die nehmen mir dann das ab, und.. [00:15:32-6](#)

I: **ENTW_DE_TL_STV_2** und **ENTW_DE_TL_STV_1**? [00:15:35-0](#)

B: Genau. **ENTW_DE_TL_STV_2** und **ENTW_DE_TL_STV_1** machen das bei uns. Und die kümmern sich um die Ecke. Und das ist auch eine Aufgabe die ich zur Vollständigkeit delegiert habe. Also da... das einzige was ich das noch tue, ist raufzucken ob das... im Groben... ein stichprobenartiges Reinschauen ist das, mehr ist das nicht. [00:15:56-0](#)

I: Auf dem CruiseControl. [00:15:56-6](#)

B: Auf dem CruiseControl, oder auch teilweise.. ob es Codereviews oder so sind, da kucke ich mir auch mal das ein oder andere an. Aber das ist minimal. Also da kann man jetzt keinen repräsentativen Schnitt daraus ziehen. [00:16:11-2](#)

I: Dann eine Frage zu deiner Sametime-Kontaktliste: wie hast du die organisiert? [00:16:19-0](#)

B: Meine Sametime-Kontaktliste ist organisiert nach Teams. Es gibt ja den Email-Verteiler bei uns im Haus, die Mailverteiler kann man in Sametime direkt reinspielen, sodass in Sametime unterhalb des Verteilers immer die angezeigt werden die irgendwie eingeloggt.. also die zumindest angemeldet sind am System, mit dem jeweiligen Status. Da habe ich die wichtigsten.. alle Entwicklerteams, mein Fachkonzepterteam, das Testteam. 00:16:47-5

B: Aufgeklappt ist bei mir eigentlich immer nur unser Entwicklerteam. Und dann haben wir noch Verteiler für Teamleiter, Abteilungsleiter - die beiden sind auch immer aufgeklappt. Dass ich dann Teamleiter, Abteilungsleiter und mein eigenes Team sehe. 00:16:59-8

B: Die anderen klappe ich bei Bedarf dazu auf. Und ganz unten habe ich halt einmal **VersicherungsGmbH** komplett drin, der ist fast immer zu. Aber wenn ich irgendjemanden suche der in irgendeinem mir exotischen Team arbeitet, kriege ich den dann auch. Also die ist recht dünn. Also da.. (unv. ich fahre eigentlich ...) offline Mitarbeiter. Also durch dieses Verteilersystem, wenn man einzelne Mitarbeiter hinzufügt, dann hat man sie auch drin wenn sie offline sind. Hatte ich auch mal ganz früher, finde ich massiv unübersichtlich. Weil dann die ganze Liste mit irgendwelchen Leuten voll ist, die... wenn man sie nicht rausnimmt, die es teilweise schon gar nicht mehr gibt im Unternehmen. 00:17:41-9

I: Wie oft kuckst du ins Ticketsystem? 00:17:49-5

B: Was zählt denn als kucken? Wenn ich ein Ticket nachkucke wenn mir irgendeiner eine Nummer zuruft und sagt dass ist.. 00:17:55-2

I: Nein, aber wie oft siehst du nach, zum Beispiel wieviele Tickets zum Beispiel bei einem Entwickler liegen oder bei einem anderen Entwickler liegen? 00:18:01-5

B: Das ist relativ einfach. Das ist ungefähr zweimal am Tag. Also es sei denn es gibt besondere Anlässe. Aber es gibt zweimal am Tag, einmal Morgens, da mache ich eine Ticketliste fertig, da ziehe ich alles komplett ab und kucke da rein und ich mache.. Normalerweise, also wenn ich es nicht vergesse oder keine Zeit habe, mache ich Abends noch so eine Gegen-Runde, nämlich ich kucke mir an, ob das was wichtig ist - nächste Auslieferung, nächster Patch oder was weiss ich was - was da noch offen ist, damit ich es mir für nächsten Tag auf den Zettel nehmen kann. Also eigentlich ist es zweimal am Tag, der Rest ist halt auf Zuruf so "kuck dir doch das 4711 an" oder "warum liegt das eine Ding da so lange herum". 00:18:42-3

I: Und schaust du dir dann zum Beispiel nur die Tickets von KMP an, oder schaust du dir an welche Tickest, oder wieviele Tickets bei anderen Teams herumliegen? 00:18:46-9

B: Also ich kucke mit eigentlich zu 95 Prozent nur KMP an. Was ich mir noch ankucke, ab und zu mal, alle paar Tage, einmal die Woche vielleicht, ist.. sind die KKZ Tickets, weil das in der Tendenz auch unsere sind. Aber Leben-Tickets interessieren mich nicht, und Provision.. alles andere interessiert mich schlichtweg nicht. Ausnahmen bestätigen die Regel. Also wenn ich jetzt gezielt auf eines raufkucke, das ist etwas anderes. Wenn mir einer sagt "ich stecke Architektur ein Ticket zu, zu diesem Problem" dann kucke ich genau auf das drauf. Aber die Gesamtsumme oder Umfang oder bei wem da welche liegen ist mir.. finde ich auch nicht wichtig. 00:19:38-1

(JIRA-Login) 00:20:53-5

I: Ok, so. 00:20:54-4

B: Ok. Das ist jetzt ein bisschen ungewöhnlich auf dem kleinen Monitor, aber... 00:20:59-4

B: Es gibt für mich zwei wichtige Dashboards, die sehen fast gleich aus. Einmal der (klickt), der das ist der für die Mitarbeiter gemachten. 00:21:07-8

I: Das hast eh du selbst eingerichtet, oder? Konfiguriert. 00:21:11-5

B: Genau, die habe ich für das Team mitkonfiguriert. Da ist eigentlich.. Dieses Dashboard funktioniert hauptsächlich so, dass der aktuelle eingeloggte Benutzer betrachtet wird. Und da steht halt drin, was für Aufgaben er komplett halt noch irgendwie liegen hat. Das sieht bei jedem dann anders aus. Bei mir sieht es momentan so aus. 00:21:29-3

B: Das für mich wichtigere ist das hier (klickt). Das sieht man jetzt hier ein bisschen schlecht, weil zu klein, aber.. Auf einem 24-Zöller sieht das vernünftig aus. 00:21:49-6

B: Da sind im Prinzip alle Aufgaben drin, die mein Team betreffen. Die Workflow-Schritte haben die mein Team betreffen. Es gibt Workflow... einzelne Aufgaben bilden einen Workflow ab, zum Beispiel Implementierungs-

Aufgaben haben den Workflow, dass sie zuerst beim Entwickler liegen, der kann sie 'in Arbeit' nehmen oder 'warten' lassen, dann nachher später, oder sie 'fertig' melden. Sobald er sie 'fertig' meldet wandern sie per Zuweisung an jemanden von Qualitätssicherung, und derjenige muss einen Testnachweis dranhängen. Und die Aufgabe kriegt den Status 'in QS' oder sowas. Das ist dann kein Workflow-Schritt mehr der bei mir liegen würde. Die sind ausgeschlossen durch die Filter. Also hier sind im Prinzip alle Aufgaben drin abgebildet die bei den Mitarbeitern liegen, und auch den Status 'geschlossen' sind hier nicht mehr drin. Man sieht hier den dazugehörigen Filter, das ist der Bursche hier (zeigt). Da kann man halt jederzeit sehen, wer, was, wo dran ist. Der hat hier noch viele Implementierungs-Aufgaben für den Meilenstein liegen, das sind... diese drei vor allen Dingen und noch ein paar andere. Man kann sehen, wieviele Aufgaben die Leute insgesamt haben. Man kann sehen.. also... (...) Und man kann halt auch sehen, welche Aufgaben dran hängen. Was den Vorteil hat, wenn man sagt "ok, ich will jetzt noch wissen was für Fachkonzept-QSen sind denn noch offen" das sind dann die Aufgaben hier. Man kann hier unten auf die Zahl drücken (klickt) und sich ankucken "ok, diese Leute haben diese Fachkonzept-QSen noch liegen, die sollten sich gefälligst mal drum kümmern". Und im Zweifelsfall muss man da mal hinterher laufen. [00:23:38-9](#)

B: (klickt) Und das ist mein Hauptfilter, das sind noch die einzelnen dazu. Hier sieht man das im einzelnen auch noch heruntergebrochen, mit den einzelnen Workflow-Schritten in denen sich das gerade befindet momentan. Wenn man sieht, meinetwegen hier bei den Implementierungs-Aufgaben sind so-und-so viele offen, die sind momentan gerade 'in Arbeit'... Das kann man sich anschauen, da kann man auch Statistiken drüber ziehen. Und was-weiss-ich nicht alles. Man sieht einen Gesamt-Meilenstein - "wie sieht das aus?", "wieviel habe ich schon erledigt?", "wieviel ist offen?". Das ist so meine... Hauptinformationsquelle. Um wirklich an Informationen heranzukommen, jetzt zu sagen "wie weit sind wir denn", oder jetzt "wo hakt es". [00:24:29-3](#)

B: Wenn ich andere Sachen brauche, dann baue ich mir die per Filter selber zusammen. Ich habe irgendwie (klickt) (...) wo sieht man das am besten (klickt). Ich habe gefühlte vier Millionen Filter hier, die halt alle irgendetwas Spezielles suchen. Also es gibt ein paar die noch aus alten - wir haben da noch ein altes JIRA-System was vor einiger Zeit.. ein altes Projekt - da habe ich mir auch noch ein paar alte gehalten, wo einfach diese Hauptaufgaben die wir mal hatten noch drinliegen. Dass ich einfach auf das alte JIRA noch immer mal zurückkomme. Und für das neue habe ich halt jede Menge einzelne, kleine Filterchens. (...) (klickt) [00:25:09-0](#)

B: Das sind alle Komposit-Team die jemals irgendwo herumgeschwirrt haben. Also das Hauptthema dazu. Wenn man dann reinkuckt.. Das ist immer wichtig für mich, wenn ich irgendwie Fragen beantworten muss, oder sonst irgendwas. "Wir hatten doch mal das-und-das", "wir hatten das implementiert" oder ich weiss nicht was noch. Das sind unzählige Beispiele (klickt). Wenn man hier mal reinkuckt "ok, da hat sich damals **FK_1** darum gekümmert". Das ist dann wichtig für mich, wenn ich überhaupt mal wieder einen Ansprechpartner suche. Also.. wer arbeitet da dran, oder wer hat denn da daran gearbeitet. Implementiert haben das scheinbar irgendwie hier.. wenn das **ENTW_DE_TL_STV_2** getestet oder **TST_1**.. das sind für mich dann schon wichtige Ansprechpartner, wo ich dann hingehen kann und sagen kann "ok Leute, hier 'Hausrat-Storno' gibt es ein Problem was du implementiert hast oder getestet - was ist denn das da an der Ecke". Deswegen sind diese ganzen.. auch alte Infos von längst erledigten Aufgaben ganz ganz wichtig.. häufig. [00:26:11-0](#)

B: Ja, und im Normalfall, die einzelnen Unteraufgaben haben halt Termine, und nach diesen Terminen kann man auch irgendwie dann filtern. Was heisst irgendwie... Kann man auch filtern, und dann kann man sich überfällige Aufgaben und ähnliches rauskriegen. Also das ist... für mich ist... [00:26:23-4](#)

B: Also was man über JIRA noch machen könnte, wäre eine Zeiterfassung. Das machen wir leider nicht, das würde ich mir ein bisschen wünschen. Dass man wirklich sagt, dass wenn ich eine Aufgabe auf 'in Arbeit' setze dann arbeite ich da dran. Und solange wie ich das Ding auf 'in Arbeit' ist, ist meine Arbeit praktisch darauf gebucht. Also das kannst du nicht so einfach machen, aber es würde schon... man kann ja über 'Arbeit protokollieren' dann seine.. 'was habe ich getan'... und... man kann einfach hinterlassen 'ok, wieviel Arbeit habe ich da reingesteckt'. [00:27:19-5](#)

I: Ok, das kann man auch mit (unv. ...) machen... Also das zählt nicht automatisch die Zeit mit, sonder man sagt dann einfach 'ich habe jetzt drei Stunden daran gearbeitet' [00:27:27-4](#)

B: Genau. "Ich habe jetzt vier Stunden daran gearbeitet, und ich habe, was weiss ich, hier unten (klickt) den Dialog XY implementiert". Dann "zack", "protokollieren", fertig. Man würde nachher eine Chance haben, von "was habe ich da mal geschätzt" und "was habe ich wirklich gebraucht" zu laufen. Und da haben wir ein ziemliches Problem mit. Also wir haben... wir schätzen im Plan immer richtig gut, und wir haben.. unser Controlling ist schwach. Also momentan ist das Controlling bei uns eher... Eigentlich schätzen wir immer so, dass es in den Meilenstein reinpasst, und arbeiten dann so, dass die mal irgendwann vorhandene Schätzung auch irgendwie passt. Und das ist eigentlich eher falsch. Also in meiner Welt eigentlich müsste man so arbeiten, dass man sagt "ich mache eine Schätzung, darauf mache ich eine Planung, und diese Planung passe ich an wenn es notwendig ist". Und mache da nicht durch Überstunden und sonstirgendwas stellt... diesen Zustand, dass ich dann am Ende der Planung auch wirklich fertig bin. Dann... welche Planung ist denn so gut, dass man... wenn man am 31.12.

fertig sein will - nach der Planung - dass man da auch wirklich fertig ist? Das ist eigentlich nie so. Und das will man mit solchen Sachen zumindest vorantreiben können. [00:28:41-9](#)

I: Wie akkurat sind jetzt diese Schätzungen die da eingetragen werden? Oder auch das Setzen von den Terminen, die werden ja auch ab und zu angepasst, oder? [00:28:50-2](#)

B: Ja, da werden Termine dann angepasst, da werden Schätzungen drüber.. also... Das System ist ein bisschen komplex momentan. Am Anfang des Meilensteines hat man einen Sack Aufgaben, also das sind... [00:29:05-0](#)

I: Also zum Beispiel hier die ganzen Tasks (zeigt auf den Bildschirm)? [00:29:06-4](#)

B: Nein, das fängt erst mal auf der Ebene so an. Man kriegt also ins Team rein "implementiere mal Storno". So, und das wird von uns dann ja nochmal rausgebrochen, und dann werden da mehrere Sachen daraus gemacht. Das ist eine alte Aufgabe die übernommen ist, deswegen sieht man das bei der nicht so richtig schick (zeigt auf den Bildschirm). [00:29:26-0](#)

B: (klickt) Bei der dürfte man das besser sehen. Und es werden unterschiedliche Aufgaben daraus gemacht. Es werden Implementierungsaufgaben.. [00:29:35-4](#)

I: Ein Teil von denen wird ja automatisch angelegt, oder? [00:29:37-3](#)

B: Genau. Es wird von jedem Typ eine automatisch angelegt. [00:29:41-4](#)

I: Aber die Implementierungstasks werden dann zusätzlich noch angelegt. [00:29:44-7](#)

B: Genau, die legen wir zusätzlich an, und zwar in der Reihenfolge, dass wir sagen "was müssen wir überhaupt für Tätigkeiten machen?". Das ganze schätzen wir, diese Schätzung geht erst mal... Also: was für Tätigkeiten müssen wir machen wird dann erst mal hier angelegt. Die Schätzung wird auch erstmal hier angelegt. Dann ziehe ich das ganze rüber zu.. über einen ganz normalen Datentransfer in MS Project. Project macht mir dann daraus eine Planung mit GANTT-Diagrammen und ähnlichem, und da purzeln Fertigstellungsdatümer heraus. Und die ziehe ich wieder rüber, und nehme die als Ende-Datum für die Tasks. [00:30:26-4](#)

I: Also das ist dann schon auch gleich priorisiert, dass man sagt "dieser Implementierungstask muss als erstes gemacht werden, bis zu dem-und-dem Zeitpunkt, danach soll das gemacht werden bis zu dem-und-dem Zeitpunkt" [00:30:38-7](#)

B: Genau. Die Abhängigkeit die man dann in Project erzeugt - und da erzeugt man ja sowas, dass man sagt "16 ist der Vorgänger von 17" - die gehen natürlich alle wieder verloren beim rückwärts (unv. rauchen?!). Also wenn man jetzt wieder rückwärts dann in JIRA reingeht, und hier.. man überträgt nach JIRA nur noch Fertigstellungsdatümer. Und die verlieren natürlich den Zusammenhang. [00:31:00-0](#)

I: Gibt es da keinen Filter oder sowas, wo man so ein Diagramm oder soetwas darstellen lassen kann? [00:31:04-4](#)

B: Nein, du kriegst diese Aufgaben nicht von der abhängig gemacht. Man kann zwar das ganze Ding verknüpfen, nur diese Verknüpfung hat keine Auswirkung. Also die Verknüpfung - da gibt es verschiedene, 'ist Voraussetzung für' das wäre ja schon das Richtige für, 'ist abhängig von', und so weiter und so weiter - die hat aber nachher keinen Wert mehr, weil selbst wenn eine Aufgabe die nicht fertig ist eine Voraussetzung ist für eine andere Aufgabe, kann diese zuerst abgeschlossen werden. Da hindert JIRA einen nicht dran. Ich weiss nicht, in wiefern das alles per Hand konfiguriert ist. Ich kenne JIRA nur so wie wir das benutzen, und das ist schon extrem konfiguriert. Also da haben irgendwelche Leute bei uns auch schon viele, viele Dinge... [00:31:51-1](#)

I: Diese Prozesse die da abgebildet werden? [00:31:52-8](#)

B: Genau. Also wir haben (tippt)... Schulung... Das sind so alte Schulungsaufgaben. Und wenn ich mir den mal rüberhole, da kann man halt sehen, was man mit so einem Ding machen kann. Da kann ich 'in Arbeit nehmen' und 'warten', und muss dann das auch noch kommentieren. Das macht alles Sinn und Zweck, weil jedesmal wenn ich, oder wer auch immer dann ein Stück weiter weg sitzt, da rein kuckt und das Ding steht auf 'warten', dann weiss ich warum. Ich kann da reinkucken, und dann steht halt drin 'weil Entwickler krank' oder ich weiss nicht was. [00:32:39-2](#)

B: Ich kann die Arbeit wieder fortsetzen und so weiter und so fort. Und je nachdem was für einen Aufgabentyp man hat, hat man halt unterschiedliche... Also der Workflow ist nur sichtbar für den, dem das Ticket gehört. [00:33:02-0](#)

B: Ich habe (unv. ...) unterschiedliche Sachen. Ich kann es zur QS freigeben, dann müsste ich hier sagen "ich habe einen Testnachweis dran", der ist verlinkt - wir können ja hier auch auf eine andere Aufgabe verlinken oder so - "nicht erforderlich weil" - da gibt es immer einen guten Grund - ich kann sagen "mach bitte eine QS bis", ich kann hier ein Datum eintragen, und ich kann sagen wer es kriegen soll. Also, was weiss ich, "TST_TL" gibt es nicht... [00:33:41-8](#)

B: Also ich kann es dann einem zuweisen und sagen "bitte QS" - also hier muss ich nicht kommentieren - und dann kann ich es zur QS freigeben. (...) Mache ich jetzt nicht, sonst bekomme ich gleich eine Email. "Testnachweis ist required"... Also es sind dann auch Plausis (Anm.: "Plausibilitätsprüfungen") drauf, also hängt überhaupt ein Testnachweis dran, oder habe ich angehakt "ist nicht erforderlich" und das dann gleichzeitig kommentiert und sowas. Dann wird es jetzt bei der Qualitätssicherung hängen, und die Qualitätssicherung wird sagen "ok, ich akzeptiere das Teil" oder "ich lasse es nachbessern". "Testnachweis ok" oder ist nicht ok. Und wenn er ok ist, gibst du die Anfrage zurück, sagst "bitte nachbessern bis" - das muss man nicht mal kommentieren - und dann hängt es wieder bei mir. Auf "in Arbeit". [00:34:29-0](#)

I: Diese Zuweisungen, die gehen dann auch immer automatisch zu den Teamleitern? [00:34:32-9](#)

B: Nein, die gehen leider nicht automatisch. Das muss du leider jedesmal mit der Hand machen. Also das ist nicht hinterlegt wer.... [00:34:39-8](#)

I: Aber die.... vom Qualitätsmanagement, die wissen praktisch wer dafür zuständig ist? [00:34:46-1](#)

B: Genau, ja. Das geht immer.. wenn die Qualitätssicherung etwas zurückgibt, gibt sie es immer demjenigen, von dem sie es gekriegt hat. Also wenn Entwickler X irgendwas rübergibt zur QS, und das ist nicht in Ordnung, dann gibt er es auch wieder an X zurück. Wenn es in irgendein Team reingeht aus dem der Task vorher noch nie rausgegangen ist - kann es ja auch geben - also ich gebe es jetzt zum Beispiel zur Qualitätssicherung rüber, das war ja noch nie da, zum ersten Mal, dann geht es immer zum Teamleiter. Das ist relativ simpel dieses System. Es sorgt zwar dafür, dass die Teamleiter immer relativ viel herumliegen haben, aber das ist relativ schnell sortiert. Also da kann man sich Filter setzen "was hängt bei mir noch", und das sortiert man morgens einmal, und dann ist es gut. [00:35:36-4](#)

I: Verwendest du zum Beispiel diese Überwachungsfunktionen oder so? [00:35:38-7](#)

B: Nein. [00:35:39-3](#)

I: RSS-Feeds oder so? [00:35:41-9](#)

B: Das nicht, und das auch nicht. Also ich stimme nicht für irgendetwas, ich überwache auch nichts, weil ich habe da ganz am Anfang mal mit rumgespielt, habe aber festgestellt, dass man eigentlich alles was man interessant findet auch filtern kann. Also wenn ich irgendwelche Dinge als interessant empfinde kann ich mir dafür einen Filter bauen. Den muss ich ja nicht bis in alle Ewigkeiten behalten. Aber ich kann immer einen... ich bin dann explizit auf den Sachen die ich wirklich haben will. Wenn ich irgendwie mit 'Überwachen' arbeiten würde, und da hätte ich jetzt zwanzig, dreissig Aufgaben drin, dann sind die Themenunabhängig. Wenn ich einen Filter habe, dann hätte ich zwar vielleicht fünf verschiedene Filter, aber die sind immer Themenabhängig. Ich möchte normal wissen "was ist denn mit den vier Aufgaben aus dem Bereich 'LEIST-SACH'?". Fand ich praktischer. Weil das ist dann mehr so eine... das ist für mich so ein Schuhkarton wo ich irgendwie Bilder drin sammle.. die so unsortiert in der Gegen rumliegen, die man sich mal wieder anucken KÖNNTE.. vielleicht. Auch total prioritätslos. Da sind dann so 'kucke-ich-mal-irgendwann-wieder-hin-Sachen' bis zu 'oh, muss ich unbedingt draufkucken'. Deswegen fand ich das so... wenig hilfreich. [00:37:03-1](#)

I: Bekommst du irgendwelche automatisch generierten Emails aus JIRA? [00:37:05-6](#)

B: Ja, die fliegen alle in den Müll. [00:37:08-7](#)

I: Die hast du herausgefiltert? [00:37:09-7](#)

B: Ja. Also.. Auch da die Feststellung meinerseits: alles was man wissen muss in JIRA, wenn man es dauerhaft benutzt und vernünftige Filter setzt, dann kann man sich in einem mir selbst gewählten Takt anucken was wichtig ist. [00:37:25-1](#)

B: Welche Aufgaben haben sich geändert kann man mit relativen Datüern herauskriegen. Also wenn ich wissen will was hat sich in meinem Team in den letzten zwei Stunden getan, kann ich jederzeit kucken welche Aufgabe wurde in (klick) den letzten zwei Stunden aktualisiert. Man kann mit relativen Datüern arbeiten in den Abfragen, und das mache ich dann auch. Also sowas will ich nicht wissen im Normalfall, das ist ein Spezialfall wenn ich wissen will "was ist denn heute hier fertiggeworden morgen". Aber wenn man sowas wissen will, kriegt man es

alles über Filter heraus. [00:37:56-8](#)

B: Die Email-Orgie sorgt eigentlich nur dafür, dass man vielleicht selektiv mal etwas findet das interessant ist, aber in den allermeisten Fällen Müll kriegt. Also ich kriege.. ich weiss nicht wieviele JIRA-Emails ich am Tag kriege. Hundert? Zweihundert? Als Teamleiter kriegst du ja alle Emails von deinem Team. Und das hat den ekligen Nachteil, dass du dann in einer Flut ertrinkst, und eigentlich gar nichts mehr liest davon. Ich glaube für den einzelnen Mitarbeiter ist es hilfreicher. Weil wenn ich nur die Email kriege zu Tasks wo ich auch als Bearbeiter oder auch Autor meinetwegen drinstehe, dann macht das vielleicht Sinn. Also das Problem an der Sache ist ja auch noch, bei den allermeisten Tasks in meinem Team bin ich auch noch Autor. Und dann kommt die ganze Flut für den Autor noch mit rein. Also die lohnt sich faktisch nicht, die Emails. [00:38:57-0](#)

I: Und weil ich vorhin gefragt habe, ob du dir Tickets von anderen Teams ansiehst: siehst du dir JIRA-Tasks von anderen Teams an? [00:39:03-5](#)

B: Nur ganz selten. Also auch nur wirklich bei Bedarf. [00:39:09-5](#)

I: Und wann besteht zum Beispiel Bedarf? [00:39:11-2](#)

B: Also Bedarf besteht bei Schnittstellen, oder Schnittmengen, Themen. So etwas wie was weiss ich, DWH, GDV-Geschichten, sowas wie (unv. ...) Leben, macht 'LEBEN', aber sind wir viel mit drin. Bedarf besteht manchmal bei solchen Themen, wo wir Leistungs-Themen für 'LEBEN' abarbeiten - wir haben ja bei uns im Team den kompletten Leistungs-Bereich hängen - und arbeiten da auch 'LEBEN' zu. Und deswegen muss man manchmal in 'LEBEN'-Tasks reinkucken, ob da irgendwelche Themen drin sind, die das Leistungssystem betreffen. Das ist aber dann schon sehr speziell immer. Dass man wirklich da in fremde JIRA-Tasks reinkuckt. Was man macht ist, wenn man (klickt) in so einer Task hier drin ist.... Es gibt ja diese Fachkonzept-Task. Hier hängt das Fachkonzept im Normalfall - hängt da jetzt bei der Schulungs-Tasks wahrscheinlich nicht - aber natürlich kucke ich da rein, obwohl das Ding nie bei mir war. Wenn ich ein Fachkonzept zum Beispiel lesen will. Dann, klar kucke ich da rein. Aber ich betrachte eigentlich - wenn das ein Komposit-Thema wäre - das heisst hier würde 'Komponente: Komposit' drinstehen - dann betrachte ich eigentlich das ganze Thema schon als mein Thema. Auch wenn hier Aufgaben dabei sind die nicht... Also alles was mich betrifft ist gelb irgendwie. Auch wenn Aufgaben dabei sind die nicht bei mir gemacht werden. Aber sonst.. selten. [00:40:57-3](#)

I: Und vergleicht man sich dann zum Beispiel mit anderen Teams oder so? Dass man sagt... [00:41:02-1](#)

B: Vergleichen? [00:41:03-0](#)

I: Ja.. wenn man so.. Ab und zu gibt es so.. wo du hörst bei 'FINANZEN' geht es gerade drunter und drüber oder so. Die haben furchtbar viele Tickets oder so. Wo man sich dann denkt "puh, bei uns momentan Gott sei Dank nicht so stressig", dass man sich irgendwie.. ich weiss nicht.. entspannen kann oder so. [00:41:25-8](#)

B: Also sowas läuft zentraler über Abteilungsleitung. Also Abteilungsleitung macht bei uns - oder Produktionsleitung wie man es auch immer nennen will - macht bei uns auch allgemeine Auswertungen. Also so wie sieht der Tickettrend für alle vier Entwicklerteams, oder auch spartenabhängig - wir haben ja mehrere Sparten hier im Team bei uns - wie entwickelt sich da etwas, oder wo gehen irgendwelche Kurven hin und sowas. Das wird alles über die Abteilungsleitung gemacht. Da gibt es dann einmal in der Woche einen Bericht, den kann man sich ankucken, wenn der gut ist pinnt man ihn an die Wand, wenn er schlecht ist wartet man auf die nächste Woche. Also.. es ist halt immer so. Mal geht es hoch und bei den anderen runter, mal ist es umgekehrt. In der Tendenz ist das... ist es aber nichts was man braucht. In meiner Welt braucht man keinen Vergleich, so "kuck mal, wir haben die da hinten endlich eingeholt" oder "die sind noch viel schlechter" oder so. Das braucht kein Mensch. Also... Wir haben die Arbeit die da ist, die ist irgendwie zu tun, und ob andere genausoviel Arbeit oder weniger oder mehr haben hilft einem im Zweifelsfall nicht weiter. Also das ist jetzt aber etwas philosophisches. Das (unv. ...) andere Leute anders. [00:42:40-2](#)

I: Ja, ich habe das nur in einer Arbeit gelesen, wo so ein ähnliches System wie JIRA untersucht wird, wo der Teamleiter eben dann für die Leute auch so eine Art Filter macht, wo quasi die offenen Tickets von den einzelnen Leuten sind. Und er halt auch sagt, dass er damit quasi so ein bisschen einen Wettbewerb anstartet oder so. [00:42:59-4](#)

B: Das geht mit der eigenen Ticketstatistik. Das machen wir auch, das läuft aber über Excel halt. Weil die aus Notes rauskommen. Das geht aber auch nur teamintern, weil andere Teams kochen anders, haben andere Themen, andere Schwierigkeiten, und, und, und. Teamintern ist es schon so, dass man sagt "Kuck mal hier, XY hat jetzt zehn Tickets, wird mal Zeit dass er davon wieder runterkommt. Und kuck dir mal im Gegenzug Entwickler soundso an, der hat jetzt von seinen zweihundert Stück nur noch fünf liegen. Die hat er jetzt alle innerhalb von ein paar Wochen runtergearbeitet." Also das kann man da schon ganz gut transparent machen, und dadurch auch... Das stimmt, da ist auch so ein Motivationseffekt drin. Dass man sagt "hier kuck mal, das war

doch irgendwie klasse", oder auch mal im Grunde mit jemandem zu reden und zu sagen "also hier das hast du wirklich spitze hingekriegt". Und "das gefällt mir total, wie du in zwei Wochen das und das weggeschafft hast". Das geht, also das kann man machen. Wenn man JIRA als Ticketsystem - also wir benutzen das ja gar nicht als Ticketsystem - nutzt, dann ist wahrscheinlich sogar normal dass man so etwas macht. Wenn man das als Workflow-Unterstützung - was wir ja hier irgendwie so.. wie wir das hier verwenden - da sagt das weniger aus. Natürlich kann man sagen zu einem Zeitpunkt X Meilenstein 'LEBEN' hat fünfzig Prozent seiner Tasks fertig und Komposit nur dreissig. Ob das eine Aussagekraft hat sei dahin gestellt. Weil du nie weisst, wie gross so eine Aufgabe hier ist. Die sind alle gleich gewertet. Wenn du nach Mengen gehst. 00:44:40-3

I: Nach welchem Ding wird das aufgebrochen, diese einzelnen Aufgaben? Oder... denkst du dir manchmal, das könnte man vielleicht noch in weitere Unteraufgaben aufbrechen, oder so irgendwie? 00:44:51-5

B: Du meinst Implementierungsaufgaben? 00:44:53-4

I: Genau, Implementierungsaufgaben, ja. 00:44:55-5

B: Also die werden runtergebrochen.. da haben wir eigentlich INZWISCHEN - also das hatten wir jetzt auch noch nicht so lange - inzwischen einen einigermaßen einen Konsens im Team gefunden. Aufgaben werden immer heruntergebrochen auf welche Ausfertigungsgründe... Also normale Implementierungsaufgaben. Es gibt auch immer so Sachen so wo die Jobs angepasst werden, die laufen ein bisschen anders. Welche "GeVos" sind betroffen, in welchen Ausfertigungsgründen, dafür gibt es dann halt jeweils einen Task. 'Neu' und vier, 'Neu' und sieben, 'Neu' und irgendwas. Die einzelnen Dialoge, in diesem Dialogtask "D001" ist dann wieder vermerkt für welchen GeVo gilt das, dafür gibt es jeweils einen Task, das heisst, irgendjemand kümmert sich darum dass dieser GeVo überhaupt funktioniert, das ist der Task, und irgendjemand kümmert sich darum, dass dieser Baustein in allen diesen GeVos funktioniert. Im Normalfall der gleiche, aber er hat zwei Änderungen daran, du musst dich um 'null' und 'fünf' kümmern, und du musst dich darum kümmern, dass der D001 in ' NEU ' und fünf gar nicht ist, in 'NEU ' und sieben ist er vorhanden, in 'NEU' und acht hat er die und die Besonderheit. Also dass wir das so aufbrechen, und da kommen dann immer noch so ein paar extra Sachen zu, sowas wie... so allgemeine Dinge. Man muss einmal Datenbankanträge stellen, und sowas, aber das ist so die grundsätzliche Struktur wenn es um Dialogentwicklung geht. Ist immer das gleiche. Also welche Ausfertigungsgründe und GeVos haben wir, und welche Dialoge sind betroffen. Dann noch ein, zwei Tasks für Hintergrundverarbeitung, und dann ist es eigentlich auch schon fertig. 00:46:41-5

B: Nur so wird man irgendwann vergleichbar. Wir kommen ja von einer grünen Wiese was das angeht. Wo man früher gar nichts gemacht hat, dann tastet man sich langsam an irgendetwas ran. Also was funktioniert denn... 00:46:57-7

I: Ja früher hat es ja schon so Listen gegeben, oder? Was alles gemacht gehört. 00:47:01-7

B: Ja genau. Es gab dann Checklisten und was weiss ich nicht alles... 00:47:05-1

I: Wo zugeordnet wurde wer was macht. 00:47:07-9

B: Genau. Sowas gab es dann. Das gab es dann in Excel und was weiss ich nicht in was für einer Form alles. Man muss nur irgendwie eine Vergleichbarkeit herstellen, damit ein Entwickler irgendwann auch blind eintragen kann "ok, alles klar hier, der Dialog hier der kostet immer zwei Stunden. Dauert immer zwei Stunden den irgendwo einzubinden." Dass der Entwickler nachher dann meinetwegen vier dafür braucht weil er neu ist, oder nur eine halbe weil er schnell ist, das sei jetzt mal dahingestellt. Aber das muss irgendwann eine Schlagzahl geben, dass man sagt "also das und das, das ist quasi immer gleich viel". 00:47:45-7

I: Und da seid ihr aber jetzt noch nicht ganz, oder wie? 00:47:47-2

B: Nein, da sind wir... 00:47:47-4

I: Aber da wollt ihr hin. 00:47:50-2

B: Da wollen wir hin. Und da müssen die Entwickler aber irgendwie noch ein Gefühl für kriegen "wie sieht denn das aus und was braucht man", damit man.. von unseligen Mondschtätzungen und sonstwas runterkommt. Früher sagen Schätzungen so aus "was machen wir, neue Sparte so-und-so, ok für die Neuanlage der Sparte brauche ich 80 Personentage, oder 100." Und das ist gut, schlecht, wie auch immer, also das ist auf jeden Fall eine Zahl, die ist auf jeden Fall viel zu grob. Weil man A nicht weiss was da alles dahintersteckt, man hat sich da vorher gar keinen Kopf darum gemacht. Und B ist die Aufgabe so auch nicht teilbar. Du kannst sie so nur in einem wegbuchen und sagen "ich geb dir jetzt mal die hundert Personentage". Und du machst.. wenn man das so macht, dass man auf relativ standardisierte Kleinaufgaben zurückgehen, dann kann man auch sagen "du die drei, du die drei, du die drei". Klar muss ein kausaler Zusammenhang wahrscheinlich sein. Und kann dann sagen, "ok,

die drei kosten immer sieben, die drei kosten immer vier, die drei kosten... Ihr seid ausgelastet für die Woche, legt los". "Du brauchst auch die nächste Woche noch und (unv. irgendjemand) kriegt dann was neues". Und das funktioniert schon ganz gut mit einigen älteren Entwicklern, mit den neueren noch nicht so richtig gut. Aber das ist halt eine Gefühlssache dass man da reinkommt. Auf jeden Fall viel besser als irgendeine Mondschatzung mit "Klar, Neuanlage immer hundert Personentage" oder so. [00:49:18-2](#)

I: Und wie gut funktioniert das dann eben.. oft ist es ja so, man kriegt das Konzept, das wird geschätzt oder so, da steht "Job so-und-so", und dann stellt sich aber heraus, dass im Fachkonzept irgendetwas noch nicht ganz fertig spezifiziert ist oder so. Woraus sich dann halt Verzögerungen ergeben. Wie wird das dann gehandelt? [00:49:39-5](#)

B: Momentan wird das so gehandelt, dass dann durch Überstunden die Verzögerung weggedrückt wird. [00:49:42-9](#)

I: (lacht) [00:49:43-8](#)

B: Jaja, ernsthaft. Also bis zu einem gewissen Maß. Irgendwann ist dann der Topf auch zu. Aber das ist das was ich meine mit "die Planung ist immer genau so dass sie eingehalten wird". Im Normalfall wird das dann neu geplant werden. Also das ist das, wozu wir jetzt so langsam übergehen. Und sagen "ok, wir haben eine Planung gemacht, wir kontrollieren das auch, und wir sind da, was weiss ich... Nach 25 Prozent der Zeit haben wir 23,2 Prozent fertiggestellt, also wir sind so ein bisschen hinterher, aber grob im Plan. Und jetzt kommt auf einmal irgendetwas rein. Was weiss ich, 20 Prozent von den 23 Prozent müssen nochmal überarbeitet werden." Weil irgendwas geändert wird. In dem Fall kannst du den ganzen alten Plan natürlich nehmen, wegschieben, sagen "ok, das war mein Plan, ab heute gibt es einen neuen, es wird neu geplant. Der Rückbau des Ganzen, der neue Einbau, und, und, und" Und man stellt dann wieder fest "ok, wir haben geplant, so und so viele Tage, und wir haben jetzt eine Fertigstellung von was weiss ich, meinerwegen nur noch drei Prozent. Da soll es hingehen. Also da sind wir gerade dabei uns da ein bisschen hin zu entwickeln. Also ein bisschen die Scheu davor abzulegen zu sagen "das schaffen wir nicht". [00:50:54-6](#)

I: Aber wird das dann zum Beispiel im JIRA festgehalten, oder wird dann neu geschätzt dort? [00:51:00-5](#)

B: Da wird neu geschätzt, und die JIRA-Aufgaben werden alle angepasst. Das ist ein bisschen mühselig. [00:51:05-0](#)

I: Gibt es dann auch ein Kommentar dazu, dass man sagt "wegen dem und dem Ding" [00:51:08-6](#)

B: Genau. Also die werden dann wieder... [00:51:09-6](#)

I: (unterbricht) ... wird die Schätzung... ich weiss nicht, wird die Dauer verlängert.. [00:51:13-8](#)

B: Es können ja ganz unterschiedliche Sachen passieren. Es kann durchaus sein, dass man eine Aufgabe wegschmeisst, also es gibt ja die Möglichkeit.. einige wenige Leute können ja löschen. Sollte man natürlich meistens nicht tun, oder ungern tun, aber du kannst eine Aufgabe auch einfach schliessen, also einmal den Workflow durchklicken mit einem entsprechenden Kommentar "nicht mehr benötigt nach Konzeptanpassung Version 4". Und es müssen.. muss ein ganzes Rudel wahrscheinlich neuer Aufgaben angepasst werden. Es werden.. was wir nicht machen ist sowas hier nochmal aufreißen. Also zumindest in der Implementierung nicht. Wenn eine Aufgabe hier jetzt hier keine (unv. Beierwagen????) dann (unv...) das Ding wieder zu, dann wird die nicht wieder aufgerissen, sondern es wird eine Neue gemacht, die heisst vermutlich genauso, mit 'Anpassung zu Konzeptversion vier', und da geht dann die Schätzung, was braucht man um also das zurückzubauen nochmal und wieder anzubauen. Letztendlich soll das ganze ja auch dokumentieren wieviel Zeit habe ich denn da wirklich reingesteckt. Da wollen wir ja irgendwann mal hin, auch mit diesen... Momentan schätzen wir da nur drauf, aber wir wollen da eigentlich schon dahin, dass wir sagen... das ist ja gut wenn wir schätzen, und das ist ja meinerwegen gut wenn wir mit Überstunden und sonstwas dann wirklich am Ende der Planung die aus der Schätzung kommt auch wirklich da stehen wo wir hin wollen. Aber es ist ja eher schlecht, wenn man dann wirklich mal draufkuckt und sagt "naja, wir haben hundert Tage geschätzt, zweihundert gebraucht, und die Qualität stimmt auch nicht meinerwegen". Ja und das ist eher schlecht. Und das passiert eigentlich wenn man die Planung nicht anpasst. Vermutlich. [00:52:56-2](#)

I: Aber so würde man dann zumindest sehen, ok, warum.. Oder, da hat es jetzt zehn Anpassungen gegeben. [00:53:01-9](#)

B: Genau. Also man würde dann wirklich sehen, also wir haben hier schon meinerwegen fünf Tage reingesteckt, das Ding ist schon zu, jetzt müssen wir hier nochmal eine aufmachen, das kostet nochmal acht Tage, vier Tage zurückbauen und vier Tage etwas neues. Das sind dann nicht mehr fünf Tage für meinerwegen diesen Teil hier, sondern es sind dann schon eben 13. Das wird man dann schon sehen. Und es wird auch einen ganzen Haufen

transparenter dadurch. JIRA ist da der Transparenzgewinn schlechthin, im Vergleich zu vorher. 00:53:27-9

I: Wird das dann auch mit dem Kunden kommuniziert? 00:53:31-3

B: Also mit dem Kunden kommuniziert wird das über.. 00:53:35-8

I: Das Reporting? 00:53:36-8

B: Ja.. Es gibt leider unterschiedliche Wege wie wir mit dem Kunden reden. Also im Normalfall ist bei uns Produktmanagement und Fachkonzeption, aber auch Test und Entwicklung reden mit dem Kunden. Das brechen wir auch momentan ein bisschen auf. Wir wollen hier alles ein bisschen umstrukturieren, von so einer Liniennummer auf eine Projektnummer. Und dann wird das über einen Projektleiter rausgehauen und gesagt "ok, Projekt X verzögert sich um X Tage, ist neu geplant, WEIL". Und dann gibt es ein paar Gründe dafür: kranker Entwickler, Bombeneinschlag, Neuanforderung, ich weiss nicht was. Und dann war es das. Und der Kunde muss es ja nicht fressen, der kann ja durchaus, was weiss ich, notfalls vor Gericht ziehen oder sonst irgendwo. Es geht schlichtweg erstmal darum, da eine seriöse Planung hinzulegen, und zu sagen "mein lieber Kunde, dein neues Produkt, oder was du gerade bei uns bestellt hast, das wird jetzt nicht zum Jahresende fertig, sondern erst ein Jahr später, WEIL". Bestes Beispiel ist hier.. wie heisst das Ding.. Elbphilharmonie? (lacht). Ich weiss gar nicht, wann das fertig sein sollte, und auch da wird ständig neu geplant, und wer auch immer sich da vertan hat.. Ich glaube jetzt zum Beispiel, letzte oder vorletzte Woche war es mal in der Zeitung.. 00:54:51-8

I: Die Kostenaufstellung? Das habe ich auch gesehen. 00:54:52-0

B: Ja, irgendwas mit den Scheiben war nicht in Ordnung glaube ich. Da sind ganz spezielle Scheiben dran, jede Scheibe ist eine Einzelanfertigung. Und irgendwas stimmt damit nicht, irgendein Schallschutz.. sonst irgendwas. Und die müssen jetzt alle Scheiben einmal rausnehmen. Und natürlich verzögert so ein Fehler - wer auch immer ihn gemacht hat - die gesamte Planung und auch die Fertigstellung. Und wenn die Scheiben nicht drin sind kann ich auch keinen Innenausbau machen, weil es regnet ja rein oder so. Das muss man neu aufplanen, und von mir aus auch einen (unv....). Aber wenn man seriös sein will, hat man ja gar keine andere Chance als zu sagen "ok, das und das ist passiert, und das resultiert da draus. Und du kannst da ein Jahr später erst deinen Richard Wagner hören." 00:55:51-8

I: Ok. Dankeschön.