Die approbierte Originalversion dieser Diplom-/Masterarbeit ist in der Hauptbibliothek der Technischen Universität Wien aufgestellt und zugänglich.
http://www.ub.tuwien.ac.at

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology.
http://www.ub.tuwien.ac.at/eng

Diplomarbeit

Free & Open Source Methods in Architecture and How They Impact Designing

ausgeführt zum Zweck der Erlangung des
akademischen Grades einer Diplom-Ingenieurin
unter der Leitung von

Ao. Univ. Prof. Dipl.-Ing. Dr.-Ing. Dörte Kuhlmann
E259 Institut für Architekturwissenschaften
E259/4 Architekturtheorie

eingereicht an der
Technischen Universität Wien
Fakultät für Architektur und Raumplanung
von

Astrid Magauer
e0326859

Wien, 03.11.2015

Free & Open Source Methods in Architecture and How They Impact Designing

Astrid Magauer

Abstract

Der Begriff „Free & Open Source“ (FOS) taucht in letzter Zeit immer öfter im Architekturkontext auf. Über eine konkrete Umsetzbarkeit von Konzepten der "Free & Open"-Bewegung in die Architektur wissen wir jedoch relativ wenig. Anhaltspunkte sind Projekte, die einen solchen Versuch unternommen haben.

Von Studien über Open Design und Open Hardware wissen wir, das speziell die Übertragung dieser Ideen in die physische Welt eigene Herausforderungen in sich birgt. Ich beschäftige mich in meiner Arbeit mit der Übernahme von FOS-Arbeitsweisen und Methoden in architektonische Projekte.

Meine Annahme ist, dass diese Arbeitsweisen und Methoden, wenn sie erfolgreich in die Architektur übertragen werden sollen, auch einen veränderten Zugang und Umgang mit Design erfordern. Diese These stützt sich auf Shirky, demzufolge FOS-Methoden grundsätzlich in jedem Feld, das eine umfassende Digitalisierung erfährt, Fuß fassen können. Jedoch nur dann wenn die „Arbeitsmaterie“ eines Feld an diese Methoden angepasst wird. Ich gehe in meiner Arbeit davon aus, dass die fragliche „Arbeitsmaterie“ als Kombination aus (1) dem Designprozess und (2) dem Design selbst fassen lässt.

Ausgehend von dieser Annahme untersuche ich, wie Design und Designprozess im Kontext von FOS-Methoden zu konzipieren sind, vor allem in Hinblick darauf welchen Herausforderungen sich Projekte zu stellen haben, die ein solches Ziel verfolgen. Freilich interessiert mich auch wie diese Herausforderungen bewältigt werden (können).

Darüber hinaus ist es mir wichtig zu zeigen, dass die Idee einer FOS-Architektur keine keineswegs ein Gedanke ist, der bloß eine Thematik anderer Branchen aufgreift. In diesem Sinne bette ich meine Untersuchungen in den aktuellen Architekturdiskurs ein, indem ich thematische Berührungspunkte des Diskurses über digitaler Architektur und des „Free & Open“-Diskurses aufzeige.

The notion “free & open source” (FOS) appears ever more frequently in architectural contexts. Yet, we know little about how feasible free & open source concepts and methods are in architecture. To change this, we should have a look at projects that have attempted to apply these methods.

From studies of open design and open hardware we know that transferring these ideas to the physical realm bears its challenges. This thesis focuses on the transfer of free & open source methods to architecture.

I assume that these methods, if they are to be transferred to architecture successfully, require changing the way in which we approach design. This assumption is based on Clay Shirky, who argues that free & open source methods can take hold in every domain that undergoes comprehensive digitization. However, this applies only if the work processes and the subject matter of that domain are adapted to these methods. In my thesis, I propose that the subject matters in question are (1) the design process and (2) the design itself.

Based on these considerations, this thesis explores how design and the design process might change when we apply free & open methods, in particular in response to the challenges that projects that want to apply these methods are likely to face. Identifying these challenges is another aim of this thesis.

Moreover, it is important to me to show that the notion of a free & open architecture is more than just a thought that picks up a notion from other fields. Thus, I embed my studies in the current architectural discourse, by pointing out where the discourses on the digital in architecture and the discourses of the free & open movement touch – and even overlap.

Acknowledgments

I would like to thank, first of all, my advisor Dörte Kuhlmann, for her open-mindedness, her valuable guidance and for granting me the freedom to pursue my research my way. Without her, this thesis would not have been possible. I would also like to thank Peter Ferschin, above all, for his enlightening comments, and Christine Hohenbüchler, who was so kind to agree to serve on my thesis defense committee on rather short notice. Furthermore, my sincere thanks go to Alireza Fadai, for being a wonderful and understanding supervisor. My deepest gratitude I owe Odin, Fleg, Elisabeth, Karin and Mikka for their active support, their care, their understanding and their reliable intuition for when help was so desperately needed. Moreover, I would like to thank Julia, Luise and all my flatmates, studio colleagues and wonderful friends, who supported me during the writing of this thesis, as well as Mikka, Flo, Nic and the Vienna Open festival for inspiring discussions on free & open source ideas. I would also like to thank my dear sisters and my parents, for their love and support and for having made possible everything that I do now. Finally, I thank the City of Vienna, for having set up park banks and tables on a street next to my apartment, who knows whether I would ever have finished this thesis, if I could not have written it in the sun?

Table of Contents

0 Introduction

0.1 Context

The book you hold in your hands is the result of an explorative journey into the open. How would architecture change if we adopted free & open source methods? More precisely, how would our designs and our design process change? This is the question to which I would like to contribute with this thesis. However, before I go on, I should probably say something about why architecture may want to adopt free & open source methods in the first place (I will call them "free & open methods" from now on, for brevity). After all, although "free & open source" has become a common enough phrase, attempts to apply its methods to architecture are quite recent and the notion that we may benefit from giving away our work for free, admittedly, may sound radical, not to say, naive. Steven Weber, a professor at the Department of Political Science at Berkeley, remarks that "[s]everal years ago when I began thinking about open source software, I had to convince just about everyone I talked to, outside of a narrow technology community, that this was a real phenomenon and something worth studying in a serious way" (Weber, 2004). I have made the same experience in architecture.

Of course, today we know that open source software works. There are high quality free & open source software products. These products play an important role in the industry (for example, Android, the most widely used operating system on smartphones, is based on the free & open source operating system Linux). And successful business models for free & open source software have emerged. What is more, concepts from the free & open movement gain more and more traction in different fields, there is an open science movement (promoting open access to scientific research), an open data movement (promoting open access to institutional data), an open source hardware movement (which will be discussed later), an open design movement (which wants to adopt free & open methods to design), the fab lab culture (which will also be discussed later) etc. The free & open source idea, as well as the methods that are associated with that idea, are also the subject of many studies, for example, by Günter Grassmuck (2004), Steven Weber (2004), Chris DiBona et al. (2005), Bernd Lutterbeck et al. (2004–2008) or Yochai Benkler (2006). We should, therefore, be careful not to dismiss the thought that free & open methods may benefit architecture out of hand.

Of course, there are important differences between software and architectural designs, the most important and most obvious of which being that, simply put, software only needs to be copied, whereas architectural designs usually need to be built. Hence, we not only need to check whether we find the benefits of free & open source software appealing, but also if they actually could be made to apply for architecture. I will say more about this in the first part of this thesis, for now suffice it to say that, following Clay Shirky (2005), associate professor at the New York University, free & open methods can take hold in any field the working methods of which have been digitized – which no doubt applies to architecture. I would like to put it the other way around, the benefits of digitization are, above all, that designs can be more easily shared, inspected, adapted, and can be utilized to higher extent by using free & open methods, because these methods lower the costs of sharing, learning from, and adapting designs even further.

0.2 Research Objectives and Theses

This thesis wants to contribute to the larger debate on the adoption of free & open methods in architecture by exploring (1) how the practice of architecture may need to change in order to adopt these methods and, conversely, (2) how adopting them may change the practice of architecture.

This exploration is guided by two theses, which correspond to the two objectives above. The theses are based on arguments by Clay Shirky and Mario Carpo, professor for architectural history at the University College London, respectively. Shirky (2005) argues that in order for a discipline to be able to adopt free & open methods, the respective subject matter of that discipline must have been adapted to digitized working methods. Carpo (2011) argues that the adoption of digitized working methods in architecture may profoundly change how we, as architects, understand and approach our role as authors. The two theses that guide my work are:

1. The subject matter that must be adapted in order for architecture to be able to apply free & open methods is design and, by extension, the design process.
2. If architecture adopts free & open methods, then this may, among other things, change how we, as architects, understand and approach our role as authors.

I would like to point out that these theses guide my research, they are not hypotheses to be tested.

With these theses in mind, the objective of this work is to explore (1) how architectural design and the architectural design process needs to be adapted for free & open methods to be applicable in architecture and, conversely, (2) how adopting free & open methods in architecture may change how we, as architects, understand and approach our role as authors. Since this research is exploratory in nature, its findings are best understood as hypotheses.

The purpose of this research is twofold: On the one hand, identifying changes to architectural design and the architectural design process that are required in order to adopt free & open methods in architecture shall make it easier for future projects to adopt these methods. On the other hand, understanding how these changes affect our role as authors will help us to better assess the impact of adopting these methods.

Piet Verschuren and Hans Doorewaard (2010, p. 33), following whom this research project is designed, argue that research, generally speaking, is either practice oriented, that is, attempts to solve specific problems, or theory oriented, that is, attempts to gain more abstract knowledge. This research project is, therefore, located somewhere between those two poles. It is practice oriented insofar as it addresses a specific problem, namely, how we can adopt free & open methods in architecture. It is theory oriented insofar as it tries to gain abstract knowledge, namely, how we should design – that is, if we want to apply free & open methods; in other words, it contributes to design theory. According to Verschuren and Doorewaard's framework, it is design oriented, in the sense that its goal is to make recommendations on how we design our design process.[1]

0.3 Research Design

How can these two objectives – that is, exploring how architectural design and the architectural design process may need to be changed in order for us to be able to adopt free & open methods and how the adoption of these methods may change the way in which we understand and approach our roles as authors – be reached? Notably, exploring how these methods may change the way in which we understand and approach our roles as authors will not only benefit from hypotheses on how architectural design and the architectural design process will need to be adapted, but we will also require a (1) *discussion of authorship in architecture*, with which the changes that free & open methods may bring can be compared. Having said that, how can these changes be identified? Abdelkafi et. al (2009) also investigate the transfer of free & open methods from the digital to the physical realm, noting that while this subject has received more attention recently, academic literature is still scarce (even more so when it comes to the transfer of free & open methods to architecture, I may add). They investigate this transfer by first analyzing the differences between the production of software and that of tangible products, and then develop a set of challenges that projects that want to apply free & open methods for the production of hardware are likely to face. I would like to build on that approach. However, the design process is a creative process, there is often more than one way to "do it right". Hence, simply spelling out differences between designing software and designing tangible goods is unlikely to be fruitful – but trying to identify problems and challenges may be. Put simply, the most relevant changes to design and the design process will be those that present themselves as challenges. What is more, the challenges that Abdelkafi et al. have identified are rather general in nature, while this study aims to identify challenges that may be specific to architecture. Moreover, if interesting challenges are unlikely be discovered by comparing the design process of software to that of tangible

1. Note that Verschuren and Doorewaard use "design" to refer to the design of processes or tasks, rather than the design of structures or artifacts.

goods, they must be discovered in another way. Therefore, this study tries to identify such challenges (2) through an *analysis of sources on open design and open architecture* and, since literature on free & open methods in architecture is scarce, (3) through a *case study*.

The (1) *discussion of authorship in architecture* will cover (a) the (dominating) notion of authorship in architecture, (b) how that notion is connected to the way in which we design, (c) how the way in which we design is affected by the tools that we use and how these tools have changed with digitization and (d) how the ideas of the free & open movement have spread, in particular to fields that have been digitized.

The (2) *analysis of sources on open design and open architecture* will draw on literature on the application of free & open methods in design in general, in what has become known as "open design", and architecture in particular (the criteria for the selection of literature will be discussed in the review itself). Moreover, since the application of free & open methods tends to require extensive documentation, which in the case of open design projects often covers design rules and specifications, the analysis will also draw on such documentation (the selection criteria for sources will be discussed more extensively in that analysis). From these sources, challenges will be extracted. Where challenges are not discussed explicitly, particularly in the case of project documentation, they will be reconstructed from how they were addressed (in a similar manner to how we can infer that there is a steep slope from spotting a serpentine road on a map). Here, a failure to master a challenge will be no less interesting for this study than a success. These challenges will then be used in the case study.

The (3) *case study* will investigate the WikiHouse project.[2] The WikiHouse project was chosen because it is the most prominent and the most successful project to date that applies free & open methods to architecture (for a survey of open design projects see (Rosada, 2012). It is relatively mature, and the only project of this kind that has an active community and that actually distributes design work among the members of that community. Studying an open architecture project, that is, a project that applies free & open methods to architecture, this thesis can discover challenges that may be particular to architecture. What is more, studying the practice of free & open projects allows to discover challenges that have not yet been discussed in scientific or scholarly literature. Typically, single case studies consist of three columns, where each column represents a step toward completing the research objective (cf. Verschuren &

2. See http://www.wikihouse.cc

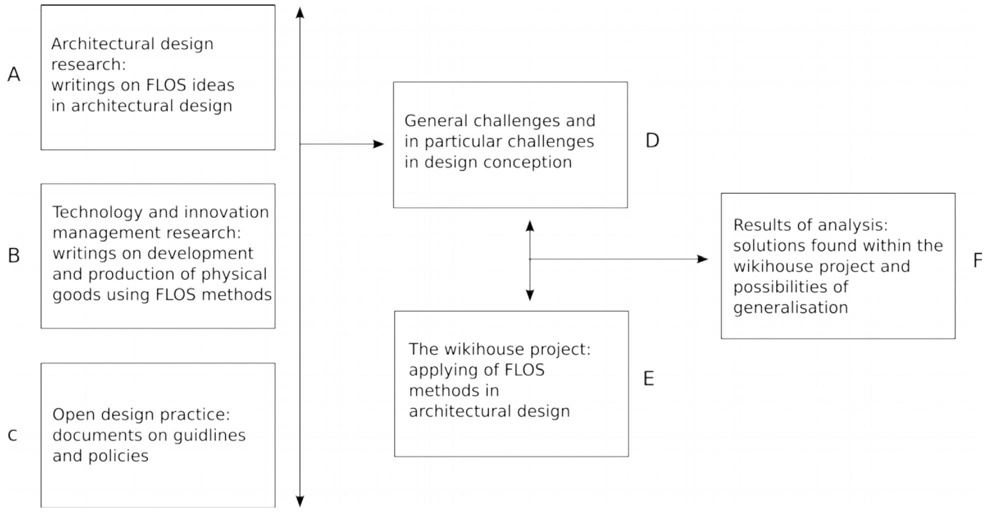

Figure 1: The columns of the case study

Doorewaard, 2010, p. 67 ff.): (a) the development of criteria with which the case in question will be confronted, (b) the confrontation of that case with those criteria and (c) the conclusion that is drawn from that confrontation (see figure 1for an overview). The (a) criteria for this case study will be challenges and categories of challenges that free & open architecture projects are likely to face; these challenges will be identified in the aforementioned analysis.

0.4 Research Questions

This thesis is guided by four sets of research questions, where the latter three sets correspond to the three columns of the case study outlined above. The first set of questions shall be answered by a discussion of authorship in architecture, mostly drawing on established literature. The first set of questions is:

1. What is the dominant notion of authorship in architecture?
2. How may digitization change that notion of authorship?
3. What are the core ideas of the free & open source movement?

The second set of questions of subquestions shall be answered by the first column of the case study, the literature review; the subquestions concern the development of criteria for the second column. What is more, because the answer to these questions will be categories of topics and challenges with which free & open architecture projects are likely to have to deal, these results will also be interesting in their own right. The second set of question and subquestions is:

1. What challenges are projects that want to apply free & open methods to architecture likely to face?
    1. What challenges or kinds of challenges are already described in the literature on open design?
    2. What challenges or kinds of challenges are already described in the literature on open architecture?
    3. What challenges that have not yet been described in the literature can be found by analyzing project documentation?
2. How can the challenges that are found in question 1 be categorized (taking the relevant literature into account)?

The third set of question shall be answered by the second column of the case study, the confrontation of the WikiHouse project with the categories of challenges that have been developed based on the literature review. The third set of question is:

1. Does the WikiHouse project encounter the kinds of challenges that have been identified in the literature review?
2. If so, how are these challenges dealt with?
3. Do these challenges impact design or the design process?
4. If so, is this related to the (attempt to) apply free & open source methods and how?

The fourth set of questions shall be answered by the third column of the case study, the conclusion drawn from column two. The fourth set of question is:

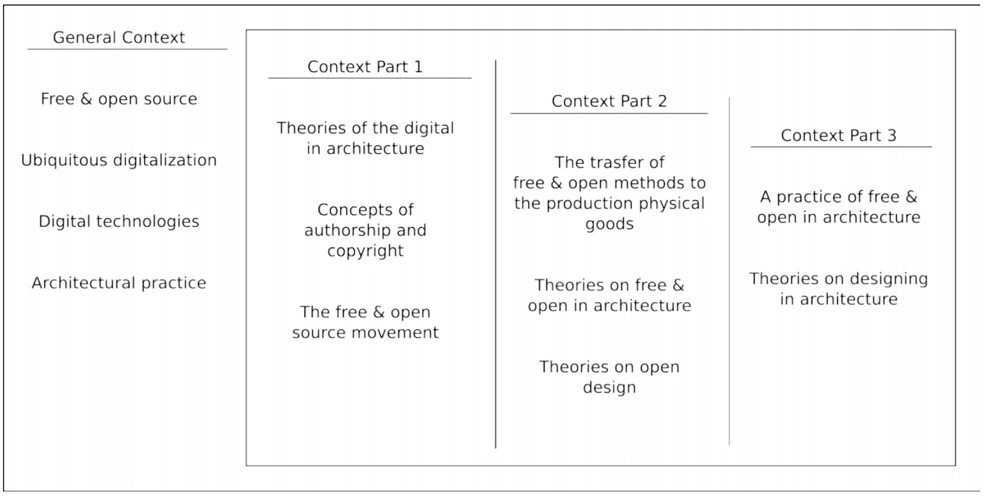

Figure 2: Context structure of this research project

1. Can the way in which the challenges that are connected to the application of free & open methods affect design or the design process, if any, be generalized? If so, to what extent?
2. Can the way in which the WikiHouse project addresses these challenges be generalized? If so, to what extent?
3. Do these challenges or the way in which the WikiHouse project addresses them affect the way in which we, as architects, should understand and approach our role as authors?

0.5 Structure and Outlook

This thesis consists of three parts. (1) The first part situates this thesis in a broader context, the discussion on the nature of authorship, and provides a foil to which the architectural authorship in free & open architecture can be compared. (2) The second part analyses literature on open design and open architecture as well as documentation of open design projects for challenges (the first column of the above research design). (3) The third part is the case study (the second column of the research design). These parts will be followed by the conclusion (the third column of the research design). I will now briefly describe and contextualize these parts further (figure 4 gives an overview of the context of each part and how they relate to the context of the overall thesis).

The first part of this research is mainly based on the work of Carpo (2011), who recounts the history of digitization in architecture and discusses the deep changes of architectural practice that digitization brought about, focusing on the consequences for the architectural notion of authorship. Carpo's work can be situated within a growing body of theory on the digital in architecture (on that growing body of theory, cf. Oxman & Oxman, 2013). This part will also discuss the history of free & open source software, how the free and open source software movement conceptualizes authorship, its model of production and to which extent those can be generalized. This discussion is mainly based on the seminal work of Eric Raymond (1999), a famous software developer, and Karl Fogel (2005), a former board member of the Open Source Initiative.

Situated in the larger debate on transferring free & open methods to fields other than software, the second part of my research project builds upon the current state of the debate on transferring free & open source methods to architecture. Andrea Rosada (2012), for example, shows a wide range of forms of attempts of using free & open ideas in architecture. Theodora Vardouli and Leah Buechley (2014), to give another example, show that – due to the different subject matter – there can be no 1:1 translation of free & open methods to architecture. Consequently, they explore what the terms "access" and "source code", terms that are central to free & open methods, can mean in architecture. All in all, however, literature is scarce and more detailed examinations of how free & open methods could be transferred to architecture are needed. However, as Rosada (2012) points out, the discipline of open design has already made some progress in establishing foundations for free & open development and production. Therefore, I will also draw on more general work on the transferability of free & open methods from the production of software to the world of tangible goods. That being said, as I have already mentioned, Nizar Abdelkafi, Thorsten Blecker and Christina Raasch (2009) who research free & open methods in design, find that literature on applying free & open methods in design is also scarce (it is, but it is still not as scarce as literature on applying these methods in architecture). Abdelkafi, Blecker and Raasch propose to investigate how free & open methods can be transferred to design by analyzing which challenges projects that try to transfer these methods to the design and the production of physical goods face. I will adopt this approach, and also try to identify challenges that such projects may face by studying the documentation these projects provide (cf. above for a more detailed discussion). Rosada (2012) gives an extensive overview of such projects. Projects that focus on establishing design rules and guidelines are of particular interest, for such rules and guidelines are likely to be responses to challenges of the sort in which this research is interested. Among these projects, the Open Source Ecology project[3] is the most important source, because it has not only been called a "best practice" model (Gibb, 2014), but also is extensively documented.

The context of the third part of my thesis is the application of free & open methods to architecture. Of course, most projects that try to apply these methods to architecture are still experimental. The most recent project to date being the WikiHouse project, which, as I have already explained above, is the most successful among these projects. And since this part confronts the WikiHouse project with the challenges that have been identified in the second part in order to explore how we need to adapt architectural design and the architectural design process, the wider context of this part are theories on designing in architecture.

3. See http://opensourceecology.org and http://opensourceecology.org/gvcs

1 The Free & Open Source Movement and the Digital in Architecture

1.1 Introduction

The history of computation and the digital in design is also a history of theories related to the digital. The use of digital tools for conceptualizing, producing and visualizing designs took roots in architectural practice in the late 1980s and early 1990s. Pioneer projects of this time, according to the exhibition *Archeology of the Digital* curated by Greg Lynn, were, for example, Peter Eisenman's Biozentrum (1987) or Frank Gehry's Lewis Residence (1989–95). Theories in architecture were also impacted by the technological changes. Eisenman, with his conceptual approaches to design, brought together terms and concepts from computational sciences, and inputs from linguistics and philosophy. In 1993 the AD Special Issue called *Folding in Architecture,* edited by Greg Lynn, was published, which is seen as a pivotal point for digital architecture in architectural design theory. *Folding in Architecture* was meant as an antithesis to the then dominant deconstructivist theory and for the first time positioned the digital in a prominent place in architectural design theory (Oxman & Oxman, 2013). For the following decade, writings on the digital in architecture focused on characterizing the new digital architecture by describing and explaining theories of digital architectural practice with a focus on formal and geometric aspects and a discursive interrelationship with philosophy and mathematics. At this time architectural design theory was mainly inspired by the generative aspects of the digital. In the mid 2000s the discourse on the digital in architecture shifted from a mainly morphological one to a discourse of theories about the concepts related to the possibilities that come with digital technology on the one hand, and a discourse about their influence on the design processes on the other. This was a shift towards more scientific, computational and technological approaches in architectural theory (parametric design, digital materialization and fabrication, performative design, integrated information, etc.) (Oxman & Oxman, 2013). More recently, the topics of theories have become broader, but at the same time also divided into more and more subdisciplines, which is not surprising, since digitality and computation became ubiquitous phenomena over the past five to ten years. Oxman and Oxman in their book *Theories of the Digital in Architecture* (2013) use the following categories to systematize today's theories, concepts and models: (a) the *Ontology,* which provides *Theories* on the nature of the digital in architecture, the relationships and impacts of the related concepts, and so on; (b) *Computational Processes:* this includes *Form and Generation, Performative Design, Parametrics*; (c) *Concepts and Models:* these are *Morphogenesis and Tectonics*; (d) *Technologies:* which include *Materialization, Fabrication and Responsive Technology*; and (e) *Epistemology* of the digital in architecture, which provides *Disciplinary Knowledge*.

Following the categorization by Oxman and Oxman, the focus of my thesis concerns *Ontology* as well as *Concepts and Models.* They are, in a way, *ontological*, since I want to sketch out what free & open source architecture might *be*. Furthermore, and more importantly, they concern how we can apply free & open source methods in architecture and are, in this sense, *conceptual*.

In this first part of my thesis my aim is to provide an approach that connects contemporary debates on the digital in architecture with free & open source concepts, in order to show that due to the nature of the digital, open source approaches should be of interest to architects as well as to further describe free & open source concepts and to introduce conditions of their generalization and thereby sketch the success conditions of free & open approaches in architecture.

To show that, today, theories on the digital in architecture already overlap with the debates on and around free & open source, I introduce the latest writings of Mario

Carpo on the digital in architecture. Oxman and Oxman locate his analysis within the ontology of the digital in architecture. Carpo, as a theoretician and historian who is interested in contemporary developments in architectural theory provides us with an analytic view on contemporary developments and is able to put them in a broader historical context. He approaches the digital and the digital turn in architecture by focusing on the impact of the according technologies on architectural practice, and combines this with the discussion on the concept of authorship in architecture. A main aspect of his argument is that the possibilities introduced by digital technologies are about to overturn paradigms that are crucial for the modern-age model of the architectural author.

1.2 Digital Architecture and Its Impact on Authorial Concepts

In the following I will review the thoughts that Mario Carpo (2011) presents in his book *The Alphabet and The Algorithm* and, subsequently, show how open source is part of an ongoing, bigger development that has been shaking up architectural practices for two decades. My aim is to show that free & open source models and their implication for architectural practice are not just some external developments that architecture now has to deal with, but something that goes along with the digital in architecture and with the current technical changes. The architectural practice is changing due to the general technological developments, which are much broader in scope than architecture. In the context of those developments, introducing free & open source practices to architecture is an interesting option.

The technological developments I refer to are digital technologies. It has become hard to imagine the world without the internet, computers, smartphones, microcontrollers, and so on, and it has become hard to imagine an architect's workplace without digital design, building information management and digital communication tools. Digital technologies have become ubiquitous and impact every profession as well as every aspect of our everyday life. And, as all earlier technological revolutions, digital technologies brought and bring massive changes, not only on a technological level, but are all-encompassing. They impact society, politics, economics and culture. In architecture, digital technologies impact the design methods as well as the design outcomes and architectural practice in general.

Carpo even argues that digital technologies impact the fundamentals on which the classical modern definition architecture, as an art and as a profession, is built. In the preface to his book *The Alphabet and The Algorithm,* he talks of a "shift in paradigms" and argues that due to technical developments, the definition and authorial conception and self conception of the architect, as they have been used for a long time, have become brittle. His argument is based on the nature of the digital. I will use his notes on the digital in architecture and its impacts on authorial concepts to argue why adopting the concepts of free & open source should be interesting for architects in the context of contemporary technological developments. Before connecting Carpo's thoughts to free & open source concepts, I will trace Carpo's arguments to give an understanding of how the nature of the digital is impacting the conception of the author in architecture.

Following Carpo, the conception of the architect as a designer that is the author of built architectural artifacts goes back to the 15th century when the architectural theoretician Leon Battista Alberti on the edge of the transition from the middle ages to the modern era published *De Re Aedificatoria.* In this treatise, which is seen as the first modern definition of architecture and its practice, architecture is defined as an allographic, authorial, notational art. Alberti's theory can be seen as the "invention" of the architect as an author, as he called for a separation between designing and making, and defined architectural authorship in a humanistic intellectual sense and in contrast to the master-builder and the artisan tradition in the building culture of pre-modern times. The building in his theory is seen as the identical copy of the architect's design. Introduced by Alberti, the idea of identical copies had its second bloom with the Industrial Revolution and the rise of mechanical mass production. Alberti's definitions are still the base for today's understanding of the architectural authorship. Although, in its absoluteness, and as a highly idealized description, it never has been completely implemented in practice (cf. Carpo, 2011, p. 117). Carpo confronts the Albertian definition of architecture as allographic, authorial, notational art with the characteristics of the digital and its consequences for architects practice in order to show that these "Albertian paradigms" are about to be overturned due to technological developments with clearly disrupting consequences for the theoretical constitution of the according concept of architectural authorship.

I will now briefly describe the confrontation Carpo makes, as well as its outcomes. These outcomes, I think, already indicate the potential of open source approaches. But firstly I shortly explain the Albertian definition of architecture by explaining what is meant by saying architecture is (1) an authorial, (2) an allographic, and (3) notational art and how this is constitutive for the authorial concept of architecture.

Calling architecture (1) an authorial art means that creatorship is not attributed to the person who physically crafts the object, no matter how artistically it is crafted, but to the person who created the idea. The architect is the originator and creator of the building as she or he is producer of design thoughts which are brought to paper and noted in a way that may be executed by others. The design is set identical with the physical building. To make this identicality conceivable, a notational system is needed that makes it possible to relate one to the other. "When this condition of notational identicality is satisfied, the author of the drawing becomes the author of the building" (Carpo, 2011, p. 23). The building is thereby the physical reproduction of the object contained in the blueprints. "In Alberti's theory, the design of a building is the original, and the building is its copy" (Carpo, 2011, p. 26). As the work which is truly done by the architect is the design, the authorship of the architect is an intellectual authorship. Further, with authorship not only intellectual creatorship is acknowledged, but also "the architect's design becomes as authoritative as any 'authorized' literary text: in Alberti's theory, a stable, permanent, authorial source not to be altered, changed, or in any way tampered with by others" (Carpo, 2011, p. 138, footnote 42). This integrity of the design is part of the moral rights, which are inalienable rights, that are assigned to the author by today's copyright laws (at least in countries that follow the droit d'auteur tradition).

Architecture is called (2) an allographic art due to the fact that architectural plans serve as blueprints for the architectural object. The very literal meaning of allographic, when used in the context of architecture, is that architecture has its notational language, the design drawing, which can be translated into the physical architectural object. The architect is the one who scripts the design while the material realization of his design is executed by others. This can be compared to the composer who is the one who produces a musical idea and notes it in a standardized way so it can be played by others (Carpo, 2011, pp. 16, 20). In Alberti's view, the architect should not be involved in the process of materialization at all. When design is finished, the builders take over. "Alberti's distinction between building and design [...] is one of the foundational principles of his entire architectural theory" (Carpo, 2011, p. 20). To be "designed by one to be constructed by others" (Carpo, 2011, p. 16), represents according to Carpo the modern allographic definition of architecture as an art. The term allographic arts was coined by Nelson Goodman in opposition to the autographic arts (Goodman, 1968). Carpo writes that "[a]ccording to Nelson Goodman, all arts were born autographic— handmade by their authors. Then, some arts became allographic: scripted by their authors in order to be materially executed by others" (Carpo, 2011, p. 16). When in Alberti's theory the material process of making was thought as pure reproduction of what is noted in the design drawings, to the effect that, in his theory, architecture could be seen as fully allographic, it was not so clear that architecture already had fully transformed into an allographic art for Goodman; not least because he found architectural plans to be something like, but not yet a fully developed standardized notational language.

Architecture is (3) a notational art as architectural plans are a kind of notational system (Giovannelli, 2010). The notational techniques that are used in architecture today were developed in the 15th century with Alberti playing a role of crucial importance. His writings mark the rise of the idea that the design idea must be translated and noted in a graphic and standardized form. This form he finds in floor plans, elevations, sections, etc. which together form the architectural design. A standardized notational system guarantees a standardized translation which is needed to argue for the identicality of the design and the built object and, consequently, for the authorship of the architect. Further, the notational unambiguity is a precondition for separating the processes of designing and building. With a notational unambiguity, the transformation into physical matter can be executed by any skilled builder and will be recognized as the object described in the design notations.

So due to the authorial, allographic and notational concept, even if the architect has never touched the walls of the building, Alberti would argue – and for us today this is no longer even a question –, that he or she is the author of the building, due to the constructed "identicality" of the design and the built object (cf. Carpo, 2011, pp. 16–23).

Now that we know what is meant by architecture being an authorial, allographic and notational art and how authorship is connected to this, we can investigate how this concept is impacted by the use of digital technologies. Carpo highlights that the "Albertian paradigms" are still very present in the picture of architecture today, but shows that, when confronted with digital technologies and their consequences for architectural practice, these concepts do not mach reality any more. In architectural practice, we have seen a shift in working methods due to digital technologies that started in the early 1990s with neglecting the drawing board and adopting the computer as main tool used in architectural design work and that continues to this day. This shift is known as architecture's digital turn. In what follows, the changes that were induced with that digital turn are recapitulated and compared to the Albertian conception of architecture as a notational, allographic and authorial art. As Carpo's argument is centered on the concept of authorship, I will describe the aspects of the digital that are of interest when asking for its consequences on authorship.

Basically there are four points effecting the concept of authorship as it is used today: (a) the three-dimensional virtual space, (b) the use of parametric functions and algorithms, (c) the digital editing and the general nature of digital design, and (d) the use of digital fabrication techniques.

(a) The three-dimensional virtual space: The change brought by 3-dimensional virtually was that it made a new range of forms accessible. Two-dimensional graphic representation is restrictive in which forms may be represented in an understandable and unambiguous way within what is seen as an appropriate effort. So architecture for a long time was restricted to what geometries could be well represented in two-dimensional drawings. Digital environments allowed to work in a three-dimensional virtual space. While it is hard to represent free-forms in a measurable and thereby buildable way in two-dimensional representations, when working with 3D virtual models, every point in a model becomes measurable.[4] In consequence "all that is digitally designed is, by definition and from the start […] geometrically defined and buildable" (Carpo, 2011, p. 34). With computer aided design and the three-dimen-

4. This brand new situation opened up a whole universe of newly accessible forms for architects. Free-form architecture of the middle and late 1990 is the result of this (cf. Carpo, 2011, pp. 40–41).

sional virtual space digital design may and do become virtual avatars of the physical object.

(b) Parametric and algorithmic design descriptions: The change brought by parametric and algorithmic descriptions is that one and the same design-algorithm may result in a variety of physical objects, a parametric object may take different forms and have a variety of different physical results. This clearly differs from a notation in which there is only one possible physical object that each blueprint may result in. Parametric design is a description of rules and parameters that define the form of an object. By designing this way, designs or design elements are turning into mathematical functions with variable parameters. Bernard Cache (1995, p. 88 ff.) and, referencing him, Gilles Deleuze (1993, p. 19), have coined the term "objectile" to describe this. They contrast the objectile with the object. The object is clearly defined in terms of its form. The objectile, in contrast, is a clearly defined rule or set of rules. Objects are the result of executing these rules. Carpo describes the objectile as follows: "the objectile is not an object but an algorithm – a parametric function which may determine an infinite variety of objects, all different (one for each set of parameters) yet all similar (as the underlying function is the same for all)" (Carpo, 2011, p. 40). When using one and the same algorithmic or parametric source the resulting instances share a similarity due their shared source which surely may, but does not need to be, a visible one. The final shape the single instance takes is partly predefined by the source script, but only ultimately decided by using that script.

(c) Digital editing: The change brought by digital design and digital editing in general is the alterability and variability of design drawings. If architectural plans are drawn by hand each plan is a piece of its own which is hard to alter. Making changes in a hand drawn plan or using elements within an other design means redrawing. In digital designs it is possible to alter the drawing or to exchange single parts of a design without redrawing the whole. Single design elements can, drawn once, be reused by simply pasting them into a new environment. If elements are defined in a parametric way, they not only can be reused as they are, but can even be adapted within the ranges of their definitions, e.g., in matters of size to fulfill different requirements. Carpo notes that "[o]pen-endedness, variability, interactivity, and participation are the technological quintessence of the digital age" (Carpo, 2011, p. 126).

(d) Digital fabrication: The change brought by digital fabrication is a variability in manufacturing that allows for machine-executed but customized production. Digital manufacturing is based upon the development from analog, mechanical, and electronic technologies to digitally controlled machines. Such digitally controlled machines are, e.g., CNC mills, laser cutters and 3D printers. While the major achievement of mechanical manufacturing was serial production of identical copies, with digital fabrication it is possible to produce single copies of varying shapes. While the mechanical era was characterized by mass production, digital production is characterized by mass customization. since digital fabrication tools can produce variations at no extra cost (cf. Carpo, 2011, p. 41).

Above, I have shortly described four of the aspects of the digital that Carpo mentions in his analysis of the impact of the digital turn on architecture as authorial, allographic and notational art. I will now sketch out, by drawing on Carpo's arguments, why these aspects are impacting it.

When looking at (1) the concept of authorship that makes architecture an authorial art, and how architecture is impacted by digital technologies, we need to look at the changes to intellectual creatorship and the moral rights of the architect as an author connected to it. Those moral rights, as we have seen, include the right of the integrity of the design, which is an unalienable right the author is given but also a claim that the author is confronted with. Here what is mentioned in (c) as a change that came with digital technologies becomes relevant: Digital files are first and foremost alterable. And moreover it is a crucial quality of digital objects that they can can be copied easily, pasted, adapted, and combined in new ways. Making use of these qualities conflicts with a concept that calls for stable, permanent and inalterable authorial sources. We have even come up with technological solutions (so called "Digital Rights Management") to restrict the variability of digital artifacts and to reinforce the authorial source as a stable object. But we also experience that this kind of protection of the authorial source, as it is understood in the theory of Alberti and embodied in today's moral rights, can hardly be maintained in the digital. Due to the nature of the digital, the possibility may not and is not to be ruled out that "some parameters [of a design] may be chosen, at some point, by someone other than the 'original' author, and possibly without his or her consent" (Carpo, 2011, p. 42) So we can say that the variability that comes with (c), the digital editing and the work with digital designs, affects (1), the authorial aspects of architecture and its practice as they were defined by Alberti, more specifically, the claim for a permanent and stable authorial source.

For (2), the allographic definition of architecture as an art, one of the points at which we should take a closer look, when talking about impacts of the digital technologies, is the clear separation of designing and making the physical artifact or, as Alberti puts it, the principle that the architect should not be involved in the actual building process. Hence, we should look at the digital design and the work flows that evolved with digital technologies. In digital design, due to the possibilities offered by (a), the three-dimensional virtual space, the physical objects can be represented *as they are*, as three dimensional objects with true lengths and widths. Moreover, with building information modeling and management (BIM), a broad range of information and properties of building parts can be included in the digital model. So digital models can be seen as avatars of the physical object. As objects are available in true size and form as digital data, such objects, such data, can be directly used for (d), digital fabrication. For these work flows, in which digital designs are used for digital fabrication, the term computer aided design and computer aided manufacturing integration (CAD-CAM integration) has been coined. Carpo argues that due to CAD-CAM integration, architects are more and more involved not only in design, but also in the actual production, because digital design and manufacturing allows to turn digital models and drawings into physical matter through machines that can be said to be controlled by the architect, for he or she is the producer of the digital blueprint. But that the digital design being is an avatar of the physical object has another consequence for architecture being an allographic art as well, namely in regard to the uncertainty that Goodman felt about calling architecture fully allographic. Carpo states that "[a] cad file would certainly satisfy all of the complex requirements that Goodman asked of notational systems" (Carpo, 2011, p. 78). So it can be said that the combination of fully geometrically defined true size digital objects provided by the possibilities of (a) the three-dimensional virtual space and (d) the possibility to transform this digital data into physical matter by the use of digital fabrication technologies turns the architect into a maker, one that is directly involved in the materialization process. Therefore, (2) the clear separation between designing and the production process, which Carpo calls essential for architectures modern allographic definition, is no longer applicable (cf. Carpo, 2011, p. 45) at the same time digital design architecture allows for a notational exactness, rendering architecture more allographic that it ever has been before (Carpo, 2011, p. 78).

Regarding architecture being (3) a notational art, it is the aspect of a standardized notation that enables unambiguous translation of the design into a built object that is of interest. This unambiguous translation is the condition on which the identicality, the sameness, of the built object and the design hinges. A precondition for this identicality, according to this conception of architecture (Carpo, 2011, p. 43), is that design descriptions that are rule based, like (b) parametric and algorithmic descriptions, are defined in a way that allows for multiple possible physical outcomes, each of which attributed to the respective rule. This is why, so Carpo (2011, 32), "[d]igital technologies inevitably break the indexical chain that, in the mechanical age, linked the matrix to its imprint" (Carpo, 2011, p. 32). So we can say that (b) the parametric and algorithmic descriptions impact (3) the concept notational identicality because of which architecture is called a notational art.

We have seen that digital technologies have an impact on the definition of architecture as authorial, allographic and notational. The confrontation of the definition of architecture as authorial, allographic and notational with (some of) the changes digital technologies brought for architectural practice has shown that: (A) The variability that comes with digital technology conflicts with the demand for a stable authorial source. (B) The fact that the architect becomes a maker by combining digital design with digital fabrication technologies conflicts with the demand that that the architect is the designer and as such not involved with the materialization. Although (C) three-dimensional digital designs and building information allow for a new quality of notational identicality of design and physical object, (D) parametric design algorithms enable us to generate many different outputs, which conflicts with the concept of notational identicality and a stable 1:1 relation between design notation and its physical translation; assumptions used to argue for architectural authorship. So we have seen now that, as Carpo argues, the definition of architecture as an authorial, allographic and notational art, that is, the "Albertian paradigm" is impacted by the digital technologies – that is, digital technologies, at least certain aspects of them, do not fit the authorial, allographic and notational model.

We have seen now, by reviewing the arguments of Carpo, that Alberti's definition of architecture does not fit an architectural practice that uses digital technologies. As already mentioned this also affects our notion of authorship, since today's concept of architectural authorship is closely related to Alberti's definitions. As Carpo notes "[m]odern architectural authorship came into being only with the rise of […] the Albertian paradigm […], this paradigm has […] inspired most of Western architecture for the last five centuries, and it is at the basis of the dominant legal framework that still regulates the global practice of the architectural profession" (Carpo, 2011, p. 117). That intellectual works, including architectural ones, have one author (or at least clearly defined authors) and one source, which must not be altered by anybody but that author, as Alberti has argued, are beliefs that are firmly embedded in our legal frameworks. In consequence, aspects of the digital that conflict with the "Albertian paradigm" also conflict with today's dominant legal framework and the corresponding concept of architectural authorship. On the one hand, there is a legal definition that calls for a stable, inalterable authorial source that can result in one type of object and one type of object only. On the other hand, there is the digital that is by concept variable and alterable and algorithmic and parametric design definitions that are meant to result in a multitude of digital artifacts and thereby, by virtue of their very nature, contain the possibility of a multiple physical artifacts. With the previous section in mind, I would conjecture that the conflicts described in (A) and (D) could prove to have disrupting consequences for today's model of architectural authorship, whereas (C) and (B), to be sure, may also affect our current legal framework, but not in a way that would require to re-think the core assumptions of that framework. (A) and (D) are about the variability of the digital; this leads me to the conclusion that it is this variability of digital technologies in particular that conflicts our current model of architecture, and that this is where we need new approaches.

I now suggest that free & open source approaches to authorship present a more suitable approach. But before I go on to argue this, I would like mention that Carpo also stresses the impact of digital technology on physical production. He argues that authorial models are closely related to the then dominant mode of production of physical goods, for the use of technologies has a broad impact on our daily lives and society in general. Carpo argues that the traditional concept of authorship, the one that informs Alberti's position, became dominant hand in hand with the rise of technologies that allowed for the mechanical production of identicals; these technological developments have shaped our perception of, and our thoughts about, authorship (cf. Carpo, 2011, pp. 20–26). The rise of technologies to produce and mass-produce identical copies began in the early modern age with the discovery of perspective as a way to "copy reality" and the invention of the movable type by Gutenberg. Later examples are photography, industrial mass production and the photocopier. Carpo (2011, p. ix) notes that "[i]dentical copies inspired a new visual culture, and prompted new social and legal practices aimed at the protection of the original and its owner or creator." With digital fabrication techniques, the influence of mechanically produced identicals on our conception of authorship is waning and so, we may hypothesize, will the thoughts that came with them (cf. Carpo, 2011, pp. ix, 1–4) Parametric functions in design and digital fabrication methods taken together develop a disruptive power, since with them comes the idea of a non-standardized seriality, and this idea challenges the authorial construct that has been defining the architectural profession from the onset of modernity on onwards (2011, pp. 41, 42); in the words of Carpo (2011, 42): "Nonstandard seriality […] already contains the seeds of a potentially different authorial approach".

So far this is what I extracted from Carpo's writing on the authorial paradigm, its rise and its predicted fall with today's digital technologies. Carpo artfully outlines a theory of authorship as well as digital architecture, and hence makes an important contribution to the discourse on the digital in architecture. He draws a relationship between architectural theory, historical accounts of architecture, digital technology and today's architectural practice. The story he is telling is the story of the paradigm of modern architectural practice from its emergence with Alberti's praise of humanistic authorship in the early modern age to its obsolescence in the past decades. However, Carpo not only wants convince us that an old model is falling, but also that a new one is inevitable and that we are about to recast our concepts – in theory as well as in practice.

Touching upon topics like horizontal integration of actors and agencies, open-endedness, interactivity and collaboration he provides us with a strong foundation to argue for new practices in design that build on strategies and authorial approaches. Some of them, I propose, will come from the free & open source movement.

Allow me to recapitulate. I have shown that contemporary debates on digital architecture in architectural theory are taking up the subject of authorship in the digital age. What is more, bringing together digital architecture with a discussion on authorship in the digital age paves the way for connecting digital architecture with open source on a theoretical level – as the topic of authorship in the digital age links architecture to the free & open source movement.

1.3 The Death of the Author

The next thing I would like to discuss, before moving on to look closer at what open source actually means, is that Carpo has not been the first in architecture to recognize that the digital and its methods have grave effects on the traditional concept of authorship. So let us go back in time, to the early days of digital architecture.

At the dawn of the digital age, Peter Eisenman already discusses eliminating the author, or at least unfavored aspects of authorship, by using digital methods (cf. Kuhlmann, 2001). The theoretical figure Eisenman used is *The Death of the Author* – an element of post-modern theory proclaimed by Roland Barthes (1967), a French literary theorist and philosopher. The artistic strategies that have been in this context proclaimed by Eisenman are based on the conscious, chosen detachment of creator and object. Eisenman was one of the first who concerned himself with the issues of new digital technologies and in the 60s and 70s developed different algorithms to put in between the author and the design. Furthermore, he worked with arbitrary input taken from different fields. Theories taken from the natural sciences as well as from philosophy served as "machines" to generate design. The concept of detachment may be seen as a position derived from a rejection of the positions proclaimed by the predominant classical architectural tradition. To break with that tradition the author, in whose mind this tradition was imprinted by education, had to be detached from the design. (Jencks, 2001; cf. Kuhlmann, 2001) By detaching the architect and, in consequence, declaring the architect to no longer be an original genius but to be replaced by the arbitrariness of a machine, the author is symbolically declared to be dead.

When we compare Carpo's argument and Eisenman's aim, we can conclude that the idea of introducing algorithms to design, as it was Eisenman's strategy, impacts authorship. And while his act of killing the author may not have been as successful as he might have hoped for, the insight that it does impact architectural authorship if the architect ceases to have full control over an output that can no longer be described as "his" or "hers" is nonetheless profound. What is more, we can see that Carpo and Eisenman both have given some thought to the matter how the advent of algorithms impacts architectural practice. Of course, they did so in different ways; it will be helpful to follow each of them. Both see a potential rupture that is a threat to classical authorship. That said, they locate that rupture at different points, in different phases of the transformation from an idea to an object. In the relation between subject (the architect) and object (the building) there are two transformations the idea has to undergo while staying the same: (1) From the architect's mind to a design and (2) from a design to a built object. Whereas Peter Eisenman aims to loosen the connection between the architects mind and the design, Carpo argues that we can no longer maintain a conception of architectural authorship that is build on an allographic relationship between design and object. Eisenman and Carpo both argue that our current conception of authorship is no longer useful, and that we need to let go of this conception in order to be able to move on. We need to open up to new concepts. Eisenman argues for an active act of getting rid of the author. Carpo takes the position that authorship in the Albertian and modern sense is a concept of the past at any rate, due to the inherent properties of digital design and fabrication (Carpo 2011, 47). When Carpo mentions the death of the author he clearly says that, in his eyes, death threatens "but one, particular, time-specific category of authors: the author of identical, mechanical copies—the modern, Albertian author" (Carpo, 2011, p. 47). So authorship in general does not vanish for Carpo.

The death of the author in its interpretation by Eisenman was undoubtedly a radical gesture and a decisive act of liberation. What is of importance to us is that Eisenman not only recognized the generative moment of design algorithms as a new way of finding forms, but that he also related this generative moment and the variety of outputs that could be generated by algorithms to the concept of the modern author – diagnosing a deep incompatibility.

To sum up, our journey back in time has shown that the new possibilities of the digital, particularly, their generative possibilities, and their impact on our traditional conception of authorship have been recognized early on in architecture.

1.4 Authorship of the Many

I have already, with reference to Carpo, discussed the changes that come with the rise of the digital in architecture, the copyability, the alterability, and the generative powers of algorithms. However, this was somewhat abstract. We should now take a look at how these new dynamics play out in practice – on the internet. There, a new culture has developed: the web culture. This culture emphasizes spreading, sharing and remixing, the power of the crowd, open-endedness, interactivity and online-collaboration. So, having seen that contemporary debates on digital architecture in architectural theory have taken up the subject of authorship in the digital age and that the disruptive powers of the digital in matters of the traditional concept of authorship have been identified at the dawn of the digital age already, the aim of this section is to take a look at the current situation and the approach on authorship that can be found in contemporary internet culture, where the possibilities of the digital, above all, its open-ended variability, are put to use to their fullest extent.

Fully making use of these new possibilities, introduced by the open-ended variability, requires opening up to new concepts, since the old concepts of authorship call for a inalterable and stable source. Of course, we are, as architects, used to this traditional concept of authorship and may even define ourselves as authors in this sense. Carpo (2011, p. 47) nonetheless finds that "[e]vidently, even among practitioners less inclined to theoretical speculation, the nagging feeling that something today is not quite right with architectural authorship has made some headway."

Looking at today's web culture, the internet is a lively place of interaction and user generated content – and it was from the beginning. Recently, however, due to a broad commercialization of the web's interactive features by large companies, and a general shift in how web pages are made and used – changes that are often labeled the "Web

2.0" – this interactive use and collaborative web tools gained broad popularity.[5] With the internet as a place of interaction, with its culture of sharing, spreading and remixing, of action and reaction, information is reused and supplemented and authorship gets a cascading and cumulative character. Information is used to produce derived works or follow-up works – that is, works based on already existing work, works that would not have been possible without them. Furthermore, there are collections and conglomerated works, works that are made by the combination of other works or fusion of other works of many, collectively produced by communities. Such conglomerated works have by definition more than one author. Cascading, collective or cumulative authorship is found in, and is the case for, a lot of digital art found on the web, for graphic design elements, for collections of scripts of 3D objects, for digital information collections like Wikipedia and for a lot of free & open source software. Moreover, it is often not customary to indicate all involved authors. Authorial traces in digital artifacts are often complex and when we try to think about new models of authorship, we need to take into account that there are things that have many authors, things that include a growing complexity of interwoven digital authorial traces or even things that have no author or where a pseudonymous author is used because of too many people have made contributions. Put simply, reminding ourselves of the impact of variability, alterability and copyability on authorship, we can say that the advent of the digital may have killed the solitary author, but has paved the way for a true authorship of the many.

When we surveyed the history of authorship with Carpo, we saw that the notion of authorship is nothing "natural," but constructed as an interwoven system of practices, technologies, believes and laws. Put in familiar lingo, authorship is socially constructed. Social constructs perform one or (more typically) a diverse set of functions and are circumscribed and reinforced by legal frameworks. Authorship, in its traditional sense, has, inter alia, the functions to define the authorized version, to define the cut-of line behind which no editing is allowed and to control who may use a work and how they may do so by restricting copying. Today, copyright, which is used to enforce an Albertian conception of authorship, is discussed controversially.

Authorship of the many or multi-authorship, as it is practiced in participatory and collaborative digital cultures, differs in its functions. Its goal is to keep content

5. For a critique of this commercialization of collaborative and interactive tools see, e.g., Mushon Zer-Aviv et al. (2010, p. 15 ff.).

broadly usable and to enable adjustments, alterations and the emergence of different versions as well as remixes.[6] To ensure these qualities and to keep artifacts variable, so called "free & open licenses" are used. A lot of freely or openly licensed artifacts may be found on the web. And there are more and more thematic collections. Everyone may, for example, put a script online under a free or open license. As such licenses explicitly allow use, editing and publishing of derived works, this script may get modified, complemented, improved and republished or may even be used in a bigger context. Over time, content published on the web under a free or open license may have many authors. This may happen, or not. But there are also more organized projects that build on the variability and open-endedness of the digital, which are more far-reaching than individual publishing under free licenses. Projects where people interact collaboratively on an idea and do that on the base of variability and multi-authorship or authorship of the many. A form of organized authorship of the many are free & open source projects where in form of a collaborative mode of production multiple people contribute to realize a common objective, often one that none of them could realize alone. A project may function as a crystallization point that attracts people. A script, or another piece of work has a function, a use. A project can offer more, since it can embody an idea. Often, in such projects, collective identities are formed and embodied in organizations, though these organization need to be very formal, for whatever is produced by the members of this organization has many origins. Also, even if a project were published as free & open source, if it were still de facto controlled by a single person, rather than an organization, that project would be much less attractive for other collaborators. It would fail to offer the prospect of being part of something. We could also say, such a project would be dominated by somebody who, in his or her role, closely resembles the role of a traditional author – and this is alien to web culture.

To sum up, with the inherent characteristics of digital artifacts (variability etc.) and the collaborative and interactive practices of internet culture, authorial realities today include diverse variants of multi-authorship. Artifacts with multiple authorial traces are produced by people acting individually as well as collectively. Building on

6. The function of multi-authorship is, for example, mentioned in an artistic research project on artificial multi-author personalty created by collaboration (cf. Snake-Beings, 2010).

multi-authorship, such organized forms of collaborative action are used to reach goals that would be out of reach for each single individual.

1.5 The Open Source Model

We now know that with digital technologies and internet culture, the reality of authorship includes different forms of multi-authorship. Today's legal basis for this kind of multi-authorship, in all its forms, was laid down by the free & open source movement to protect a mode of work that essentially builds on the possibilities opened up by digital technologies, above all, variability, alterability, and copyability. We have seen that our concept of authorship in architecture no longer fits reality. The free & open source methodology, by contrast, has been developed in step with the growth of digital technologies and may hence offer a concept of authorship that fits with the inherent variability of the digital. That being so, architects should be interested in free & open source concepts. Put simply, they are essential when talking about authorial approaches fitting the digital age. Hence, the aim of the following section is to have a look at the free & open source movement, its concepts and its methods.

As already mentioned, the free & open source movement has enabled projects to evolve that build on organized multi-authorship approaches that openly invite – enabled by the internet – users to collaborate asynchronously and over great distances. Such projects are often open-ended, their creations have a dynamic character and are characterized by revising and adaption. Furthermore, their output is a collaborative effort, where transformative authorship, multi-authorship and collaborative authorship are combined with non-proprietary accounts of authorship. Processes like revising, adaption and open-ended refinement are enabled by free & open licenses, which waive most rights granted by copyright laws, since these rights, in their usual form, would hinder these processes. Projects like Linux or Wikipedia are examples of such collaborative communities that work together under free commitment to produce something that is a conglomerate of the work of many. The method of working together that these and similar projects use has grown from the practices of the free & open source software community. The principles of this movement were laid down in the mid 1980s and today are essential not only for producing software.[7] By looking at the principles of the free & open software movement and relating them to the qualities inherent to the digital, it becomes clear that those qualities occupy a central position in the free & open software movement. The principles of the free & open software movement claim that everybody is free to use, alter and redistribute the original as well as altered versions of the software.[8] That is, by securing for everybody the freedom to alter the original, the free & open software movement embraces the variability of the digital; and by securing for everybody the right to distribute copies of the original they embrace the copyability of the digital. They secure these rights for everybody by releasing their programs under special licenses, so-called free licenses, that enable the multi-authorship approaches I have discussed above. The principles of free & open software were put in place to protect a working process that has developed in the early days of digital age (Tai, 2001). This working process was based on free sharing and building communities. Eric Raymond (1999) describes this process in his essay "The Cathedral and the Bazaar," which has contributed to establishing free & open source methods as useful and effective methods to develop software.

In the literature on free & open source software development it soon becomes apparent that free & open source is not only about the accessibility of information (i.e., the code) but about a mode in which what we may call intellectual goods are developed. A mode that is implemented by FLOSS projects. According to Raasch et al. (2009) "[i]n the scholarly literature, OSS [i.e., open source software] is identified as an example of a 'new innovation model' beyond markets, hierarchies and strategic alliances (Osterloh & Rota, 2007) that has also been referred to as the 'community-based model' (Shah, 2005), the "open source method" (Osterloh & Rota, 2007), or 'open-sourcing' (Ågerfalk & Fitzgerald, 2008). Open source development is an example of the private-collective model (von Hippel & von Krogh, 2003) and one form of open technology (Nuvolari & Rullani, 2007)." Clay Shirky (2005) talks about the open source method or pattern as being "part collaborative creativity, part organizational

7. The principles of the free & open source software movement have inspired communities in other fields that formulated their own principles (Freedom Defined, 2008; OD+H Group, 2013; OHANDA, 2009; Open Knowledge Foundation, 2009; cf. OSHWA, 2013b).

8. "The four freedoms of free software: 0.: The freedom to run the program, for any purpose; 1.: the freedom to study how the program works, and adapt it to your needs; 2.: the freedom to redistribute copies; 3.: the freedom to improve the program, and release your improvements to the public, so that the whole community benefits." (Free Software Foundation 2013)

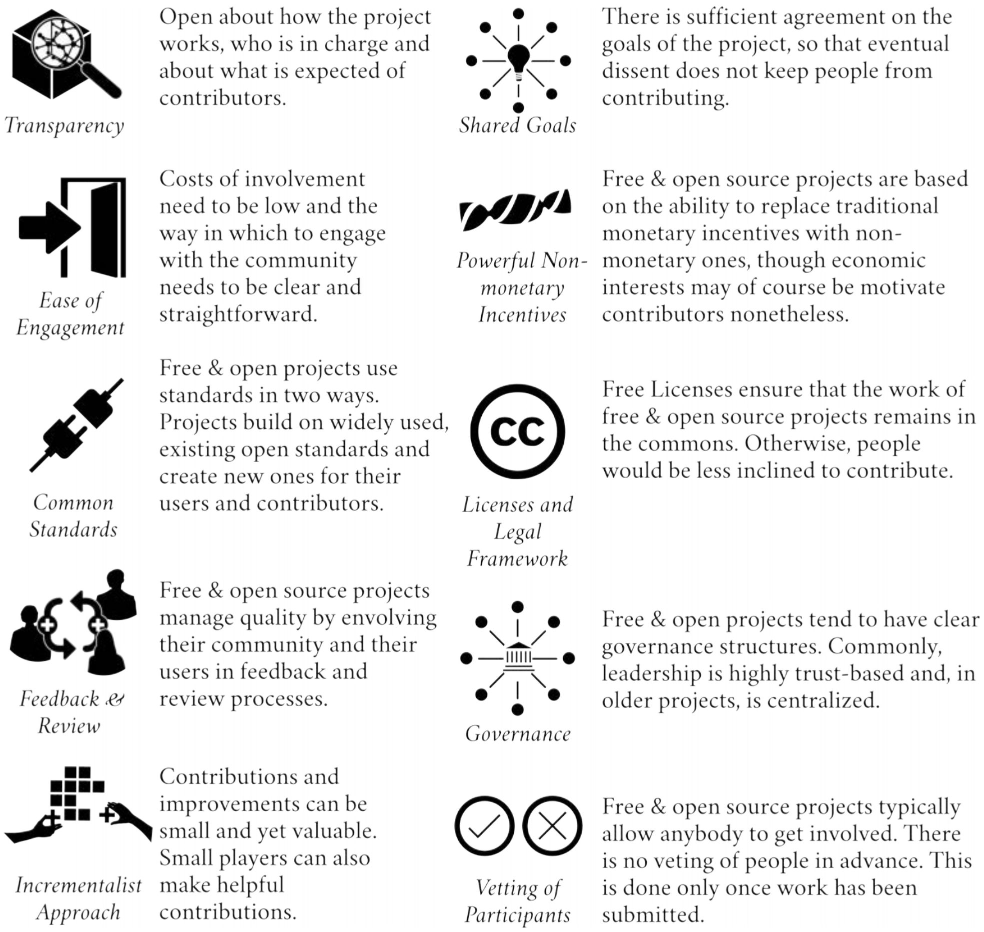

Figure 3: Characteristics of free & open source development, as found in most major open source projects (cf. Mulgan, Geoff, Steinberg, Tom, & Salem, Omar, 2005)

This figure has been composed using the following works: Network by Brennan Novak, Cube created by José Manuel de Laá, Entrance by Pedro Lalli, Connection by Pedro Ramalho, Feedback by Attilio Baghino, Help by Moveable Feast Collective : GDA 01, Share Idea by Kelig Le Luron, Candy by Julia Soderberg, Museum by Anbileru Adaleru. All from the Noun Project.

style, and part manufacturing process." The free & open source movement had a special working process from its very beginning, and its social practices and technological approaches have grown with it. Fogel (2005) who, in *Producing Open Source Software How to Run a Successful Free Software Project,* described those social practices and refers to them as the technical and political infrastructure of free & open source.

This section has given a brief overview over the concepts that free & open source projects use as well as their origins. We have seen that, as a movement that emerged with the internet and digital technologies, the free & open source movement has developed approaches to authorship and authorial rights that fit the characteristics of digital artifacts. Considering the situation described by Carpo, the principles and concepts of free & open source can be an inspiration to architects. Furthermore, we have seen that not only have the authorial concepts of free & open source – transformative, multiple and collaborative authorship combined with non-proprietary authorial approaches – emerged together with digital technologies, but so did new concepts, methods and modes of production – as Carpo posited, technological revolutions tend to have far-reaching effects.

1.6 Generalizing the Lessons From Free & Open Source

Over the last fifteen years, the interest in free & open source methods outside of the software sector has grown considerably. Although some people doubted that open source could cross the border from software development into other domains, these doubts have been disproven by practice. Today many people believe that free & open source approaches can work in other areas than software development. In a time in which the traditional authorial model in and of architecture is tottering, we may be would seem to be well-advised as architects to take a closer look at the areas in which free & open source methods have spread – so that we can assess whether we may want to join them.

Shirky (2005, p. 483) stated in *Perspectives on Free and Open Source Software* that "the patterns implicit in the production of Open Source software are more broadly applicable than many of us believed even five years ago." He goes on to argue that "[n]ow that we have identified Open Source as a pattern, and [are] armed with the analytical work appearing here and elsewhere, we can start asking ourselves where that pattern might be applied outside its original domain." Five years after Raymond (Raymond, 1999 but first presented in 1997, at the Linux Kongress Würzburg) presented his essay "The Cathedral and the Bazaar", Yochai Benkler (2002, p. 369) suggested that "the phenomenon has broad implications throughout the information, knowledge, and culture economy, well beyond software development" and introduced the more general term commons based peer production. Benkler describes "instances of peer production of content […] as parallels to peer production of software" and argues that the rise of digital means of communication and the declining cost for the production of sophisticated information give advantages to ways of organization over the internet, such as commons based peer production. Critical requirements for such communities to succeed, according to Benkler, are getting a sufficient number of contributors and managing to organize collaboration.

When it comes to modes of authorship and a mode of production that allows for variability that may be inspiring for architectural production, Carpo notes that just because something seemed unthinkable in the past, it still may become normal in some years. He notes that "we can already count plenty of instances where the new digital media are fast unmaking established traditions of authorship that, until a few years ago, would have been deemed indispensable—both intellectually and economically. Who could have anticipated the meteoric rise of a universal encyclopedia that has no author (because it has too many)?" (Carpo, 2011, p. 43) The so-called Web 2.0 has fostered a less consumerist, more engaged and more productive attitude toward the internet among many. Carpo (2011, p. 119) notes that "[t]he open source movement […] famously pioneered the […] principles of commonality and the spirit of collective creative endeavor now pervasive in the world of social media and user-generated content." However, we also need a critical approach towards the Web 2.0. Think of how we have grown dependent on particular platforms such as Google, Facebook, Twitter, etc., platforms that employ closed source programs about the inner workings of which we know little if anything. What we do know, of course, is that these platforms collect immense amounts of personal data. Yet, one may argue, that with the Web 2.0, a basic understanding of the power of online communities found its way into the mainstream. Now, this awareness of the potentials of online communities would seem to be a good basis for reaching out to new fields in which to apply free & open source methods.

When talking about applying open source strategies to new fields, we must keep in mind that a direct transfer of free & open source methods often does not yield satisfactory results and that some projects, because of their goals and their more general make-up, are more likely to succeed than others. Shirky puts this well:

> Open Source is not pixie dust, to be sprinkled at random, but if we concentrate on giving other sorts of work the characteristics of software production, Open Source methods are apt to be a much better fit. (Shirky, 2005, p. 5).
>
> Instead of asking "How can we apply Open Source methods to the rest of the world?" we can ask "How much of the rest of the world [can] be made to work like a software project?" (Shirky, 2005, p. 5).

The internet has changed the way software is written. Shirky (2005, p. 2) sees this as one of the key observations provided by Raymond (1999). The internet enabled asynchronous collaboration over great distances as well as the formation of deterritorial digital communities. This, as Raymond has shown, has fundamentally affected the way software is written. Today, with the ubiquity of the digital, it affects almost every part of every day life. Shirky (2005, p. 5) notes in this context that "[e]very time some pursuit or profession gets computerized, data begins to build up in digital form, and every time the computers holding that data are networked, that data can be traded, rated, and collated. The Open Source pattern, part collaborative creativity, part organizational style, and part manufacturing process, can take hold in these environments whenever users can read and contribute to the recipes on their own."

When Shirky wrote about the perspectives of open source outside the software industries in 2005, he mainly concentrated on projects dealing with forms of text. Today, 10 years later, the field has broadened. Open design and Open Source Hardware projects are now active parts of the Open landscape. Ideas to extend free & open source methods to hardware manufacturing already came up in the 1990s. In the late 1990s, even some licenses for, and definitions of, open source hardware were created but "[d]espite this initial burst of activity around the nascent concept of open source hardware, most of the initiatives […] faded out within a year or two and only by the mid 2000s would open source hardware again become a hub of activity. This was mostly due to the emergence of several major open source hardware projects and companies, such as OpenCores, RepRap, Arduino, Adafruit and SparkFun" (OSHWA, 2013a).

While the first activities in the open source hardware community focused on licenses and definitions, it is the mode of production that receives most attention today. Raasch et al. state that "the assumption that the OSS [open source software] model may be transferable to other industries usually refers not only to the fundamental licensing conditions, but also to […] practices and 'ingredients'" (Raasch, Herstatt, & Abdelkafi, 2008). Raasch et al. are specifically interested in come up with a generalization of free & open source approaches that is applicable to the production of tangible goods of every kind. For this they coin the term open source innovation (OSI), which they define as "an innovation, which is (1) generated through volunteer contributions and (2) characterized by a non-market transfer of knowledge between the actors involved in invention and those involved in exploitation. Actors involved in invention provide open access to their results for anyone wishing to exploit them, allowing utilization, modification, and re-distribution" (Raasch et al., 2008).

Introducing free & open source methods as useful methods for architects requires knowledge about the possibilities of applying these concepts in fields other than software. In the section above, an overview on the generalizability of free & open source strategies was given. Basically, in every profession that gets computerized, and starts using digital networks and digital data, free & open source practices can take hold – if users do have the possibility to contribute and to exchange, combine and adapt such data. As computerization required adaptations of work processes, in order to make full use of the possibilities of digital variability we need to shape our concepts and working processes so that they suit collaborative endeavors, thereby raising our chances to successfully apply free & open source methods.

1.7 Conclusion

In this part, we have seen that, with the advent of digital technologies and the digital turn in architecture, the Albertian conception of architecture no longer fits the ways in which we work and the same goes for our authorial concepts. We have seen that this became a topic for architectural theory not just recently with Carpo's book *The Alphabet and the Algorithm,* but already with Peter Eisenman, who understood early on that the digital made modern-age authorship a vulnerable concept. With digital media, the internet and free & open source culture, different approaches to authorship have gained ground. These approaches have in common that they give up the hard separation between mine and thine that is implied by traditional, modern authorship in order to allow for a culture of copying, remixing, adaption and open-ended variability. Discussing digital architecture together with authorship in the era of the digital paves the ground for thinking about free & open source concepts in architecture, for a discussion on authorship in the digital age is, per se, a discussion on transformative, multiple and collaborative authorship and non-proprietary authorial approaches, like the ones used by free & open source projects. Having seen that free & open source approaches, contrary to initial doubts, are used more and more in fields other than software production, the next topic I will talk about is how these methods are adopted in architecture. But as it has been said above, free & open source methods and concepts cannot be directly transferred from software production to other realms – some adaptations are needed, to our methods, to what we do and to how we understand ourselves. What I want to argue is that there are forms of design and ways in which we design that can foster collaborative action and, therefore, enable the adoption of free & open source approaches in architecture.

The topic of this thesis are conceptions of design processes and design, in the context of the assumption that free & open methods can be transferred to architecture even though some adaptions will be needed. The contribution of this part was to give an introduction to the topic by discussing how contemporary debates on digital architecture in architectural theory are connected to free & open source concepts. I hope I have made clear why adopting design and designing processes that allow for free & open source methods might be interesting for architects in the context of the contemporary technical developments. Furthermore, this parts provides some insights on the concepts, origins, applications and possible generalization of open source methods. Above all, this should enable us to identify success criteria for free & open source approaches in architecture.

2 Challenges in Implementing Free & Open Methods in Architecture

2.1 Introduction

Taking a look back to part 1, we will find that there are working methods, namely, the working methods of free & open source software production, that have grown out of the possibilities offered by the digital, above all: out of the variability of digital matter and the deterritorial collaboration that came with the internet. When we discuss how to apply open source strategies to architecture, we should be mindful of Shirky's advice (2005, p. 487) on the application of free & open source methods to fields other than software. On the one hand, he states that every pursuit, every field or profession that gets computerized holds in its hands the variability of the digital that open source methods build on, but, on the other hand, we have to be aware that not every process can be successfully open-sourced. First of all, open source needs adequate processes, which may still be in need of being created in architecture. This means that, for the transfer of open source strategies to architecture, some interpretative steps may need to be made and it may be necessary to think outside of the box. Second, we should bear in mind that creating, applying and establishing new methods in a field is always a process of trial and error. And, as Kaspori (2003) puts it: "[O]pen source is not a model to be developed and rolled out on a large scale. It must have a chance to evolve gradually. It entails an experimental process of adjustment".

In this part of my thesis, I will investigate challenges and topics that are of interest when free & open concepts are applied to architecture, and in particular when it comes to the question of how to adapt design and designing in order to suite free & open source methods. I do this by drawing on selected textual sources, that is, sources on introducing free & open source ideas and methods to architecture and sources on open design.[9] I will do so based on the assumption that where problems and challenges are not discussed as such explicitly, the very fact that something is seen as a topic worthy of discussion may indicate that this topic poses a challenge. Furthermore, I suggest that topics and challenges that require consideration in design in general, also require consideration in architecture, since in either case open source methods need to be applied, ultimately, to the production of tangible goods. This part of this thesis will identify such topics and challenges, providing a broad and structured collection of aspects and challenges that need to be considered when free & open concepts are applied to architecture. I will structure these aspects and challenges with the aim to identify those aspects and challenges that concern either design or the design process, for these are the aspects and challenges that I will use in my case study in the third part of this thesis. At the same time, this second part presents the first results of my research. A broad overview over possibly challenging aspects will benefit future free & open architecture projects, helping them to consider problems in theory – before being confronted with them in practice.

I will, of course, build on previous research on free & open source and architecture. However, since academic literature on applying free & open source to architecture is still scarce – it is, after all, a rather young topic –, I will also draw on experiences in the application of free & open source concepts to the production of tangible goods in general. I will give a short overview of these sources before I go on to extract from them the aspects and challenges that need to be considered when applying free & open methods to architecture.

9. I use the term "open design" in the way it has been defined by the Open Design Foundation (2000), following the Interpretation of Balka (2011), who notes that open design "describes open hardware as well as other physical objects being developed in accordance with the open source model. A multitude of open design projects has constituted itself, ranging from bicycles to microchips and from MP3 players to manufacturing equipment." Cf. also Vallance et al. (2001).

2.1.1 Sources Used

As stated at the end of part 1 of this thesis, applying free & open source methods in fields other than the production of software, requires a (re-)structuring of the respective subject matter of that field, so that it becomes more suitable for working with free & open source methods. If we aim to apply free & open source methods to architecture and to adjust the subject matter of architecture, then, so one of my main theses, we need to talk about design and the design process. And we need to identify the problems that we face when we try to mold design and the design process so that they become suitable for free & open source methods. As I have explained in part 0, I will analyze (a) articles on applying free & open source methods to architecture, (b) articles on open design and (c) documents from the field of open design in order to do so. The aim of this section is to give an overview over the selected articles and documents.

I will, given that we should not lose sight of the actual topic of this chapter, focus on contributions that, in the case of scientific or scholarly texts, either appear to me to have shaped the academic debate or, in the case of non academic texts, to have shaped either the academic or the public debate at large. I have ignored non academic texts if their arguments were too vague or too far fetched. I have also limited myself to contributions that are available in English, unless a contribution appeared to me to be of particular importance. I did not use a particular definition of "free & open architecture" to select my sources, simply because there is no agreed upon definition yet (I will review the contributions to the debate that try to define what "free & open source" could mean in architecture in the next section), so that settling for any particular definition would have risked to arbitrarily exclude important contributions.

Articles on free & open architecture:

Dennis Kaspori, "A Communism of Ideas. Towards an Architectural Open Source Practice" (Kaspori, 2003)

Usman Haque, "Distinguishing Concepts: Lexicons of Interactive Art and Architecture" (Haque, 2007)

Carlo Ratti et al., "Open Source Architecture (OSArc) – Op-Ed" (Ratti et al., 2011)

Andrea Rosada, "L'open source come modello di sviluppo dell'abitare contemporaneo" (Rosada, 2012)

Theodora Vardouli and Leah Buechley, "Open Source Architecture: An Exploration of Source Code and Access in Architectural Design" (Vardouli & Buechley, 2014)

Articles and documents on the field of open design:

Christina Raasch, Cornelius Herstatt and Kerstin Balka, "On the open design of tangible goods" (Raasch et al., 2009)

Christina Raasch, Cornelius Herstatt and Nizar Abdelkafi, "Creating Open Source Innovation: Outside the Software Industry" (Raasch et al., 2008)

Nizar Abdelkafi, Thorsten Blecker and Christina Raasch, "From open source in the digital to the physical world: a smooth transfer?" (Abdelkafi et al., 2009)

Documents from open design practice:

Open Source Ecology, "OSE Specifications" (OSE, 2013b)

The articles on the application of free & open source ideas and methods to architecture that I use in the following will be reviewed more extensively in the next section. The articles on open design that I have selected are extracts of the research conducted within the *Open Source Innovation* project, a project that, among other things, investigates "factors that drive or complicate the application of [Open Source Innovation] in different contexts" (Open Source Innovation Project, 2014). The term "open source innovation" here is used to refer to various kinds of innovation driven by free & open source projects regardless of their area of activity. More precisely, I have selected those articles that investigate open source innovation in the field of open design. I have also selected a document from open design practice, namely, the specifications that are used in the Open Source Ecology (OSE) project. I have added them since they give comprehensive insights into the conception of design and designing in one of the larger open design projects. Unfortunately, the field of open design is too vast for a more comprehensive selection of sources. I have chosen sources that should be representative, under the assumption that the topics and challenges that are discussed in those sources are broadly applicable and can be generalized.

This section listed the sources that I have selected for investigating possible challenges and topics of interest in the application of free & open source methods in architecture. All selected sources are text documents and will be analyzed in what follows, sorted by categories.

2.1.2 Excursus I: A Brief History of Free & Open Ideas in Architecture

The application of free & open source ideas to architecture is quite young a topic, hence there are, as of yet, no comprehensive overviews over its development. That being so, it will be helpful, before diving deeper into the actual subject matter of this chapter, to give a brief introduction into the debate by recapitulating the discussions from 2003 to 2014.

The first broadly visible appearance of free & open source as a topic in architecture is in 2003, when the Dutch magazine *Archis* publishes a special issue on open source (Archis, 2003). Within this issue, the article *"A Communism of Ideas: Towards an Architectural Open Source Practice"* by Daniel Kaspori is probably the most important one (Kaspori, 2003). In this article, Kaspori argues for an organizational renewal of architectural practice and a turn towards a more collaborative practice. Based on references from art and free & open source software production, he considers an open practice for architecture that embraces the idea of innovation, produced through distribution of ideas and broad active participation. In 2006, Cameron Sinclair (2006), co-founder of *Architecture for Humanity,* holds a talk at that year's official TED conference[10], where he explicitly connects the terms "open-source" and "architecture". Winning the 2006 TED Price, *Architecture for Humanity* founds the *Open Architecture Net-*

10. "TED (Technology, Entertainment, Design) is a global set of conferences run by the private non-profit Sapling Foundation, under the slogan 'Ideas Worth Spreading'. TED was founded in 1984 as a one-off event; the annual conference series began in 1990. TED's early emphasis was technology and design, consistent with its Silicon Valley origins, but it has since broadened its focus to include talks on many scientific, cultural, and academic topics." (Wikipedia contributors, 2015)

work, a platform allowing architects and activists to share architectural ideas, designs and plans. Also in 2006, on the *Game Set and Match* conference Ole Bouman, the chief editor of the magazine *Volume,* which is the successor of *Archis* since 2005, according to Usman Haque (2007) purports "that architecture has long been open source because buildings have always been constructed by borrowing technology and techniques developed by other designers and disciplines. […] Bouman described an open-source society as one 'where everybody grabs what they can' […] and portrayed the magazine Archis as open source because it redistributed recipes taken from the Internet within its pages." In 2007, Haque's article "Distinguishing Concepts: Lexicons of Interactive Art and Architecture" is published in *Architectural Design* (Haque, 2007). Haque observes that in interactive art and architecture, fields where concepts of different disciplines are mixed, terms are often used in a misleading way – terms that have become fashionable and ubiquitous are used without knowledge of the original concepts they represent. He explicitly mentions open source as one of these misused terms, referring to Bouman's talk at the *Game Set and Match* conference. Further, he points out that we need to have a discussion in architecture about the difference between the practice of copying/stealing ideas and the practice of sharing as well as about what open source techniques propose in matters of how we organize our work – contrasting the traditional top-down approach in architecture with open source approaches. In 2010, the agency Ecosistema Urbano (2010b) announces that the project *Air Tree Shanghai* (see figure 4), a pavilion at the Expo Shanghai representing the City of Madrid, is released under a Creative Commons[11] license – "[t]he project's construction plans and specifications were published under a Creative Commons license, making this project one of the first 'open source' contemporary architectural designs". In their Blog, Ecosistema Urbano (2010a, my trans.) states that "[e]l Árbol de Aire pasará a llamarse 'Air Tree Commons', puesto que a partir de hoy el proyecto entrará a formar parte del Procomún: cualquier persona, entidad o empresa podrá copiarlo, construirlo, venderlo y modificarlo en total libertad"/"The Air Tree will be renamed 'Air Tree Commons', from today on the project will be part of the common good: any person, entity or company may copy, build, sell and modify in total freedom." In the following year, in 2011, the Italian magazine *Domus* devotes its issue 948 (Domus,

11. Creative Commons is an NPO that provides standardized licenses for free cultural works. These licenses generally allow for people to use and redistribute work that is licensed under them, as long as they credit the original creator, though the exact terms of each license vary.

Figure 4: Madrid pavilion at the Expo Shanghai by Ecosistema Urbano (2010)

2011a) “to the open-source movement and to the activities of a number of theorists, designers, architects and inventors who embrace an entirely new, collaborative attitude towards authorship” (Domus, 2011b). In this issue of *Domus,* an op-ed with the title *Open Source Architecture (OSArc)* is published, stating that “Domus approached Carlo Ratti to write an op-ed on the theme of open-source architecture. He responded with an unusual suggestion: why not write it collaboratively, as an open-source document? Within a few hours a page was started on Wikipedia, and an invitation sent to an initial network of contributors. The outcome of this collaborative effort is presented below. The article is a capture of the text as of 11 May 2011” (Ratti et al., 2011).[12] The article is a collection of topics that the authors associate with the term “open source” and presents an attempt to outline what open source architecture could be. The authors call open source architecture an “emerging paradigm” and a “proposition for a different approach […] to succeed the single-author model [that] includes tools from disparate sources to create new paradigms for thinking and building” (Ratti et al., 2011). In 2012, Andrea Rosada (2012) presented the article “L’open source come modello di sviluppo dell’abitare contemporaneo”/“Open source as development model for contemporary living” (my trans.) at the international workshop *Inhabiting the new/inhabiting again in times of crisis* at Naples. He presents a collection of selected projects and initiatives that he sees as relevant when talking about open source architecture and thereby gives a good overview of what – in 2012 – is seen to be, meant to be or simply called by the name of open source in architecture and construction. Moreover, he gives a first analytic approach to a possible definition of open source architecture by identifying three main directions of action found within the projects. One of the projects he mentions is presented one year later at the 2013 official TED conference by Alastair Parvin (2013): The WikiHouse project, which was founded in 2011 and is described by Parvin as an open-source construction system (see figure 5 for an early prototype). The WikiHouse project collects 3D models that, one the one hand, can be assembled into simple buildings as if they were IKEA kits and, on the other hand, can be cut out from digital plans. What is more, these 3D models, these plans can be created by everybody, are free to be adapted, and are shared via an online platform under Creative Commons licenses. A further con-

12. Surprisingly, when looking at the Open-source architecture article’s revision history on Wikipedia, one will find that the page was created on the 18th of May 2011. I did not investigate this further, however, since it does not contribute to my research question.

Figure 5: Early WikiHouse prototype by Team 00 in London (Team 00, 2011)

tribution to the subject of defining what open source architecture is and what "open source" means in architecture is the article "Open Source Architecture: An Exploration of Source Code and Access in Architectural Design" by Theodora Vardouli and Leah Buechley (2014), published in *Leonardo* in 2014. Based on the assumption that while open source found its way into the architectural discourse, there is no consensus yet about what that term and related terms actually mean. The article discusses possible interpretations and translations of the terms "access" and "source code" when these concepts are transferred from the digital realm of free & open source software to architectural design and the production of tangible artifacts. Furthermore, they also touch upon how the free & open source method might be transferred from the digital to the physical world.

This concludes my brief overview of the history of applying free & open source topics to architecture from 2003 to 2014. For the interested reader it shall be mentioned that most of these sources are available online and that Carlo Ratti has just recently, in June 2015, published another book, *Open Source Architecture*, which could not be reviewed here, since the review process for this thesis had already been finished by that time.

In the section above, I have described the state of the debate and the practices of applying free & open ideas to architecture as of 2014 (newer publications cannot be considered in this thesis). We should now have a basic knowledge about the debate as well as about practical attempts to transfer and apply free & open ideas to architecture. Further, we have seen that there is a growing interest in the topic – two well known architectural magazines published special issues on open source. I will now move on to give a brief overview over what "free & open source" is taken to mean in architecture.

2.1.3 Excursus II: Attempts to Define "Free & Open Source" For and In Architecture

Having given you an overview of the history of the debate on free & open source architecture, I will now focus on discussions of what "free & open source architecture" may mean. I should note that my own understanding of "free & open source architecture" or "open architecture", that is, the understanding on which I have drawn to select the WikiHouse project for my case study, is independent from the definitions that I will discuss here, even though it is of course informed by them. I should also remind you that my selection of sources was *not* guided by any particular definition of "free & open source architecture" or "open architecture," but deliberately left open, accepting the self-proclamations of the respective authors.

When talking about adopting free & open source methods in architecture, the term used most often is "open source architecture". I nevertheless have decided to use the terms "free & open source architecture" or "free & open architecture" instead, since I want to reference the free & open source software movement with its two major actors, the Free Software Foundation and the Open Source Initiative. Because there are ideological disputes between those two actors, disputes that cannot easily be resolved, the term Free and Open Source Software (FOSS) was coined in software production. Today, another term has become popular, namely, "Free, Libre and Open Source Software" (FLOSS); this term is usually used to emphasize that free software is "'free' as in 'free speech', not as in 'free beer'" (Free Software Foundation, 2015). I will stick with the term "free & open" in this thesis, save for citations.

As already mentioned in the previous section of this chapter, there is still no agreed upon definition for free & open architecture. The definition given by Ratti et al. was a first attempt to outline what the term "open source architecture" might mean. However, what we are confronted with by Ratti et al. is a highly associative mix of topics that is more of a brainstorming than a definition, especially when Ratti et al. mention referential inputs and when it comes to the question of what open source architecture means in practice. Moreover, the picture they draw bears little if any connection to the definitions of free & open source given in the free & open software (Free Software Foundation, 2013; Open Source Initiative, 2002), as Rosada (2012) points out. Yet, among all the things that Ratti at al. mention, one may also find elements of free & open source principles; for example, they mention (2011) that open source architecture relies on a digital commons. Yet other statements appear to conform with free & open principles at first sight, but turn out to be insufficient upon close examination; for example "is typically democratic, enshrining principles of open access and participation" (Ratti et al., 2011) That there is a digital commons and this commons can be accessed without restrictions ("open access") does not, by itself, imply that there are source files that are available in a useful (and usable) file format and that can be edited. That said, Ratti et al. have, with their associative approach, opened the field for a discussion on what actually could be a free & open source architecture.

Rosada (2012) approaches the problem of definition by observing practice. By analyzing approaches of different initiatives that have, according to their self-understanding, committed themselves to open source architecture, he comes up with a snapshot of the self-proclaimed open source architecture scene at the time he wrote his article. Based on this analysis, he defines three categories. These categories evince different interpretations of the meaning of "open source" in architecture. Those three categories are: firstly, building concepts that focus on the concept of the *open building*, which, among other things, is about functional openness and flexibility of built structures; secondly, *platform approaches,* which focus on the *spread of free knowledge* and project ideas, for example, project collections; and thirdly, planning *tool kits* and exchange platforms, which develop construction concepts that are *free to use* and target the *production of design families* and inter-compatible artifacts (cf. Rosada, 2012).

Vardouli and Buechley are the first who actually focus on what open source terminology could mean in architecture. In view of the principles articulated by the FOSS movement, they explore different literal and metaphorical interpretations of the terms "source code" and "access" in architecture. Literally translated, the source code in architecture would be "the digital files that encode information on built artifacts" (Vardouli & Buechley, 2014, p. 53). But, so they argue, the source code is part of a process, the process of transforming that very code (the source) into the actual software (the product). We should consider this when we are looking for its equivalent in architecture, hence we need to discuss how information is transformed into a tangible good. Unlike compilation, that is, the transformation, of code into software, the transformation of designs into buildings is an ambiguous process, Vardouli and Buechley argue. They approach the term "access" via the debate on the democratization of design and the integration of laypeople in the designing process. They mention the historical and contemporary ideas on computer-aided participatory design support, but find that that those proposals are lacking an openness in the sense of open source. They also, similarly to Haque, mention that the keenness of many architects to find inspirations in other fields carries a certain risk that the concepts of those fields are misinterpreted. Vardouli and Buechley then argue that when we transfer terms and concepts, we need to discuss their meaning, for, on the one hand, there is no direct translation for every concepts and, on the other hand, terms may already be in use – but with a different meaning.

We have learned that there is no agreement on what a free & open source architecture actually is. However, there is a debate on what the terminology of free & open source software may mean when its applied in architecture. This I take as an indication that the translation and the transfer of free & open source methods from the realm of software to that of architecture is not without its difficulties.

2.1.4 How Aspects and Challenges Were Extracted and Categorized

In the previous sections, I have introduced the sources that I use in this part of my thesis. Moreover, I have given a brief overview over debates on practical attempts to apply free & open source concepts in architecture. The aim of this thesis is to gain a deeper understanding on how to best apply free & open source methods in architecture. So, taking for granted that (a) open source methods can take hold in every field that gets gets computerized (cf. this thesis, part 1; Shirky, 2005), (b) introducing free & open source methods to architecture is reasonable, in light of recent technological developments and their impact on how we do architecture (cf. this thesis, part 1), and (c) we need to consider reshaping our conception of the design process and designs in order to apply free & open source concepts in architecture successfully (cf. this thesis, part 1; Shirky, 2005). Investigating how we may transfer free & open methods to architecture, in particular, which challenges we will face in doing so, is an important contribution to the contemporary debate. This second part of my thesis focuses on identifying such challenges. This is done by analyzing the selected sources and structuring the results. This section describes the methods used in analyzing these sources and in structuring the results. In the following, I firstly describe my methodical approach in analyzing the sources, and secondly explain how I structure the results of this analysis.

The sources that I use in this second part of my thesis are all textual sources. My basic method is textual interpretation. I have divided my sources into three categories, each of which I approach differently. The first category are the aforementioned articles on open architecture. They are analyzed with the aim to identify topics that are brought up in the context of open source in architecture. The units of investigation are text passages that cope with architecture and free & open source or with the architectural branch, its actors and free & open source. The topics that were mentioned in those passages were collected. Passages that deal with urbanism, urban open space and landscape planing were not taken into account. The second category consists of the aforementioned articles on open design and on the transfer of free & open source concepts to the production of tangible goods in general. These focus on difficulties in transfer. Hence, my analysis and my collection of challenges and topics is focused on aspects that are mentioned as critical, on problems, relevant differences and challenges in that transfer. The third category of sources are the aforementioned guidelines from open design practice. These guidelines are analyzed with a focus on points that indicate designing approaches that are a result of using free & open source methods. The points identified as doing so are collected. I hope to thereby arrive at a broad collection of topics and challenges that are of relevance for the application of free & open source methods in architecture.

Doing so, I have found a vast landscape of topics. Trying to sort, structure and connect them was by no means an easy task. The resulting collection is a list of more or less precise statements. Since nine different sources are analyzed it is to be expected that the points that they mention will overlap or represent different aspects of the same topic.

The result of my analysis is a collection of points sorted by origin. These are then categorized, using categories that have been established in previous research on free & open source methods as well as categories that have been construed with the aims of: one, highlighting aspects and challenges that are pertinent to design and the design process, which according to my main thesis is the relevant subject matter to be adapted in architecture; and two, providing a foundation for the case study in part 3 of this thesis.

The categories that have already be established are those employed by Lessig (1999) and adopted by Braun (2007) and Raasch et. al. (2008): *economic aspects, technical aspects, legal aspects* and *social aspects*. Adopting these basic categories facilitates comparing and contrasting this work with former research. As not all aspects found in this analysis are covered by this framework, further categories are introduced for this study. These categories result form a thematic clustering of the points not yet assigned to a category. Moreover, the categories *cultural aspects* and *definitorial aspects* quickly became apparent; to avoid confusion, by "social aspects" I mean aspects that pertain to patters of interaction, norms, and values of particular projects, whereas by "cultural aspects" I mean those aspects that pertain to patterns of interaction, norms, and values of particular projects of whole disciplines or fields. In defining further categories several attempts were needed. Due to multiple interrelationships, aspects were hard to disentangle and multiple options for categorizing needed to be tested. Finally the categories *tools and technical infrastructure*, *information transfer and legibility*, *the process, complexity, modularity, automation, fabrication,* and *the artifact* were found to be the most suitable with respect to my research objective. I have adopted the category *tools and technical infrastructure* from Raasch et al. (2009), who speak of "technical prerequisites" and "technical aspects," which however struck me as too broad. The remaining categories are informed by those used by Georg von Krogh and Eric von Hippel (2006, p. 977), which however do not concern design, and those used by Andrea Rosada (2012, cf. above), which however are also to broad for this study. Unfortunately, many topics and challenges that are discussed in my sources are cross-cutting concerns that do not fit neatly into broader categories (e.g., modularity applies to the design process as well as to fabrication), hence it was impossible to keep the number of categories smaller. In brief the categories used to structure the collection of topics and challenges are the following: (a) *aspects of definition*, (b) *economic aspects*, (c) *legal aspects*, (d) *social aspects*, (e) *cultural aspects*, (f) *tools and technical infrastructure*, (g) *information transfer and legibility*, (h) *the process*, (i) *complexity*, (j) *modularity*, (k) *automation*, (l) *fabrication*, and (m) *the artifact*.

Given the objective of this thesis, my focus lies on rethinking and adapting conceptions of design and the design process. Thus, categories were sorted in respect to their impact on the design processes as well as on the conception of design. The points collected in the categories (f) to (m) where found to be those with major influence on the design processes and the conception of design. Presenting them is the main focus of this part. They also are the basis for the case study conducted in part 3. The categories (a) to (e) are considered to contain points of secondary interest. What is more, some of the points discussed in categories (b), (c) and (d) appear to be well established and are widely discussed in the literature on free & open software. Put simply, they seem to be quite similar for the vast majority of FOS projects. Hence, the categories (b), (c), (d) and (e) are presented in short form in the section *Economical, Legal, Social and Cultural Aspects.* The findings in category (a) have already been presented – they are the foundation of the section on the definition of free & open source architecture. Before moving on, I should mention that some of the points collected are neither discussed nor considered in part 3, since they either did not fit any category or were too vague.

2.2 Economical, Legal, Social and Cultural Aspects

In the previous section the textual sources that are analyzed in this part of my thesis have been introduced, and we have seen how the results of this analysis have been categorized. In the following sections, the results of my analysis are presented.

In this section I give an overview over the categories (b) to (e), that is, *economical*, *legal*, *social* and *cultural aspects.* As mentioned previously, these categories will not be used in the case study in part 3 of this thesis. But since the aim of this second part is not only to identify a set of criteria for the third part, but also to broadly map the topics that are commonly discussed, these categories are discussed here. Moreover, they provide interesting insights into the challenges that projects trying to apply free & open source methods in architecture are likely to face, even though these challenges do not pertain to design or the design process. I will discuss these categories in the order in which they are given in the title.

The challenges that can be subsumed under "economic aspects" are, to a large extent, already known from free & open source software projects, however some of them are specific to open design. These latter challenges can be divided into those *with* and those *without* relation to the physical nature of the products of open design. Furthermore, since economic concerns are closely related to the feasibility of projects, I include feasibility aspects here. While most of the points that have been categorized as economic aspects have been found in the sources on open design, only funding was mentioned in the sources on free & open architecture.

I start with an aspect that is well known from software development, the *economics of motivation*. What may motivate potential contributors to actually contribute to a project is a recurring theme in the given sources. Potential contributors, so those sources assume, are, knowingly or not, guided by economic considerations of cost and benefits. They also, again according to my sources, consider possible gains and possible risks of sharing, for example, the risk of openness, noting that "[c]ommercial contributors are more likely to participate in OSI projects not closely related to their competitive advantage" (Raasch et al., 2008), and the risk of project failure, nothing that "particularly projects with commercial contributors, are more likely to occur when third parties set the rules and bear part of the cost" (Raasch et al., 2008). A further point Raasch et al. have found to be relevant for some projects is the regulation of access, especially in early stages and if commercial contributors are involved. They point out that it may happen that potential participants "refuse to contribute, e.g. for fear of competitors obtaining crucial knowledge at an early stage" (Raasch et al., 2008). These aspects mentioned by Raasch et al. are mainly affecting projects dealing with commercial actors due to their typically profit-oriented approach to gains and losses (Raasch et al., 2008). For individual contributors, by contrasts, those benefits are not necessarily monetary. But they also note that potential contributors are, in general, only motivated to contribute if the "perceived benefits more than outweigh the costs of contributing" (Raasch et al., 2008). The economics of motivation affect free & open projects in general – regardless of whether they produce digital or tangible goods. For a broader discussion of the economic aspects of free & open source software see Hippel (2005) and Weber (2004).

It is crucial for free & open projects to attract a sufficient number of contributors, because without enough people to join the effort, many projects simply are not feasible. Project feasibility is not only related to a project's ability to attract single contributors, but also to the *absolute number of potential contributors*, that is, to the size of the pool of people that can be attracted. Raasch et al. argue that this number of potential contributors is dependent on the *field of activity* and that a project is more likely to succeed when knowledge about the artifact under development is widely dispersed, because this allows for more potential contributors (Raasch et al., 2008). Another point related to the field of activity is mentioned by Abdelkafi et al. (2009), namely, that in free & open projects a "getting things done" approach is much more popular than deadlines. Abdelkafi et al., referring to interviews that they conducted, note that their "respondents find it difficult to stick to time plans; in fact, some of the projects did not even develop a schedule" (Abdelkafi et al., 2009, p. 1622). In some fields of activity, especially in fast-moving ones, where high market-responsiveness is required, this *slow pace of development* is seen as problematic. Abdelkafi et al. regard this as relevant first and foremost for hardware projects, but, it seems to me, that this affects software projects as well.

While the points mentioned above are about contributors and motivation, the following focuses on monetary aspects. Abdelkafi et al. note that one viable business concept in free & open source software is providing services around a product, while the product itself often is free of charge. This is possible due to replication costs being virtually equal to zero. Abdelkafi et al. (2009), as a thought experiment, transfer this concept to the production of tangible goods. The question asked is whether it is possible to give tangible products away for free. They suppose that "[t]his depends on the fact if the revenues over the product lifecycle compensate initial product costs. Furthermore, at the start of the business capital is necessary to bridge the first period of no revenues, when products are manufactured and shipped free of charge" (Abdelkafi et al., 2009). However, in my experience, tangible goods are not given away for free in practice, and Abdelkafi et al. also point out later that "[i]n the physical realm, however, a price is charged […]" (Abdelkafi et al., 2009). Maybe people do not expect to get tangible products for free – but they do expect price transparency. The *cost structure of production* of tangible products clearly differs from that of digital products. Abdelkafi et al. (2009) note that, for software, the costs of development are very high, while the marginal costs, that is, the costs for every subsequent copy of the software, are extremely low. "Physical products are different in this regard. While there is also high up-front investment and high design cost, marginal costs still represent a substantial portion of total costs, in spite of scale economies and learning curve effects", (Abdelkafi et al., 2009). Raasch et al. (2008) also describe the *costs of transforming information into a tangible good* (prototyping and replication) as a critical aspect in open design. "To run, test, and debug a software application, developers need only a computer and a compiler. Building and testing product prototypes, however, may be rather costly" (Raasch et al., 2008). How to acquire the necessary funding is a topic for free & open source projects in general, but due to the fact that open design projects develop tangible goods, they tend to need more funding, hence the acquisition of sufficient funds tends to be a greater challenge. Ratti et al. (2011) draw attention to the internet based financing models that have become popular in recent years. "New economic models, exemplified by incremental microdonations and crowd-funding strategies like Sponsume and Kickstarter, offer new modes of project initiation and development […]" (Ratti et al., 2011). They even conjecture that crow-funding strategies may fundamentally change the way in which building projects are financed.

I should note here that all challenges and aspects that are connected with the conception of design can be condensed into the challenge of how to cover the costs of transforming information into a tangible artifact. That being so, economic aspects will recur in my discussion of challenges in fabrication and challenges in the conception of the actual, tangible artifact, however they will only be touched upon briefly.

We now will look at the *legal aspects*. These are not discussed in my sources on free & open architecture, all of the following points are taken from the articles on open design.

Referring to legal matters Raasch et al. (2008) see a striking difference between software and hardware production. There are well established and useful licensing schemes for free & open source software, whereas this is not the case for free & open source projects that try to develop tangible goods. Furthermore, free & open source software projects usually only have to cope with copyright law, whereas free & open source projects that try to develop and produce tangible goods have to deal with copyright law *and* patent law (Raasch et al., 2008).[13] Abdelkafi et al. (2009) also mention patents, but find that they are seen as disadvantageous in software production[14] and may be seen as less problematic when it comes to the production of tangible goods. Abdelkafi et al. (2009) do not elaborate this more precisely.[15]

There are no established free & open licensing schemes for tangible goods yet. As Abdelkafi et al. note, these are still under development and different approaches are being discussed; the most important inputs, in my opinion, have been given by the Open Source Hardware Association (OSHWA, 2014), the Open Source Hardware and Design Alliance (OHANDA, 2009), the CERN – European Organization for Nuclear Research (CERN, 2011) and the TAPR – Tucson Amateur Packet Radio (TAPR, 2007) Abdelkafi et al. (2009) note that licensing schemes for the development and the pro-

13. This finding of Raasch et al. stands in need of further comments, for FOSS projects also have to deal with patents, at least at first sight. However, whether patents can, in fact, be granted in the US has always been somewhat unclear, even though such patents have been granted in the past, and more recent court rulings, namely, *Alice Corp. v. CLS Bank International* (134 S. Ct. 2347, 2014) and *Allvoice Developments US, LLC v. Microsoft Corp.* (Fed. Cir. May 22, 2015) would seem to suggest that software is only patentable under certain conditions. EU law is not much clearer, allowing patents for computer-aided inventions but not for software "as such" (see Article 52 of the European Patent Convention). What is more, software patents are opposed by significant (though by no means all) parts of the industry. That being so, software patents may be less of a concern for FOSS projects than patents in general for projects that try to produce tangible goods, but this is just a conjecture.

14. This is the position of Abdelkafi et al. However, although the view that software patents impede development is wide-spread in the FOSS movement and among the industry at large, some (though by no means all) software companies, particularly larger ones, see this differently.

15. I find the positions of Raasch et al. (2008) and Abdelkafi et al. (2009) difficult to reconcile. However, Bessen and Meurer (2007) have studied the impact of patents and have found that patents are a significant obstacle to developing new products in almost all fields of activity, though software patents appear to be particularly troublesome.

duction of tangible goods are a very complex topic, pointing out that, for example, Markus Merz, the initiator of the OScar (open source car) project, finds that "it is often difficult to determine, whether the license should exclude product commercialization or not".[16] Raasch et al. (2008) note that "if an industrial firm freely reveals its designs and initiates an open source project, competitors may use the project outcome to produce and sell goods, while making profits. Competitors cannot be prevented from marketing these products and asking for a price".

Another legal aspect that is often discussed are warranties. Abdelkafi et al. (2009) see a major difference between software and tangible goods in matters of lifetime and the possibility of fixing defects. Warranty services and replacement of tangible goods may cause additional costs that may be hard to handle for free & open source projects. By contrast, defects in software products can be fixed via patches, which can simply be downloaded from the internet. For hardware, every single unit has to be repaired or replaced.

The legal aspects mentioned, similarly to the economic aspects, for the better part do not influence the conception of design directly. Just some of the questions that are raised by the qualities of physical objects are relevant for the conception of design. Patents and licensing schemes are relevant only in deciding whether particular pre-existing designs or design patterns may be reused by any specific project or whether other projects are allowed to reuse designs or design patterns that have been developed in that project, and though these are issues of great importance, they do not concern the concept of the design itself. Hence, patents and licensing schemes are not part of my research objective. The only legal aspect that intersects with design conception aspects are warranties, because the lifetime of the product and the ways in which defects can be avoided or at least handled has to be kept in mind during the design process.

However, this is just to say that the quality of the product has to be kept in mind – which it would have to be at any rate. And since warranties raise no relevant challenges apart from those connected to the quality of the product, they will not be investigated further in this thesis.

Social aspects are mainly about how to build and sustain communities. Hence, the aspects that have been identified are not particular to the production of hardware, but

16. Unfortunately, they do not quote him verbatim, and the website to which they refer as a source for Merz' statement is no longer online.

rather affect all kinds of free & open source projects, regardless of whether they produce digital or tangible goods. Most of the points that have been categorized under social aspects where found in articles on open design. Community building is the only point that is also mentioned in articles on free & open architecture.

Kaspori (Kaspori, 2003) points out the essential requirements for community building, when he talks about the consequences of open source for architectural practice. He opposes the predominant hierarchical model with the free & open source model of an open community, which he, with reference to Raymond, calls the bazaar model. "The sole requirement for this [the bazaar or free & open source] type of cooperation is the same as for all other types of community, namely a shared interest. That interest leads to knowledge being shared between different disciplines and also between professionals and hobbyists. The identification of this user base is accordingly an important step in the development of an open-source architectural practice" (Kaspori, 2003). Further, Kaspori cites Raymond (Raymond, 1999), who wrote: "It [the software] can be crude, buggy, incomplete and poorly documented. What it must not fail to do is convince potential co-developers that it can be evolved into something really neat in the foreseeable future".

The ability to attract new and hold existing contributors is critical to all free & open source projects, regardless of whether they produce digital or tangible goods. Following Raasch et al. (2008) "any OSI [Open Source Innovation] project must be able to draw on a sufficient number of potential contributors who not only have access to the knowledge and equipment required to participate, but also have the motives and interests the project appeals to". There are many free & open source projects which, due to little participation, never take off. However, a general statement on how many contributors a projects needs cannot be made, since this number differs from project to project (Raasch et al., 2008).

Motivational aspect are important for community building. Raasch et a. (2008) argue that basic designs, which are ready for others to build upon them, are an important motivational factor for new projects in order to attract contributors. Abdelkafi et al. (2009) find that a basic design, ideally one that is already marketable, is of crucial importance, at least for profit-driven projects. For non-profit projects they note that a non-marketable prototypes or even just a list of requirements, just so that there is some guidance for future development, may be sufficient.

However, especially for community-based projects, it is not only import to involve new members, but also to keep those that they already have. In open communities people can always leave – and they will, if they are not happy with how things are done or feel that things are done in a non-transparent way. Of course, loosing contributors due to frustrating dynamics within the community is something most projects seek to avoid. Unfortunately, this is easier said than done. It is a challenge for most free & open projects to develop a workable governance model and to reach social agreements that are acceptable to most of their members, especially once a project reaches a certain size. The social dynamics and the social structures of a project are, besides personal interests, the decisive factor for whether people commit to contributing to a project long term. Raasch et al. note that in matters of governance, free & open source software communities strongly dislike outside influence, that is, influence by people who are not members of the project community. If a project is not self-governed, members are much more motivated if there is at least a perspective for the project to become self-governed in the future (Raasch et al., 2008). See Fogel (2005) for a detailed description of governance models in free & open source projects. Furthermore, free & open source projects, like almost all groups have their unique social rules. Just letting those rules develop over time often causes problems, for the social dynamics of a group are hard to change once they have been established and non-transparent structures make them difficult to understand for people who join the group later on. Transparent rules of social interaction, communication and conflict resolution as well as transparent power structures are therefore of crucial importance for groups that want to attract new contributors. That is why free & open source projects usually install a written down code of conduct, including rules and mechanisms for conflict resolution, and often even a coordinator or group which is responsible for general coordination and matters of general project direction (Raasch et al., 2008). Again, see Fogel (2005) on the political infrastructures within free & open source software projects.

The last social aspect that is mentioned in my sources is how work is distributed. Free & open source communities are usually based on the work of volunteers and have developed a system of self-assignment to distribute open tasks. In this context, Raasch et al. (2008) mention the problem of uninteresting tasks that no one wants to take care of, that nobody wants to assign to themselves. This problem is often mentioned in literature for all kind of communities that work with concepts of self-organization and self-assignment. Once more, see Fogel (2005) for reports on practical experiences.

The social aspects mentioned in the sources examined revolve around community, community building, and participation and are all known from software development, where they have been discussed extensively. A social aspect that might be more interesting to discuss when talking about how to apply free & open source concepts to the production of tangible artifacts and to architecture is how regional “offline” communities impact the social interaction in an online community. However, this is not within the scope of this thesis. When looking at the social aspects from the viewpoint of the conception of design and the design process there is but one point that directly touches matters of design: the initial design that may foster the motivation to contribute to the project. But since the need for an initial design does not have any implications on how to design, this is of no further interest either. However, I would like to add here that the motivation to contribute can not only be fostered by providing an initial design, but also by the way the design is conceptualized. Put simply, the motivation to take care of a task may also depend on the absolute amount of work that is required to finish it. Small tasks that can be finished relatively quickly require far less motivation than large ones that are fairly time-consuming. And how tasks can be broken down so that each of them each of them appears doable is a question for design and its conception. Further, I would like to mention a method that is common in software production, has motivational impact and might well suit the way architects are trained to work: so-called hackathons or, more formally, code sprints. A code sprint is gathering of programmers over a short period of time, usually one to three weeks, during which the programmers focus on the tasks at hand. I have the impression that architects also like to do their work in short but intense phases, or are at least trained to work in this manner, so sprints may be an interesting method for free & open source architecture projects. However, this is also beyond the scope of my thesis.

The last category in this section is that of *cultural aspects*. Knowledge about free & open source culture, about the way things are done, is an important topic when considering to how to apply free & open source methods in architecture. Cultural knowledge is also mentioned as a challenge in my sources on free & open architecture as well as in those on open design in general.

Abdelkafi et al. note that the fields of software and tangible design differ therein whether there is a shared culture of the field. In software production, there is a shared culture, which includes an ethos of free & open source. Abdelkafi et al. "[…] believe that the absence of such a culture promoting openness and free revealing of ideas makes the transfer of the open source principles to the physical realm less evident" (Abdelkafi et al., 2009).

Kaspori (2003) also points out that it may be challenging to establish a shared notion of a more collaborative practice, that is, of a working method that differs from traditional top-down approaches. In software production, such approaches, are called the "cathedral model". This model stands for top-down hierarchy, "closed-ness" and non-collaborating competition. Kaspori states that this form of competition has a strong tradition in architecture and is also a proven generator of innovation, but also leads to enormous fragmentation. The "bazaar model", as it is called in software production, by contrast, stands for the way most open source software is produced, through open collaboration, where innovation is generated by the free distribution of knowledge and testing ideas in different situations. This approach emphasizes continuous development and improvement (Kaspori, 2003). Kaspori thinks that this approach may be an attractive model for architectural and spacial planning, but also notes that "the idea of a collaborative practice presupposes a complete reversal of the existing organizational model of a discipline that is very keen on its autonomy and the concept of copyright" and that "[o]pen source is […] a turn-around in thinking about the fundamental organizational principles of architectural practice" (Kaspori, 2003). Furthermore, you will remember, he finds that "open source is not a model to be developed and rolled out on a large scale. It must have a chance to evolve gradually. It entails an experimental process of adjustment" (Kaspori, 2003). A similar picture is drawn by Haque (2007), who contrasts open source techniques with traditional top-down hierarchy. "Several designers and researchers have been particularly interested in how these concepts might be applied to the field […] of architecture. There are problems with such a translation, but it does seem that the collaborative means of production offered by an open-source approach might have much to contribute to a discipline that is known, particularly in the West, for its top-down authoritarian approach" (Haque, 2007).

Furthermore, Haque mentions a lack of awareness of the differences between copying, stealing and sharing among architects. He (2007) also notes a widespread belief that I have often been confronted with myself when talking with other architects about architecture and free & open source principles, namely, that architecture were open sourced already because of the extensive copying practiced in the branch. This reflects a very vague and blurry picture of what free & open source principles stand for and a very self-indulgent position that, as Haque (2007) notes, "diverts us from exploring a radically different means of architectural production, one that is explicitly designed for sharing with others – the most exciting notion behind open source in the first place". He illustrates this with a quote from Ole Bouman who "suggested […] that architecture has long been open source because buildings have always been constructed by borrowing technology and techniques developed by other designers and disciplines. […] and described an open-source society as one 'where everybody grabs what they can' […]" (Haque, 2007). Haque (2007) describes that as an example of a point of view that fosters "the inversion […] of the productive features of open source [in architecture]".

Cultural aspects are important when talking about free & open source methods in architecture because a lack of knowledge on free & open methods and their culture has already lead to misunderstandings and misinterpretations. Having said that, while clarifying basic notions of free & open source and its methods would be important, this would not broaden our understanding of how design needs to be conceptualized in order to be suitable for such organizational strategies. Therefore, the cultural aspects and challenges will not be explored further in this thesis.

As already hinted at earlier, the economical, legal, social and cultural aspects have, in conclusion, been found to be only marginally relevant to the conception of design and the design process. Hence, they will not be used in the case study in part 3. Of course, we have seen that many of the points that have been discussed in this section should nonetheless be given some thought when conceptualizing design and design processes, to wit: (a) the relationship between transforming information into a tangible artifact and the costs a project has to bear, (b) the relationship between warranties and product lifetime as well as repair costs and (c) how initial designs and other organizational decisions in the conception of the design and design process affect the motivation of potential contributors.

Furthermore, it can be noted that some points and challenges mentioned in this section only affect free & open architecture or open design projects, but not free &

open source software projects, whereas others affect free & open software projects as well. The cultural aspects that have been discussed in this section affect projects that introduce free & open concepts to the physical realm in particular ways. This is because free & open source culture is well established in the field of software production, so that most contributors have basic knowledge of how free & open source culture works and how free & open source software is produced. This cannot be said for free & open architecture or open design, at least not yet. A similar situation we find in legal matters. The legal situation is relatively clear in software production, legal concepts have been tested and established. Again, this is not the case for the development and design of tangible goods. By contrast, some economic aspects concern projects that produce tangible goods in particular or at least to a greater extent than those producing digital ones, but others affect both kinds of projects in the same manner. Costs that arise because of the tangible nature of goods (production, distribution and repair costs) as a matter of course only affect projects that attempt to produce tangible goods. What is needed to offer potential contributors a favorable cost-benefits ratio is something all projects need to figure out. Finally, the social aspects that have been discussed, that is, how to build and sustain a community, are important for all free & open projects alike.

In this section an overview over economical, legal, social and cultural topics and challenges has been given. However, these challenges have been found to be at best of minor importance for the conceptualization of design and design processes. Of course, projects should keep them in mind nonetheless, if they are considering to apply free & open concepts to architecture. The remainder of this part of my thesis focuses on topics that evolve around and have high impact on design conceptions and the design process.

2.3 Tools and Technical Infrastructure

When we think about designing, we must not forget to also think about the tools that enable us to design in the ways we do. During my studies, there was a thought-provoking sentence written on the wall of studio I worked in, at least it was thought-provoking for me, though not because I agreed. Freely translated, It said: “The [only] limits of my architecture are the limits of my creativity." That sentence engraved itself into my memory – together with a strong feeling that this is not the whole truth. The architect is always bound to its tools. The drawing board and the Albertian conception of architecture correspond to an architecture that could be drawn in two dimensions. The burst of free forms in architecture only came with the rise of the virtual three dimensional space. However, not only the creation of forms is strongly connected to its tools, the ways in which we work together are as well. Working together is not only a matter of will, but also one of tools and techniques. Tools are often at the same time the precondition for as well as the result of particular styles of work and/or modes of collaboration. For the free & open source software movement it can be said that with the working methods of free & open source, a group of tools that support and foster a collaborative work that is based on a shared commons grew simultaneously.

This section illustrates what is said about the topic of tools in the context of free & open architecture and free & open digital design collaboration, with the aim of sketching challenges and bringing up points of discussion.

In matters of tools and technical infrastructure the following aspects and topics are mentioned in my sources: (a) *editing tools*, (b) *collaboration tools*, (c) *repository solutions and version control tools*, (d) *expectations regarding the impact of tools – mass-customization and the integration of laypeople*.

The first topic are (a) *editing tools*. Editing tools are tools that are used to manipulate digital designs, which are stored in corresponding files. Design files are only as accessible and hence editable as the tools to open and edit them are. Therefore, it is essential that all potential participants have free access to the means and tools to create and modify product designs. Raasch et. al. (2008) find that "[e]xamples of OSI in the automotive industry show that closed access to tools (in this case CAD software) put a significant strain on OSI projects" (Raasch et al., 2008). In software development, a broad range of free & open software to edit code is available. In computer aided design (CAD), by contrast, free & open source software is often either unavailable or at least not available in the quality that is expected by designers. Ratti et al. (2011) also mention rapid prototyping in the context of digital design tools. This, I take it, suggests a deeply interwoven relationship between digital design tools and digital fabrication (this relation will be discussed in more detail below, in the section of fabrication).

Another aspect of digital design are (b) *tools for collaboration*. Tools that enable and facilitate sharing and working together are brought up by as a topic by Ratti et al. (2011) in 'Open Source Architecture (OSArc)' under the point "design". They (Ratti et al., 2011) mention that "BIM (Building Information Modeling) and related collaboration tools and practices enable cross-disciplinary co-location of design information and integration of a range of platforms […]", but merely touch the topic.

As another technical aspect crucial for collaboration Raasch et. al. mention (c) *repository solutions*. They point out that "[t]he advantages of OSI projects cannot be realized, unless each member can build upon the ideas of others. This requires that he [or she] can find, understand and revise the blueprints of his [or her] peers. […] [T]herefore, there must be a repository for current as well as preceding designs including detailed documentation" (Raasch et al., 2008). In software development, a repository is a database in which changes that have been made to files are stored. It is usually integrated with a version control system that keeps track of these changes and their dependencies. See Fogel (2005) for details. While free & open solutions for these tasks are not yet commonly in use either in free & open architecture or in open design, there is already a free & open server application: *BIMserver*.[17]

Further aspects mentioned in my sources with respect to design tools are expectations, possibilities and hopes that are connected to digital tools. They are grouped under (d) *expectations regarding the impact of tools,* namely, *mass-customization* and *integration of laypeople*. Ratti et. al. prognosticate that in open source architecture "[m]ass customization replaces standardization as algorithms enable the generation of related but differentiated species of design objects" (Ratti et al., 2011). However, the tools that they mention are all proprietary[18] software and focus on design professionals. Ratti et al. highlight algorithmic and objectile-like characteristics as fostering for mass customization and connect this to the integration of new groups into decision making processes in design. Mass customization is also mentioned by Vardouli and Buechley (2014, p. 52). Like Ratti et al., Vardouli and Buechley connect open source architec-

17. See http://bimserver.org

18. The term "proprietary" has been coined by the FOSS movement to describe software that is *not* free (in the sense of freedom, not price), but rather, and in this sense, owned by an individual or a company. See, e.g., http://www.linfo.org/proprietary.html

ture to user empowerment and the integration of laypeople in design. With reference to traditions longing for the democratization of design, in the eyes of Vardouli and Buechley, the user that has to be empowered is an *end* user without design skills. In this tradition, the aspired goal is to give laypeople access to design and to enable them to participate in the design process. According to the notions of access and accessibility as they are used in the FOSS movement, the accessibility of the source is primarily a quality that fosters and supports collaborative work. Professionals are the primary target group, even though the same rights of access are granted to everybody, which of course includes laypeople (cf. Vardouli & Buechley, 2014, pp. 53–54). Vardouli and Buechley (2014, pp. 54–55) hence present approaches for computer aided designing environments that can easily be used by non professionals, so that lay users can be given access to design and be enabled to participate in the design process. Thereby, differing end user demands in matters of taste shall be met. Criticizing existing design environments that are designed to be used by non-professionals for being conceptualized as non-open systems and, therefore, being black boxes, they (Vardouli and Buechley 2014, 54) propose to "combine the ideas and practices of FLOSS with established frameworks of computer-aided participatory design to produce a hybrid structure that contains multiple sources and multiple layers of openness and accessibility". Ratti et. al. (2011) draw attention to existing (proprietary) tools that do not primarily address non-professionals but which, by virtue of parametric design, may "enable new user groups to interact with, navigate and modify the virtual designs, and to test and experience arrays of options at unprecedented low cost" (Ratti et al., 2011). In this context they also argue that we should recognize "laypeople as design decision-making agents rather than just consumers" (Ratti et al., 2011).

These are my findings of how tools and technical infrastructure are discussed in my sources. In the following, I briefly discuss these findings and provide some additional information that strikes me as important. It can be conjectured that the availability, or rather the lack of free & open digital tools and work environments that adequately support free &open working methods is a challenge to free & open architecture projects. Furthermore, there are different and at times even conflicting expectations of what a software should be able to do. On the one hand, proponents of free & open architecture call for complex design-software that fulfills the needs of highly specialized professionals. On the other hand, they call for software that is easy to handle, even for non-professionals. However, the lack of adequate free & open software solutions is not that surprising. The idea of a free & open architecture is comparatively young. So tools need time to be developed, and developing high-end CAD software is no trivial task.

Having said that, we need to keep in mind that the only way to guarantee equal access to design as well as its openness for the long term are to not only use free & open source software, but to also have that software operate with free & open file formats, which in turn need to be governed and protected by free & open licenses. I mention formats here, because they are closely related to software. Whereas software that is available to everybody is required in order to guarantee open access and free editing, free file formats are needed to guarantee that the information that is so created is not lost if the software with which it has been created is no longer (freely) available (software developers may decide to monetize their software, may lose interest in development, no longer find the time, etc.). It helps to take a look at the principles of the free & open movement to understand what the essential requirements for environments, tools and formats are in order for them to be suitable for free & open source methods and form them to guarantee free & open access and editing not only for the time being but also in the long term. These principles include the freedom to study and modify the source code. While the free & open software movement had no need to talk about editing tools and file formats (source codes are usually plain text and thus require no particular editing tools[19]), they are an important topic for other fields that want to implement free & open methods. The definition of "free cultural goods" by freedomdefined.org, for example, states: "For digital files, the format in which the work is made available should not be protected by patents, unless a world-wide, unlimited and irrevocable royalty-free grant is given to make use of the patented technology. While non-free formats may sometimes be used for practical reasons, a free format copy must be available for the work to be considered free" (Freedom Defined, 2008). Michael Avital (Michel Avital, 2011) notes that the distinct features of open design are that "open design is […] specified by a common notational language" and that "open design is not […] specified by proprietary notation". Furthermore, the Open Design Definition (OD+H Group, 2013) notes:

19. Of course, they do require compilers, and the GNU C Compiler (C and its derivates are the most important programming languages in operating system development) was indeed one of the first projects of the Free Software Foundation. See https://gcc.gnu.org/wiki/History

> Open Design is a design artifact project whose source documentation is made publicly available so that anyone can study, modify, distribute, make, prototype and sell the artifact based on that design. The artifact's source, the design documentation from which it is made, is available in the preferred format for making modifications to it. Ideally (but not exclusively necessary), Open Design uses […] open infrastructure, unrestricted content, and open-source design tools to maximize the ability of individuals to make and use hardware.

These statements make clear that proprietary software and file formats may restrict the freedom to modify as well as to study the source, for they may leave files inaccessible to most users. Of course, this may not always we possible, as it is implicitly acknowledged in the Open Design Definition, when the use of "open-source design tools" is described as "ideal", rather than as crucial.

We have seen that the topic of digital design tools presents challenges for open architecture projects, at least at the moment. And since tools, working methods and results tend to be closely intertwined, these challenges also affect design and the design process.

2.4 Information Transfer and Legibility

The studying and editing of designs requires tools. This is what the previous section was all about. However, the studying and editing of designs also require that the source files, the digital designs are "legible", that is, that they can be parsed by the tools that have been chosen. This topic has already been touched at the end of the previous section, where free & openly licensed file formats have been mentioned together with the need for free & open design software.

In this section, I will present topics and challenges that relate to legibility in a broader sense, namely, legibility as general and fundamental condition for enabling people to use open designs, or at least for enabling people to use open design as intended. Put simply, if something cannot be understood, cannot be read, then it cannot be used. Whether the reading is done by a person or by a computer program does not make much of a difference. Whether something can be read depends on all parties following the same conventions, these conventions can be about language, interfaces, etc. and may relate to digital artifacts as well as tangible ones. I found two such topics: (a) *standards* and (b) *documentation*.

(a) *Standards* are brought up by Ratti et al. (2011). They talk about them primarily because they see them as pivotal when it comes to how a design can ensure that all its tangible elements are compatible with each other. They recognize standards as an important aspect of distributed collaborative work processes and propose that "[t]he establishment of common, open, modular standards (such as the grid proposed by the OpenStructures project) addresses the problem of hardware compatibility and the interface between components, allowing collaborative efforts across networks in which everyone designs for everyone" (Ratti et al., 2011). But they also find that standards might have a more general importance for free & open source concepts, stating that "[u]niversal standards also encourage the growth of networks of non-monetary exchange (knowledge, parts, components, ideas) and remote collaboration" (Ratti et al., 2011).

(b) *Documentation* is mentioned by the OSE project. They state, in the OSE specifications, that "[d]ocumentation is the key to the replicability of practices and of hardware" (OSE, 2013b). The OSE project has comprehensive explanations on what and how to document (OSE, 2012, 2014). The importance that they give to documentation is underlined by the statement: "If it's not documented on the wiki, it doesn't exist" (OSE, 2012). Besides replicability, the OSE project highlights the empowering aspect of documentation: "[D]ocumentation allows the user to comprehend, take apart, modify, service, maintain, and fix tools readily without the need to rely on expensive repairmen" (OSE, 2013b).

This is what I found in my sources. Put another way, they did not have a whole lot to say about legibility. However, keep in mind that legibility was implicitly discussed by many sources when it came to editing tools. This would suggest that many projects *do* discuss legibility, but in practical, applied terms, rather than in abstract, theoretical ones. Still, the absence of a more theoretical discussion strikes me as somewhat surprising.

Standards and documentation are not explicitly mentioned to be challenging, but, considering the formulations with which they are discussed, it is safe to assume that they present challenges. Ratti et al. discuss what they think would be desirable regarding broad free & open collaboration. They imagine a future. And since the thing they imagine, universal standards of free & open architecture, does not exist yet, this is a challenge. Moreover, the statement by the OSE project, that things only exist if they are documented, suggests that people tend to be negligent in providing all the information that would be necessary in order to fully understand, use or replicate their design – which also indicates a challenge.

For collaboration, as well as the production of interrelated, follow-up and derived works, building on the work of others and, therefore, understanding their work is necessary. Hence, those works need not only be legible in a broad sense, but must allow for gaining a deeper understanding of how they function and of which reasons motivated which design decisions. Usually, the contributors to free & open projects tend to fluctuate and cooperate over great distances – and that they can do this and still be successful is one of the strengths of free & open methods that may be fruitful for architecture to adopt, hence free & open methods require techniques that allow to transfer information in a non-personalized and non-time-bound way. At this point, the two most important ways to do so, that is, in the free & open movement, are documentation and commenting. Often, different documentation is provided and maintained for different target audiences. In software design, for example, there is usually different documentation for users and developers. In open design and open source hardware, yet other forms of documentation may become necessary, for example, documentation for manufacturers. User documentation, simply put, concerns the finished product. When the product is a computer program, design algorithm or a parametric object, the user documentation describes how to use that software, script or object. When the product is a DIY kit, the user manual may help with assembly, and so on. A developer documentation explains how to develop or to built on a particular piece of software or a particular design and is intended for those who want to adapt the software or the design, or want to join the developer team. If a design process requires the use of computer programs, those programs' user documentation is part of the design developer documentation. Another part of the developer documentation are the standards used, for example, drawing standards that simplify the exchange of drawings and improve their legibility or technical standards that foster inter-compatibility. In short, documentation provides the basic information that is needed to get familiar with a project. Good documentations may even serve as didactic material.

Another form of transferring information are comments. Typically they are used to convey information about design decisions and are directly embedded in the code or the design file. Comments are, usually, used to make a design or a piece of code easier to read or to document why particular solutions were chosen, rather than to explain how to use something (though they are also used to provide developer documentation in software projects). Documentation and comments are ways to make something legible, comprehensible and usable. They must be understood as strongly connected to the artifact and an important part of the design process.

Another point I would like to add is the problem of deliberate obfuscation. Although one would expect that collaborators of free & open projects would not deliberately render their contributions difficult to understand, the Open Hardware Association (OSHWA) and the Open Design and Hardware Group (OD+H Group) both find it necessary to discuss the issue. In their definitions, the OSHWA as well as the OD+H Group explicitly broach the issue of deliberate obfuscation, pointing out explicitly that "[d]eliberately obfuscated design files are not allowed" (OD+H Group, 2013; OSHWA, 2013b). In highly competitive markets, where exclusive knowledge confers advantages over competitors, not releasing all information is often seen as crucial. In free & open projects, by contrast, making a design comprehensible and usable for others is a central goal.

Of course, a lot more could be said about legibility, but I have intentionally limited myself to what could be called its "technical" aspects. Still, it should have become clear that legibility may pose challenges for free & open architecture projects. Since legibility is a crucial aspect of the concept of designing in free & open projects and essential for any design that either is intended as foundation for follow-up works or builds upon the work of others, it needs to be taken into account in the context of open design as well as of free & open architecture.

2.5 The Process

This section presents the results from my analysis that deal with the conceptions of the design process in free & open design and architecture projects. Topics I found are: (a) *design as an evolutionary process* and (b) *feedback*.

(a) *Design as an evolutionary process* is mentioned by Kaspori (2003), Ratti et al. (2011) and the OSE project (2013b). Kaspori (2003) sets out from the assumption that planing issues are usually complex issues and thus can only be solved by working together. Free & open source provides organizational approaches for collaborative endeavors. Kaspori (2003) notes that further "[o]pen source presupposes that […] ideas are disclosed and made available to others, who in turn can improve on them". In consequence "design change[s] […] from a one-off action into a kind of evolutionary process" (Kaspori, 2003)

Ratti et. al. (2011) speculate that "design [is] becom[ing] an ongoing, evolutionary process, as opposed to the one-off, disjointed fire-and-forget methodology of traditional architecture". They say this in the context of building automation, mentioning buzzwords such as "the internet of things" or "kinetic or sensor-based environments" that tightly integrate software, hardware and mechanisms. They also mention free & open electronic and digital hardware as well as open platforms for everything from design over construction to occupancy – it seems they imagine some kind of digital monitoring of every step from design to demolition.

Furthermore, Ratti et al. (2011) also talk about (b) *feedback*, referring to sensor based, responsive systems and techniques that have been intensively discussed recently in the context of so-called "smart environments" and "smart cities". They hope to gather large volumes of real-time information by crowd-sourcing such feedback mechanisms: "Real-time monitoring, feedback and ambient display become integral elements to the ongoing life of spaces and objects" (Ratti et al., 2011).

In hardware development, problems are often discovered only in the actual, tangible artifact, sometimes only after it has been used for a prolonged period of time. Therefore, reports on such problems and a continuous process of development are essential to optimization. Continuous development and maintenance are also important to the vitality of a project. Put simply, free & open projects and the things that they develop need to be looked after. If there is no visible online activity, a project is quickly perceived as dead, not only by outsiders, but also by contributors, who, after all, are connected to each other only by online channels. As a consequence, such a project is typically unable to attract new contributors and may even struggle to keep the ones it already has. In the OSE specifications it is indicated, under the point "product evolution", that "[a] process should be in place for continued maintenance and development of a product. This could be a support community, foundation, or users" (OSE, 2013b).

In the sources on free & open architecture, design is seen as an evolutionary process by Kaspori and Ratti et al. Moreover, Ratti et al. display an attitude that is very enthusiastic about technology, a tad too enthusiastic if I may say so. What they mean precisely, when they talk about design as an evolutionary process, is difficult to discern, but they make an interesting point when they mention feedback processes. How to include feedback loops in the design process is an interesting question worthy of discussion. Kaspori describes evolutionary design as enabling and being enabled by the improvement of ideas that is made possible by openness – this corresponds with the point that the OSE project makes when they demand continued development and maintenance. Ratti et al. and Kaspori find that our current approach to design, which they describe as "one-off action" or "disjointed fire-and-forget methodology of traditional architecture", needs to change. Of course, this requires that we, architects, have to change the way in which we think about design and design projects as well. This, I assume, is also a challenge.

2.6 Complexity

In this section, I present topics and challenges that pertain the complexity of projects. These are clustered under: (a) *effects of complexity* and (b) *simplicity.*

The (a) *effects of complexity* are mentioned by Raasch et. al. (2008) and Abdelkafi et. al. (2009) when they talk about the feasibility of projects: Projects of high complexity may be very demanding to realize, due to the large amount of development and coordination, which may well endanger the success of a project, especially if the community of that project is (still) small, as Raasch et. al. (2008) note. Put the other way round, the more development is needed and the larger the effort that is required to coordinate that development, the less feasible the project. A way to reduce complexity is to “narrow the project scope to crucial components by incorporating existing proprietary solutions into the design, at least during the earlier stages” (Raasch et al., 2008). When integrating non-open but already available components, a working product can be developed more quickly. Later on, those components may step by step be replaced by free & open components (Raasch et al., 2008).

The OSE project regards (b) simplicity as essential to design. They argue (OSE, 2013b) that keeping a design or concept clear and simple may be the design goal that is hardest to attain. However, simplicity is identified as one of three key features that makes design replicable, besides making that design open and favoring low cost approaches. Designs and implementations should focus on simplicity in order to keep fabrication procedures simple and costs low. What is more, simplicity is, besides designing for disassembly, and proper documentation, noted as a feature that fosters user-friendliness (OSE, 2013b) That said, they also make clear that the aim of making designs simple should not lead developers to compromise performance standards, arguing that simplicity “does not imply substandard or economically-insignificant production” (OSE, 2013b).

Complexity is a challenge for projects in general, but for free & open projects in particular, given that the majority of contributors tend to be volunteers who may be put off by overly complex projects. Complex systems are hard to understand and difficult to organize due to the interplay of various factors. What is more, it is not only the problem at hand or a particular design that may be (overly) complex, projects may also (and usually do) grow more complex over time. One way to deal with complex systems, as we are about to see in the next section, is modularity, which reduces the number of things that need to be understood “at once,” so to speak. Understanding of a design is the foundation for working together and employing free & open methods. The simplicity that is required to make such an understanding easier to attain is achieved by reducing problems to their essentials and making something easy to understand “by design”. Thereby, simplicity fosters collaboration and the reuse of previous work.

2.7 Modularity

Put simply, modularity means to compose a system out of smaller units. Those single units, building blocks or modules are intended to have as few inter-module dependencies as possible. Minimizing and specifying the inter-dependencies between individual units, that is, modules, allows to distribute the work that is to be done between contributors in a way that enables each of them to work autonomously to a high extent. What is more, designing a module with clearly defined dependencies presents itself as a far less arduous or complex task than the respective project as a whole or solving the same design problem within that project but without modularization.

In this section, my aspects and challenges that relate to modularity are presented. Those are: (a) *Properties of modular design in artifacts* and (b) *conditions for modularity in software and hardware.*

The (a) *properties of modular des*ign are emphasized in the specifications of the OSE project. According to these specifications (OSE, 2013b) "[o]bjects should be designed so that they are made as building blocks, or modules, of other or larger objects". Put the other way round, large or complex designs should be composed out of different, independently functioning modules. The OSE specifications focus on the advantages of modular designs, namely: flexible, modifiable and adaptable designs for flexible, modifiable, adaptable and easily serviceable artifacts, which give the user a high degree of control. The OSE specifications appear to adhere to the philosophy that the modularity of a design reflects the modularity of the artifact. Moreover, the OSE project also list requirements for individual modules or components within a modular artifact, namely: modules should be inter-operable, scalable, exchangeable and interchangeable (OSE, 2013b) They place particular emphasis on scalability; component designs should be apt for smaller as well as larger version of that component. The systematic implications of modularity are described with reference to Christopher Alexander et al.'s book *A Pattern Language* (1977): "If modular design is followed, then the type of interoperability of using building blocks leads us to a Pattern Language of technology. In this pattern language, the modules or building blocks serve as the sentences of a larger language, or technology infrastructure" (OSE, 2013b).

(b) *Conditions for modularity in software and hardware* is raised as a topic by Abdelkafi et al. They (2009) see differences in how modularity should be achieved between software and tangible goods, finding that in software production, modern programming languages provide a framework that supports and fosters modularity, in particular object-oriented and aspect-oriented programming paradigms. They compare the role of programming language as the medium in which software is developed to the role of matter in the design of tangible goods, calling upon us, in order to picture the advantage that software has in respect to modularity as a consequence of this medium, to "imagine a kind of matter that naturally improves the modularity of physical products" (Abdelkafi et al., 2009). Research on modularity in tangible goods focuses on methods that enable object modularity (2009). Almost all methods, they note, split up existing designs into modules and are *not* meant to be applied at the beginning of the design process. Raasch et. al. (2008) mention that high level modularity is probably only required for highly complex products. For simple products, modularity is not necessary.

These are my results for aspects and challenges that concern the modularity of designs or the that of the designed artifacts respectively. In my sources, modularity is strongly connected to the actual, tangible artifacts. This is not very surprising, but it should be mentioned that modularity exists on many levels. Of course, artifacts are a good example for modularity, but modularity can also be found in design or in fabrication. What is more, modularity in design does not necessarily result in modularity in fabrication or in a modular artifact. That is, a modular approach at one level does not imply a modular approach let alone a modular outcome in another. That said, even though this is not necessarily so, modules that are defined in light of the needs either of the finished tangible good or the fabrication process are often a good starting point for the modules that should be defined in the design. Abdelkafi et al.'s comparison between programing languages and building materials struck me as misleading at first sight, for I would argue that programming and programming languages are better compared to designing and the ways in which we build up our designs than with materiality. Yet, even though this comparison is somewhat misleading, Abdelkafi et al. have a point, namely, that the physical material, for example, the building material, influences what kind and what degree of modularity can be achieved in a tangible good. Put simply, in-situ concrete would be a less than ideal the choice for a modular artifact, wood or prefab elements would be preferable. Furthermore, there are also modular construction techniques. However, modularity in design is only partly, and not necessarily, connected to modular construction methods. What is more, I think we should keep in mind that the terms "modular" and "modularity" can mean somewhat different things in architecture and software design. The term "module" in architecture is used for a unit of measurement that, while referring to some kind of modules, is used to express the desired proportions of buildings. Of course, the term "module" may also refer to standardized building units that are used to compose larger objects. Collaborative design and development may need to rethink what "modularity" means in architecture.

Put another way, the challenge for free & open architecture is to answer the question: "What does or could a modularity that fosters free & open working methods look like in architecture?" I think in order to answer this question, we need to consider the different levels at which modular approaches can be applied. More importantly, we must keep in mind that if we want to apply free & open source methods to architecture, then the goal of applying modular approaches should be to make it easier to divide tasks into sub-tasks. This is because dividing tasks into sub-tasks by defining modules and their inter-dependencies enables contributors to work, to a high extent, autonomously and in parallel, rendering even complex projects manageable without too much coordinative overhead.

2.8 Automation

Automation means to set up a system in a way that allows it to perform a series of operational steps with little or no oversight, intervention or even involvement of a human agent. Automation can be applied to and found within tangible artifacts, in fabrication and in design.

In this section, I present topics and challenges that I found in my sources that are connected to automation, namely: (a) *digital automation in hardware,* (b) *digital methods in design and fabrication* and (c) *unrewarding tasks.*

(a) *digital automation in hardware* is discussed by Abdelkafi et al. (2009): "With programmable hardware, it is possible to translate some hardware design tasks into software development activities. It is a way to take advantage of the inherent characteristics of software that are conducive to open source development." At a first glance, programmable hardware does not seem to be too important for architects, but when we remind ourselves of Ratti et al.'s remarks about smart environments, building automation, etc., we can see that programmable hardware matters to us architects as well. Furthermore, programmable hardware can be put to use in many fields, from automated climate control to controlling media facades. Put simply, digital automation will be useful wherever behavior can be digitally changed to meet different conditions and needs. Whereas Abdelkafi et al. focus on digital automation that is integrated in actual, tangible artifacts, discussions of automation can also found concerning design and fabrication.

(b) *digital methods in design and fabrication* are discussed by Ratti et al. as well as Vardouli and Buechley. Those methods, parametric and algorithmic design, mass customization, BIM and digital fabrication, are all forms of digital automation, hence I include them here as well, even though they are also discussed in the sections on *tools* and on *fabrication*.

How to avoid (c) *unrewarding tasks* through automation is mentioned by the OSE project (2013b). They find that all repetitive tasks are candidates for automation. A task is repetitive either if few people need to do it frequently or if several people need to do it, even if they have to do it only rarely. Automation is favored whenever "repetitive, difficult, dangerous, or otherwise unrewarding tasks can be carried out with computer assistance instead of human labor" (OSE 2013b). Automation is first and foremost discussed in the context of fabrication, but also with respect to other tasks, which however are not specified any further (OSE, 2013b).

Design can be automated by transforming a design task into a rule-governed process that can be executed by a computer. By automating repetitive tasks, work loads can be reduced and uninteresting or time-consuming work can be avoided, as found by the OSE project. Basically, every rule-based design task can be automated. Generative processes, algorithms and objectiles may be used (and reused) to produce a quantity of different objects, employing parametric variability. Of course, the more often a design task needs to be performed, the more advantageous its automation is.

However, automating a design task also opens up the possibility of discovering new forms, for example, by experimenting with its parameters. That being so, automation may well be worth the effort, even if the task at hand needs to be performed only infrequently. In design, as in other fields, there are a lot of rule-guided tasks. Yet, for many tasks, automating them does not seem worth the effort, even if they are highly rule-guided and therefore apt for automation. However, free & open architecture and design builds on using, improving, adapting and altering designs from and by many different people. Hence automation may be a rational choice, even if this is not apparent at first sight. When we leave the restrictions that have been imposed upon us by traditional authorial approaches behind, then automating tasks becomes reasonable even for individual developers, because the tasks that they may have to do only once may need to be done by others as well. And if we establish a commons of techniques that automate these tasks, based on free & open methods, then such techniques can be shared.

Furthermore, automation gives people the ability to manipulate designs or generate forms without necessarily knowing how to do it "manually". Hence, automation is related to a point that was mentioned earlier, in the discussion of design tools: how parametric design may help to integrate laypeople in the design process.

Unfortunately, traditional architectural education neither teaches us to think in formal relationships nor to recognize and formalize the, often quite complex, rules behind rule-guided processes. These skills are not yet seen as crucial for designers. Hence, defining relationships and formalizing the rules behind intuitively proposed solutions is probably challenging to most architects. However, doing so, which would allow to easily come up with variations of a design. And this is highly interesting for free & open working methods, because it allows the adaption and thereby the reuse of designs, even under different circumstances.

2.9 Fabrication

Fabrication is the translation of the information provided by a design drawing and associated information into a tangible artifact. Thus, the design needs to be legible for the entity that shall produce the physical artifact. That being so, that entity is, in a way, co-determining in which form its design information needs to be provided. Hence, the properties of that entity limit the possible forms of design. This is true for machines as well as for humans or companies. Therefore, we should ask who the "producers" of free & open designs are, who the users are, and how they fabricate before we discuss how fabrication impacts design in free & open source culture. Hence, we need to keep in mind how free & open source methods differ from traditional methods; in fact, the so-called community-based model of production that is employed by many free & open projects is often linked to social or non-market ways of production. If free & open methods are not only about access to source data, but also about a community-driven approach to production and development, an idea that this thesis builds on, then this may not only be relevant for the production of digital artifacts, but also for the fabrication of tangible goods. And since architecture is not only about design but also about tangible artifacts, any discussion of free & open source architecture must include community-driven approaches to production and fabrication, at least as a possibility.

Regarding fabrication, my sources mention the following aspects and topics: (a) *interpretation and ambiguity,* (b) *loci of production*, (c) *design for community based production,* (d) *means of production and supply* and (e) *prototyping and replication.*

Vardouli and Buechley discuss (a) *interpretation and ambiguity* and bring up that, usually, the "output" in architecture does not only depend on the quality of the design, but also on the abilities of those who interpret it, that is, of those who build. They (2014) also describe, in their discussion of the meaning of the terms "source" and "access" the process of transforming code into software and building information to built artifacts. As they point out, in software production the transformation from source code into compiled software is conducted by a machine. No human interpretation is needed. Transforming the design information into tangible architectural artifacts, by contrast, does require, at least in traditional approaches, human interpretation. The final result is highly contingent on human interpreters. Hence, ambiguity is an inherent characteristic of how design information in processed in architecture. This may be seen as a source of creative potential. Vardouli and Buechley draw and embrace the picture of builders that are interpreters, much like musicians. The quality of the final building depends on the quality of the builder's interpretation and work. This embrace runs counter to the fully pre-determined and "mindless" translation of code into software that is done by compilers and welcomes the creative potential of the infinity of translations. Software is not a human interpretation of source code, but a building is a human interpretation of design, plans and other building information. This is a crucial difference between software production and the construction of buildings. In addition to the quality of ambiguity they also discuss developments that may reduce spaces for ambiguity in human interpretation, for example, BIM. They argue that BIM, which aims to convey as much information about a building as possible, allows to share highly detailed building information and therefore reduces ambiguity. However, they find, an abundance of information does not necessarily increase quality. Furthermore, they identify automation and digital fabrication techniques (e.g., 3D printing) as trends that may further reduce ambiguity, by replacing builders with machines. This may, one day, enable an unambiguous translation of digital descriptions into tangible artifacts (cf. Vardouli und Buechley 2014, 53).

The question of the actual (b) *loci of production*, that is, of who manages and executes the process by which a product that has just been designed becomes (physical) reality, is raised by Raasch et al. (2009). To be sure, Shirky has a point when he finds that "[a]n increasing number of physical products are becoming so data-centric that the physical aspects are simply executional steps at the end of a chain of digital manipulation" (Shirky 2007 gtd. in Raasch et al., 2009). And so does Hippel when he says that "hardware is becoming much more like software" (Hippel qtd. in Raasch et al., 2009). But physical products require physical production. And this is a challenge that all open design projects face in one way or the other (Raasch et al., 2009). Raasch et. al. (2009) identify three ideal type approaches to who does (or is intended to do) the actual production: (1) *external manufacturers*, (2) *the community* and (3) *the manufacturing company that has initiated or coordinates the project* (and hence is not "external"). Not all of the points that are discussed in what follows apply (or apply in the same way) to all of these approaches.

The OSE project subscribes to fabrication by the community. Hence, they also discuss how to (c) *design for community based production*. They start out from what the cal "design-for-fabrication," which according to the OSE specifications is characterized by three features: (1) simplicity of design, (2) modularity that allows to use or reuse already available components and (3) the use of off-the-shelf, standard, preferably low-cost components. Needless to say, the goal of these features is to make fabrication as simple, efficient and consuming as little time as possible (OSE, 2013b). They then elaborate that "[d]esign-for-fabrication may take on the form of design-for-Collaborative-Production" and, in connection therewith, that combining collaborative production and digital fabrication may increase efficiency (OSE, 2013b). The term "collaborative production" is not described any further in the OSE specifications, but the more detailed explanation found within the OSE wiki is interesting. "Collaborative production" as understood by the OSE project, describes a form of production in which a complex artifact is fabricated in a short time by a group of community members in a flexible workshop, where individual teams build components or modules in parallel. This production model is called "production run" (OSE, 2013a), in analogy to what is known as "sprint" or "code sprint" in software production (cf. the section on economical, legal, social and cultural Aspects above, where "code sprints" are discussed as a working method that may motivate contributors).

The topic of (d) *means of production and supply* is raised by Abdelkafi et. al. (2009). Focusing on the differences between software and hardware, they state that software production, for the better part, requires no costly infrastructure. Also, software production is not tied to a specific place. Usually, all that is needed is a hosting provider, the basic services of which tend to be available at low cost, and a computer, which contributors usually own themselves. With tangible goods, this is different. When producing tangible goods, one always has to deal with supply chain management and daily business operations, which may also be a relevant cost factor (Abdelkafi et al., 2009). Supply chain management is of particular relevance for companies that develop and produce physical open source innovation products.

Aspects of (e) *prototyping, production and costs* are further points where differences between software production and the production of tangible goods can be found. One difference between digital and tangible goods is how well they can be reproduced. Whereas the physical features and hence the quality of every unit of a tangible product will, if only slightly, differ from that of any other unit and production may be costly, digital products can be reproduced without any loss in quality and reproduction comes *de facto* at no cost; any additional unit of a digital product will, in all relevant aspects, be exactly the same as the one from which it has been copied and the marginal costs for each of these units are close to zero (Abdelkafi et al., 2009).[20] Moreover, software can be distributed almost instantly via the internet. This allows to

20. The different cost structure of software production and the production of tangible goods is discussed in more detail above, in the section on economic aspects.

adopt a practice that is called “soft release,” where developers publish few or small improvements frequently in short time intervals, rather than bigger or more numerous improvements in longer time intervals. In Hardware design, drawings may be released frequently and be improved continuously, but the tangible artifact, once produced, cannot be changed that easily. However, as Abdelkafi et. al. (Abdelkafi et al., 2009) note, “many design problems and errors in production are not detected until the product is manufactured”. Raasch et. al. (2008) also point out that the costs of transforming information into a tangible good, of prototyping and (re-)production, may be challenging. “To run, test, and debug a software application, developers need only a computer and a compiler. Building and testing product prototypes, however, may be rather costly” (Raasch et al., 2008). That said, they also note that digital simulations may, in some cases, reduce the amount of testing required and, thereby, alleviate costs (Raasch et al., 2008).

How a product will be constructed or fabricated, that is, how a design will be translated into matter, must be kept in mind when designing. Doing so, the notion of “designing for fabrication”, to design in a way that enables communities of peers or even laypeople to fabricate that design on their own, sure enough requires the biggest change in perspective. To design “for fabrication” in this sense, requires us to take a closer look on fabrication techniques and how accessible they are, that is, whether they require hard to come by or expensive raw materials, whether they require hard to come by or expensive tools, and whether they require hard to learn skills. Although low-tech buildings are discussed in architecture, particularly in the self-building sector, a more vibrant culture of sharing, where design files are distributed widely, can be found in the fabbing community,[21] which grew around high-tech tools of digital fabrication. We should look at that community and how it operates, though this is beyond the scope of this thesis.

“Designing for production” and, thereby, enabling and building a community that manufactures the product is an interesting and feasible approach for free & open projects. Usually in such projects, the community consists of producers, developers and users alike, with many people will combining all three roles at the same time. Thereby, experiences from fabrication can be drawn upon to further develop the

21. The term 'fabbing' has been coined for forms of personal fabrication that are assisted by software (cf., e.g., Troxler, 2010). The fabbing culture is strongly tied to sharing platforms for digital models for digital fabrication, so-called Fab Labs, hacker spaces, etc.

product – and this feedback loop is important for development. This overlap between designers, producers and users is what Carpo argues may have disrupting consequences for the well-established practices in our field, which have emerged around an Albertian architect-author. By contrast, free & open source approaches often rely on the close relation between designers, producers and users.

Hence, building and enabling a community to do the physical production may be a feasible strategy for free & open design and architecture projects, even though these projects need to keep in mind *that* such communities need to be built and nurtured first *as well as* enabled to actually produce the good in question by "designing for production."

2.10 The Artifact

This section is about the conception of the actual, tangible artifact, that is, the product, itself. Indeed, the form, the material, the functions and other aspects of the actual, tangible object may be the first things most people think of when they think about design. Carliss Baldwin and Kim Clark (2000, p. 21) defines the design as "a complete description of an artifact", pointing out, with reference to Alexander (1964), that "[d]esign is the process of inventing objects – or 'things' – that perform specific functions". When we make design decisions, we, so to speak, narrow down the range of possible outcomes, possible objects. So, when we talk about the artifact, then we talk about its qualities or functions. Importantly, there may be qualities that people expect from artifacts that were designed and developed with free & open source ideals in mind.

In this section, I list aspects and challenges that are mentioned in my sources and concern the qualities that appear to be expected from artifacts that have been designed and developed using free & open source methods and which, therefore, need to be considered when designing them. The following aspects are mentioned: (a) *disassembly*, (b) *lifetime*, (c) *material aspects* and (d) *performance aspects*. All those topics are brought up by the OSE project.

The OSE project emphasizes the qualities of artifacts which are designed for (a) *disassembly* as, so they argue, repair and hardware maintenance are based on the possibility to remove and replace broken parts. Design for disassembly describes a way to design artifacts that allows replacing parts of modules as well as whole modules – this is fostered by a modular approach in the composition of the tangible artifacts. By supporting disassembly by design, the lifetime of an artifact may be significantly extended. In the OSE specifications design for disassembly is further stated to be essential for user-friendliness because the user may easily take the artifact apart (OSE, 2013b). The OSE project states that "[d]esign-for-disassembly is synonymous with user ability to 'look under the hood' of a certain device" (OSE, 2013b). This reminds of the "freedom 1"[22] of the Free Software Definition by the Free Software Foundation (cf. the section of aspects of definition above), which mandates that user must have the freedom to study how programs work, which requires access to the source code.

(b) Product *lifetime* is mentioned together with "design for disassembly", maintenance and serviceability of an artifact. Design is a key factor for product lifetime, and the OSE project states that the "value of a product [should] not depreciate over time" (OSE, 2013b) as an explicit design goal. This goal should be reached through design that is solid, robust and allows the user to service and repair the artifact. Design that allows users to do so has three properties: (1) Information about the design is accessible. This is guaranteed by keeping the source files open, by providing documentation, and by nurturing an active user community. (2) The artifact needs to be designed for disassembly, that is, the user must be able to dis- and reassemble the artifact without damaging it. (3) The artifact needs to be modular and modules need to be replaceable (OSE, 2013b).

In the design of tangible artifacts (c) *material aspects* always play a major role. Needless to say, material aspects are closely related to fabrication. The OSE project argues for using materials and resources that are not scarce but commonly, and preferably, locally available; scarce resources should, wherever reasonable, be substituted by more common ones (OSE, 2013b). Also, costs should also be kept low (2013b), which, of course, influences material decisions. What is more, all materials and parts that are needed to built a product should be listed in a "bill of materials" (BOM), which should

22. To avoid confusion, programers tend to count from 0, rather than from 1, hence "freedom 1" is the *second* freedom expressed in that definition.

include information on where these parts and materials can be obtained and how much they cost (OSE, 2013b). The BOM is a part of the design documentations.

Further (d) *performance aspects* are mentioned. High performance and efficiency are target standards of the OSE project. Performative competitiveness with industry standards and products that "match or exceed standards of industrial counterparts are envisioned in order to provide a viable, comparable or better alternative" (OSE, 2013b). It is highlighted that high performance is not contradictory to a DIY approach "as long as the enabling tools and techniques are accessible. There is no limit to the tooling available in the DIY context – as long as the tools are appropriate, open source, and user-centered" (OSE, 2013b). The aim of the OSE project is to keep the design straightforward, to stay clear from adding superfluous features, and to focus on what is sufficient for a certain level of performance. Furthermore, the OSE project emphasizes that designers should employ trusted and proven techniques, simply because these techniques tend to be common knowledge and hence are more accessible (OSE, 2013b).

Many tasks in design boil down to making decisions about the properties and parameters of a future artifact (cf. Baldwin & Clark, 2000, chap. 2). Different parameters, in turn, are tied together by an often complex web of inter-dependencies. When we make a choice about one parameter, then this choice will limit the number of choices that are available to us when it comes to other parameters, and these choices, too, have their consequences. The choice of material, for example, has consequences for choices regarding construction, fabrication, etc. The question is whether a free & open source approach in design or architecture affects the properties and functions of the designed artifact. We can also ask this question the other way round, are there properties or functions that make some kinds of artifacts particularly suitable for free & open source approaches, commons-based collaboration and community-based ways of designing and developing, and if so, what are they? Put simply, do free & open methods "preconfigure" certain design decisions?

We have seen, when looking at what is expected from the actual, tangible artifact, that the ideals of the free & open movement shall also be embodied by that artifact. The right to disassemble and open, the right to alter, to fix or improve (cf. aspects of *lifetime* and *disassembly*) all refer to the freedoms that are required for a software to be free & open software. Furthermore, making the product in question available to as many people as possible by working with widely available, affordable materials (cf. *material aspects*) is also an aspect that can be traced back to the functioning of free & open source. However, broad access is not necessarily something that is aimed for because of lofty ideals, free & open methods simply work better with a large pool of potential contributors.

To sum up, free & open methods may not strictly require, but probably are easier to apply if the physical object meets certain conditions, which result from the principles of free & open – in particular the freedom to study and edit – and from striving to lower the threshold in matters of access. These conditions are challenges in applying free & open source concepts to architecture, especially if a project decides that its product should be produced by the community.

2.11 Conclusion

In this part, I have given an overview of the debate on free & open source architecture and extracted aspects and challenges that need to be considered when free & open methods shall be applied in architecture. I have distinguished between design-related and non-design-related challenges and, even though they are not my main topic, have compiled a list of non-design-related challenges, namely concerning economic, legal, social and cultural aspects. However, the main goal of this chapter was not to simply compile a list of aspects and challenges, not even of design-related ones, but to come up with a list of categories under which design-related aspects and challenges can be subsumed – these are the main findings of this chapter. Of course, that is not to say that non-design-related aspects and challenges were unimportant (I would not have collected them if they were), all free & open architecture project would seem well advised to consider them as early as possible. Having said that, I have found the following categories (I exclude from this list those categories that were adopted from literature, *the economical, legal and social aspects*):

Table 1: Categories of challenges faced by open design projects

Design-related	Not design-related
1. Tools and Technical Infrastructure 2. Information Transfer and Legibility 3. The Process 4. Complexity 5. Modularity 6. Automation 7. Fabrication 8. The Artifact	1. Aspects of Definition 2. Cultural Aspects

Unfortunately, many issues where cross-cutting concerns, so that it was difficult to limit the number of categories. Moreover, I opted to prefer a longer list of more concrete properties over a shorter list of more abstract ones, since more abstract categories would have been far less informative (e.g., points 3–8 could all have been subsumed under "Design aspects", but that would not have been very telling). Since my thesis is based on the assumption that the subject matter that needs to be adapted in order for architecture to be able to make more extensive use of free & open methods is design, I have chosen not to discuss the *aspects of definition* and the *cultural aspects* and further the *economical, legal and social aspects* in detail, due to their subordinate impact.

Under these categories I have subsumed the following aspects and challenges, with some simplification (the aspects and challenges that I have found are numerous and manifold and resist easy compilation into a précis):

Table 2: Challenges faced by open design projects, by category

Tools and Technical Infrastructure	Editing tools
	Collaboration tools
	Repository solutions and version control tools
	Expectations regarding the impact of tools – mass-customization and the integration of laypeople
Information Transfer and Legibility	Standards
	Documentation
The Process	Design as an evolutionary process
	Feedback
Complexity	Effects of complexity
	Simplicity
Modularity	Properties of modular design and artifacts
	Conditions for modularity in software and hardware
Automation	Digital automation in hardware
	Digital methods in design and fabrication
	Unrewarding tasks
Fabrication	Interpretation and ambiguity
	Loci of production
	Design for community-based production
	Means of production and supply
	Prototyping and replication
The Artifact	Disassembly
	Lifetime
	Material aspects
	Performance aspects

These categories and, to a lesser extent, aspects and challenges will be confronted with the WikiHouse project in the next part of my thesis. Of course, not all of these categories, aspects and challenges will apply or be found in that project though. However, I did not only come up with these categories in order to have an instrument for my analysis of the WikiHouse project – they are an interesting finding in their own right.

That being so, I also want to draw attention to two important avenues for future research. (1) I have mentioned that we need to rethink what "modularity" means in architecture, considering all levels at which "modularity" can be applied to a design, not only artifacts; Abdelkafi et al. (2009) appear to agree (cf. the section on modularity above). (2) I have not discussed economic aspects in depth because my sources do not identify economic aspects that have more than minor implications for the design process (to the effect that much of the corresponding discussion in current literature is beyond the scope of my thesis). However, whenever I discuss the possibilities of free & open architecture with colleagues, concerns regarding the economic viability of free & open architecture (and its impact on "ordinary" architecture) are usually quick to arise, often making for a heated debate. Hence, I think we need research (and experimentation) on which economic challenges are particular for open design and open architecture and how to overcome, including how viable business models would look like or what role free & open architecture can play in the construction sector.

I have argued, in part 1 on this thesis, that adopting free & open source methods may impact and even require to change the way in which we think about and treat authorship in architecture. We may want to and even need to switch from the Albertian architect-author to the authorship of the many, which may even include contributions made by laypeople. Doing so will, of course, change the way in which we design – in a way, that is the whole point. Put simply, as Albertian author-architects we know how to build Raymond's "cathedral," but if we (also) want to be a different kind of author, then we will have to learn how to build "bazaars". On this, the findings of my analysis correspond to MacCormack et al. (2006, p. 1015) who note in their study, which seeks to characterize the differences in design structure of proprietary and FOS software, that their "results, while exploratory, are consistent with a view that different modes of organization are associated with designs that possess different structures". The categories of aspects and challenges that I have developed in this section are, in a way, both a list of things that we need to learn and a map of obstacles that we may have to overcome along the way, for even with these categories to guide us, we will, in the end, have to learn many things the hard way, by experiment, by trial and error. We need to learn how to best share our works, how to maintain evolutionary design, how do design in a modular way, how to recognize and formalize rule-based aspects, how to put algorithmic and parametric design to use in collaborative networks, etc. Often, this is not simply a learning process, rather we will also have to establish best practices and platforms, maybe even institutions in order to enable these practices. For all this, that is, the practices, the institutions, the platforms etc., WikiHouse is, to this date, our best example. Therefore, I will, in the next and last section of this thesis, confront the WikiHouse project with the categories that I have just developed.

3 The WikiHouse Project, a Case Study

3.1 Introduction

About a decade ago, free & open architecture was merely a cloud of ideas. The first project that brought broader attention to the idea of introducing free & open source ideas into architecture was, as we have seen in part two, the Open Architecture Network, which was founded by Architecture for Humanity after Cameron Sinclair (2006), one of the founders of Architecture for Humanity, gave a TED talk and won the the TED price. Then, in 2010, another event generated attention for free & open approaches in architecture: the Madrid Air Tree Pavilion at the Expo Shanghai, designed by Ecosystema Urbano, the plans for which were released under a Creative Commons license (Ecosistema Urbano, 2010a). Today, the Open Architecture Network no longer exists. And around the Air Tree, no community ever evolved.

The WikiHouse project was born in 2011. In 2013, Alastair Parvin (2013) presented the project at a TED talk, and the project got a lot of attention and attracted a growing number of participators since then. Today, approximately 900 people are members of the WikiHouse Google group (WikiHouse, 2015g). WikiHouse is the first free & open architecture project that has been able to attract such a large community that is working together on the same idea and designing on the basis of the same design concept. Presently, the project is healthy and the community is lively and growing (recently, one has even been built in Vienna, as part of the *Vienna Open 2014*; see figure 6). WikiHouse, it seems, has been doing things right. This makes the project well-suited for learning something about applying free & open methods to the physical realm in general and to architecture in particular.

This third part of my thesis builds on the findings from part two. In part two it was investigated, on a theoretical level, which challenges projects that want to apply free & open source methods to architecture are likely to have to face. My aim in this part is to check if the topics and challenges I found in my theoretical investigations are actually important in the real world. When looking at real world attempts of applying free & open methods in architecture, there are only few projects that do so and their approaches are quite diverse (cf. Rosada, 2012). Therefore, I decided to study the project that, in my view, is the most promising – WikiHouse. The WikiHouse project appears to be the most successful project of this sort – it has an active, lively, and growing community, which is one of the most important markers for success for free and open source projects, and it gets public attention.

For my investigations I have chosen the form of a single case study, focusing on the conception of design and design processes and closely related aspects in the WikiHouse project. The sources that I have used for my investigation are the materials that have been available on the WikiHouse website between October 2013 and June 2015. Contributors to free & open projects tend to be widely dispersed, living in different cities, often different countries. Consequently, what is not shared online is not accessible to all community members. The OSE project (OSE 2012) even quips that what is not documented online, in its Wiki, "doesn't exist". That being so, I can safely presume that documents that are not online have little impact on the overall project and can hence be disregarded. I will compare these sources with the findings from part two, using the categories that I have developed there. Thereby I want to answer the following questions: (1) Are the topics and challenges that have been identified in part two relevant for the WikiHouse project? (2) If so, then how are they dealt with or implemented in the project? Further points of interest are (3) if and how design and the design process are influenced by these topics and challenges and (4) if this is connected to the WikiHouse project being a free & open project? I thereby hope to achieve a better understanding of the topics and challenges that have been identified in part two and of how free & open methods impact design and the design process. Given my

Figure 6, top: A WikiHouse Studio is assembled for the *Vienna Open 2014*
Bottom: The fully assembled WikiHouse Studio at the Karlsplatz, Vienna
(Bast, 2015a, 2015b)

methodology, these findings are not representative in a statistical sense, but the ways in which the WikiHouse project deals with the aforementioned topics and challenges may nonetheless be applicable to many other projects and will be useful evidence to inform the ongoing discussion on how to apply free & open source methods to architecture. I can, of course, only form hypotheses to the extent to which my results (the ways in which the WikiHouse project deals with these topics and challenges) are applicable to other projects. Testing these hypotheses would be an interesting topic for future research.

I will now look at the WikiHouse project under the viewpoint of the categories, which I have developed in part two, using individual sections to cover each of the following categories: *Tools and Technical Infrastructure, Information Transfer and Legibility, The Process, Complexity, Modularity, Automation, Fabrication*, and *The Artifact.* For each category the respective aspects and challenges are named and the corresponding statements of the WikiHouse project and/or the situation of the WikiHouse project are discussed.

3.2 Tools and Technical Infrastructure

In this section, I will discuss the WikiHouse project's decisions regarding its technical infrastructure. In part two, we have seen that the following aspects and topics are important in this respect: (a) *editing tools,* (b) *collaboration tools*, (c) *repository solutions and version control tools*, (d) *expectations regarding the impact of tools*, above all regarding *mass-customization and the integration of laypeople*. We have also seen that free & open source infrastructure, environments and tools, while preferable, are still rare, so that finding appropriate infrastructure providers, environments and tools is a challenge in and of itself. In the following, we will have a look at how the WikiHouse project deals with this situation.

I start with (a) *editing tools*. The founders of the WikiHouse project decided to opt for SketchUp,[23] a 3D modeling software that is rather popular amongst professionals as well as hobbyists, as the preferred editing tool. They not only recommend that software for creating and saving designs, but also have developed their own WikiHouse plugin (which is discussed in more detail below) exclusively for that software.[24] They have done so, they argue, "simply because it is free and relatively easy to learn and use" (WikiHouse, 2013c). When SketchUp is called "free", this refers to the fact that the basic version of that software, SketchUp Make, can be downloaded free of charge. This must not be confused with a software product being free & open source (which would require the source code of that software to be available for everybody to inspect and build upon). SketchUp is proprietary software. Still, SketchUp Make is available for free and easy to use, lowering the threshold to participate and allowing many people to easily view and edit design files.[25] Considering the 3D modeling software available to date, the decision of the WikiHouse project is reasonable. There are almost no free & open source 3D CAD applications that are mature enough to provide the necessary features. The most advanced 3D modeling software is Blender,[26] a mesh-modeling software, which offers many features and is fit for professional usage, but is rather difficult to use. Of course, settling for a particular tool has consequences (i.e., beyond lowering the threshold for participation). What are these consequences?

First, there are problems regarding compatibility between different software products. These are typical for proprietary software and their corresponding proprietary file formats.[27] When you save a design in the file format of software product A, then you will often will have problems in opening or editing it with software product B. A post in the WikiHouse forums reports such a case, which hindered people to contribute their designs to the Open Library of the WikiHouse project:

23. SketchUp Make is Freeware, SketchUp Pro is a commercial software product. Neither SketchUp Make nor SketchUp Pro is a free & open source software product.

24. This plugin is in the public domain and available online at: https://github.com/tav/wikihouse-plugin

25. Provided they use either Microsoft Windows or Apple Mac OS X. There is no Linux version.

26. See https://www.blender.org

27. However there is a trend towards open standardized file formats in proprietary software.

> We are using rhino for our ongoing modelling now and have imported our original SketchUp model of our latest release in as we needed to fillet some corners to specific radii. When we exported the rhino model back to SketchUp we found that the WikiHouse plug-in would not generate the cutting sheets thus making the SketchUp model useless for publishing into the library. (Fisher, Parvin, Luff, & Squires, 2013)

The WikiHouse Design Guide version 3.0 also mentions this issue:

> Teams are using all kinds of [modeling] software […]. Please do share these [files] too in the [WikiHouse forum] groups, but if possible export to the SketchUp library as an exchange/sharing format. (WikiHouse, 2013c)

A second consequence becomes apparent when we consider algorithmic and parametric modeling. At the time, a very popular software for algorithmic design is Grasshopper[28] for Rhino[29]. A lot of architects have learned to use this graphic programing language, which however only works with Rhino, but are unable to, for example, use programing languages like Python or Ruby, which are supported by many other applications (including SketchUp, which supports Ruby). That being so, SketchUp is not an ideal choice for algorithmic design. What is more, SketchUp supports neither parametric modeling nor BIM by default (though plugins from others vendors are available). Hence, it will be difficult to realize the aim of a parametric WikiHouse using SketchUp (the WikiHouse project would have to implement this feature via its plugin).

Regarding (b) *tools for collaboration*, we have seen, in part two, that Ratti et al. (2011) speak of "BIM and related collaboration tools and practices". The tool that the WikiHouse project has chosen to organize collaboration is a web platform (2013c, 2013d, 2014g, 2014m). They aim for a "web platform for sharing and collaboration through the commons" (WikiHouse, 2013d). Consequently, creating an online platform is a declared development focus of WikiHouse (2013d, 2014g). They want this platform to provide three types of functions – it should facilitate: (1) sharing, collaborative edit-

28. See http://www.grasshopper3d.com

29. See https://www.rhino3d.com

ing and online collaboration on designs, (2) governance and communication, and (3) a framework for financial and economic endeavors. All of these functions concern collaboration, only the kind of collaboration differs. I will focus on the functions that are subsumed under (1) sharing, collaborative editing and online collaboration, for requirements on how (2) governance and communication of a project are organized and (3) financial and economic endeavors are beyond the scope of my thesis. The WikiHouse project (2014g) hopes to develop the following kinds of tools/platforms that fall under (1): a "model repo" (a repository for models), an "App library" (a repository for parametric apps) and a Wiki. This Wiki should have "Wiki-pages for hardware [–] A clean, editable wiki page & catalogue structure for clean sharing of design solutions, with overall CMS [content management system] and editing protocols", with pages that include, among other things, comment and feedback function. Moreover that platform should provide "indexing for a Wikipedia of things [–] A fully developed and comprehensive system for indexing [...] design solutions" and "a Github of things [–] Engines to manage pushing/pulling and 'diffing' of files through forks". These development goals are assigned to different development stages. The WikiHouse project is still in an early development phase and within the points grouped under (1) only the model repository is scheduled for this early phase. It actually has already been realized, on the old website. Yet, the WikiHouse project notes, in a document on the history, present and future of the project from early 2014, that "the website and platform is [sic!] still debilitatingly basic, severely limiting the ability to share files and collaborate, resulting in too much of the project still flowing through inboxes. There is an urgent need [...] to build a better web collaboration space [...] and to develop the first iteration of the 'Wikipedia for stuff', with full classification and documentation" (WikiHouse, 2014h, section 2). Lately, in 2014, the project changed its repository system, now using Google Drive as a repository for 3D design files as well as for other file types. This Google Drive repository is called the "WikiHouse Commons". The WikiHouse Commons seem to be a good enough solution for the time being. The other points mentioned above are still open challenges. This indicates that they are somewhat difficult to implement. And there are at least for some no appropriate ready-made solutions to date.

I will now take a closer look at another part of the infrastructure of free & open source projects: (c) *repository solutions and version control tools*. That is, the tools that are used to store information and track changes. Raasch et al. (2008) argue that an infra-

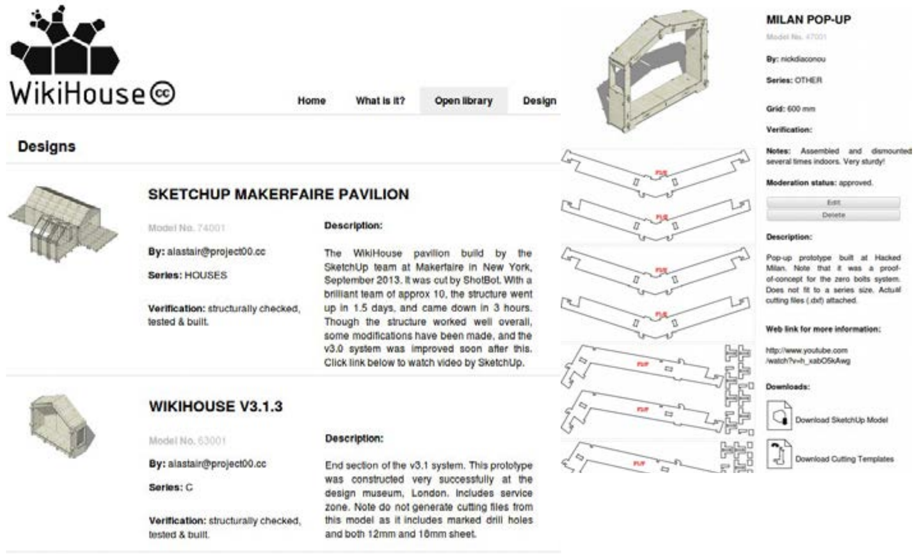

Figure 7, left: Presentation of uploaded models in the Open Library.
Right: Detailed view of a model stored in the Open Library
(WikiHouse, 2014b, 2014e)

structure that enables distributed and collaborative working, a place where all information on designs is stored, so that each member of the community can build upon the work and ideas of all others, is pivotal.

When I first visited the WikiHouse homepage, the project used their own online storing system to publish and store models. They called it Open Library (cf. WikiHouse, 2014b). In the Open Library only SketchUp models that were uploaded via the WikiHouse SketchUp plugin could be stored. These models could be downloaded either via the plugin or the Open Library on the homepage (cf. figure 7). This process relied heavily on the plugin to act as an interface between SketchUp and the Open Library. If the upload via the plugin failed, then there was no alternative way to upload a design. That said, the Open Library also made it easy to get a first impression of the uploaded SketchUp models, for it provided semi-automatically generated preview picture and a short description for each model. Hence, the Open Library allowed some control over what was uploaded. What was not accepted by the plugin could not find its way into the library. However, this also made it difficult to include additional data connected to a model, for example, spreadsheets or documentation. What is more, since the Open Library software had been developed by the WikiHouse project team, it also had to be maintained by the WikiHouse project team, and software maintenance tends to eat up a significant amount of resources.

Now, after having launched a new homepage, the project has moved the 3D models and all other data that is relevant for the community to Google Drive. This new Google Drive repository is called the "WikiHouse Commons", access to which can be requested via the WikiHouse homepage (WikiHouse, 2015c). Google Drive is a ready to use storing solution, is maintained by a professional company, allows for storing all kinds of information and even provides basic office tools that are tailored for online collaboration. WikiHouse notes in its introduction of the WikiHouse commons: "You don't have to store everything relating to your project in the WikiHouse Commons, but it can be very helpful to enable collaboration" (WikiHouse, 2014m). This seems to be a big advantage. However, Google Drive is, at its core, a folder system, and folder systems tend to be difficult to document and, as a consequence, hard to navigate for the uninitiated. Yet, all things considered, Google drive seems to be a good enough solution for the time being.

So, the WikiHouse project has found a workable repository solution, but what about version management? Google Drive is a less than ideal solution for version management, in particular when used with file formats other than those provided by Google Drive, such as SketchUp (and SketchUp does not offer strong version management either). This is because, if data is not stored in a native Google Drive format, then Google Drives only stores the last 100 versions and all versions that were created within the last 30 days. This may seem more than sufficient, but many people use an auto-save feature to save the file that they are currently working on regularly, to the effect that the 100 version limit may be reached within days. Having said that, the kind of version control that a lot of people do manually, namely, saving a new version every time they work on a file, keeping track of the date and the version in the filename, may be sufficient for the time being, but for the long term a more sophisticated form of version control, one that is able to highlight changes in a design, that allows users easily accept or reject such changes and so on would be useful. Such would be workable solutions for collaborating online that are tailored to for design teams and free & open architecture. This would probably require not only version management, but also complete project hosting.

Put simply, an important part of working together in teams is not to step on each others toes. This can be achieved by making projects modular (as will be discussed later) and by using strong version management.

Which (d) *expectations regarding the impact of tools* does the WikiHouse project have, in particular regarding *mass-customization and the integration of laypeople*? We have already seen that the WikiHouse project wants to parametrize its design in the long term. This reminds us of the possibilities that Raasch et al. (2011) see for free & open architecture. As noted in part two, Ratti et al. (2011) hope that algorithmic design could foster mass customization and the inclusion of laypeople in the design decision making process; the latter point is seconded by Vardouli and Buechley (2014) as well.

When we look at the decisions that the WikiHouse project has made regarding its tools, from the perspective of making the threshold for using these designs and participating in the design and the development process as low as possible, then SketchUp is a good choice for an editing tool. SketchUp is easy to use, even for people who are not familiar with 3D modeling. Hence, the modeling software is no significant barrier for laypeople. However, the kind of parametric design that is imagined by Ratti et al. cannot yet be done with SketchUp (as we have already seen above). Having said that, the WikiHouse project aims to provide a web interface for parametrization, naming SpokeCreator, which has in the meantime been renamed to "MatterMachine", as a model (Kettle, 2013; WikiHouse, 2014l). There have also been experiments with Rhino and Grasshopper (Taylor, 2012).

At this point, I think it is important to emphasize that, at least at the current stage of development, WikiHouse as an idea is not dependent on any particular piece of software. As has already been mentioned, designers use all kind of software tools to design with the WikiHouse system. What needs to be considered however, is how it can be made sure that models can be shared – if software tools are not freely available; for example, while professional architects usually have access to Rhino, which is quite expensive, laypeople usual have not. If the WikiHouse projects aims to be open to everybody, which of course is one of their declared goals, and if they want to avoid that their community is split in two camps, professionals and hobbyists, then the project must be careful which software is chosen and how new features, among them the parametrization of designs, are implemented. If parametric design is implemented in a way that precludes access by laypeople, then this would run counter to the goal to empower laypeople to make design decisions, a goal that of course is currently thought to be furthered by parametrization.

3.3 Information Transfer and Legibility

This section deals with information transfer and legibility, for which the following topics and challenges have been found in part two (please not that I have reversed the order of presentation in this part): (a) *documentation* and (b) *standards*.

I first will have a look at (a) *documentation*. To do so, I have a look at the documentation provided by the project, that is, before and after the launch of the new WikiHouse website and the WikiHouse Commons in summer/autumn 2014. However, I did not review documents published later than June 2015. I will only briefly describe what kinds of documentation are available and which subjects the available documentation covers, for an in-depth analysis would only be mildly interesting and contribute little to my research topic.

I will now proceed to the documents and information that the WikiHouse project offers. A very general introduction to the project is given right on the start page of the WikiHouse website, where a couple of slides explain what WikiHouse is and how a WikiHouse is basically made, using schematic drawings (WikiHouse, 2013a, 2015a). This introduction was on the old website and can still be found on the new one, in a slightly different version. The new website also has a section for frequently asked questions (FAQs), aimed at people or companies that are interested in building or designing a WikiHouse as well as at those who are interested in getting involved with the project.

The Design Principles, which also can be found on the old as well as the new website (with some adaptions), are basic statements about the ideals that WikiHouse wants to represent and, at the same, advice on how to design (2014c, p. 5, 2014d, 2015a). The ideals of the WikiHouse project, generally speaking, are those of the free & open movement (cf. part 1) as well as ecological ones (cf. the section on the artifact below).

A more general documentation for designers who want to use the WikiHouse building system is found in the Design Guide (as opposed to the Design *Principles*) and the Modeling Standards. The Design Guide (WikiHouse, 2013c, 2014d) explains how to design with the WikiHouse system. For example, it recommends reasonable gauges and spans and shows how frames and joints are designed. The Modeling Standards (WikiHouse, 2014f) are about configuring and structuring the drawings in a way that (1) makes 3D models easy to understand and easy to work with for other people and that (2) allows them to be parsed by the WikiHouse SketchUp plugin, so that design information can be processed correctly. For example, the Modeling Standards cover how parts and layers should be named and how precise drawings need to be. Interestingly, only the Design Guide has been transferred to the WikiHouse Commons (WikiHouse, 2015f), whereas the Modeling Standards that could be found on the old WikiHouse website have, by spring 2015[30], been transferred neither to the new website nor the WikiHouse Commons.

During the data collection phase of my work on this thesis, only limited documentation was available for individual WikiHouse models.[31] That is, a short description of each model can be found on the old website (WikiHouse, 2013b). In the WikiHouse Commons, where each individual WikiHouse design project gets a project folder, there are, as of April 2015, only subfolders for manuals, the vast majority of which are empty (WikiHouse, 2014m).

There are also several documents on the organizational structures of the WikiHouse project (WikiHouse, 2014c, pp. 3–4, 2014h, section 3, 2014i, 2014j). The structure of the WikiHouse Foundation is documented in the WikiHouse Constitution, which has only been added with the new website. The constitution also covers legal terms, legal conditions, licenses and trademark policies. (WikiHouse, 2014c, pp. 6–9).

A link to the development goals of WikiHouse also can be found in the constitution (WikiHouse, 2014g, p. 10). These goals are not particularly detailed, restricting themselves to milestones. The WikiHouse project also keeps a challenge map (WikiHouse, 2015h), which lists long-term challenges or development ideas, only some of which

30. Now, in autumn 2014 a Design Guide for the latest version of the construction system has been published in the WikiHouse Commons (see WikiHouse Foundation 2015b). Modeling Standards do not exist in the WikiHouse Commons, to date.

31. In the meantime IKEA-like assembly instructions have been added for some WikiHouse models (Parvin, Gold, & WikiHouse Design Team 00, 2014) and a general guide for assemblers has also been released (see WikiHouse, 2015f).

are also development goals. The WikiHouse project provides no information on how the Development Goals and the Challenge Map are related. Presumably, the Development Goals lists more important objectives, whereas the Challenge Map lists long term objectives that are less pivotal to the core project, many of which concern other fields than architecture.

Finally, there is also documentation on how to use the WikiHouse Commons (WikiHouse, 2014m), for example, where to find and store files or how to name them.

To sum up, I found: (1) A general introduction and FAQs, (2) Design Principles, (3) a Design Guide, (4) the Modeling Standards, (5) (the aim for) manuals for individual WikiHouses, (6) information on organizational structures, (7) information on legal aspects, and (8) instructions on how to use the WikiHouse Commons. I will now assess whether these documents address the challenges that I have identified in part two. To do so, it will be helpful to remind ourselves that different kinds of documentation address different target audiences, namely, in the case of WikiHouse: users, designers and developers. By "users" I mean those who download existing designs to manufacture and build them. By "designers" I mean those who use the WikiHouse construction system to design tangible artifacts. And by "developers" I mean those who get involved in the project with the purpose to contribute to the development of the overall building system. I will now assess the quality of the documentation for each of these groups in turn.

Users need documentation on the object level, that is, in the case of a WikiHouse design, a fabrication and assembly manual. Additional information that is important for users are the terms of use, licenses and legal disclaimers as well as information on how to get help from the community and or to find associated professionals. I have already said that there were no assembly manuals at the time at which I collected my data, though the WikiHouse project has improved in this matter in the meantime (not so far as to provide manuals for all models however). Without manuals the users are left with the 3D model, which provides some orientation, but fails to provide step by step instruction. What is more, such instruction may not be easily derivable from the model (at least for laypeople), and if users, so to speak, start at the wrong end of the 3D puzzle, then this can complicate assembly.

Designers need information that covers the essentials of how to design within the project and how to use the building system as well as Modeling Standards. The Wiki-

House project, by and large, provides (or used to provide) that documentation in their Design Principles, their Design Guide and their Modeling Standards. However, the Modeling Standards appear to have been discontinued; they could be found on the old WikiHouse website, but have vanished from the new one. What is more, there are no standards for designers on how *they* should document their work, which may lead to a lack of (good) documentation in the long term.

What kind of information would be needed by developers? Of course, in order to develop and improve a system, one needs to know how it is (supposed to be) used, so the documentation for users and designers may also be interesting for developers. Needless to say, however, developers need more information than that, they need: (1) information about previous developments that allows them to understand previous decisions and the path that development took; (2) standards for how they should document their research and their developments;[32] (3) documentation on how to do testing, in particular whether there is any protocol that should or needs to be followed;[33] (4) a roadmap of future design goals and their priority to the project; (5) information about how the project is structured and governed. The WikiHouse project only has a roadmap (i.e., their Development Goals and Challenges Map, thus meeting (4)), and limited information about project governance (in their constitution, partially meeting (5)). This may be the cause why development in the WikiHouse project is done in different branches that work in parallel and pursue different goals. For without the necessary documentation is it rather difficult to attain an overview of the state of development in general or to acquire a better understanding of how the developments of the different branches are connected, particularly for outsiders or newcomers.

To conclude, documentation in the WikiHouse is rare, especially documentation that targets developers. This could be improved by implementing standards on documentation, covering what needs to be documented (test results, arguments, decisions, activities, etc.) and who is responsible for doing so. More transparent development processes and better information on how people can contribute to the project seem to be required, even more so since the WikiHouse project is a free & open source project

32. Gibb, Adabie, and Baafi (2014, chap. 14 and appendix E) give some advice on how to deal with this kind of documentation. However, as Marcin (2015) notes on the OSE project blog "[…] a coherent set of development standards is yet to be defined".

33. For many technical procedures there are national or international standards that should be considered. These standards need to be highlighted by documentation; particularly in a project where laypeople or professional from fields other than architecture may contribute.

that, on the one hand, has to attract contributors and, on the other hand, holds the potential for distributed development. Without proper documentation, doing either will be increasingly difficult in the long run.

I have just mentioned (b) *standards* in the context of documentation and will now analyze how the WikiHouse project discusses them and what standards they have adopted within the project itself. As I have already mentioned, some of the sources that I am using here are from the previous version of the WikiHouse website and could neither be found on the new website nor in the WikiHouse Commons at the time I have finished my data collection. The WikiHouse Foundation wants to "[…] to provide a lean framework for an open community; setting simple rules, protocols and standards that allow anyone to 'do'" (WikiHouse, 2014c, p. 3). They also say, in their design principles: "Open standards[.] Share and make shareable. Where possible, work to existing standards" (WikiHouse, 2014c, p. 5, 2014d, 2015a). However, these two statements are very abstract in nature. They get more concrete only in three areas: (1) naming conventions in the WikiHouse commons, (2) their modeling standards (which, however, have been discontinued); and (3) file formats.

The WikiHouse project has adopted naming conventions in the WikiHouse Commons because "[…] standardising file management across project makes it much easier for teams to collaborate […]" (WikiHouse, 2014m, section "Welcome to the WikiHouse Commons"). Apart from that and the very fact that they exist, however, there is little to say about these naming conventions.

There used to be Modeling Standards (WikiHouse, 2014f) on the website of the WikiHouse project, but those were not migrated to the WikiHouse Commons. These standards mainly provided guidance on how information in design file should be structured and how design information needs to be prepared in order for it to meet the requirements of the plugin (i.e. work precision, grouping and components, parts and layer naming, notes on unify visualization, notes on the plugin's function and reasons for script errors).[34]

I have already discussed file formats earlier (in the section on tools and technical infrastructure), but since standardized file formats are pivotal for the legibility of designs, I mention the role file formats as standards here, if briefly. As already noted, the

34. In the Design Guide v4.2.1, which has been published after I had finished my data collection, some of the information found in the former modeling standards has been included.

WikiHouse projects suggests to use open standards wherever possible. However, the file formats for digital designs that are currently used do not comply to such an open standard.[35]

To conclude, the WikiHouse project underlines the general importance of standards for sharing designs collaboration. This includes general standards as well as standards defined within the project. When it comes to the use of open standards, however, the aim for openness conflicts with the pragmatic decisions WikiHouse has made regarding its design tools. There is little to say about the standards that have been defined within the project, since they are few and not too extensive. This is not too surprising, since the project as a whole is still in an early development phase. Probably, the current practices are not yet stable enough to be standardized. What is surprising to me is there are no standards for documentation, as I have already pointed out above.

3.4 The Process

Many free & open source projects understand their work as an ongoing development process. This section looks for such approaches in the WikiHouse project. I will do this by confronting the WikiHouse project with the topics that I have found in the analysis, which I presented in part two of this thesis. Those topics are: (a) *design as an evolutionary process* and (b) *feedback*.

The first of these two topics I will look at, is (a) design as an evolutionary process. In its constitution, the WikiHouse foundation writes: "Wikihouse [sic!] is an experimental, continually developing concept in which designs, drawings, and information […] are made available under open source licence to individuals […] (WikiHouse, 2014c). And in a press release from Winter 2014 they state: "WikiHouse provides an open platform for inventors, designers and citizens to build a global commons: a continuously improving library, or 'Wikipedia for things', of low-cost, high-performance

35. However, the specifications for the Drawing Interchange Format (DXF), which has been developed by Autodesk and is used for cutting files (cf. e.g. WikiHouse 2015e, section "download files"), are available on the website of Autodesk (Autodesk, 2011).

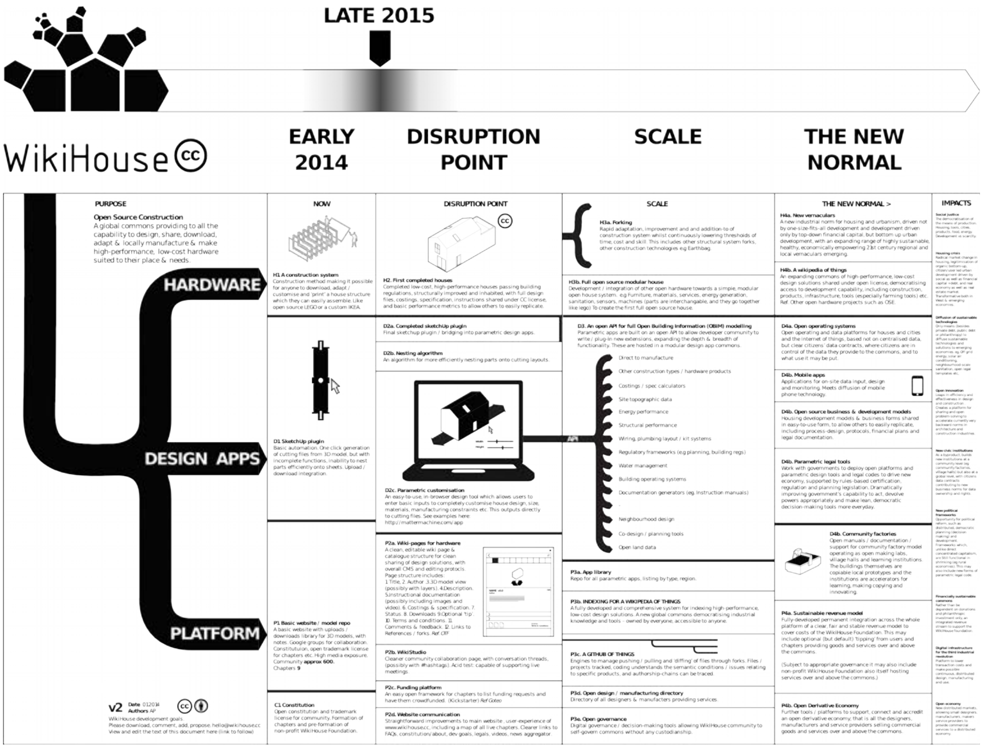

Figure 8: Development goals v2 of the WikiHouse project, with approximate development state marked (figure adopted from WikiHouse, 2014g)

design solutions for housing, neighbourhoods and cities" (WikiHouse, 2014a). Furthermore, in its challenge map, the project notes: "The power of open source is that once solved, each problem will always be solved for everyone, forever; and will continue to evolve as it is improved and adapted" (WikiHouse, 2015h). The key phrases I want to draw attention to here are: *continually developing concept, continuously improving* and *will continue to evolve*. The WikiHouse projects sees itself as a long term project that evolves over time. Looking at the WikiHouse development goals, shown in figure 8, one realizes that the project is still in its early development phase (WikiHouse, 2013c, p. 11, 2013d, 2014g). That design and its development are evolutionary processes in the WikiHouse project is beyond doubt. This is also illustrated by the version map the project created in 2012 (see figure 9).

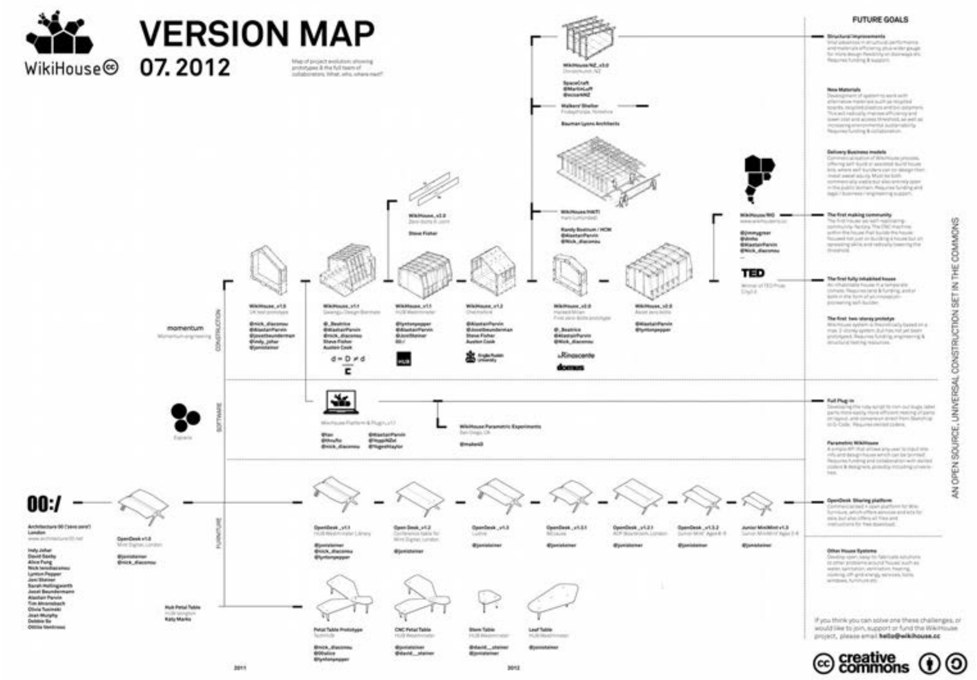

Figure 9: WikiHouse development tree, development level summer 2012.

What is more, the project is explicitly inviting improvements and understand its work as work in progress. With regard to the project's development goals they note, for example: "This document is permanently unfinished, but shared for all to see. If you have ideas or would like to improve it, please join the WikiHouse project group" (WikiHouse, 2013d). People who use information provided by the WikiHouse project are at various places encouraged to participate and improve them. For example, WikiHouse states: "All designers are invited not only to use these rules, but to develop and change them as they improve on the construction set" (WikiHouse, 2014d) and "[…] if you can help us improve it [the plugin] faster, please do join the WikiHouse software group" (WikiHouse, 2013c).

However, there is little information on how this evolutionary process is (or shall be) organized, save for the aim to develop a GitHub-like version management system for design files (WikiHouse, 2013d, 2014g), that is: "A GITHUB OF THINGS[:] Engines to manage pushing/pulling and 'diffing' of files through forks. Files/projects tracked, coding understands the semantic conditions/issues relating to specific products, and authorship-chains can be traced" (WikiHouse, 2014g). But this is only a goal. Yet, even if there are no predefined, transparent processes in place, using Google Drive for the WikiHouse Commons made it possible that everybody can post her or his improvements – in whatever form they deem reasonable.

(b) *Feedback* is hardly ever mentioned by the WikiHouse project. They only state that they want the pages of the yet to be realized Wiki for hardware (cf. above) to include comment and feedback mechanisms and to have reporting tools included in their online platform (2013d, 2014g). Typical feedback mechanisms that are known from free & open source software projects would be bug trackers, which allow to keep track of everything from error reports (i.e., "bugs") to feature requests. That said, since what I found on feedback is too thin a basis for an informed comment, I will conclude this subsection here.

To sum up, the WikiHouse project understands design as an evolutionary process. The project strives to evolve over time. Yet, when we look for formal processes that support collaborative development, iteration and improvement, we find that no such processes exist. The situation for feedback is similar, if somewhat worse.

3.5 Modularity

In part two the following topics have been found for modularity: (a) *Properties and benefits of modular designs and artifacts* and the (b) different conditions for *modularity in software and hardware*. As I also have mentioned in part two, modularity is a concept that can be used in digital design, in the production processes and in the composition of tangible objects. Moreover, it can be applied in software as well as in hardware production. So when we talk about modularity it is important to make clear which modularity is actually meant. That being so, I should provide a definition of what I mean by these terms in the following. I will call anything a "module" that (a) can be a *part* of a greater whole that performs some function (broadly defined), (b) can perform (one or more) *functions* that contribute to the overall function of the whole and (c) can

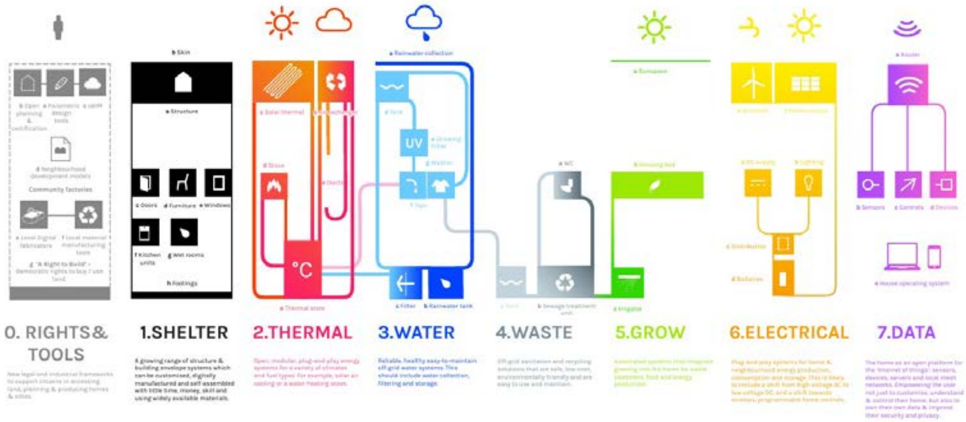

Figure 10: The WikiHouse Catalogue: The WikiHouse Challenge Map – WikiHouseMap v4 (WikiHouse, 2015h)

do so in a largely *self-contained* way, where "self-contained" means that it can perform its functions (i) with no or only minor side-effects, (ii) under conditions that can be met easily and that it can be (iii) easily replaced. I will correspondingly call the property of being, to a significant extent, made up of such modules the property of being "modular". These definitions are intentionally broad, since they should cover modules in the digital as well as in the physical realm.

I will investigate how the WikiHouse project approaches topics and challenges concerning the (a) *properties and benefits of modular designs and artifacts* by analyzing the discussions of modularity within the project as well as their designs and tangible artifacts. I should, in this context, point out that the WikiHouse project not only develops and designs tangible artifacts, but it is also involved in software production and, accordingly, discusses modularity in both contexts. Hence, I will also briefly discuss their notes about software, before focusing on their design and tangible artifacts.

The WikiHouse project currently maintains only one piece of software continuously, namely, its plugin for SketchUp. Moreover the WikiHouse project, in the their development goals, version 2, aims for two other pieces of software that have yet to be developed, namely, (1) an advanced programming interface (API) that enables other people to easily write software that allows for parametric customization of the

WikiHouse designs and (2) an in-browser tool that is based on that API and does so (WikiHouse, 2014g). In general, software and hardware should be designed to be modular according to the WikiHouse design principles. The projects aims for "hardware and software that is interoperable, product-agnostic and flexible, so elements can be independently altered, substituted, mended or improved" (WikiHouse, 2014c, 2015b). Regarding hardware, according to the development goals, version 2, the goal is a "modular open house system", in which "parts are interchangeable" and "go together like lego" (WikiHouse, 2014g). They also mention the development of modular and open plug-and-play systems (e.g. solar housing technology systems) as an open challenge in their challenge map, which is shown in figure 10 (WikiHouse, 2015h).

To get an overview over the modules that the WikiHouse project envisions for the future, we can look at the WikiHouse challenge map, where seven thematic modules are defined: (0) rights and tools, (1) shelter, (2) thermal, (3) water, (4) waste, (5) grow, (6) electrical and (7) data. Each of these modules addresses another aspect of (or concerning) construction and technology – and therefore different people, with different qualifications and knowledge. Moreover, each of those modules is divided into further modules, and all of these modules are well suited to be developed separately. For example, the (1) shelter module comprises the following sub-modules: (a) structure, (b) skin, (c) doors, (d) furniture, (e) windows, (f) kitchen units (g) wet rooms and (h) footings. Most design and development efforts at the moment focus on module (1), the shelter, and within this module on the sub-module (a), the structure. I am first and foremost interested in the role that modularity plays in these design and developments efforts. To be specific, all buildings that have been been designed by the WikiHouse project to date, we could call them "WikiHouses", have a timber structure at their core. This timber structure acts as a kind of chassis module and is then complemented by other modules in order to form a complete house – I will focus on this core structure. That is, wherever in the following I talk about the WikiHouse "building system" or "construction system", I talk about this chassis or core structure, rather than other elements such as the footing, skin, windows, etc. I will have a look both at the structure of actual, tangible artifacts and the structure of the respective designs. Hence, I will in the following focus on this sub-module.

First, I will have a look on the modularity of the tangible artifacts. What does modularity of the chassis structure look like in the WikiHouse project and what advantages does it offer?

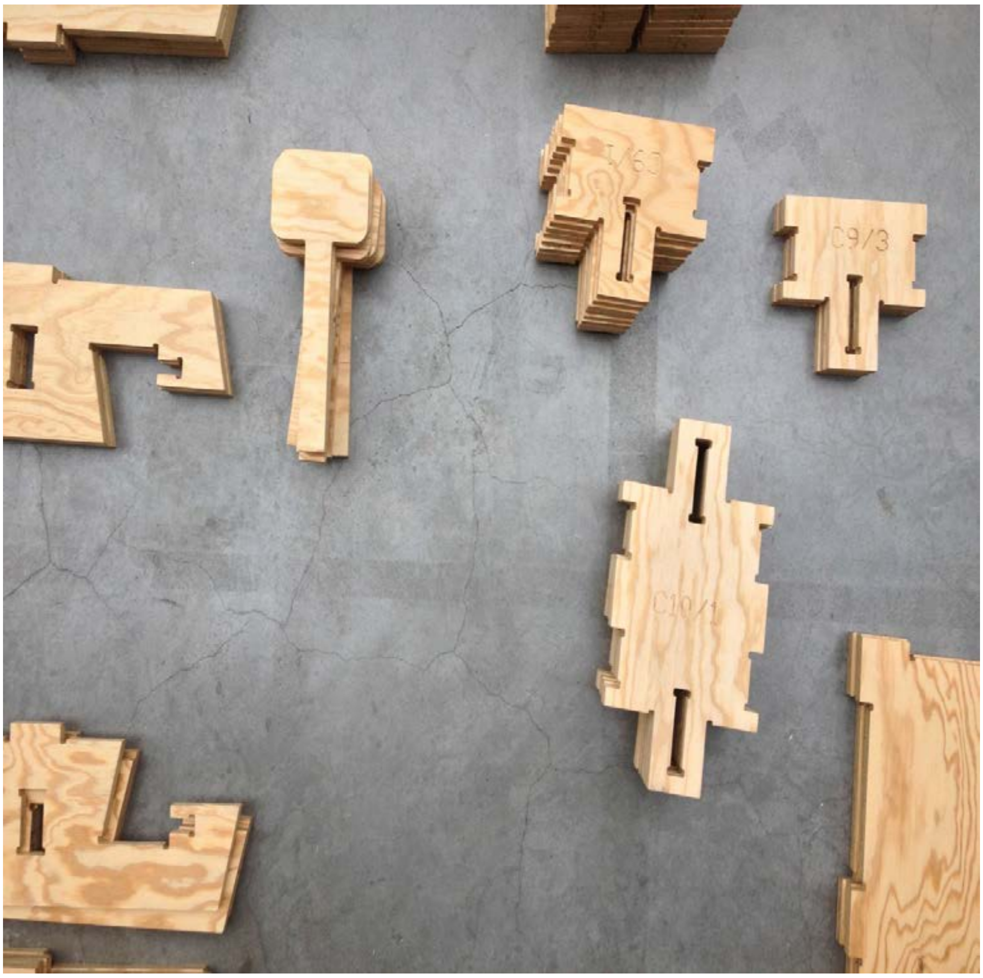

Figure 11: Examples of the plywood elements used to construct a WikiHouse, the maximum size of a single element equals the size of the used standard sheet material (1200mm × 2400mm) (WikiHouse & Chatainger, 2013).

The building system that has been developed by the WikiHouse project uses small elements to form larger timber frame[36] elements (see figures 11 and 12). These timber

36. I am using the term “frame” here because the WikiHouse project calls these elements by that name.

Figure 12: Fin of a two story WikiHouse building, assembled from plywood elements (WikiHouse & Knight, 2015).

frames are connected to a linearly arranged frame system. Drawing upon my broad definition of modularity, a system is modular if it is, to a significant extent, made up of parts that perform functions and do so largely by themselves. If we apply this definition, we will find that buildings that are constructed using the WikiHouse system are modular. Small plywood elements are used to make timber frames and the timber frames are again elements of the basic WikiHouse construction. In case of the WikiHouse system, that fact that the whole house is built of small plywood elements also has the advantage that all parts fit into a standard sized van. Also, the single elements are quite handy. What is more, the prefabrication of the basic elements can be done in a garage sized fabrication facility. All of this contributes to making the WikiHouse system more accessible, since small fabrication facilities are more likely to be available in the neighborhood and standard sized vans are available to more people than large vehicles. Therefore, the way in which the modules of WikiHouses are designed contributes to realizing a free & open architecture, lowering the threshold for participation and enlarging the number of potential contributors and developers. However, many of the aspects and qualities of tangible modular systems that have been identified are not present in the WikiHouse structure, at least not on this level, namely that modules should be easy to adapt and replace, in order for the whole system to be flexible and easy to repair as well as in order to allow for a high degree of control by the end user. And this, sure enough, cannot be said of a timber frame that is simply put together from smaller timber parts. That said, this kind of modularity can be found in the system presented in the challenge map (figure 10). There, the overall structure is made out of many modules that together form a complete house system, a structure module that together with a skin module, footing modules, kitchen modules etc. forms a basic shelter, which can then be complemented with other modules to provide further functions. Of course, this kind of modularity is, in fact, a standard approach in architecture in general.

I will now move on the modularity of the digital designs. Again, I am mainly interested in the basic structure of a WikiHouse. What modules can be found in the digital design? How does modularity in these designs look like? What advantages does it offer? I will, in search for the answers to these questions, first have a look at the big picture, before proceeding to the details. Doing so, I will draw on the digital modules that I have in found, one, in the so-called Open Library, the old repository for WikiHouse SketchUp models, which has been discontinued with the launch of their new homepage, and, two, the WikiHouse Commons, the current, Google Drive based repository for these models.

However, before I move on, I should clarify why and wherein the modularity of the digital design differs from that of actual, tangible WikiHouses. After all, one might wonder why there should be different modules in the digital and the physical realm, if the latter mirrors the former. So allow me to explain this briefly. When you try to split something up into modules, you aim to reduce interrelations between those modules. Hence, if you are designing a tangible artifact, you need to respect certain rules for how modules work in the physical realm. However, as Baldwin and Clark (2000) argue, the modularity of the design itself (i.e., the drawings, the digital model, etc.) follows other rules entirely. For example, you may want your design to allow for easy replacement of parts and you may want to design in a way that allows different people to work on different parts of the design simultaneously and without (too

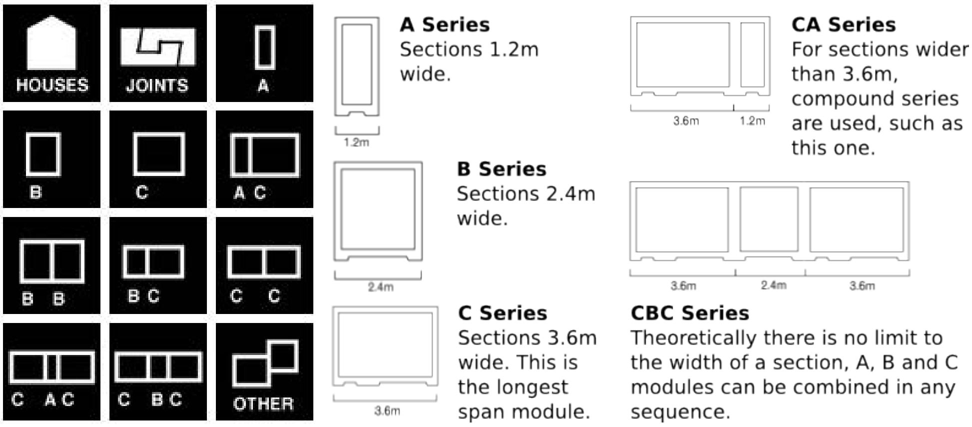

Figure 13: WikiHouse, Open Library, Series (WikiHouse, 2013b, 2014d)

much) coordination, you may want to reuse older designs, etc. How does this play out for the designs of the WikiHouse project? We are about to find out.

In the Open Library digital models were grouped into categories called "series", with the following series available: (a) houses, (b) joints, (c) frame profile series, sub-categorized by their span (series A, B, C),[37] (d) combined types (AC, BB, BC, CC, CAC etc.) and (e) other, a category for everything that does not fit into any of the other series (see figure 13 and WikiHouse, 2013b). However, of these series, only (a) houses and (b) joints in the C sub-series of the house series as well as (e) others have actually been populated with designs. From the viewpoint of modularity, two of these series are of particular interest: (c) the frame profile series – for, as the drawings in figure 14 indicate, the objects in that series are meant to be combined in order to form larger objects; (b) the joints – for reasons that will become apparent later. I will now discuss these two in turn.

Just like the actual, tangible WikiHouses, the digital designs of the basic structure of WikiHouses are separated into smaller elements. This, of course, is not particularly surprising, since these designs are the blueprints for the plywood parts out of which

37. The A, B and C series were planed to be segments with fin widths of 1.2 m, 2.4 m and 3.6 m (WikiHouse, 2014d). This grid was chosen since "in most countries, sheet materials [eg. plywood sheets] come in 1200mm x 2400mm sheets" (WikiHouse, 2013c).

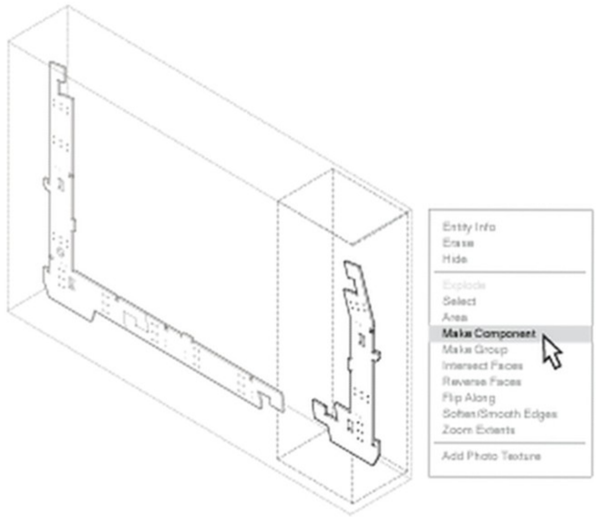

Figure 14: Parts grouped in the digital drawing in a way so that they represent building parts of the physical object (WikiHouse, 2014f)

WikiHouses are build. WikiHouses are in a way physical copies of these designs. The very fact that the WikiHouse project intended to have an A, a B, and a C series, even though they did not (yet) materialize, can be seen as evidence for modular design, since the objects in these series were intended to be combinable in order to build objects of (d) combined types; in fact, the C series has been called "the longest span module" (WikiHouse, 2014d). This notion of a combinable series mirrors the combination of different modules in the physical realm. Furthermore, a WikiHouse basically consist of a row of repeating frames. The frames in the digital model resemble their counterparts in the physical realm. So the modularity of the digital design resembles that of the tangible artifact down to the smallest elements. Again, this is not surprising. However, the modularity of the digital designs has other functions than that of the tangible artifact, namely: That the digital modules are exact counterparts of the plywood parts of the actual, tangible artifact is a condition for the WikiHouse SketchUp plugin to work (cf. figure 14). What is more, the project notes that "[t]his also saves a great deal of work when editing repeated components. You may also wish to then nest collections of parts (components) into further larger components, for example to re-use standard fin profiles" (WikiHouse, 2014f). What is more, this makes the models

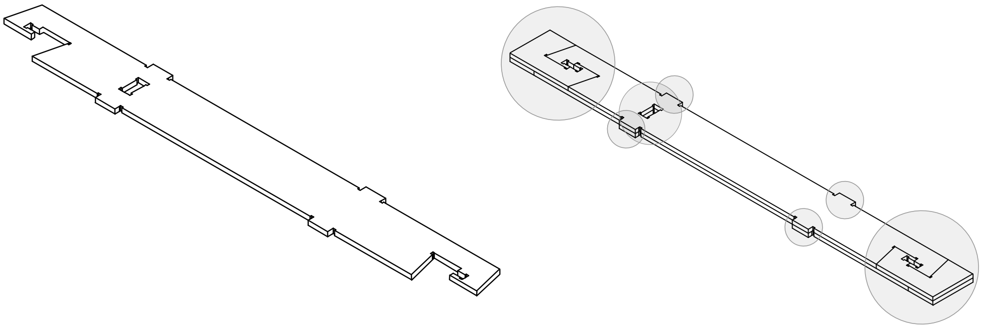

Figure 15, left: Example for a construction part of a fin designed for WikiHouse v3.0.
Right: Jointing points of a fin like they where used in the WikiHouse v3.0
(adopted from WikiHouse, 2013c, p. 4)

easier to understand, which is important because they do not only function as blueprints for digital production, but also as plans for human builders. Hence, every component of the physical artifact should be easily found and "grabbed" in the digital model (if you recall, the avatar-like relationship between the digital model and the physical artifact was underlined by Carpo). To sum up, we can say that (1) reusing digital modules or using them multiple times in the same design reduces the workload for designers. Moreover, (2) using digital modules that are exact counterparts of tangible ones is useful for digital fabrication and also (3) allows to use the digital model as a construction manual.

However, things get more interesting when we take a closer look at the (b) joints. They are, in the physical realm, inseparable parts of the plywood elements of which WikiHouses are assembled, but modules in the digital realm. Joints are essential parts of the WikiHouse construction system. The plywood elements that form the core construction need to be held together and, needless to say, the connections between them need to be of a certain strength. The category "joints", as I understand it, was meant as a repository for jointing solutions. Joints differ from the modules discussed previously, because they are purely *functional* modules that can be developed almost independently of the rest of the structure, yet are no independent parts in the actual, tangible artifact. In fact, a joint consists of parts of *different* tangible modules. Therefore, joints can be defined as modules in the digital model – but not in the tangible artifact (cf. figure 15). How important the digital modularity of the joints is to the WikiHouse project is evidenced by the fact that they are cited as an example for the kind of modularity that is needed in the WikiHouse working paper on implementing parametric design, where it is stated that "[a]s far as possible, molecules [i.e, elements of the WikiHouse construction system] should be 'modular'. So for example the 'S-joint' is a subscript which could be modified independently of the entire rest of the model" (WikiHouse, 2014l).

I will now move on to the (b) *different conditions for modularity in software and hardware*. However, while this may be a fruitful topic in (and for) theory, there is, as of yet, little to say about that difference in practice, save for acknowledging that it exists. Hence, this section will be brief. I have mentioned in part two that Abdelkafi et al. (2009) find that software has an advantage when it comes to the application of free & open methods since, so they argue, programming languages are a medium that fosters modularity; they even ask us to imagine a physical material that does the same, so that we get a better impression of that advantage. I have already pointed out that I find that analogy misleading and would suggest to rather compare programming languages to the design tools that we use (see the corresponding section in part two). We may, of course, ask ourselves what kind of materials foster modularity in the physical realm. However, this is beyond the scope of my thesis.

These are my findings on the modularity of the WikiHouse project. However, the difference between modularity of digital designs and tangible artifacts may use some further discussion. At first sight, it seems that the modules of a design are just the modules of the designed artifact and that the distinction of modularity in digital designs and modularity in tangible artifacts does not make much sense. Yet, upon closer examination, we have found that, when we think about how we can, for any given design, reduce the interrelations between its parts there are cases in which the modules in the design will be the same as in the designed artifact as well as cases in which they will be different. This is because the modularity of the design serves different purposes than the modularity of the artifact and, furthermore, because the extent to which objects can be divided into smaller parts may differ substantially. The smallest parts in which physical artifacts can reasonably be divided are, in case of the WikiHouse construction set, the single pieces out of which the fins are constructed (cf. figure 15). These units are also modules in the digital design (cf. figure 14). This, of course, is reasonable (for the reasons given above). However, if you consider the design process, that is, the altering, the adapting, the experimenting, etc., then you will find that a different kind of granularity may be useful, that there different rules according to which you can structure your design into modules. Of course, the rules according to which tangible artifacts are structured into modules still apply; for example you will still want to be able to easily exchange parts in the artifact or to use the same kind of part multiple times. However, in the digital realm, there are more things to keep in mind, for example, you may want to be able to design different parts of the artifact together with other designers in parallel, without needing to coordinate (too much) with these other designers. We can use the WikiHouse system as an example to illustrate this point and discuss what may have motivated the choice of design modules that we find in that system, focusing on the structural framework, the fins, the primary and secondary connectors, etc. Consider two scenarios: (i) reusing components of an older model for a new design, (ii) adapting an older model to make use of advancements that have been made in newer designs. For scenario (i), we will focus on things that are similar in different models (which have been designed using the same version of the building system). When we look at a single fin, these are not the parts that the fin is made of, but the way in which these parts are connected. When we look at the gauge, the gauge too is not necessarily always the same. The same can be said for the connectors. However, what will occur in all of these models (provided they use the same version of the building system) are the jointing points. For scenario (ii), consider that the joints are continuously under high development, to the effect that older designs often need to be updated to make use of the new jointing systems. However, the primary connectors, the fins, etc. change as well. Hence, it is useful to allow for all of them to be developed separately. This in turn makes joints ideal candidates for design modules.

3.6 Complexity and Simplicity

This section is about the way in which the WikiHouse project discusses complexity and simplicity, whether it applies to design itself or the way in which the design process is organized. Designs as well as projects as a whole can grow in complexity for various reasons, including growing inter-dependencies between different aspects of the project, unknown parameters, or simple the size of the project. In the analysis of part two, the topics (a) *effects of complexity* and (b) *simplicity* where found to be of interest. In the following I review the results of confronting these topics with WikiHouse.

When we talk about (a) *effects of complexity*, then the main effect that we need to talk about is that overburdening complexity can endanger the realization of projects (Abdelkafi et al., 2009; Raasch et al., 2008). Correspondingly, when people talk about complexity, they typically talk about how to deal with it. However, or so it seems to me, complexity is often difficult to anticipate in the abstract, rather it is discussed only when some particular aspect of a project is about to become too complex to handle. That may be the case with the WikiHouse project. At least, when I searched the core assets of the WikiHouse commons for the terms "complex" and "complexity", I found just a single entry dealing explicitly with complexity and design: a discussion of parametric design. Parametric design is an aim of WikiHouse project that has not yet been realized, though there are experiments to develop in-browser applications. Under the title "*How complex?*", in an early document on how to best approach the sub-project of parametrizing models, it is noted that this sub-project should start with rather simple parameters; with regard to the aim of having a fully parametric WikiHouse, it is stated that "[w]ithout making some things impossible, nothing at all is really possible! The basic first objectives should be to only expose very basic variables such as house length, width and roof shape" (WikiHouse, 2014l). That is, the WikiHouse project, at this point, takes into account that too much complexity might overstrain their resources and that narrowing down the scope for the time being, as suggested by Raasch et al. (2008), may help to reduce complexity. When we look at at the WikiHouse project in general, which we must, given that our topic at hand receives little discussion by the project team, the project, in my opinion, approaches the problem of complexity quite sensibly. While the overall visions are ambitious, their actual approach was (and is) down-to-earth. They started with small proof-of-concept structural prototypes, went on to develop a tiny house module as well as further building modules then continued to develop a complete house, albeit a small one, – and they keep on going (WikiHouse, 2013d). Keeping it essential, taking away everything that is not needed is generally a good approach, as long as one keeps in mind that customization needs to be possible. If customization is possible, then variants and additional features it can be designed by the users.

(b) *Simplicity*, understood as an approach to deal with complex systems, is discussed implicitly in different versions of the WikiHouse design principles (WikiHouse, 2013c, 2014c, 2015b) as well as in the design guide that could be found on the old homepage (WikiHouse, 2014d). All of these discussions take place in the context of advice on how to design. Most of them focus on the tangible artifact, but one also mentions the design process and documentation. “Design products, processes and documents that are accessible and intuitive” (WikiHouse, 2015b). The gist of that advice is that designs should be kept simple, so that they are easier to comprehend for laypeople who want to build and assemble a WikiHouse. The following are examples: “Try to design components which either make it impossible for the assembler to get it wrong or are designed in such a way that it doesn't matter if they do” (WikiHouse, 2014d). “The easier to dismantle structures or replace individual parts, the better” (WikiHouse, 2013c). Moreover, only the assemblers are mentioned as a particular group to keep in mind when it comes to making things easier in the modeling standards (WikiHouse, 2014f).

These are the considerations of complexity and simplicity that I have found in the WikiHouse project. Interestingly, none of the statements I found are about keeping things such as reusing, altering, or remixing the designs as easy as possible for other designers or developers. However, given that the focus of this thesis is the way in which free & open methods impact designs and the design process, I would like to take a closer look on simplicity of the digital design and the designing and development.

What does simplicity mean in digital design and for the design process? Needless to say, the digital design needs to be simple to understand, simple to use and simple to adapt as well. This requires, above all, that the designs are simple enough to be easily understood. So how easy to understand are the designs of the WikiHouse project? I found the following statement in a blog post on SketchThis.NET (Schimelpfenig, 2014). “I started out with one of the WikiHouse modules. Since I needed to resize it I ended up using their model as a guide, and I have drawn this entire model from scratch. I’ve re-worked many of the joints based on my experience of building one in

NYC" (Schimelpfenig, 2014). Indeed, after inspecting WikiHouse design files, I have often reached a similar conclusion, namely, that coming up with a new model that is based on these designs, which include re-drawing everything from scratch, may often be easier than adapting the existing model. This is not because the structure of the models was hard to comprehend, quite to the contrary, their structure is quite clear, but because the structure of the model was geared towards being useful as a building guide, rather than towards being easy to adapt.

We should also have a look at the overall concept of the WikiHouse building system. To get a better view, we need to go back in time, to 2013. Back then, the proposed building system and its documentation were clearly structured and simple. There were guidelines on how to design besides the digital models. These guidelines also explained many of the design decisions of the WikiHouse team (e.g. the position of the joints, seizes, grids, etc.). Moreover, there where modeling standards that contained basic guidelines for drawing and structuring digital designs. In short, understanding the design and how to design was quite easy. Since then, development has moved on and the first house can be downloaded, a big step forward. However, another thing that has changed is that design decisions are not that easy to understand anymore. Sure enough, the WikiHouse building system advanced, but how exactly it did so is hard to see at first sight. At the time I finished collecting data, no documentation bundle, which would contain design guides and explanations for the design decisions that have been made, exists for the newest version of the building system.[38] Just to avoid misinterpretation, the information that is provided by the WikiHouse project is quite sufficient to build a WikiHouse, but more information would be needed for the design to be easily comprehensible to other designers or for these designers to easily reuse, adapt or modify the WikiHouse designs. What is more, the structural design that has been developed by the WikiHouse project is quite complex, it is influenced by various interrelated parameters, most of which are far from obvious to outsiders. These parameters depend on various factors, for example, the materials used. How these parameters interact is difficult to understand without further information.

To sum up: (1) The WikiHouse projects aims for simplicity in the fabrication and building process, rather than in the design and development process. (2) There is a

38. However, as already mentioned in the section on documentation, a new Design Guide, covering recent developments, has been uploaded in the meantime.

lack of transparency in the development of the WikiHouse building system; neither the motivations behind particular changes to the building system nor the advantages of new developments are documented, neither publicly nor on the mailing list or in the forum.

Maybe this focus on builders reflects the discourse on making designs easier to use for end users or laypeople (cf. Vardouli & Buechley, 2014). To be sure, making things easy to build and assembly instructions are well-suited to achieve this aim, yet in order to make designs easy to read or adapt for other designs, documentation of a different kind would be needed, namely release notes, that is, notes that explain why something was changed in comparison to the previous release and why these changes are improvements, as well as scientific reports. In order to apply the WikiHouse building system in general, designers will need need information about the more general rules of that design as well as some drawings and models. In order to participate in the development of the WikiHouse building system, designers will need information about past developments as well as future goals.

The complexity that appears to be most challenging for the WikiHouse project is its own organizational complexity, which has grown with the progress that they have made and the number of people who have joined. Of course, the fact that the WikiHouse project has many contributors also implies that it should have the necessary resources to properly document their designs as well as the overall development.

That said, complexities of any kind are usually difficult to untangle and simplicity is hard to achieve, and it is the norm rather than the exception that projects get more complex over time. Making things as easy to comprehend as possible is a hard work. Hence, I would like to emphasize that the WikiHouse project has done well in making WikiHouses easy to build by design.

It is crucial for free & open projects that comprehensive documentation is available to potential contributors. What form of documentation is the most suitable depends on the subject matter in question – for architectural design drawings and 3D models would seem to be obvious choices. That said, a lot of information can be conveyed by making design drawings available, but there are types of information that cannot easily be expressed in graphical form; for example, the reasons behind design decisions will require textual comments. Furthermore, not all information that can be expressed in graphical form is best conveyed in that form, for example, videos may be more suitable to show how parts are assembled. The choice of the medium through which information is conveyed should be guided by how the respective information can be expressed in the simplest and most comprehensible way. This might be a sketch, a formula, a video tutorial, or a wiki. However, it is important to keep in mind that searching for information still more often than not means searching for some text. Thus, text-based information is highly important.

3.7 Automation

In this section, I discuss how the WikiHouse project approaches automation. The WikiHouse project itself presents automation as a concept crucial to the project. "WikiHouse is a product of three massive trends: open design, automation and digital manufacturing – catalysts for the third industrial revolution" (WikiHouse, 2015b). Of course, given that I am researching how the application of free & open source methods in architecture may impact design and the design process, the relation between automation and design is of particular interest. In part two I found the following topics to be relevant for automation: (a) *digital automation in hardware*, (b) *digital methods in design and fabrication* and (c) *unrewarding tasks.* However, (a) *digital automation in hardware*, for example, home automation, is seen as an open challenge by the WikiHouse project team, using buzzwords such as "house operating system", "the home as an open platform", "the internet of things", "sensors", "devices" and "automated system" (WikiHouse, 2015h). However developing home automation systems is not a challenge in architectural design but in engineering and information technology. Hence, I will not discuss this topic further, but will focus on the (b) *digital methods in design and fabrication* and (c) *unrewarding tasks*, in each case I will differentiate between (1) *automation in design* and (2) *automation in hardware*. I should note that I will also mention the (d) *integration of laypeople*, since this topic is intertwined with automation.

(b) *Digital methods in design and fabrication* can be found at the point at which digital designs of the WikiHouse building system are to be translated into their physical counterparts. There are two steps that are automated in particular: (1) The extraction of the cutting lines that are needed for producing the physical building parts and the preparation of the cutting files. (2) The process by which single parts are cut out from plywood or other suitable sheet material itself. (1) is an *automation in design*, (2) is an *automation in fabrication*.

When I talk about (b.1) *automation in design*, I am referring only to automation that goes beyond the mere use of design software. This kind of automation in design can, at the moment, be found only in a single design step, which is located at the very end of the design process, when all decisions regarding the form of the future object have already been made and the only thing that is missing is a description of the building parts that can be used for fabrication. At this point, there is a finished 3D model, including all building parts and 3D descriptions of these parts, which now needs to be translated into machine-readable form. Doing so, requires to create 2D cutting files from this 3D model – and this final step is performed by the WikiHouse SketchUp plugin.[39] That is, the plugin first identifies the main face of each single building part and extracts that main face as a closed polylines. Then, it locates internal and external cutting lines, applies the offset that is required as tolerance to allow the plywood parts to slot together, and labels each of the resulting 2D parts. Finally, it arranges these 2D parts on a plywood sheets, so that each of these sheets is used as efficiently as possible, avoiding unnecessary waste (cf. WikiHouse, 2014f, section "Preparing cutting files"). The result of this automated process are executional design drawings in the Drawing Interchange Format (DXF), which can be used for automated fabrication.

(b.2) *Automation in fabrication* is only found at the first step of the fabrication of a WikiHouse, namely, the fabrication of the plywood elements out of which WikiHouses are assembled. These elements are cut out from plywood sheets by a Computerized Numerical Control (CNC) router[40] that reads the aforementioned executional design drawings.[41] Due to their form, these elements would be difficult to cut out manually, but CNC routers but can do so automatically and with high precision, so

39. The plugin includes other functions as well, namely, "Get Models" and "Share Model", which used to facilitate the up- and download of WikiHouse models. Yet, these functions were designed for the old homepage and are, as of yet, defunct. Compared to the "Make This House" function discussed above they are however of minor importance at any rate.

40. Carpo (2011) finds that CNC routers are one of the digitally controlled fabrication techniques that may have disruptive impacts on architecture. CNC routers are also one of the core tools that fab labs provide (Carpo, 2011; The Center for Bits and Atoms, MIT, 2015).

41. CNC routers are part of the digitally controlled fabrication techniques that have disruptive impacts on the production of architecture, Carpo (2011) argues.

that the cut out elements strictly match their digital counterparts. The outcome of this process is a ready-to-assemble building kit.

(c) *Unrewarding tasks* is the next topic to be considered. I will again start with (c.1) *the automation in design*. There are different situations in which the automation of a design process may be useful, tasks that are either unrewarding, repetitive, tedious, or simply could be done more reliably by a machine are always good candidates. That is, as long as the task can be reduced to a purely rule-governed process. Creating 2D executional drawings from a complex 3D model is good example. This task is not only unrewarding, tedious, and error-prone, but also has to be done whenever a 3D model in the WikiHouse building system is altered. What is more, were it to be done manually, this task would be the more time-consuming, tedious and error-prone the more complex the 3D model gets. Fortunately, this task can be reduced to a rule-governed process. Consequently, it can be represented by code and carried out by a script or a piece of software. And this is what the WikiHouse plugin is: a translation of a task into code, so that it can be executed by a machine ever and ever again. It is a piece of software that performs work which is cumbersome for humans.

Regarding (c.2) *automation in fabrication*, I can only repeat that cutting wooden parts by hand is a task that cannot be done by unskilled workers (and even skilled workers will find it hard to be as precise as a CNC router). This touches the topic of (d) *integrating laypeople* that has already been mentioned in the context of digital design tools. With digital fabrication, complex non-standard parts can be produced automatically. This allows laypeople to fabricate complex parts and opens advanced woodworking to people who could not make a chair by hand (and probably would not want to). This is also a declared goal of the WikiHouse project. If you recall, Ratti et al. (2011) find that automation in design may enable us to integrate laypeople in the design process. They mention parametric design in particular. We have already seen that parametric design is an aim of the WikiHouse project (cf. the section on tools above, especially the subsection on the integration of laypeople via tools), though one that has not yet been realized.

How does automation affect design? When we look at the WikiHouse project, in its current state, we see that the decision to use automated digital fabrication constrains the number of permissible design decisions, we would not even exaggerate to say that the overall design of the construction system is defined by this decision. By contrast, the automation of the design process through the use of plugins does not interfere at all with the creative aspects of designing – that is, apart from relieving designers from having to perform a tedious task. However, automation in design can be implemented in different ways, which will affect the design process in different ways. We have, for example, seen that the WikiHouse project aims to provide parametric models, which may constrain the range of permissible design decisions even further – for the benefit of providing models that are easier to adapt. Put another way, the answer to the questions of how and to which extent designs should be automated will depend on whether this automation should benefit the designer or the end user. If this automation should benefit the designer, then the aim will be to "automate away" tedious or repetitive tasks, while at the same time trying to leave as many design decisions as possible in the hands of the designer. If this automation should benefit the end user, then the aim will be to parametrize as many design decisions as possible and may limit the range of permissible choices, especially if it needs to be ensured that the various choices, which the end user can make by virtue of changing the values of these parameters, are compatible with each other. That said, parametrized designs may also empower designers, since they may make some design task easier or faster to perform, so that more time is left to be spent on more complex design tasks. These are questions every project that wants to automatize (parts of) its design or its design process must answer. What is more, in doing so, a project should be aware of how adopting design automation changes design, the design process and the role of the designer. For example, it should be aware of the kinds of skills that designers need (or need to acquire) in order to automate design decisions or where it needs (or would be well advised) to cooperate with programmers.

To conclude, I would like to highlight that the way in which the WikiHouse project employs digital automation allows to integrate laypeople in the building process, because it shifts the need for skilled labor from the physical to the digital realm. The precision work that happens in the physical realm is done by machines. The precision work that is done by humans happens in the digital realm.

Why is this preferable? Why is it preferable to solve a problem in the digital realm? For two reasons: (1) Once a problem is solved in the digital realm, the solution can be shared much more easily. Hence, solving a problem in the digital realm will usually reduce the amount of work we need to perform in the physical realm. (2) Digital solutions tend to be flexible, they can be abstracted, customized and parametrized. If you solve a particular problem in the digital realm, it is usually easy to modify that solution so that it solves a more general problem of which your particular problem is just an instance. To sum up, we can reduce the amount of skilled labor that is required for design or fabrication by moving the shifting the skilled labor from the physical to the digital realm. This is, at least in part, a consequence of the fact that digital solutions can be shared easily. What is more, this sharing of digital solutions is precisely what free & open methods facilitate. This is what makes the WikiHouse project well-suited for the application of such methods.

3.8 Fabrication

This section outlines the approach that the WikiHouse project has chosen for the production of the actual, physical houses. It examines the translation of design information into tangible objects. I have argued, in part two, that the decision on how artifacts shall be manufactured affects the design of those artifacts, particularly if the these artifacts shall be manufactured by the community. In the following, I address the topics found in my review in part two, namely: (a) *interpretation and ambiguity,* (b) *loci of production*, (c) *design for community based production*, (d) *means of production and supply,* (e) *prototyping and replication.*

Doing so, I will limit myself to those aspects of the fabrication of WikiHouses that are the same for all of them; that is, the fabrication of the basic plywood elements and their assembly. Put the other way round, I will ignore post-assembly aspects of the fabrication (or "building") of WikiHouse, for example, insulation, cladding or other interior or exterior finishing work. So, what does the project itself note on production and fabrication?

Is there room for (a) *interpretation and ambiguity* in building WikiHouses? If you recall, one topic that I have found in the second part is that the translation of design information into actual, tangible artifacts used to (and still usually does) require interpretation by a producer, which, of course, is an ambiguous process (cf. Vardouli & Buechley, 2014). By contrast, there appears to be little if any room for interpretation in transforming the design of a basic WikiHouse construction into its physical counterpart. All the parts required are precision manufactured via digital manufacturing tools (WikiHouse, 2015a slide 3) and are put together as if they were a puzzle in 3D. It does not matter who assembles the building kit, the result will always be the same. All design decisions concerning the basic construction are made in the design process. However, this is different for the interior construction completion as well as for the outer skin of the assembled building – these design choices are left to the end user. So even those who prefer not to design themselves can customize their WikiHouse.

Raasch et. al. (2009) identify three different approaches regarding the (b) *loci of production*: (1) production by external manufacturers, (2) production with the community and (3) production by the manufacturing company that coordinates the project. Which of these approaches is chosen also affects which business model are viable for a project (or other parties that wish to offer products or services related to the project). When we look at the WikiHouse project, we will find their approach is a hybrid of (1) and (2). Type (3) does not apply – the group who coordinates the project is not a manufacturing company, but an architectural firm. In fact, WikiHouse is open to the concept of production by external manufacturers. The projects regards it as part of its mission to host an open, fair marketplace for sale of services over and above the commons (WikiHouse, 2014c). WikiHouse plans to have special WikiHouse trademark badges for designers, certifiers, manufacturers and builders who list their services in the WikiHouse catalog (WikiHouse, 2014c). Originally, in version 4 of their constitution, the WikiHouse project intended to limit commercial activities with the aim to prohibit speculation, stating that – "[w]e will not use WikiHouse to mass produce houses whose primary function is as a speculative real-estate asset, rather than as a place to live" (WikiHouse, 2013e). However, in version 6 of their constitution, the wording of the constitution has been adapted to align with free culture licenses[42] and now focuses on establishing fairness and collaboration as cultural values for commercial activities around the WikiHouse project, saying that "we are all free to compete with one another, and we will always seek to avoid monopolies, and seek to make competition open, collaborative and fair" (WikiHouse, 2014c). WikiHouse encourages the use of licenses that that allow commercial use, suggesting the use of the

42. Free culture licenses are licenses that allow the licensed work to be "most readily used, shared, and remixed by others, and go furthest toward creating a commons of freely reusable materials" (Creative Commons, 2015). This does not allow for prohibiting of commercial use.

Creative Commons Attribution-ShareAlike 2.5 Generic license[43] or similar licenses in order to keep works and derived works freely accessible.[44] The goal is to "[s]hare global, print local" (WikiHouse 2015a). The digital designs are shared online, but to produce the actual parts the project recommends to contact a local workshop. "It is easier to ship recipes than cakes and biscuits" (WikiHouse, 2014d) they note, citing Keynes. That being so, the WikiHouse project is fond of saying that the designs that the WikiHouse community produces are "manufactured through the world's biggest factory. A distributed network of local workshops using tools such as 3D printers and CNC machines" (WikiHouse, 2015a, slide 1). The design and manufacturing concept of WikiHouse strongly builds on CNC routing technology. This technology is usually employed by timber processing companies, but is also used in community workshops and is one of the core technologies provided by fab labs. So as fabrication can and *is* done by private individuals, WikiHouses can also be offered by professional manufacturers. Hence, the WikiHouse approach to production includes (1) production by external manufacturers as well as (2) production with the community.

How does the WikiHouse project (c) *design for community based production*? The community, as WikiHouse defines it, "encompasses any individual, team or company who downloads from, uses or contributes to the WikiHouse Commons, or volunteers to support WikiHouse projects" (WikiHouse, 2014c). Community based production or production in the community is based on the accessibility of production facilities and on a low threshold for entry. The WikiHouse Foundation regards it as part of their mission to "engage with governments, non-profits and private companies to promote citizen-led production world wide" (WikiHouse, 2014c). In their paper on open challenges they call for a "[n]ew legal and industrial frameworks to support citizens in accessing land, planning & producing homes & cities" (WikiHouse, 2014k). In this context, they highlight community factories, local digital manufacturing and local materials as examples and mention open manuals, documentation and support for the community factory model (WikiHouse, 2014g, 2014k). In version 2 of their development goals, community factories are introduced as "operating as open making labs,

43. https://creativecommons.org/licenses/by-sa/2.5

44. "The [WikiHouse] Foundation strongly discourages the use of non-commercial licences for any content" (WikiHouse, 2014c). "[WikiHouse] [d]esigns and related information are shared under a Creative Commons Attribution Sharealike 4.0 license. This means you are free to use them for commercial or non-commercial use, however any derivative works built onto that information must be re-shared under the same license" (WikiHouse, 2014c).

village halls and learning institutions. The buildings themselves are copiable local prototypes and the institutions are accelerators for learning, making copying and innovating." (WikiHouse, 2014g) In short, community based manufacturing is a core principle of the WikiHouse project. Remember, the *fabrication and construction process* of a WikiHouse proceeds as follows: 3D models and the cutting files that are derived from them are shared online, free to use. "The aim is to allow anyone to design, download, and 'print' CNC-milled houses and components, which can be assembled with minimal formal skill or training" (WikiHouse, 2013a). Using a CNC router, the parts out of which a WikiHouse is constructed can be cut from a standard sheet material like OSB or plywood. The single parts are numbered to ease identifying them during the construction. On site, the parts are set out and joined to frames. The frames are raised by hand and connected to form a house structure, using pegs and wedges, no bolts are needed. To close the structure the inner and outer panels are screwed in place. I reiterate this here in order to emphasize how easy it is to assemble a WikiHouse. Indeed, the "system is designed to be as easy and safe as possible to assemble. For example, using the structure itself to make a safe working platform. This eliminates the need for a scaffolding framework, so only mobile scaffolding is needed" (WikiHouse, 2015a slide 6). According to WikiHouse, a team of amateurs can assemble a WikiHouse in a couple of days, depending on its size. The only tools needed are mallets and basic DIY tools (e.g., cordless screwdrivers) (WikiHouse, 2013a, 2015a). The WikiHouse project, as has probably become clear at that point, understands the potential changes that this approach may bring about quite well – In the guide for designers, which could be found on the old WikiHouse homepage, they boldly proclaim: "Design is disruptive when it lowers the threshold. Design structures which can be assembled with minimal formal skill or training, and without the use of power tools" (WikiHouse, 2014d).

What does the WikiHouse project say about (d) *means of production and supply*? In physical production, the production facilities and materials as well as their availability are points that need to be considered. In free & open software production, a public good is produced using private means of production, which is possible because contributors usually possess the necessary means of production already – a computer, the required software and an internet connection. In the production of digital design, the situation is similar, save for the fact that the required software is not as widely available, simply because it tends to be proprietary.[45] What is more, when it comes to producing these designs, professional production facilities are not widely available. Alastair Parvin, in his talk at the 2013 TED conference, stated that the facilities, the tools, and the infrastructure that is required to produce tangible goods is, to date, mostly owned by what he calls the "monetary economy". However, he argues that with today's digital fabrication technologies this changes due to the current efforts of the open design and hardware community in developing free & open machines. Parvin adds: "What these technologies [3D printers and CNC machines] are doing is radically lowering the thresholds of time and cost and skill. […] And they're distributing massively really complex manufacturing capabilities" (Parvin, 2013). The developments to which Parvin refers are not only furthered by people who develop free & open hardware, but also by people who host and maintain the necessary machines and make them accessible to a broader public. In their FAQ, the WikiHouse project answers the question of where to find a CNC router to fabricate a WikiHouse kit as follows: "CNC machines are increasingly available and affordable. The most affordable CNC machines include ShopBot, Marchant Dice and Blackfoot. However, many people choose to hire the services of a CNC workshop instead. You might be able to find one near you using FabHub or 100k Garages" (WikiHouse, 2015b).[46] WikiHouse, in other words, hopes to solve the problem that the means of production and supply that are required to manufacture tangible goods tend to be not easily available by using digital fabrication tools that are increasingly available to broader audiences and internationally standardized materials. Parvin even finds that these technologies amount to "an industrial revolution. And when we think that the major ideological conflicts that we inherited were all based around this question of who should control the means of production, and these technologies are coming back with a solution: actually, maybe no one. All of us" (Parvin, 2013).

The WikiHouse project develops a new construction system; needless to say (e) *prototyping and replication* are important to them. In fact, the hardware development in the WikiHouse project is driven by learning from prototypes. During the early period

45. I have already pointed out (in the section on tools) that we, as professional architects, must not forget that many of the professional software products that we use are quite expensive and, therefore, not easily available to amateurs, hobbyists or even professionals from other fields.

46. Troxler (2010) also names tech shops, hacker spaces and fab labs.

of the project "hardware development was driven by a sequence of small exhibition prototypes and events. Each of these prototypes allowed for continual iterative development of the structural system […]" (WikiHouse, 2014h). And "there continued to be a continually growing number of prototypes by 00 [the architectural firm mentioned earlier] and others around the world […]" (WikiHouse, 2014h). Furthermore, the WikiHouse project states in version 6 of its constitution that a new WikiHouse prototype should be presented annually, at an event with debates, talks and conversations that should be documented in an annual report (WikiHouse, 2014c). Of course, prototyping is costly. But while the WikiHouse project mentions development costs, they do not do so in detail; for example, in their development goals, listing milestones and the costs for reaching them, they estimate: "Module 1. Tiny House A complete test module tiny-house prototype, with footings, skin, insulation, windows. Sensors to monitor performance. Fully documented and shared. Team00 [again, the aforementioned architectural firm] Build £15k R&D £5k Total £20k" (WikiHouse, 2013d). They provide more detailed information on replication costs on their website, in the respective bills of materials (BOM; cf. above); for example, according to the BOM of the Studio (version 4.1), a tiny house can be build for about £14500 (WikiHouse, 2015d). (I will compare these costs to that of a comparable building later in this section.)

This is what I found on how the WikiHouse project approaches the challenges of fabrication that I have found in part two of this thesis. In part two I noted that when we want to discuss how particular ways of production affect the way in which design is conceptualized, we need to take into account who translates the design information into a tangible artifact, which skills and which infrastructure are required, and what costs will be incurred in the process. The WikiHouse project considers professionals as well as do-it-yourself (DIY) builders their target group. The WikiHouse Foundation explicitly encourages companies to offer paid services around the WikiHouse Commons, even though the whole production concept of WikiHouse is geared towards DIY manufacturing and construction. The concept that the WikiHouse foundation promotes thereby reminds of business models known from free & open source software. What makes the WikiHouse concept so apt for self-fabricating and self-building[47] are its fabrication and building techniques. That is, that its basic ele-

47. I use the term self-building here in the sense of "doing the work yourself". In the UK the term self-build home is in general used for individually planned and build homes in contrast to generic

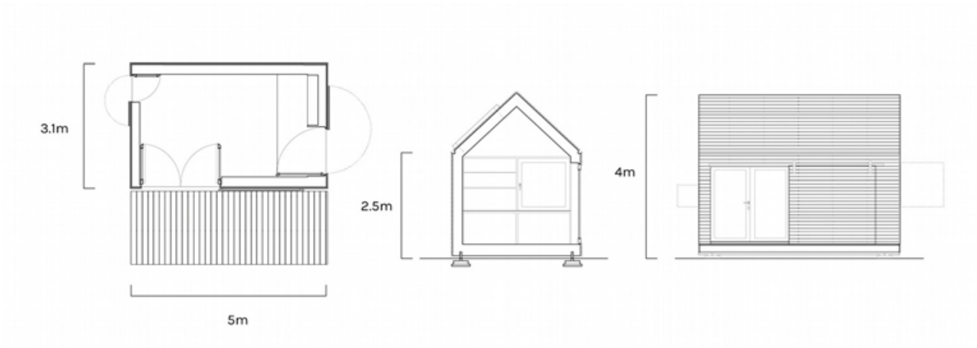

Figure 16: Design Drawings of the WikiHouse Studio, Version 4.1 BETA release, effective area 11.6 m², external dimensions w 3.1 m × 5 m × h 4 m, approximate U-value of 0.15 W/(m²K) for roof, floor and walls (Design team 00 & WikiHouse, 2015; WikiHouse, 2015d).

ments can be fabricated from a single base material, which can be chosen from a rage of standardized and widely available sheet materials; that these elements can be fabricated using CNC technology, which is also increasingly available to broad audiences; and that a small team of amateurs can assemble these elements into a full-blown WikiHouse. This is how WikiHouse pursues its aim to make building a WikiHouse as broadly accessible as possible.

So far the theory, but is the high-tech infrastructure, which is needed to fabricate the parts of a WikiHouse, really that accessible? That is, are there really that many places that allow access to infrastructure for DIY fabrication, for example, fab labs? For example, in Vienna? At the moment, there are two fab labs in Vienna: the Happy Lab and the Maker's Space Vienna. The Happy Lab already has a CNC router, the Maker's Space Vienna is building one at the time of writing. However, both CNC routers would be too small for cutting the standard sheet seize WikiHouse uses. This is also true for the fab labs in Bratislava, Graz and Salzburg. That said, the situation may change in the future. For the time being, DIY-builders need to use the services of commercial timber processing companies to fabricate WikiHouse building parts. However, access to production is not only a matter of infrastructure but also a matter of costs. For the WikiHouse Studio (see figure 16 for design drawings) material costs are estimated to be about £14500, which are about €17980, costs for on site assembly

homes built by real-estate developers.

are estimated to be about £2000, which are about €2480 (average exchange rates for 2014 according to the OANDA Corporation, 2015).

To put this into perspective, I will compare the costs of the WikiHouse Studio with the average cost for one-dwelling residential buildings. A comparison to residential buildings is appropriate, in my opinion, because the Studios construction, in regard to its U-value, meets passive house standards and also because the Studio is a prototype for a system that is suitable for residential buildings. To be sure, the WikiHouse Studio is much smaller, but to the best of my knowledge, there are no statistical data about micro houses. Table 3 to table 6 show the, admittedly rough, calculations and data on which I base my comparison.

Table 3: Conversion factors for the comparing building costs in selected countries

Regional factors for building costs in Europe

Germany (GER) to Austria (AUT)*	Germany to United Kingdom (UK)*	United Kingdom to Austria
1.001	0.866	1.156

* Baukosteninformationszentrum Deutscher Architektenkammern (2013)

Table 4: Dimensions of the WikiHouse Studio

Dimensions of the WikiHouse Studio Version 4.1 BETA release

Width* w [m]	Length* l [m]	Wall height* h1 [m]	Roof height* h2 [m]	Roof angle* α [°]	Enclosed volume w·l·(h1+h2/2) [m³]
3.1	5	2.5	1.5	45	50375

* Design team 00 and WikiHouse (2015)

Table 5: Cost for the WikiHouse Studio in the United Kingdom and Austria

Costs with on site assembly for the WikiHouse Studio Version 4.1 BETA release

Costs in the UK* [£]	Average exchange rate for Jan.-Dec. 2014**, £ to € [-]	Costs in the UK [€]	Costs in the UK [€/m³]	Conversion factor for UK to AUT [-]	Costs in AUT [€/m³]
16500	1.24	20460	406.15	1.156	469.47

* WikiHouse (2015d)
** average exchange rates for 2014 according to the OANDA Corporation (2015)

Table 6: Costs for the WikiHouse Studio in comparison to the average costs for one-dwelling residential buildings in Austria

Costs for on site assembly of the WikiHouse Studio Version 4.1 BETA release in comparison with average building costs for one-dwelling residential buildings

Average costs in GER for one-dwelling residential buildings* [€/m³]	Conversion factor for GER to AUT** [-]	Average costs in AUT for one-dwelling residential buildings [€/m³]	WikiHouse Studio costs in AUT [€/m³]	Costs for the WikiHouse Studio expressed in one-dwelling residential buildings equivalents
281	1.001	281.28	469.47	1.67

* Statistisches Bundesamt (2015a)
** Baukosteninformationszentrum Deutscher Architektenkammern (2013)

In table 6 we can see that the cost per cubic meter for the WikiHouse Studio Version 4.1 BETA release are 1.67 times as high as the average costs per cubic meter for a one-dwelling residential building. Buildings meeting the passive house standard are 1.02 to 1.05 times as expensive as a standard building (Diem & IG Passivhaus Vorarlberg, 2011, pp. 22–23). The average costs for one-dwelling residential buildings that are mainly built of wood are about 1.04 times as high as the overall average costs of one-dwelling residential buildings (Statistisches Bundesamt, 2015b, factor calculated based on the data of 2014 for Germany). So, the WikiHouse Studio Version 4.1 BETA

Figure 17: Community map of WikiHouse,
indicating the current WikiHouse chapters (WikiHouse, 2015i)

is rather expensive in comparison. However, the WikiHouse Studio has a disadvantageous wall/roof-volume to total-enclosed-volume ratio due to its very small size. This would need to be considered for making a more solid assessment. Moreover, the WikiHouse Studio Version 4.1 BETA is a prototype that is still under high development and, therefore, not fully optimized yet. That said, my goal here was only to put the costs of the WikiHouse into perspective, for which the rough calculations above are more than sufficient. We have learned that, while the WikiHouse construction system is easy to build and, as a system, relatively accessible, the project still has some way to go before reaching its goals in this respect.

Yet, despite the relatively high costs of building a WikiHouse prototype, there are WikiHouse groups all over the world (see figure 17 for an overview and figure 18 for some of the WikiHouses that have been built). Most of them emerged in recent years and have built different prototypes. Considering the cost, funding probably has been a challenge (or still is) for most of these groups. Interestingly, neither the costs nor the required high-tech infrastructure appears to put people off. The WikiHouse community is growing. I would conjecture that this is connected to the fact that the WikiHouse project has chosen CNC technology for fabrication. However, my point here is not that there are so many WikiHouse chapters because CNC technology is so

Figure 18, top: FOUNDhouse in Bluff, Utah by Patrick Beseda and Lacy Williams (Beseda & Williams, 2013a, 2013b); middle: A-Barn in Scotland by Team 00 (Carse, 2014; Parvin, 2014; Pepper, 2014); bottom left: Farmhouse in England by Team 00 (de Rivera Costaguti, 2015); bottom right: tower shaped demo pavilion in Rotterdam by WikiHouseNL and the Innovation Centre for Sustainable Building (WikiHouseNL, 2015) cost-effective and widely available (to date, it sure enough is neither), but that there are so many WikiHouse chapters because CNC technology is one of the technologies that fab labs provide and there is a growing scene around these fab labs – the fabbing community. That is, there happens to be a community of people who are interested in digital fabrication techniques, and *for members of that community* building a WikiHouse is not only an interesting project, the fabrication technique used by the WikiHouse project is also quite accessible. Put simply, the WikiHouse project targets a larger, already existing community. Building a community, needless to say, is crucial to get a community-oriented project off the ground and targeting an already existing community that is enthusiastic about the technologies employed makes that a lot easier. The use of digital fabrication techniques attracts enthusiastic tinkerers and potential innovators, who are gathered in the fabbing scene, as well as people who are knowledgeable about and interested in free & open source methods. Hence, locating the project in the fabbing scene, rather in than only in the field of architecture, was probably a good strategic decision. What is more, we also see that choices in fabrication techniques can and do affect community building. Hence, community building should be taken into account when choosing a fabrication technique.

That said, my focus is on how free & open methods affect design. If we start out from a design idea, one usual question to arise is: "How can I (physically) produce what I have in mind?" Yet, in our case another approach might be more fruitful. We start out from asking ourselves: "What are the resources are available? What is the desired mode of production? And who are the relevant actors?" That is, we can understand the fabrication capabilities and the way in which we want an artifact to be produced as reasons for our design decisions.

3.9 The Artifact

In the following, I present my findings from confronting the WikiHouse project with the topics and challenges regarding the qualities of tangible artifact. I will investigate the following types of aspects and challenges, all of which have been found in part two: the (a) *disassembly,* (b) *lifetime* of the actual, tangible artifact, (c) *material aspects* and

(d) *performance aspects*. Unsurprisingly, the artifacts that I will study are WikiHouses. However, I will not study particular buildings, but only WikiHouse designs and specifications. This is because actual, tangible WikiHouses will share few properties as *WikiHouses* beyond those that are mandated by these designs and specifications (if you recall, the finishing of a WikiHouse is left to the end user; cf. above). And these designs and specification are detailed enough for such an analysis to be meaningful. Having made that clear, I will now present which goals the WikiHouses project formulates and which challenges it discusses in relation to the physical features of actual, tangible WikiHouses.

The WikiHouse project, perhaps unsurprisingly, given that their construction system is highly modular, does discuss (a) *disassembly* at many places. If you recall, the OSE project assigns high importance to designing products so that they are easy to disassemble (and reassemble). The WikiHouse project also did so, regarding "design for disassembly" or "design to dismantle" as one of its design principles: "The easier to dismantle structures or replace individual parts, the better" (WikiHouse, 2013c, 2014d). However, that passage has since been removed from the design principles. Yet, even though the WikiHouse construction system is intended to be easy to disassemble (and reassemble), this has turned out to be somewhat more difficult in practice. WikiHouse, in the FAQs on their new website, answers the question of whether a WikiHouse can be disassembled as follows: "Yes, but not forever. With moisture in the air, the panels tend to expand slightly over time. This is good news for the strength of the structure, but means that over time the chassis will be harder to dismantle without breaking some parts" (WikiHouse, 2015b). That said, that WikiHouse cannot only be disassembled in theory, but also in practice. It has been shown at the London Design Festival in the summer of 2014. "At the end of the festival [the WikiHouse] 4.0 was carefully disassembled and sent to its new home in Liverpool […]" (WikiHouse, 2015e).

(b) The *lifetime* of WikiHouses is only mentioned by the WikiHouse project in their FAQs. There they assume that the lifetime of a WikiHouse is "[i]ndefinite, providing that it is properly built, protected and maintained. WikiHouse is a timber frame system; a construction technique established over centuries. Further still, because WikiHouses are modular, they are much easier to mend & maintain than most houses" (WikiHouse, 2015b).

(c) *Material aspects* are also a topic that is discussed by the WikiHouse project. The way in which WikiHouses are designed and fabricated places certain constraints on which materials can be used for construction. The basic construction of a WikiHouse is all made of one material. This material has to be a sheet material that can be cut using CNC mills and that provides sufficient structural strength (WikiHouse, 2013a, 2014d). The WikiHouse projects names birch plywood as one standard material; newer designs also use OSB. Moreover, they recommended to use international standard seizes (18 mm × 2400 mm × 1200 mm) (WikiHouse, 2013a, 2014d, 2015d). Decisions on the insulation, outer building skin and interior construction completion are left to the user. However, WikiHouse seems to consider to add other, additional systems, for they list the development of "[a] growing range of structure & building envelope systems which can be customised, digitally manufactured and self-assembled with little time, money, skill and using widely available materials" (WikiHouse 2015e) under their open challenges. Moreover, the WikiHouse project formulates more general demands concerning the choice of material, namely, that materials should be: reasonably cheap, standardized, abundant, local, sustainable, low-carbon, recyclable or biodegradable, 'circular', if possible (WikiHouse, 2014c, 2014d, 2014k). What is more, the WikiHouse considers it a challenge to "[e]xplore more open materials, developing a library of materials and construction techniques" (WikiHouse, 2013c). Material efficiency by design is also touched upon as a topic in the design guide (WikiHouse, 2013c). Local statutory requirements regarding materials are seen as a responsibility of the end user (WikiHouse, 2014c). The demands that have been discussed in this paragraph are of a more general nature and, with the exception of wide availability, are a consequence of the chosen fabrication technique, overall construction, skill level, and cost structure.

The WikiHouse project does discuss (d) *performance aspects*, especially ecological ones. The WikiHouse project, for examples, aims to reach "well being performance benchmarks (eg Passivhaus)" (WikiHouse, 2013d). The WikiHouse project tries to bring high performance, efficiency and low energy consumption together with being low-cost, cheap to replicate and being accessible to everyone (WikiHouse, 2013c, 2013d, 2014b, 2015a, 2015h). We have seen that OSE finds that DIY approaches are compatible with high performance standards. The WikiHouse projects makes quite similar statements: "Higher performance, lower thresholds[:] Design to lower thresholds of time, cost, skill, energy & resources in manufacture, assembly and use" (Wiki-

House, 2014c, 2015a). That is, high performance and a low threshold to entry are both seen as goals. Moreover, they name the "develop[ing] a global catalogue of high performance, low-cost, low-energy solutions for sustainable homes & neighbourhoods; accessible to everyone" (WikiHouse, 2015h) as open challenges. They reiterate this point, that high performance and a low threshold should go hand in hand: "Design for the next Normal[:] Design beautiful, high-quality products that lower cultural barriers and make radically sustainable, sociable design 'normal', rather than 'alternative' or 'fashionable'" (WikiHouse, 2014c, 2015b).

Before continuing to my interpretation of these findings, I would like to add another aspect that I have found in my analysis of the WikiHouse documents, which was not mentioned by the sources that I have analyzed in part two and which affects architectural design in particular – the decision between location specific or location neutral architecture. Which of these two options is chosen affects how reusable a design is. An object that is tailored to a particular location is probably difficult to adapt or reuse and at the very least an unattractive candidate for doing so. WikiHouse is not a specific design but a construction system. This allows the WikiHouse project to at least aim for a balanced approach, striving for variability rather than one-size-fits-all designs. For example, the WikiHouse projects wants to allow for designs to be adapted to different climates, economies and delivery models (WikiHouse, 2014g). Moreover, WikiHouses are designed to be ground agnostic. "WikiHouses rest onto timber 'rail' joists. Any kind of foundation can be used to support these […]. This allows the system to deal with a variety of ground types, including slopes" (WikiHouse, 2015b). Having said that, one design principle of the WikiHouse project is to "[s]tart somewhere[:] You can't solve everyone's problems in one go. Design something useful for a context you know and understand, then share so others can adapt to their […] context. Release small, iterate and 'fork'" (2014c, p. 5, 2015a). Moreover, you, as a designer, are advised to "design for the climate, culture, economy and legal/planning framework in which you live, and you know best. Others will then be able to adapt the design to suit their environment" (WikiHouse, 2014d). Put another way, adapting the WikiHouse construction system, or particular WikiHouse designs to different climates, cultures or economies is a long term goal and has not been realized, but prototypes have been built in, for example, the UK, New Zealand and Haiti.

We have seen that all aspects that I have found in my theoretical analysis are topics and challenges that the WikiHouse project either discusses or deals with. In my theoretical considerations, I have argued that applying free & open methods to architecture has consequences on the conception of the tangible artifact. We have found this to be the case for the WikiHouse project, the open source idea behind that project motivates particular design decisions. What is more, we have also seen that the WikiHouse project is very concerned about meeting ecological standards. At the first sight this appears to be unrelated to the application of free & open methods, since sustainability, energy performance etc. are challenges for architecture in general. However, it is interesting that self-building and high performance are explicitly and repeatedly mentioned together. It seems to be a matter of particular concern to the WikiHouse project to emphasize the importance of this combination. I would conjecture that this may be the case because self-building is sometimes associated with improvisation and poor quality. I would, furthermore, conjecture that it is particularly important for free & open architecture projects to be recognized by architects in order to be able to exist as architectural projects (by comparison, free and open source software projects are typically carried out by experts at any rate). Put simply, it might not be very appealing for professional architects to participate in a community that is dominated by self-builders, that is, by amateurs. Put the other way round, getting professional architects to participate in a free & open architecture project will be easier if they can gain reputation by participating. And they can gain reputation if their design meets high performance standards, and probably even more so if it does so while satisfying the constraints of free & open fabrication and construction methods at the same time.

Furthermore, prototypes in architecture can happen to be be quite big. Hence, it is an advantage if they can be assembled, disassembled and reassembled easily to, for example, present them at design festivals, maker spaces or simply in a public space. Moreover, it is much easier to find a public space if your prototype can act as a *temporary* construction, which can be disassembled and moved easily. Furthermore, easy disassembly (together with lightweight construction and the dry constructions system) contributes to facilitate prototyping in private workshops. This, in turn, enhances the chances that people participate in the project.

To sum up, free & open methods in architecture place certain constraints on material and, therefore, design choices. These constraints also imply that free & open methods will not suit every project. Of course, nobody said otherwise, but having identified some particular constraints will be helpful for projects to make a more well-informed choice on whether free & open methods suit their goals.

3.10 Conclusion

The WikiHouse project is, in my opinion, the most successful open architecture project to date. So, when we ask how the adoption of free & open source methods may impact design and the design process, we are well advised to take a close look at this project. I have done so in this part of my thesis, analyzing the efforts as well as the written materials of the WikiHouse project in order to compare them to the topics and challenges with which projects that apply free & open source methods to architecture most likely have to deal (as I have found in part two of this thesis). This comparison was guided by the following questions: (1) Is a topic or challenge actually a concern in practice? (2) If so, how this topic or challenge dealt with? (3) Does that topic or challenge have an impact on design or the design process? (4) If so, is this related to the (attempt to) apply free & open source methods and how? Since I can only answer these questions on the base of a single case study, my results are not representative in a statistical sense, yet they may nonetheless be applicable to other projects as well. I will first summarize my findings for all categories that I have found in part two, addressing the research questions (2), (3), and (4), as applicable (see table 7 for (1) and an overview). As a reminder, these categories are: (a) *Tools and Technical Infrastructure*, (b) *Information Transfer and Legibility*, (c) *The Process*, (d) *Modularity*, (e) *Complexity and Simplicity*, (f) *Automation*, (g) *Fabrication*, and (h) *The Artifact*. After this summary, I will form hypotheses on what we may learn for the adoption of free & open source methods in architecture and on how the adoption of these methods may impact design and the design process. These hypotheses will, likely, apply to other projects, however testing whether (or to what extent) they do so requires another methodology than that of an explorative study and is left to future research.

Table 7: Summary of findings

Category	(1) Actual concern?	(3) Impacts design/ the design process?	(4) Impact related to free & open methods?
Tools and Technical Infrastructure	Yes	Yes, but as impediment	Yes
Information Transfer and Legibility	Yes	Yes, but as impediment	No
The Process	Yes, but marginal	Yes, but limited	*Not enough data*
Complexity	Yes, but marginal	Yes	No
Modularity	Yes	Yes	Yes, though not necessarily
Automation	Yes	Yes	No
Fabrication	Yes	Yes	Depending on *locus* of production
The Artifact	Yes	Design only	Maybe, but indirectly

(a) *Tools and Technical Infrastructure*: (2) The WikiHouse project has opted for SketchUp as its editing tool of choice, because it is freely available and easy to use. Unfortunately, this tool is not free & open source, uses a proprietary file format that may impede cooperation (and has, in fact, done so) and lacks native support for parametric design and BIM. Furthermore, the WikiHouse project is looking for collaboration and version control tools. Unfortunately, adequate collaboration and version control solutions for SketchUp are currently unavailable. (3) All of this impacts design and the design process, but in a rather technical fashion, namely either, in the case of proprietary file formats and lacking features, by simply impeding the design process or, in the case of lacking collaboration and version control tools, by simply not facilitating it in the way that is known from other disciplines. (4) This is related to the adoption of free & open source methods insofar as lowering the threshold to using designs as well as making those designs legible requires the use of well-documented, non-proprietary file formats as well as of freely available and easy to use software products; since proprietary software products tend to use proprietary file formats, fully reaching these goals may, de facto, even require the use free & open source software tools.

(b) *Information Transfer and Legibility*: (2) While the WikiHouse project appears to strive to offer documentation, this documentation, at this point, focuses on builders (the group of people I termed "users" above), all but ignoring its own project developers. That is, while building a WikiHouse is documented well enough (even though there is room for improvement here as well), there is little documentation on how to design something based on the WikiHouse building system or on how to contribute to the development of the project, though the documentation for designers (but not for developers) is growing. The situation for standards is similar. The WikiHouse project emphasizes the importance of standards and asks designers to design in a way that meets established ones. Yet, it establishes few standards of its own. And the standards that it does establish do not pertain to design (but to file formats and naming conventions). Of course, the project is quite young, so it may simply need more time to experiment in order to transform its experiences and practices into standards. (3) This impacts the design process, but again only by impeding it. (4) And while this, too, is not caused by the adoption of free & open source methods, it may also impede future growth, since good documentation is pivotal for attracting new contributors and maintaining complex projects.

(c) *The Process*: (2) The WikiHouse understands itself as a project that is evolving over time and is committed to the idea of an evolutionary design and development process. However, regular or formal feedback processes are as of yet uncommon in the WikiHouse community and have only been marginally mentioned in the development goals. Apparently, formal feedback, review and bug tracking processes are deemed to be only of minor importance. (3) That being so, and despite the fact that the development of the WikiHouse project can be described as evolutionary in a way, the notion of the development process as evolutionary has little impact on the actual design process. (4) Usually, we would expect that the adoption of free & open source methods suggests an evolutionary approach to design and development, because evolutionary design and development is based on continuous improvement processes and such improvement processes are fostered by free & open source methods. Publishing early and often allows more people to inspect and test a design, to the effect that potential problems and improvements are discovered sooner and in greater number. However, given that the WikiHouse project all but mentions evolutionary design and development as goals, there is not enough evidence to say anything of substance on the matter.

(d) *Complexity*: (2) The WikiHouse project says little about complexity. However, when we take the overall approach of the WikiHouse project into account, then it is clear that the project has deliberately chosen to "start small" and to evolve step-by-step. The same goes for simplicity, which they discuss mainly in relation to the designed artifacts and the assembly process, which should be intuitive and difficult to get wrong. (3) Of course, this does affect design and, particularly if achieving simplicity is difficult, the design process as well.

(e) *Modularity*: (2) The WikiHouse project strives for its design as well as its houses to be modular. However, which modules make up a WikiHouse or a WikiHouse design and which modules are envisioned by the project are too complex to be easily summarized (see the section on modularity above for details). I will focus here on the distinction between digital and tangible modules. I have argued (in that subsection) that designs and tangible artifacts can (and often should) be made up of different modules; for example, the joints of different (tangible) modules of a WikiHouse are (and should be) distinct modules in design, but cannot be distinct modules of tangible artifacts (because they consist of intersections of different tangible modules). (3) This has implications for designs, which need to be modular and often should be so in a way that goes beyond merely reflecting the tangible modules, as well as for the design process, which is facilitated by modular digital design. Above all, modular design facilitates cooperation between teams, because these teams may then work in parallel with little coordination. (4) Modularity may be important for projects that apply free & open methods, because it enables people to contribute to a project without having to coordinate their efforts with that project. This lowers the threshold for contributing to the project, which is a key aspect of free & open methods.

(f) *Automation*: (2) Automation is a particularly important topic to the WikiHouse project. The WikiHouse project tries to automate some stops of the design process, by providing a plugin for SketchUp, their editing tool of choice, that takes care of generating cut files from designs (a repetitive and error-prone task); automation is also related to the parametrization of designs, which was already mentioned above (when discussing tools). What is more, the WikiHouse project builds on digital automation in manufacturing, which will however be discussed extensively in the next paragraph. Hence, I will limit myself here to the automation of the design process. (3) Needless to say, automating (parts of) the design process will make many design tasks easier and faster. More importantly, this may, in the future, allow laypeople to customize designs themselves.

(g) *Fabrication*: (2) The WikiHouse project is, so to speak, built upon digital fabrication. I have argued above that digital fabrication enables the WikiHouse project to shift the need for skilled labor from the physical realm into the digital realm. This shift is significant because digital design can be easily inspected, shared and modified. That is, the skilled labor, ideally, needs to be done only once. These digital designs can then be downloaded and, provided the necessary infrastructure (above all, CNC routers) is available, be "printed" and assembled by laypeople. Of course, this also means that, at least for the basic structure (the finishing of WikiHouses is left to the end user at any rate), there is no room for interpretation or creativity of the builder. (3) This has a huge impact on the design as well as the design process. (i) The design must consist of modules that can be cut out with a CNC router and then assembled by amateurs. (ii) This restricts the choice of materials, even more so, if these materials shall be economical, easily available and environmentally friendly (criteria that the WikiHouse project aims to meet). This, in turn, restricts the available choices in design. I have argued above that these restrictions suggest a particular perspective on design: Put simply, rather than designing first and then looking for the right materials, designers should, in this situation, first acquaint themselves with the resources that are available as well as the restriction with which they need to deal – and start from there. Having said that, these restrictions owe themselves to the fact that the WikiHouse has chosen "production by the community" as one *locos* of production. And Raasch et al. (2009) point out that there are other possibilities. (4) This digital fabrication is pivotal for the way in which the WikiHouse project applies free & open source methods to architecture, namely, by providing designs that are then manufactured by the community. If a free & open project chooses this *locus* of production, then it must make sure that its development can in fact be produced by a community, which requires that materials and equipment must be widely available and reasonably cheap.

(h) *The Artifact*: (2) The WikiHouse projects wants WikiHouses to be easy to disassemble, which should also make them easy to service and repair, contributing to a long product lifetime. What is more interesting, however, are the material and the performance aspects of the WikiHouse. Having said that, the relevant material aspects of the WikiHouse are all connected to its fabrication and have therefore been already discussed just above. This leaves us with the performance aspects. We have seen that the WikiHouse project aims to satisfy high performance standards, above all, regarding sustainability and being environmentally friendly. (3) These high performance standards clearly impact the design. (4) I have reasoned above that, while free & open methods do not require to meet these standards, attracting professional architects to a project will be easier if they can acquire a reputation by participating. They can do so if their design meets high performance standards, and probably even more so if it does so while satisfying the constraints of free & open fabrication at the same time.

We have seen that all categories that have been identified in part two are important to the WikiHouse project, albeit to varying degrees. This corroborates my findings of part two. We have also seen that dealing with many of the topics and challenges within that categories affects the design process in many different ways. More importantly, insofar as these changes to design and the design process are connected to (a) employing easily accessible software tools and non-proprietary file formats, (b) modularity in design, (c) digital fabrication and (d) performance standards, they have been found to be connected to the adoption of free & open methods by the WikiHouse project, again, to varying extents. I have given the following explanations for these connections:

(a) If the designs of a project shall be easy to use and built upon, then they must be available in a file format that is well-documented and can be read by software that is easy to use and available free of charge.

(b) Although free & open projects do not have to make their designs modular, this may help them to lower the threshold for people to participate in their development. Needless to say, making a design modular impacts design and the design process.

(c) All free & open projects that aim to produce a tangible good must address the question of how that good shall be produced. If a project chooses its community as (one) *locus* of production (possibly among others) and wants to enable its community to produce its artifact by developing for digital fabrication, then it must design in a way that allows its design to be manufactured using widely available and reasonably cheap infrastructure and materials. Needless to say, this impacts the design. However, I have furthermore argued that it also impacts the design process, since the constraints are so numerous that the design process should start with these constraints, rather than from a particular design idea.

(d) I have conjectured that free & open architecture projects may have difficulties to attract professional architects, since they may not want to participate in projects that could be associated with amateurism. Hence, it may be particularly important for such projects to adhere to high performance standards.

These explanations are hypotheses that I put forward, and the core findings of this part. Testing them is a subject for future research, since doing would require a different methodology.

4 Conclusion

In 2006, the term "open source software" was added to the German dictionary *Duden*. This addition, however, not only reflects the growing importance of free & open source software, but also the fact that more and more people are acquainted with the ideas and concepts of the free & open source software movement. Of course, the growing importance of free & open source software goes hand in hand with the growing importance of digital tools in our everyday lives as well as our professional endeavors. Shirky (2005) points out that free & open methods can take hold in every field if the subject matter of that field has been digitized. So what about architecture? We have seen (in part two) that there have been different attempts to apply these methods in architecture, of which the WikiHouse project is the only truly successful one, which has developed an active and lively community. Why is that? What are the difficulties in applying free & open methods to architecture? I have put forward the theses that: (1) The subject matters of architecture that needs to be adapted in order to enable architects to apply free & open methods are design and the design process. (2) In order to be able to adopt free & open methods we may need to adapt the way in which we, as architects, understand and approach our role as authors. Hence, the objective of this thesis has been to shed some light on the question of how projects that apply free & open methods in architecture could be facilitated, by investigating which challenges projects that try to apply free & open methods in architecture are likely to face as well as how adopting these methods may change our architectural practice, that is, above all, the design, the design process and our conception of authorship. More precisely, the objective of this thesis has been to identify how we need to change our design concepts, our design process and the way in which we understand

and approach our role as authors in order to be able to apply these methods. What is more, identifying the challenges that projects that try to adopt these methods are likely to face should also make it easier for future projects to adopt these methods, if they choose to do so. Thereby, I want to contribute to the larger debate on how these methods may be adopted in architecture. To address these questions, I will now summarize the findings of each part, before moving on to drawing a final conclusion.

The first part of this thesis has been concerned with the predominant conception of authorship in architecture, how digitization may change this conception, how the free & open source software movement emerged and the central ideas of that movement. If you recall, the first part was guided by the following questions:

1. What is the dominant notion of authorship in architecture?
2. How may digitization change that notion of authorship?
3. What are the core ideas of the free & open source movement?

I have argued in part one that (1) the currently dominant notion of architectural authorship is still based on the Albertian conception of design, even though the way in which we design, the tools that we use and architectural practice in general have changed a good deal with digitization. Alberti assumes that architecture is an allographic art, that is, an art in which the artist expresses their ideas through some form of notation; in our case the artist is the architect and the notation in question are design drawings. He also assumes that architects are authors in a humanist sense, that is, that they are particularly creative and skillful people who express themselves through their works. These works, being the result of particularly creative and skilled labor as well as the expression of their authors' personality, are taken to be stable once created, rather than being subjected to further change. (2) However, digitization enables us to easily share and alter designs and, thereby, fosters open-ended editing. Moreover, it facilitates rule-based design techniques, which make it easy to adapt designs or create different variations. Therefore, digitization also allows for a new kind of authorship, one which allows for a multitude of interwoven authorial traces, one where many people make small contributions, rather than a few big ones. This new kind of authorship is an authorship by the many, works can be copied, inspected, modified and extended. (3) And it is this kind of authorship that the free & open source movement explicitly embraces. What is more, we have also seen that the ideas of the free & open source software movement are spreading to other fields, and even though it is a bit late to the party, architecture is one of them.

The focus of the second part of this thesis has been to investigate aspects and challenges that we need to take into account if we want to successfully introduce free & open concepts to architecture – from a theoretical point of view. This has been done by an analysis of textual sources of three different kinds: (a) articles on free & open architecture, (b) articles on open design and (c) documentation from open design practice. This part has been guided by the following questions:

1. What challenges are projects that want to apply free & open methods to architecture likely to face?
    1. What challenges or kinds of challenges are already described in the literature on open design?
    2. What challenges or kinds of challenges are already described in the literature on open architecture?
    3. What challenges that have not yet been described in the literature can be found by analyzing project documentation?
2. How can the challenges that are found in question 1 be categorized (taking the relevant literature into account)?

Table 8 presents the topics and challenges and where they have been found. These topics and challenges have been iteratively clustered in order to develop a list of meaningful categories that may provide some guidance on which kinds of challenges we need to consider when we want to apply free & open methods in architecture. Table 9 shows these categories, distinguished by whether challenges that fall under that category are related to design or the design process. Moreover, those of the challenges that are related to design or the design process have been used as criteria with which to address the WikiHouse project. Table 10 shows these challenges, arranged along their respective categories.

Table 8: Challenges and topics sorted by type of text source and categories

Articles on Free & Open Architecture	**Articles on Open Design**	**Project Documentation Open Design**
Economical aspects • Funding Social aspects • Community building Cultural aspects • Awareness – collaborative practices • Awareness – differences between copying and sharing Aspects of definition • Meaning of "open source architecture" • Meaning of open source terms in architecture Tools and Technological Infrastructure • Tools for collaboration • Expectations regarding the impact – mass-customization and integration of laypeople Information transfer and legibility • Standards The process • Design as an evolutionary process • Feedback Automation • Digital automation methods in design & fabrication • Smart environments, building automation Production/fabrication • Interpretation & ambiguity • Rapid prototyping	Economical aspects • Economics of motivation • Aspects related to the physical object Concepts of making money • Field of activity and project feasibility Legal aspects • Patent law and copyright law • Warranty issues Social aspects • Participation • Governance and social agreements • Distribution of tasks & responsibilities • Motivational function of initial design Cultural aspects • Common culture & ethics Tools and technological Infrastructure • Editing tools • Repository solutions & version control • Communication Complexity • Effects of complexity Modularity • Differences in software and hardware Automation • Automation in hardware Production/fabrication • Means of production & supply • Prototyping, production & costs • Loci of production	Information transfer and legibility • Documentation The process • Continued maintenance and development Complexity • Simplicity Modularity • Properties of modular design Automation • Unrewarding tasks Production/fabrication • Design for fabrication The artifact • Design for disassembly • Lifetime design • Material aspects • Performance

Table 9: Categories of challenges free & open architecture projects are likely to face

<table>
<tr><th>Design-related</th><th>Not design-related</th></tr>
<tr><td rowspan="3">1. Tools and Technological Infrastructure
2. Information Transfer and Legibility
3. The Process
4. Complexity
5. Modularity
6. Automation
7. Fabrication
8. The Artifact</td><td>1. Aspects of Definition
2. Cultural Aspects</td></tr>
<tr><td>Not design-related &
adopted from literature</td></tr>
<tr><td>3. Economical aspects
4. Legal aspects
5. Social aspects</td></tr>
</table>

The third part of this thesis has been the case study of the WikiHouse project; in this study the WikiHouse project was confronted with the topics and challenges that have been identified in the second part. The questions guiding this confrontation were:

1. Does the WikiHouse project encounter the kinds of challenges that have been identified in the literature review?
2. If so, how are these challenges dealt with?
3. Do these challenges impact design or the design process?
4. If so, is this related to the (attempted) application of free & open source methods and how?

We have seen that (1) the WikiHouse project does encounter the kinds of challenges that have been identified in part two. Moreover, "observing," so to speak, how these kinds of challenges arise in an actual project and how they are dealt with contributes to our understanding of these challenges, given that the discussion in the literature is at times quite vague. (2) However, the ways in which it deals with these challenges are quite diverse (please see the conclusion of the second part for details). In general, they try to strike the right balance between keeping the threshold for building a WikiHouse or participating in the project low, and making concessions to the difficulties of being a pioneer project in architecture, for example, the lack of appropriate free & open software tools, as well as the difficulties of applying free & open methods to the production of tangible goods, above all, the need to find a way in

Table 10: Design-related challenges that free & open architecture projects are likely to face, by category

Tools and Technological Infrastructure	Editing tools
	Collaboration tools
	Repository solutions and version control tools
	Expectations regarding the impact of tools – mass-customization & the integration of laypeople
Information Transfer and Legibility	Standards
	Documentation
The Process	Design as an evolutionary process
	Feedback
Complexity	Effects of complexity
	Simplicity
Modularity	Properties of modular design and artifacts
	Conditions for modularity in software & hardware
Automation	Digital automation in hardware
	Digital methods in design and fabrication
	Unrewarding tasks.
Fabrication	Interpretation and ambiguity
	Loci of production
	Design for community-based production
	Means of production and supply
	Prototyping and replication
The Artifact	Disassembly
	Lifetime
	Material aspects
	Performance aspects

which WikiHouse can be fabricated and built. (3) These challenges, and the way in which the WikiHouse projects deals with them, were all found to impact design or the design process, though to varying degrees. The challenges that were found to have a profound impact either on design or the design process were: the goal that designs should be simple, the goal that designs should be modular, the desire to automate the design process, the employment of digital fabrication techniques and the goal that WikiHouses should be easy to disassemble and environmentally friendly (again, see the conclusion of part three for a more detailed picture). (4) However, the fact that these challenges affect design or the design process is not always related to the fact that the WikiHouse project tries to apply free & open methods (to avoid confusion, a challenge can be important to deal with for projects that apply free & open methods and yet not be strongly related to the application of these methods). The impact that a challenge has on design or the design process could, so I have argued, be related to the application of free & open methods only for (a) employing easily accessible software tools and non-proprietary file formats, (b) the goal that design should be modular, (c) the use of digital fabrication techniques and (d) the goal that WikiHouses should be easy to disassemble as well as environmentally friendly. I have put forward the following hypotheses on how the adoption of free & open source methods may be related to these challenges or the way in which these challenges are addressed: (a) If a projects wants to lower the threshold for its designs to be used and built upon by others, it is imperative that these designs can be easily read and altered. Furthermore, ensuring this, strictly speaking, requires the use of open, non-proprietary file formats and easy to use software that is available free of charge; since proprietary software tends to use proprietary file formats, this means that projects that want to adopt free & open methods are well-advised to use free & open source software wherever possible. (b) Of course, projects that want to apply free & open methods in architecture need not make their designs modular, but doing so may help to distribute design work among all members of the project, even if these members live in different cities or even different countries, in particular if there is lack of good documentation for developers; and given that maintaining good documentation requires considerable effort on its own, this is somewhat likely. (c) If such a project chooses its community as (one) *locus*[48] of production (possibly among others) and wants to enable its community to produce its artifact by developing for digital fabrication, then it must design in a

48. Just as a reminder, I adopt the concept of "*loci* of production" from Raasch et al. (2009), who use it to describe by what kind of organization or community a good is produced, e.g., by the project's community, by the manufacturing company who runs the project or by an "external" manufacturing company.

way that allows its design to be manufactured using widely available and reasonably cheap infrastructure and materials. Needless to say, this impacts the design. What is more, it also impacts the design process, because designing in such way is subject to various constraints and these are so numerous that the design process should start with taking them into account in general, rather than regarding a particular design idea. (d) Finally, I have reasoned that free & open architecture projects may find it difficult to attract professional architects, because professional architects may not want to participate in projects that might be associated with amateurism. If this is so, then it may be particularly important for such projects to meet high performance standards, of which being environmentally friendly is just one example. These hypotheses are the main findings of the third part and, together with the findings of part one, the foundation for the overall conclusion of this thesis (see table 11 for an overview of the findings of part three).

Table 11: Summary of findings of part three

Category	**(1) Actual concern?**	**(3) Impacts design/ the design process?**	**(4) Impact related to free & open methods?**
Tools and Technical Infrastructure	Yes	Yes, but as impediment	Yes
Information Transfer and Legibility	Yes	Yes, but as impediment	No
The Process	Yes, but marginal	Yes, but limited	*Not enough data*
Complexity	Yes, but marginal	Yes	No
Modularity	Yes	Yes	Yes, though not necessarily
Automation	Yes	Yes	No
Fabrication	Yes	Yes	Depending on *locus* of production
The Artifact	Yes	Design only	Maybe, but indirectly

After having discussed the findings of each part of this thesis, it is time to draw an overall conclusion, by generalizing the results of the case study and relating these results to the discussion of authorship in part one. If you recall, this conclusion shall be guided by the following questions:

1. If a challenge (or the way in which it is addressed) impacts design or the design process and that impact is related to the (attempted) application of free & open source methods, are we justified to assume that this would be the case in general?
2. Do these challenges or the way in which the WikiHouse project addresses them affect the way in which we, as architects, should understand and approach our role as authors?

I have just described the connections between applying free & open methods and (a) employing easily accessible software tools and non-proprietary file formats, (b) the goal that design should be modular, (c) the use of digital fabrication techniques and (d) the goal that WikiHouses should be easy to disassemble as well as environmentally friendly. Can these connections be generalized? Do these challenges or the way in which the WikiHouse project addresses them affect the way in which we should understand and approach our role as authors?

(1) Can these connections be generalized? (a) If a projects wants its designs to be open and accessible, it needs to use open and accessible file formats and software tools; there would seem to be no reason why this should be particular to the WikiHouse project. (b) Modular designs facilitate the distribution of design and development work between contributors of a project. While this is not only a benefit to projects that apply free & open methods, such projects often need to coordinate team members who live in different cities or even different countries, so that they tend to be in great need of solutions for coordination problems. That said, designing and developing modular designs comes with costs of its own, hence projects will need to decide on whether the effort to make designs modular is worth the reward of being able to design or develop different parts in parallel. Hence, the connection between free & open methods and the impact of the goal of keeping design modular is limited to projects that benefit from being able to design or develop different parts of their design in parallel (provided that the projects make rational choices). (c) Projects that apply free & open methods to the design and production of tangible goods, typically, lack the resources to manufacture their product themselves (save for those that are run by manufacturing companies). Hence, the WikiHouse project has chosen to let its community do the production, enabling them to do so by designing for digital fabrication (which can be done in fab labs). Of course, designing for digital fabrication has

numerous consequences for design as well as the design process. However, digital fabrication is just one way to enable a community to produce a product. What needs to be kept in mind during the design process are the resources that are available to a community; those resources may, as in the WikiHouse projects, be CNC routers, but it could just as well be simpler tools – or more complex ones. Hence, we can say that the relation between the application of free & open source methods to the design and production of tangible goods, and designing for production by the community applies to all projects that chose to their community as (one) *locus* of production, regardless of what fabrication technique is employed. (d) I have argued that free & open architecture projects may find it easier to attract professionals if they try to meet high performance standards; again there would seem to be no reason why this should be particular to the WikiHouse project.

(2) Do these challenges or the way in which they are addressed affect how we understand and approach our role as authors? (a) I have argued (in part one as well as in part two) that editing tools, version control systems and repository software are pivotal for the kind of authorship by the many which most free & open projects aim at. This is so because such software not only allows to read and change designs, but also keeps track of those changes, makes all version of a design available online and allows people to add their own version to the repository. However, we have seen that, while the WikiHouse project, too, appears to aim for this kind of authorship, at least they declare that the development of such tools is one of their goals, there are, as of now, no free & open source software tools that provide all of these functions and suit the choices of the WikiHouse projects. There are, of course, proprietary software products that provide at least some of those functions, but these software products are not widely available, in the sense that everybody has easy access to them. Hence, using them would undermine the very goal of applying free & open methods in the first place. (b)–(d) However, for all the other aforementioned challenges it is hard to see how they could impact how we understand and approach our role as authors, that is, our notion of architectural authorship. This may come as a surprise, at least it did to me. The WikiHouse project, arguably the most successful open architecture project to date, does little that questions or challenges our preconceived, traditional notion of architectural authorship. That is, save from requiring contributors to put the designs in the public domain or license them under a Creative Commons license. Are free & open methods not supposed to run counter to that notion? Why is the WikiHouse

project different? Because it did not, so far, establish an evolutionary design process or include feedback loops into that process. As I have argued in part three, the WikiHouse project lists these as future aims, but as of yet has not pushed the subject or established the necessary infrastructure for that kind of development process. The fact that the WikiHouse projects also lacks extensive documentation for developers further contributes to that problem. Of course, if the project survives long enough to reach these goals, these things may change. And given its success so far, there is no reason to suppose otherwise. The WikiHouse project may simply need more time to become a Wikipedia of architectural designs. Once it has reached this goal, we should make sure to revisit the project – and its impact on architectural authorship.

References

Abdelkafi, N., Blecker, T., & Raasch, C. (2009). From open source in the digital to the physical world: a smooth transfer? *Management Decision*, *Vol. 47*(No. 10), 1610–1632. http://doi.org/10.1108/00251740911004718

Ågerfalk, P. J., & Fitzgerald, B. (2008). Outsourcing to an Unknown Workforce: Exploring Opensurcing as a Global Sourcing Strategy. *MIS Quarterly*, 385–409. Retrieved from http://www.jstor.org/stable/25148845

Alexander, C. (1964). *Notes on the Synthesis of Form* (Vol. 5). Harvard University Press. Retrieved from http://books.google.at/books?hl=en&lr=&id=Kh3T3XFUfPQC&oi=fnd&pg=PA1&dq=Notes+on+the+Synthesis+of+Form&ots=_C4ZMDnqPv&sig=Ne2JeT7tBWBCKT3Xbny9YxJU-VI

Alexander, C., Ishikawa, S., & Silverstein, M. (1977). *A pattern language: towns, buildings, construction*. New York: Oxford University Press.

Archis. (2003). *Archis No. 3 - Open Source*. Utrecht: Van Loghum Slaterus.

Autodesk (Ed.). (2011, February). AutoCAD 2012 DXF Reference. Retrieved from http://images.autodesk.com/adsk/files/autocad_2012_pdf_dxf-reference_enu.pdf

Baldwin, C. Y., & Clark, K. B. (2000). *Design Rules : The Power of Modularity*. Cambridge, Massua: MIT Press.

Balka, K. (2011). *Open source product development : the meaning and relevance of openness* (1. ed.). Wiesbaden: Gabler.

Barthes, R. (1967). The death of the author. *Aspen No. 5+6*. Retrieved from http://www.ubu.com/aspen/aspen5and6/threeEssays.html#barthes

Bast, L. (2015a). *WikiHouseVienna WikiHouse Studio construction* [JPEG Image].

Bast, L. (2015b). *WikiHouseVienna WikiHouse Studio outside view* [JPEG Image].

Baukosteninformationszentrum Deutscher Architektenkammern. (2013). BKI-Baukosten : Regionalfaktoren 2013 für Deutschland und Europa. Stuttgart: BKI.

Benkler, Y. (2002). Coase's Penguin, or, Linux and" The Nature of the Firm." *Yale Law Journal*, 369–446. Retrieved from http://www.jstor.org/stable/10.2307/1562247

Benkler, Y. (2006). *The wealth of networks how social production transforms markets and freedom*. New Haven [Conn.]: Yale University Press. Retrieved from http://site.ebrary.com/id/10170022

Beseda, P., & Williams, L. (2013a). *FOUNDhouse inside view* [JPEG Image]. Retrieved from zotero://attachment/1843/tumblr_mylcvq4rPt1swgup1o3_1280.jpg

Beseda, P., & Williams, L. (2013b). *FOUNDhouse outside view landscape* [JPEG Image]. Retrieved from zotero://attachment/1843/tumblr_mylcvq4rPt1swgup1o9_1280.jpg

Bessen, J., & Meurer, M. J. (2007). What's wrong with the patent system? Fuzzy boundaries and the patent tax. *First Monday, 12*(6). Retrieved from http://www.firstmonday.dk/ojs/index.php/fm/article/view/1867/1750

Braun, V. R. (2007). *Barriers to user-innovation & the paradigm of licensing to innovate*.

Cache, B., & Speaks, M. (1995). *Earth moves: the furnishing of territories*. Cambridge, Mass.: MIT Press.

Carpo, M. (2011). *The alphabet and the algorithm*. Cambridge, Mass.: MIT Press. Retrieved from http://site.ebrary.com/id/10453040

Carse, J. (2014). *A-Barn outside view with finished cladding* [JPEG Image].

CERN. (2011, July 13). CERN Open Hardware Licence - CERN OHL version 1.1. Retrieved August 24, 2013, from http://www.ohwr.org/documents/88

Creative Commons. (2015). Understanding Free Cultural Works. Retrieved from http://creativecommons.org/freeworks

Deleuze, G. (1993). *The fold: Leibniz and the Baroque*. Minneapolis: University of Minnesota Press.

de Rivera Costaguti, A. A. (2015). *Farmhouse chassis outide view* [JPEG Image].

Design team 00, & WikiHouse. (2015). *StudioDrawings* [JPEG Image]. Retrieved from http://wikihouse.wpengine.com/wp-content/uploads/2014/10/StudioDrawings.jpg

DiBona, C., Stone, M., & Cooper, D. (2005). *Open Sources 2.0* (Auflage: 1., Aufl.). Beijing ; Sebastopol, CA: O'Reilly & Associates.

Diem, G., & IG Passivhaus Vorarlberg (Eds.). (2011). Sie fragen. Wir antworten. *Passivhaustrend - Zukunftsgerechtes Wohnen*, (magazin 3), 22–23. Retrieved from http://www.innovativegebaeude.at/fileadmin/media/ig_vorarlberg/magazin_3.pdf

Domus. (2011a). *Domus 948*. Milano: Domus. Retrieved from https://www.domusweb.it/en/news/2011/06/06/domus-948-in-newsstands-now.html

Domus. (2011b, June 6). Domus 948 in newsstands now. Retrieved August 1, 2015, from https://www.domusweb.it/en/news/2011/06/06/domus-948-in-newsstands-now.html

Ecosistema Urbano. (2010a). Air Tree Commons - descarga, copia, modifica y comparte | ecosistema urbano. Retrieved October 16, 2013, from http://ecosistemaurbano.org/castellano/air-tree-commons-descarga-copia-modifica-y-comparte/

Ecosistema Urbano. (2010b). AIR TREE SHANGHAI | ecosistema urbano . portfolio. Retrieved August 1, 2015, from http://ecosistemaurbano.com/portfolio/air-tree/

Ecosistma Urbano. (2010). Air Tree Commons - Architecture. Retrieved from http://ecosistemaurbano.org/download/Ecosistema-Urbano-Air-Tree-Commons-Architecture.pdf

Fisher, C., Parvin, A., Luff, M., & Squires, D. (2013, March 23). Rhino Users - a clean .3DM file of WikiHouse v3.0 - Google Groups. Retrieved from https://groups.google.com/forum/?fromgroups#!topic/wikihouse-software/RE3g_FkfRac

Fogel, K. (2005). *Producing Open Source Software How to Run a Successful Free Software Project*. O'Reilly Media. Retrieved from http://producingoss.com/

Freedom Defined. (2008, December). Definition of Free Cultural Works v.1.1. Retrieved November 23, 2012, from http://freedomdefined.org/Definition

Free Software Foundation. (2013, May). Free Software Definition v. 1.121. Retrieved November 18, 2012, from http://www.gnu.org/philosophy/free-sw.html

Free Software Foundation. (2015, September 1). What is free software? The Free Software Definition. Retrieved October 1, 2015, from http://www.gnu.org/philosophy/free-sw.en.html

Gibb, A. (2014). OSHW Definition and Best Practices. In *Building Open Source Hardware: DIY Manufacturing for Hackers and Makers*. Addison-Wesley Professional. Retrieved from http://proquest.tech.safaribooksonline.de/book/hardware/9780133373912/idot-open-source-hardware-theory/ch02_html

Gibb, A., Adabie, S., & Baafi, E. (2014). *Building Open Source Hardware: DIY Manufacturing for Hackers and Makers*. Addison-Wesley Professional. Retrieved from http://proquest.tech.safaribooksonline.de/book/hardware/9780133373912

Giovannelli, A. (2010). Goodman's Aesthetics. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy* (Summer 2010). Retrieved from http://plato.stanford.edu/archives/sum2010/entries/goodman-aesthetics/

Goodman, N. (1968). *Languages of art: an approach to a theory of symbols*. Indianapolis: Bobbs-Merrill.

Grassmuck, V. (2004). *Freie Software - Zwischen Privat- und Gemeineigentum*. Bonn: Bundeszentrale für politische Bildung. Retrieved from http://freie-software.bpb.de/Grassmuck.pdf

Haque, U. (2007). Distinguishing Concepts: Lexicons of Interactive Art and Architecture. *Architectural Design*, *77*(4), 24–31. Retrieved from http://onlinelibrary.wiley.com/doi/10.1002/ad.484/abstract

Hippel, E. von. (2005). *Democratizing innovation*. Cambridge, Mass.: MIT Press.

Jencks, C. (2001). Das neue Paradigma. In *Cybertecture: die 4. Dimension in der Architektur*. Wien: Löcker.

Kaspori, D. (2003). A communism of ideas. Towards an architectural open source practice. *Archis*, *Open Source*(2003, 3), 13–18.

Kettle, R. (2013, November 12). /*made on jupiter*/ – Goodbye Spoke Creator, Hello MatterMachine! Retrieved from http://www.madeonjupiter.com/goodbye-spoke-creator-hello-mattermachine/

Kuhlmann, D. (2001). Der Tod des Architekten. In *Cybertecture: die 4. Dimension in der Architektur*. Wien: Löcker.

Lessig, L. (1999). *Code and other laws of cyberspace*. New York: Basic Books.

Lutterbeck, B., Bärwolff, M., & Gehring, R. A. (Eds.). (2004). *Open Source Jahrbuch*. Berlin: Lehmanns Media. Retrieved from http://www.opensourcejahrbuch.de/project

MacCormack, A., Rusnak, J., & Baldwin, C. Y. (2006). Exploring the structure of complex software designs: An empirical study of open source and proprietary code. *Management Science*, *52*(7), 1015–1030. Retrieved from http://pubsonline.informs.org/doi/abs/10.1287/mnsc.1060.0552

Marcin. (2015, February 27). Open Source Hardware Development Method | Open Source Ecology. Retrieved from http://opensourceecology.org/open-source-hardware-development-method/

Michel Avital. (2011). THE GENERATIVE BEDROCK OF OPEN DESIGN | Open Design Now. Retrieved October 9, 2013, from http://opendesignnow.org/index.php/article/the-generative-bedrock-of-open-desig n-michel-avital/

Mulgan, Geoff, Steinberg, Tom, & Salem, Omar. (2005). *Wide Open – Open source methods and their future potential*. London: Demos. Retrieved from http://dlc.dlib.indiana.edu/dlc/bitstream/handle/10535/11/wideopen.pdf? sequence=1

Mushon Zer-Aviv, Mike Linksvayer, Michael Mandiberg, Marta Peirano, Alan Toner, Adam Hyde, … Astra Taylor. (2010). *Collaborative Futures*. Retrieved from https://www.createspace.com/3480309?ref=1147694&utm_id=6026

Nuvolari, A., & Rullani, F. (2007). Curious Exceptions? *Handbook of Research on Open Source Software: Technological, Economic, and*, 227. Retrieved from http://books.google.at/books?hl=en&lr=&id=75KT6GdcWbYC&oi=fnd&pg=PA227&dq=Nuvolari,+A.,+Rullani,+F.+(2007)+Curious+exceptions%3F+Open+source+software+a&ots=s53dsvYZ-q&sig=crg2TembhWgk2ri5mYn5Frd

h7io

OANDA Corporation. (2015). Average Exchange Rates Jan.-Dec. 2014. Retrieved September 25, 2015, from http://www.oanda.com/currency/average

OD+H Group. (2013). Open Design Definition v. 0.3 [GitHub]. Retrieved August 28, 2013, from https://github.com/OpenDesign-WorkingGroup/Open-Design-Definition

OHANDA. (2009, July). OHANDA - 4 Freedoms. Retrieved August 28, 2013, from http://reables.ohanda.org/

Open Design Foundation. (2000). The Open Design Definition V. 0.2. Retrieved September 30, 2015, from http://www.opendesign.org/odd.html

Open Knowledge Foundation. (2009, November). Open Definition v1.1. Retrieved November 23, 2012, from http://opendefinition.org/okd/

Open Source Initiative. (2002). The Open Source Definition v.1.9. Retrieved September 4, 2013, from http://opensource.org/osd

Open Source Innovation Project. (2014). Open Source Innovation. Retrieved November 11, 2014, from https://www.tuhh.de/tim/open_source_innovation/index.html

OSE. (2012, November 30). Work Log Guidelines - Open Source Ecology [Wiki]. Retrieved January 13, 2013, from http://opensourceecology.org/wiki/Work_Log_Guidelines

OSE. (2013a, January 23). Collaborative Production - Open Source Ecology [Wiki]. Retrieved November 24, 2014, from http://opensourceecology.org/wiki/Collaborative_Production

OSE. (2013b, August 2). OSE Specifications - Open Source Ecology [Wiki]. Retrieved January 13, 2013, from http://opensourceecology.org/wiki/OSE_Specifications

OSE. (2014, January 14). Documentation Standards - Open Source Ecology [Wiki]. Retrieved November 13, 2014, from http://opensourceecology.org/wiki/Documentation_Standards

OSHWA. (2013a, July 9). OSHWA - Brief History of Open Source Hardware Organizations and Definitions. Retrieved January 29, 2015, from http://www.oshwa.org/research/brief-history-of-open-source-hardware-organizations-and-definitions/

OSHWA. (2013b, September 23). Open-Source Hardware Statement of Principles and Definition v.1.0. Retrieved August 22, 2013, from http://www.oshwa.org/definition/

OSHWA. (2014, November 3). OSHW - Definition of Free Cultural Works. Retrieved November 19, 2014, from http://freedomdefined.org/OSHW

Osterloh, M., & Rota, S. (2007). Open source software development—Just another case of collective invention? *Research Policy*, *36*(2), 157–171.

http://doi.org/10.1016/j.respol.2006.10.004

Oxman, R., & Oxman, R. (2013). *Theories of the Digital in Architecture*. London ; New York: Taylor & Francis Ltd.

Parvin, A. (2013). *Alastair Parvin: Architecture for the people by the people*. Coference Talk presented at the official TED conference. Retrieved from http://www.ted.com/talks/alastair_parvin_architecture_for_the_people_by_the_people

Parvin, A. (2014). *A-Barn chassis outside view* [JPEG Image].

Parvin, A., Gold, S., & WikiHouse Design Team 00. (2014, July). A-Barn, Basic Assembly Manual v1.

Pepper, L. (2014). *A-Barn frame rising* [JPEG Image].

Raasch, C. (2011). Product Development In Open Design Communities: A Process Perspective. *International Journal of Innovation and Technology Management*, *8*(04), 557–575. Retrieved from http://www.worldscientific.com/doi/abs/10.1142/S021987701100260X

Raasch, C., Herstatt, C., & Abdelkafi, N. (2008). Creating Open Source Innovation: Outside the Software Industry. In *PICMET 2008 Proceedings*. Cape Town, South Africa: 2008 PICMET. Retrieved from http://catalogplus.tuwien.ac.at/UTW:UTW:TN_ieee10.1109/PICMET.2008.4599652

Raasch, C., Herstatt, C., & Balka, K. (2009). On the open design of tangible goods. *R&d Management*, *39*(4), 382–393. Retrieved from http://onlinelibrary.wiley.com/doi/10.1111/j.1467-9310.2009.00567.x/full

Ratti, C., Antonelli, P., Bly, A., Dietrich, L., Grima, J., Hill, D., … Sterling, B. (2011). Open Source Architecture (OSArc) - Op-Ed - Domus. *Domus*, (948). Retrieved from http://www.domusweb.it/en/op-ed/2011/06/15/open-source-architecture-osarc-.html

Raymond, E. S. (1999). *The cathedral & the bazaar musings on Linux and open source by an accidental revolutionary*. O'Reilly. Retrieved from http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=24619

Rosada, A. (2012). L'open source come modello di sviluppo dell'abitare contemporaneo. In *Abitare il nuovo/abitare di nuovo ai tempi della crisi* (pp. 431–442). Napoli: CLEAN Edizioni. Retrieved from http://porto.polito.it/2506435/

Schimelpfenig, E. (2014, January 2). WikiHouse Project [Blog]. Retrieved from http://www.sketchthis.net/blog/?

offset=1388951335000&category=Wikihouse+Project

Shah, S. K. (2005). Open Beyond Software. In *Open Sources 2.0*. O'Reilly Media.

Shirky, C. (2005). Epilogue: Open Source outside the domain of software. In *Perspectives on free and open source software* (pp. 483–488). The MIT Press.

Sinclair, C. (2006). *My wish: A call for open-source architecture*. Conference Talk presented at the official TED conference. Retrieved from http://www.ted.com/talks/cameron_sinclair_on_open_source_architecture

Snake-Beings, E. (2010). The construction of Karen Karnak: The multi-author-function. The Unive rsity of Waikato. Retrieved from http://researchcommons.waikato.ac.nz/bitstream/handle/10289/4327/thesis.pdf?sequence=1&isAllowed=y

Statistisches Bundesamt. (2015a). *Bauen und Wohnen Baugenehmigungen /Baufertigstellungen Baukosten Lange Reihen z. T. ab 1962* (No. Artikelnummer: 5311103147004). Wiesbaden. Retrieved from https://www.destatis.de/DE/Publikationen/Thematisch/Bauen/BautaetigkeitWohnungsbau/BaugenehmigungenBaukostenPDF_5311103.pdf?__blob=publicationFile

Statistisches Bundesamt. (2015b). *Baufertigstellungen von Wohn- und Nichtwohngebäuden (Neubau) nach überwiegend verwendetem Baustoff Lange Reihen ab 2000* (No. Artikelnummer: 5311202147004). Wiesbaden. Retrieved from https://www.destatis.de/DE/Publikationen/Thematisch/Bauen/BautaetigkeitWohnungsbau/BaufertigstellungenBaustoffPDF_5311202.pdf?__blob=publicationFile

Tai, L.-C. (Andy). (2001, July 4). The History of the GNU General Public License. Retrieved August 7, 2015, from http://www.free-soft.org/gpl_history/

TAPR. (2007, Ma y). The TAPR Open Hardware License v1.0. Retrieved August 24, 2013, from http://www.tapr.org/ohl.html

Taylor, P. (2012). *Parametric Wikihouse*. Retrieved from http://vimeo.com/41183870

Team 00. (2011). *Early WikiHouse Prototype* [JPEG Image].

The Center for Bits and Atoms, MIT. (2015). Fab Lab FAQ. Retrieved May 11, 2015, from http://fab.cba.mit.edu/about/faq/

Troxler, P. (2010). Commons-based peer-production of physical goods: Is there room for a hybrid innovation ecology? In *3rd Free culture research conference*. Berlin. Retrieved from http://papers.ssrn.com/sol3/papers.cfm?abstract_id=1692617

Vallance, R., Kiani, S., & Nayfeh, S. (2001). Open design of manufacturing equipment. In *CIRP 1st International conference on agile, reconfigurable manufacturing, Ann Arbor*. Retrieved from http://diyhpl.us/~bryan/papers2/open-source/Open%20design%20of%20manufacturing%20equipment.pdf

Vardouli, T., & Buechley, L. (2014). Open source architecture: An exploration of source code and access in architectural design. *Leonardo*, *47*(1), 51–55. Retrieved from http://www.mitpressjournals.org/doi/abs/10.1162/LEON_a_00470

Verschuren, P., & Doorewaard, H. (2010). *Designing a Research Project: Second Edition* (Auflage: 2). The Hague: Boom Eleven International.

von Hippel, E., & von Krogh, G. (2003). Open source software and the "private-collective" innovation model: Issues for organization science. *Organization Science*, *14*(2), 209–223. Retrieved from http://pubsonline.informs.org/doi/abs/10.1287/orsc.14.2.209.14992

von Krogh, G., & von Hippel, E. (2006). The Promise of Research on Open Source Software. *Management Science*, *52*(7), 975–983. http://doi.org/10.1287/mnsc.1060.0560

Weber, S. (2004). *The success of open source*. Cambridge, MA: Harvard University Press.

WikiHouse. (2013a). WikiHouse / How it Works. Retrieved October 18, 2013, from http://www.wikihouse.cc/about

WikiHouse. (2013b). WikiHouse / Open Library. Retrieved October 19, 2013, from http://www.wikihouse.cc/libra

WikiHouse. (2013c, May). WikiHouse v3.0 Design Guide. Retrieved from http://www.wikihouse.cc/static/doc/75d156345c28a296c1dd67ed1ea213d204bdef15-WikiHouse_Design_Guide.pdf

WikiHouse. (2013d, May 24). WikiHouse Development Goals. Retrieved from https://github.com/00/wikihouse-resources/blob/master/doc/WikiHouseDevelopmentGoals.pdf

WikiHouse. (2013e, July). Constitution_draft_v4.pdf.

WikiHouse. (2014a). Press release - Winter 2014. Retrieved June 2, 2015, from https://docs.google.com/document/d/1kYN2R60mLKLiCj1Au-HhgEtmG6wv4gAUTyflG3OaEWQ/edit?usp=drive_web&usp=embed_facebook

WikiHouse. (2014b). WikiHouse / Browse Library. Retrieved March 13, 2014, from http://www.wikihouse.cc/library

WikiHouse. (2014c, Okt). WikiHouse Constitution v6.0 , October 2014. Retrieved from https://drive.google.com/open?id=0BxKrrlaHSn_UVzhEWDRkYXlCSHM&authuser=0

WikiHouse. (2014d). WikiHouse Guide for Designers web. Retrieved March 13, 2014, from http://www.wikihouse.cc/guide

WikiHouse. (2014e). WikiHouse Milan Pop-up. Retrieved January 29, 2014, from http://www.wikihouse.cc/library/designs/47001

WikiHouse. (2014f). WikiHouse Modelling Standards web. Retrieved March 13, 2014, from http://www.wikihouse.cc/guide/standards

WikiHouse. (2014g, January). WikiHouse Development Goals v2. Retrieved from https://dl.dropboxusercontent.com/u/1850356/WikiHouse%20Public%20Docs/WikiHouseDevGoals.pdf

WikiHouse. (2014h, February 5). WikiHouse. Where are we? Retrieved from https://drive.google.com/open?id=17VyyknU8sxAAB8PkPiTEFapTVtfJxZng7LF3wG1IN0M&authuser=0

WikiHouse. (2014i, March 5). Chapters Directory. Retrieved from https://drive.google.com/open?id=1qV0DWvAGHG6i91-oOvJwq6dQITAq7rAqwpBpggrVbkg&authuser=0

WikiHouse. (2014j, August 29). Contributors. Retrieved from https://docs.google.com/document/d/1nXDk_4ZDk4IX4HdK6WdfYeNOFPzipvvvSED0XX0TeRk/edit

WikiHouse. (2014k, November). The WikiHouse Catalogue. Retrieved from https://doc-0o-9s-docs.googleusercontent.com/docs/securesc/j0qf4lo0tqb9eairo1e5l4gbinrjf2rt/mibuqn57dmdmsggi1rr74gej3628g7sc/1434038400000/10348834673217211408/15215029581124015448/0B18DH6S_7dR6YllnLS1iNVNXUGM?e=download

WikiHouse. (2014l, December). WikiHouse + Spoke Open Brief. Retrieved from https://docs.google.com/document/export?format=odt&id=17dlAoBriESSR55bMf5dh1gvFjthyoCk3t8VZD4wSTwA&token=AC4w5ViL6qvo1W-g7ZE_klnw2DF9_-EEOg%3A1430487235543

WikiHouse. (2014m, December 19). How to use the WikiHouse Commons. Retrieved from https://drive.google.com/open?id=1UTYpXb-wODmS9tyNgUPB64QQLO4T_FgidJnQg2NiWks&authuser=0

WikiHouse. (2015a). About | WikiHouse. Retrieved April 2, 2015, from http://www.wikihouse.cc/about/

WikiHouse. (2015b). About - FAQ | WikiHouse. Retrieved June 22, 2015, from http://www.wikihouse.cc/about/#FAQ

WikiHouse. (2015c). Commons | WikiHouse. Retrieved April 2, 2015, from http://www.wikihouse.cc/commons/

WikiHouse. (2015d). Studio | WikiHouse. Retrieved July 1, 2015, from http://www.wikihouse.cc/studio/

WikiHouse. (2015e). WikiHouse 4.0 | WikiHouse. Retrieved June 17, 2015, from http://www.wikihouse.cc/wikihouse-4-0/

WikiHouse. (2015f, July 6). A General Guide for Assemblers. WIKIHOUSE CHASSIS SYSTEM Version 4.2.1. Retrieved from https://docs.google.com/document/u/0/d/1yn-hqHkR_xNXEVyG0xcyZCPhejClhc6S-zdFu7W7_jQ/export?format=docx

WikiHouse. (2015g, October 19). WikiHouse - Google Groups. Retrieved October 19, 2015, from https://groups.google.com/forum/#!aboutgroup/wikihouse

WikiHouse. (2015h, November 2). WikiHouseMap_v4. Retrieved from https://drive.google.com/file/d/0B18DH6S_7dR6bk1mTXpwMXFkd3M/view

WikiHouse. (2015i, December 9). Community. Retrieved from http://www.wikihouse.cc/community/

WikiHouse, & Chatainger, M. (2013). *MarcChatainger_WikiHouse_5_CC_NC.jpg* [JPEG Image]. Retrieved from https://drive.google.com/open?id=0BxKrrlaHSn_UVEt0V2RvQzlPR3c

WikiHouse, & Knight, H. (2015). *Farmhouse18_WikiHouseFoundation_CC_BY.jpg* [JPEG Image]. Retrieved from https://drive.google.com/open?id=0B12i40YU63NXTHZpRFNzUjVFVVU

WikiHouseNL. (2015). *WikiHouseN tower at Rotterdam* [JPEG Image].

Wikipedia contributors. (2015, September 18). TED (conference). In *Wikipedia, the free encyclopedia*. Retrieved from https://en.wikipedia.org/w/index.php?title=TED_(conference)&oldid=681584810