FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

# Web Data Extraction of University Staff Competencies

DIPLOMARBEIT

zur Erlangung des akademischen Grades

**Diplom-Ingenieur**

im Rahmen des Studiums

**Software Engineering und Internet Computing**

eingereicht von

**Edin Zildzo**
Matrikelnummer 1125449

an der Fakultät für Informatik
der Technischen Universität Wien

Betreuung: Ao.Univ.Prof.Dr. Jürgen Dorn

Wien, 13. September 2015

| Edin Zildzo | Jürgen Dorn |
|---|---|

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

# Web Data Extraction of University Staff Competencies

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

**Diplom-Ingenieur**

in

**Software Engineering and Internet Computing**

by

**Edin Zildzo**
Registration Number 1125449

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Ao.Univ.Prof.Dr. Jürgen Dorn

Vienna, 13th September, 2015

Edin Zildzo

Jürgen Dorn

# Erklärung zur Verfassung der Arbeit

Edin Zildzo
Leibnizgasse 51/7 1100 Wien

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

Wien, 13. September 2015

Edin Zildzo

# Acknowledgements

I would like especially to thank my advisor Prof. Dr. Jürgen Dorn for his support, remarks, guidance and his whole engagement for this thesis.

I would like also to thank my family and my friends for their support. My aunt Nermina Zildzo was always giving me useful advices and with those advices kept me motivated to finish my thesis.

Then, I would like to mention my good friend Ermin Hasic who was studying together with me and helping me to overcome tough situations and solve problems.

I dedicate this thesis to my parents Jasmin and Aida and my brother Semir, and I would like to thank them for their patience, encouragement and support during my studies.

# Kurzfassung

Diese Arbeit präsentiert einen Ansatz um Daten von Mitarbeitern einer Universität und ihre berufliche Kompetenzen zu extrahieren, während man sich mit Themen wie Web-Datenextraktion Probleme wie z.B. Seitenstruktur Probleme, Dynamic Data, Unstrukturierte Daten usw. auseinandersetzen muss. Die Software die in dieser Arbeit vorgeschlagen versucht Daten zu extrahieren von Universitätsmitarbeitern und ihre berufliche Kompetenzen und das soweit möglich von den meisten Universitäten. Die Professionellen Kompetenzen der Universitätsmitarbeiter werden durch die Verwendung einer bestehenden Ontologie ermittelt, die erweitert wird um die Domäne eines Anwendungsfalls abzudecken.

Diese Software wird aus zwei Teilen zusammengesetzt sein. Der erste Teil wird sich damit beschäftigen um alle nötigen Daten zu besorgen die ein Input für den zweiten Teil sein wird, welches ein Datenextrahieren ist. Es wird den SelectorGadget Bookmarklet verwenden, welches die CSS-selector Elemente bereitstellt mit denen Benutzer auswählen können welche Daten sie für die Extraktion möchten. Das andere Bookmarklet wird den Input von SelectorGadget bekommen und es dann an das Data extraction Software weiterleiten.

Um das vorgeschlagene Verfahren zur Extraktion von den oben angeführten Daten zu evaluieren wird eine Umfrage durchgeführt mit Mitarbeitern eines Anwendungsfalls. Die Genauigkeit der extrahierten Kompetenzen die durch diese Arbeit ermittelt wurden wird durch den Vergleich mit Kompetenzen die aus der Umfrage gewonnen wurden gemessen.

# Abstract

This thesis presents an approach to extract University staff professional competences while dealing with Web data extraction issues like page structure problems, dynamic data, unstructured data, etc... The software which is proposed in this thesis will tend to extract University staff professional competences from most of the Universities and if it is possible to work for all. University staff professional competences will be determined by using an existing ontology which will be extended to cover the domain of an use case.

That software will be composed of two parts. First part will be dealing with gathering required data which will be an input for the second part which will be a data extractor. It will use SelectorGadget bookmarklet which provides CSS-selector elements with which users can select desired data for extraction. The other bookmarklet will get the input from SelectorGadget and pass it to data extraction software.

In order to evaluate the proposed method for extracting University staff professional competences the survey will be conducted with staff members of an use case. The accuracy of extracted competences obtained by the method from this thesis will be measured by comparing them with competences obtained from the survey.

# Contents

# CHAPTER 1

# Introduction

## 1.1 Introduction and motivation

Web data extraction is a challenging process due to complex data structures and unstructured data on various Web pages. Web pages with a wide variety in the styles, code and violations of standards are considered as unstructured and complex for data extraction process. Web pages are categorized based on the information represented in them, some web pages display static text, whereas others extract the information from the backend database dynamically during runtime, even some run complex scripts to generate data at the time of display. Finally the complete web page can be viewed as a combination of different types of content displayed in the form of visual blocks inside the Web browser window.[Nar13]

The common problem for Web data extraction tools is the structure of data on the Website. When data is written as a plain text without using the classifiers, it is very complex to identify what those text sections represent. The content of the majority of Web pages is not just a plain text, it is organized into various structures which makes it challenging for Web data extraction process. In order to navigate deeply into the Website to extract the data various techniques are used such as Xpath expressions and CSS selectors for selection and navigation of Web page elements.

All those Web data extraction tools have their advantages and disadvantages where some of those tools can extract data from pages with particular structure and other tools cannot and the other way around. From this fact we can conclude that it is very hard task to create a tool which will work for all Web data extraction cases.

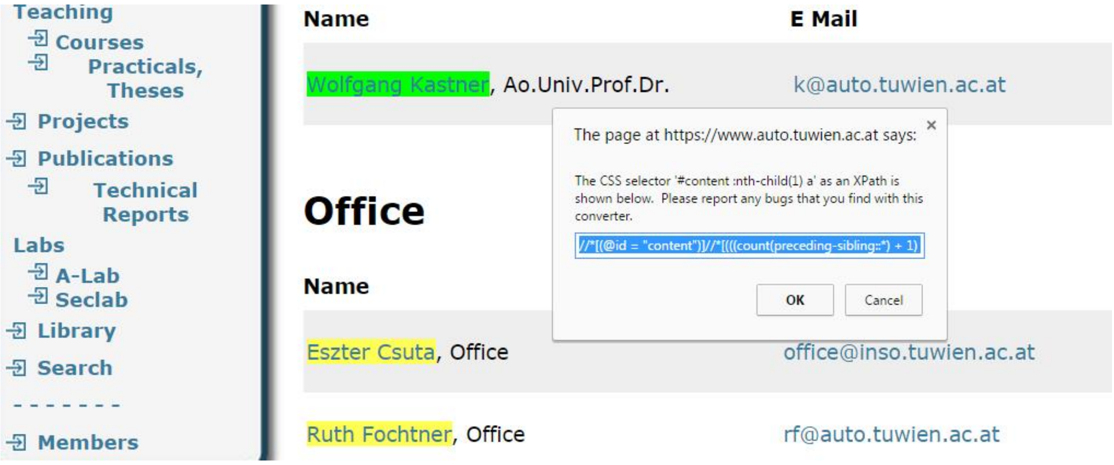
As almost everything can be found on the Web, user demand to search and query Web data arises which leads to particular difficulties. Users can browse for data very easily and can search for particular data but problems arise when there is a large amount of data which becomes cumbersome to locate and to search for. It is not practical to click on several next links when there is a large amount of data because users get lost in the whole process. Although most of Websites provide keyword search which in some sense simplifies the process of searching for data, those search results can contain undesired results. In order to solve these problems wrappers emerged which extracts data more efficiently and stores them in various formats like JSON, XML, CSV and other formats for further processing.

A Web data extraction system, interacts with a Web source and extracts data from it. If the source is in a form of a HTML Web page, the extracted information can be in form of elements of a page or the full text of the page. The extracted data could be latter processed and stored for further usage. The importance of Web data extraction systems is increasing because large amount of data is continuously produced online. When we think about commercial field, the data which can be found on Web can be very useful to some companies. In the *Competitive Intelligence* field, the acquiring and analysis of data about company competitors is very important. HTML is the dominant language for implementation of Web pages. HTML pages provide semi-structured data which is in a nested structure. [FMFB14]

In the early days of Web data extraction discipline, *learning-based* and *rule-based* approaches were used in data extraction process. Those approaches tend to delevop systems by using human expertise to define extraction rules. There was a requirement that users which define those extraction rules should have a profound programming knowledge and they should be domain experts in that domain where extraction process is done. But, nowadays many approaches have emerged which tend to leverage human involvement in the Web data extraction. Some of them are based on *Artificial Intelligence* in the sense that they adopt some algorithms which analyze structure of Web pages and extract data. *Machine learning supervised* and semi-supervised techniques are used to design systems which are able to function by learning from examples which then enables those systems to extract data from similar domains.[FMFB14]

Semi-structure of Web pages can be represented as *labeled ordered rooted trees*, where labels represent tags of the HTML language syntax and the tree hierarchy represents levels of elements in a tree structure of Web page. [FMFB14]

DOM is a representation of a Web page by labeled ordered tree. DOM vision is that a Web page structure is represented as plain text which contatins HTML tags which are interpreted by browser to represents elements of a page and those tags can be in a hierachical structure. DOM provides a possibility to exploit tools of XML languages. XPath is used for querying elements in a XML document and elements of a HTML. XPath can be used to identify a single element in a document tree or to identify multiple occurences of the same element. There could be a case in which content of some Web page is generated by a script which could cause problems to XPath expressions. In that case page structure changes and there is a need for human involvement in order to adapt XPath expressions to new structure. In order to deal with that issues *wrapper robustness* concept is introduced which finds XPath expressions which are less influenced by changes of a page structure. Figure 1.1 shows an example of XPath of some selected elements on a page. [FMFB14]



| Name | E Mail |
|---|---|
| Wolfgang Kastner, Ao.Univ.Prof.Dr. | k@auto.tuwien.ac.at |

Office

| Name | |
|---|---|
| Eszter Csuta, Office | office@inso.tuwien.ac.at |
| Ruth Fochtner, Office | rf@auto.tuwien.ac.at |

The page at https://www.auto.tuwien.ac.at says:

The CSS selector '#content :nth-child(1) a' as an XPath is shown below. Please report any bugs that you find with this converter.

```
//*[(@id = "content")]//*[(((count(preceding-sibling::*) + 1)
```

Figure 1.1: Xpath Example

Web wrappers are composed of the following steps:

1. Wrapper generation
2. Wrapper execution
3. Wrapper maintenance [FMFB14]

In first step wrapper is defined by using some techniques. It was a case earlier that wrappers were generated only manually which means experts were writing those wrappers which were able to identify target information and to extract it. But later on, some advanced techniques were developed in wrapper generation field which enabled users to use graphical user interfaces to define and execute wrappers. That method made a wrapper generation process more user friendly because users could make wrappers without their expertise in programming language in which wrapper is implemented. Wrappers might be executed by using a *wrapper induction* which uses high level automation strategies. [FMFB14]

There are three significant approaches used for wrapper generation:

1. *Regular-expression-based approach*
2. *Wrapper programming languages*
3. *Tree-based approach*

Regular expressions are a formal language used for identify patterns in unstructured text while using some matching criteria. Wrappers which are based on regular expressions dynamically generate rules for data extraction. Those regular expressions are based on some criteria like word boundaries, HTML tags, etc . . . The advantage of using regular expressions is when users select desired elements on a Web page, the system can automatically infer regular expressions to locate those elements. [FMFB14]

W4F is a tool which uses regular expressions for extracting data. W4F eases wrapper design by using a *wizard procedure*. This procedure allows users to select and annotate elements on a Web page and then W4F produces regular expressions for annotated items. Then then optimization process of those regular expressions is done by some experts due to accuracy issues. The well known issue regarding use of regular expressions is its sensitivity to structural changes of a Web page. When there is a change in a Web page structure, in most cases regular expressions stop working and there is a need for their maintenance. [FMFB14]

The other approach is *Wrapper programming languages*. Tools which use those languages consider Web pages as semi-structured tree documents where DOM represents page structure where nodes are elements which have properties and content. The advantage of this apprach over *regular-expression based approach* is that this approach uses wrapper programming languages to exploit semi-structure of a Web page and their content.

The *tree-based approach* is based on an assumption that information in Web pages are collected in contiguous regions of a page. This approach identifies and extracts those regions.

The fact is that Web pages change their structures. Some Web pages change it quite often and some quite rarely. This cause a wrapper to stop functioning correctly or even to stop working. There is a need for wrapper maintenance. It is very important to have some automatic strategies for wrapper maintenance in order to have more reliability in them, to have them working even when Web pages change structures and that they provide correct data. We can argue that wrapper maintenance is very important step in Web data extraction field. In the earlier stages, wrapper maintenance was done manually which means that wrapper creators were maintaining wrappers as Web sources change. That was working well for small amount of Web sources since there was a not much work to be done but difficulties emerged in the cases where there were many Web sources. The method which arised from the scientific field in order to handle this issue is *wrapper verification*. This method should be a required step in the execution of a wrapper which checks if wrappers will function correctly or they will have some issues due to modifications of Web page structure. [FMFB14]

A method which tries to automatize the wrapper maintenance process is SGWRAM. It is based on the definition of XML schemas during wrapper generation phase. SGWRAM is buit on top of following assumptions:

- Syntactic features such as data patterns or string lengths are preserved.
- Hyperlinks are rarely removed in Web page modifications.
- Annotations : descriptive information representing the semantic meaning of a piece of information in its context is usually maintained.[FMFB14]

Based on those assumptions the system was developed which create schemas during the wrapper generation phase which will be used in wrapper maintenance. During the wrapper generation the user provides HTML documents and XML schemas which are mapped. Then the system generates extraction rules and then wrapper is executed. The XML document is built according to a defined schema. Additional process is introduced which is known as *wrapper maintainer* which checks if wrappers have any errors in Web data extraction process and it provides an automatic procedure for wrappers which have errors which arises from structural modifications of a Web page. If this procedure has successfully succeeded then extraction process will continue and if there is a opposite case then it will issue a warning that there are errors. DTD is used to define XML schemas and HTML documents are represented as DOM trees. SGWRAM system builds mapping between XML schemas and HTML documents and generates extraction rules.

*Automatic wrapper adaptation* is another method for automatic maintenance of wrappers. This method is based on comparison of an original Web page structure with the modified one. Elements are identified and represented as sub-trees of the DOM tree and they can be used to find similarities between two structures of the same page. The *candidates* are elements which have higher degree in similarity. The algorithm which does this matching among the DOM trees of the HTML documents is *weighted tree matching*. This approach can be extended to show that it is possible to identify multiple elements sharing a similar structure with a treshold of similarity. Lixto commercial tool has a method which allows wrappers to automatically detect and change their functionality due to structural changes on a Web page. *Signatures* are used which represents a DOM sub-tree of elements from the original Web page. If there are any issues in the data extraction process, the algorithm is executed in order to adapt the wrapper to the new structure.

On the Web pages of Universities, most of the faculty members have their own list of publications which we assume that they show their expertise in a particular area. Some of the publications are not available on the University Web page, so it will be necessary to browse the digital libraries in order to get a complete list of publications for a particular author. Publications data will be mapped with concepts in an ontology in order to get technical/professional competences of faculty members.

Competency management can be seen as one of the foundations of learning activities in knowledge intensive organizations. As a critical point in the functioning of knowledge management, competencies require a representational framework that is rich enough to support effective and efficient processes of competency search, matching and analysis.[Sic06]

## 1.2 Problem statement and objectives

Web sites differ in a large manner by their structure. All Websites look pretty similar on the presentation layer although their structures are different. An ideal form would be if all pages were built up on similar structure and that the data they have is nearly same structured. This is not the case in the real world, and this is on purpose. Many

Web administrators are trying to get their job right and make the data extraction process as hard as possible for crawlers and data extractors. This goes from simple data rearrangements to more complex ajax requests which try to hide the dynamic behavior of the site. The most common challenge that crawlers encounter is the arrangement of data in the sites, when the whole data is written as plain text without any classifiers of what each text section represents. Unfortunately there is no other option then heavily adjusting the crawler for breaking down such data step by step. The structure which can cause problems to Web data extraction software is when there is a need to link some data which is divided into separate paragraphs. Most likely, in this case Web data extraction software will need human intervention in order to set it up to link this data.

The outcome of this thesis is to have an implemented software which will extract list of faculty members from various Universities and their publications data from ACM and IEEE Xplore digital libraries since they contain publications in Informatics field.

The goal is to have a one for all approach in the sense that this software can extract data from many Universities and if it is possible that it works for all Universities. The other goal also is to have all this as an automatic process as much as it is possible, to have a minor user input.

The extracted publications data of faculty staff will be used to get their technical/professional competences. In order to determine their professional/technical competences, the ontology with technical competence concepts(example: java programming, network security, databases, web development, etc..) will run in background and the publications data will be mapped to those concepts in an ontology in order to determine their technical/professional competences. So, for example if they publish ten papers in some technical field, we can assume that they have technical competence in that field. The technical competences will have its strength and trust which will be measured in some way which will be determined during the work on the thesis.

The ontology which will be used is an existing ontology in [Pic08] which was extended in [Hoc12] which will be extended to cover domain of the use case.

## 1.3 Methodological approach

The Methodology consists of:

- **Search and Analysis of Literature**

  Literature, which provides a profound information in the area of Web data extraction.

- **Designing a Software for Web data extraction**

  The software is designed based on requirements to have it as much as an automatic process and that it works for majority of Universities(if possible for all). In order to have a software which will extract data from various Web pages, the issues of unstructured data, dynamic pages, data without classifiers, etc... need to be taken into consideration. In that sense, existing Web data extraction tools are analyzed and compared in order to see how they deal with mentioned issues in the area of Web data extraction.

The expected results of this research and analysis of these tools are used to get some ideas on how to design and implement a software which will deal with all the mentioned issues because this software should be able to extract data from various Web pages which could have a complex structure.

This software will need to have some inputs like Cascading Style Sheet (CSS) selectors, XPaths or other methods for locating some elements on the page(location of staff members, publication data) and to have a possibility to load an Ontology from some source like a file or a server. Use case for this software will be some specific Institute of the Faculty of Informatics at the Vienna University of Technology. In general, this software will target Faculties of Informatics.

- **Extending an existing Ontology with domain of an use case**

  Ontology which is used for this software is an existing ontology in [Pic08] which is extended in [Hoc12] and it is extended with domain of an use case. That is done based on a research on what are the institute research areas from the use case. For example, if that institute is dealing with security then probably some of the nodes in an ontology will be application security, network security, data security etc... Also, those nodes have subnodes which are linked to its parent nodes. (Example: Data security(parent), Cryptography(child)).

  Ontology is used for mapping of publications data with technical competence concepts in an ontology in order to determine technical competences of faculty staff.

  Ontology is extended using Web Ontology Language (OWL).

- **Implementation of a Software**

  Technologies which are used for implementing a software are Selenium, CSS selector (for data navigation), Java Script, .NET/C#, Web Services,HTML, SPARQL queries.

  Selenium enables browser automation from Java, it acts as a Web browser out of java code and gives the possibility to read and manipulate data from Websites.

- **Evaluation of Results**

  The technical competences which are provided by implemented software are evaluated in order to check if they are correct. In order to do evaluation the questionnarie/survery is carried out among University staff from University institute from the use case in order to check if the technical competences provided by the software match with technical competences of staff members which are provided from survey. The question in the survey is in the form of bullet list where participants are asked to select which technical competences and subcompetences they believe to have.

The technical competences are extracted for around fifty staff members of the University institute which is a use case. Those staff members are from Institute of Computer Aided Automation at the Faculty of Informatics which is an use case and it is not limited to some particular areas in Informatics. They could be from institute of automation, institute of information systems, etc...

Ontology for the use case is an existing ontology in [Pic08] and [Hoc12] which is extended to cover the domain of the use case. Ontology is used as an input to the program in order to have a possibility to extend it to some other domains.

## 1.4 Outline

The thesis has the following structure:

Chapter 2 presents an existing work in the area of Web data extraction. Some state-of-the-art approaches are presented and analyzed. Existing Web data extraction tools and their fucntionality is presented.

Chapter 3 discusses how to deal with Web data extraction issues and how to successfully extract data from Web pages.

Chapter 4 shows an use case which is used for the Web data extraction software.

Chapter 5 lists and describes used technologies for implementing a software. It is described how it is done and what were the issues.

Chapter 6 presents a survey results which show what are the professional competencies of the use case University staff members and compares them with the professional competencies which are obtained from implemented software.

Chapter 7 summarize the work done in this thesis and presents some ideas what could be a future work in this field.

# CHAPTER 2

# Existing Work

Nowadays, there are a lot of commercial Web data extraction tools and and mostly their functionality is similar. Some tools provide more functionality than the others but the core problem remains which is the structure of various Web pages. Most of the tools can detect already common structures and extract data efficiently but the problem arise when there are some unordinary cases like page sections not properly marked, text sections not classified, dynamic data on a page generated in a complex way.

The Web is constantly evolving. Web pages use many technologies to present, modify and load content on the page. CSS are used for separation of page style from its content, Asynchronous JavaScript Requests are used for dynamic load of the page content, client side scripts which are used for modification of a page for a client. Web browsers are becoming more and more complex because they constantly add a lot of new features and with those additions they use much computation power to render Web pages. Modern pages use AJAX requests to load a content. For example, due to performance issues some pages use AJAX pagination in the case when there is a "next" link. A Web browser will not load a whole page when "next" link is clicked, it will load just asynchronously received data from Web server. [GT13]

Cookies stored in browsers are very useful technology for page navigation and also data can be stored in cookies which could be very useful for data extraction software. If some data extraction software opens one page and extracts some data which is needed for data extraction on another page, that data could be stored in a cookie and during the run of a program it could be retrieved from a cookie and used on another page.

It is a very hard task for modern data extraction tools to fully emulate behavior of a modern Web browser in order to present and extract data accuratelly because of mentioned new features of which Web pages are composed of. The Deep Web page navigation provides a new challenge for data extraction. Data extraction software needs to execute queries in order to retrieve information behind the Web forms which means that it is becoming very difficult to code the data extraction software. [GT13]

Web pages are differently designed and they have a different structures. Some pages have a similar structure but that is not a case for all Web pages. If we want to extract data from many Web pages we need to have that as an automatic process because it is not practical and it is inefficient to have many programs for each site. The field of automatic Web data extraction is still under active research. Web data extraction systems should be able to extract data from many sources and to automatically adapt to Website changes. [GT13]

## 2.1 Web data extraction system

Web data extraction system is a software system that automatically and repeatedly extracts data from Web pages with changing content and delivers the extracted data to a database or some other application. [GT13]

Web data extraction tasks can be divided into five phases:

1. Website interaction, which is a navigation to pages which contatin desired data;
2. wrapper generation and execution;
3. scheduling, which allows repeating data extracting tasks by constantly revisiting target Web pages;
4. data transformation, which includes filtering, transforming, refining, and integrating data extracted from one or more sources and structuring the result according to a desired output format
5. data provision, which is delivering the extracted structured data to external applications such as database management systems, data warehouses, business intelligence systems, decision support systems, etc . . . [GT13]

Figure 2.1 shows a typical architecture of the Web data extraction system. The wrapper generator is composed of visual interface and program generator. Visual interface is used for providing a rendered Web pages to the user, where user marks data which needs to be extracted. Program generator generates data extraction rules. Wrapper executor runs previously generated wrappers from repository. The data transformation and integration unit cleans, combines, transforms and integrates the extracted data. The data delivery unit delivers data via appropriate channels such as FTP, HTTP, E-mail, etc. Data delivery unit delivers data to the target application. [GT13]

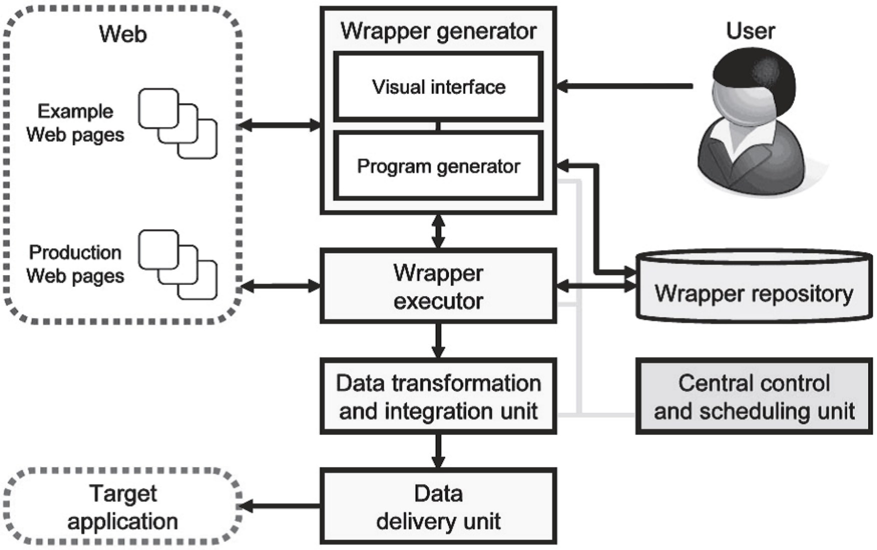


Figure 2.1: Architecture of a typical state-of-the-art Web data extraction system [GT13]

## 2.2 Web data extraction tools

There are many Web data extraction systems available today as commercial tools but there are also some open-source tools. The users of those tools are mostly experts but there are some non-expert users interested in some information.

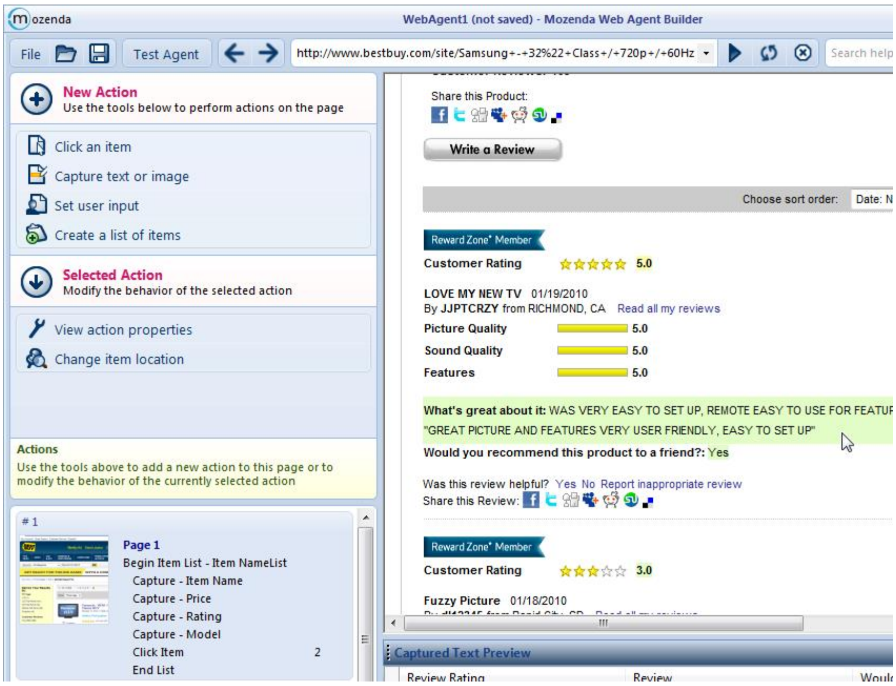
In the scientific literature there are numerous approaches for Web data extraction but they are not yet fully implemented. One of the most prominent examples of systems coming from the academic research field is Lixto. Lixto is a typical visually aided state-of-the-art Web data extraction system in which the user is asked to simply visually select the data that should be extracted. Usually, no programming knowledge is required. In the Lixto Visual Developer software, wrappers are created in an entirely visual and interactive fashion [GT13]

In the Lixto user interface, users can see recorded navigation steps when they browse some website. Users are also provided with extraction configuration tab where they set the rules for data extraction.

Some commercial tools like Mozenda, Visual Web Ripper, Lixto provide a good functionality and they are user oriented.

Mozenda is a practical tool for basic users. This tool has a nice interactive user interface and a powerful browser from which data is selected for extraction process. With Mozenda it is possible to make scheduled extractions and it also provides several data output formats. By using this tool it is also easy to create a pager for next link and to capture a list of items. Form filling is also done nicely, it is possible to provide a list of various inputs for some forms to complete and proceed the extraction according to these results. AJAX requests are fairly good handled with a sufficient timeout delay.

The output formats of the extracted data may be CSV, XML or it can deliver the results via email or to a FTP server. Iterations and loop operations work very well in Mozenda, since the framework keeps track of each predefined steps, and by some missing values it is possible to make the data field optional so it does not crash because of some missing value. The application seems very robust and due to the ability to specify optional data fields it is pretty adaptive. Regarding performance, it's greatest advantage is that it uses Mozenda cloud framework and its computing power for doing the extraction, so its possible to make scheduled extractions, and then just to download the data. There are some difficulties while extracting data with Mozenda tool. Using Mozenda only entire table row could be extracted because there is no option to separate data in a row from other data. If it is a case that for example name of some person, address, location, e-mail was written in the same table row and we want to extract only e-mail, it will not work with Mozenda tool. The other problem with Mozenda tool is that if the data is organized in separate paragraphs and we want to connect data to a single field it is not possible.[Moz15b]



http://www.bestbuy.com/site/Samsung+-+32%22+Class+/+720p+/+60Hz

Use the tools below to perform actions on the page

Modify the behavior of the selected action

Use the tools above to add a new action to this page or to modify the behavior of the currently selected action

# 1

Page 1
Begin Item List - Item NameList
Capture - Item Name
Capture - Price
Capture - Rating
Capture - Model
Click Item 2
End List

Customer Rating 5.0

LOVE MY NEW TV 01/19/2010
By JJPTCRZY from RICHMOND, CA Read all my reviews
Picture Quality 5.0
Sound Quality 5.0
Features 5.0

What's great about it: WAS VERY EASY TO SET UP, REMOTE EASY TO USE FOR FEATUF
"GREAT PICTURE AND FEATURES VERY USER FRIENDLY, EASY TO SET UP"
Would you recommend this product to a friend?: Yes

Was this review helpful? Yes No Report inappropriate review
Share this Review:

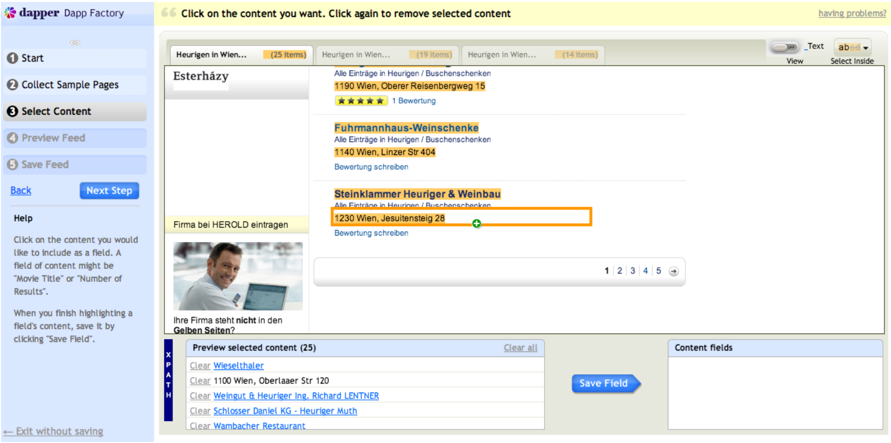
Customer Rating 3.0

Fuzzy Picture 01/18/2010

Figure 2.2: Mozenda User Interface [Moz15a]

Dapper tool is also easy to use like Mozenda with Web based interface. Dapper is useful for processing unstructured information from HTML.It can produce a basic XML feed from any page and can produce CSV, RSS and other formats. Dapper tool works well with Yahoo pipes which is a easy to use tool for data integration but pretty unstable. To integrate the data from Dapper the pipes make use of the provided Fetch Data module, which is capable of fetching XML, JSON or similar formats from a given

Web source. Also the pipes are making use of a string builder functionality to build more personalized search terms, so that it is possible to give different search term for every Dapper. In the end the results from different Dappers are merged into one structure using the union module which is also provided in Yahoo pipes. As a final result the data can be represented as a RSS feed or JSON.



Click on the content you want. Click again to remove selected content

Click on the content you would like to include as a field. A field of content might be "Movie Title" or "Number of Results".

When you finish highlighting a field's content, save it by clicking "Save Field".

Esterházy

Firma bei HEROLD eintragen

Ihre Firma steht **nicht** in den **Gelben Seiten**?

Alle Einträge in Heurigen / Buschenschenken
1190 Wien, Oberer Reisenbergweg 15
★★★★★ 1 Bewertung

Fuhrmannhaus-Weinschenke
Alle Einträge in Heurigen / Buschenschenken
1140 Wien, Linzer Str 404
Bewertung schreiben

Steinklammer Heuriger & Weinbau
Alle Einträge in Heurigen / Buschenschenken
1230 Wien, Jesuitensteig 28
Bewertung schreiben

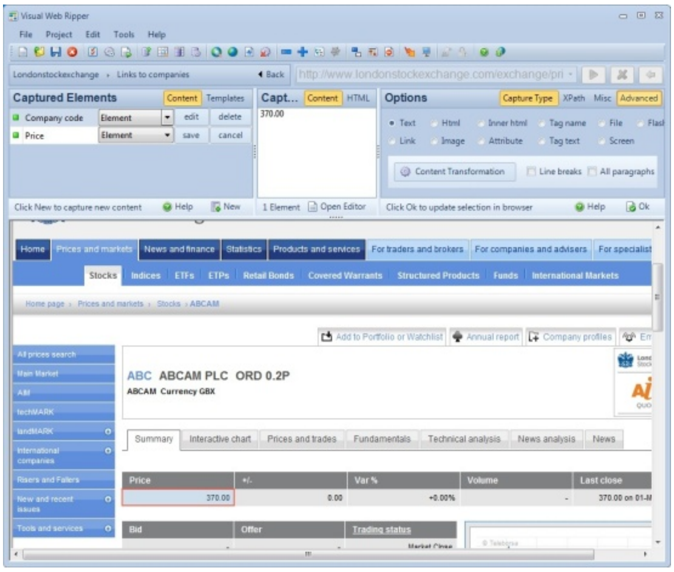
Preview selected content (25)
Clear Wieselthaler
Clear 1100 Wien, Oberlaaer Str 120
Clear Weingut & Heuriger Ing. Richard LENTNER
Clear Schlosser Daniel KG - Heuriger Muth
Clear Wambacher Restaurant

Figure 2.3: Dapper User Interface

Visual Web Ripper is an excellent tool for automated web scraping. This tool extracts complete data structures, such as product catalogues. If needed Visual Web Ripper may repeatedly submit forms for all possible input values which is important for a multiple search.[Vis15] This tool is useful for tough to scrape Websites because it supports many features like AJAX, it provides feature to submit forms of various input values, it can bypass CAPTCHAs, connecting to websites with random time delay, hiding IP-address by using proxy. It supports various data export formats like CSV, Excel, XML, SQL Server, MySQL, SQLite, Oracle as it is shown in Figure 2.5.

Web Harvest is a tool written in Java. It offers a way to collect desired Web pages and extract useful data from them. In order to do that, it leverages well established techniques and technologies for text/xml manipulation such as XSLT, XQuery and Regular Expressions. Web Harvest mainly focuses on HTML/XML based Websites which still make vast majority of the Web content. On the other hand, it could be easily supplemented by custom Java libraries in order to augment its extraction capabilities.[Web15] Web Harvest uses XML configuration files to describe how to scrape a site and with a few lines of Java code you can run any XML configuration and have access to any properties that the script identified from the page. This is definitely the safest way to scrape data, as it decouples the code from the web page markup, so if the site you are scraping goes through a redesign, you can quickly adjust the config files without recompiling the code they pass data to.[Web10]

In the Web Harvest tool most of the functionality need to be coded from scratch. It is easy integrated and executed with Java. The core of the Web Harvest is its configuration


Visual Web Ripper
File Project Edit Tools Help
Londonstockexchange › Links to companies
Back
http://www.londonstockexchange.com/exchange/pr
Captured Elements
Content Templates
Capt...
Content HTML
Options
Capture Type XPath Misc Advanced
Text Html Inner html Tag name File Flasl
Link Image Attribute Tag text Screen
Content Transformation
Line breaks All paragraphs
Click New to capture new content
Help New
1 Element Open Editor
Click Ok to update selection in browser
Help Ok
Home Prices and markets News and finance Statistics Products and services For traders and brokers For companies and advisers For specialist
Stocks Indices ETFs ETPs Retail Bonds Covered Warrants Structured Products Funds International Markets
Home page › Prices and markets › Stocks › ABCAM
Add to Portfolio or Watchlist Annual report Company profiles
All prices search
Main Market
AIM
techMARK
landMARK
International companies
Risers and Fallers
New and recent issues
Tools and services
ABC ABCAM PLC ORD 0.2P
ABCAM Currency GBX
Summary Interactive chart Prices and trades Fundamentals Technical analysis News analysis News
Price +/- Var % Volume Last close
Bid Offer Trading status

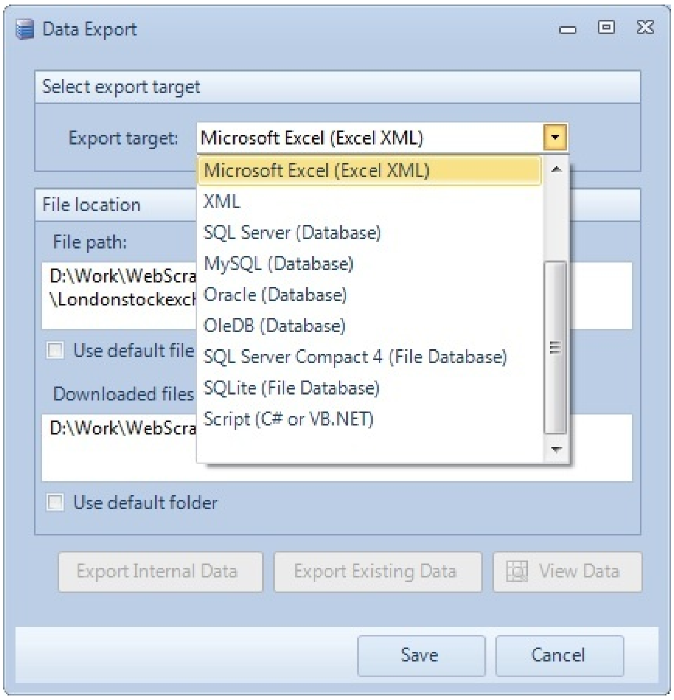
Company code Element

Price Element

370.00

| Price | +/- | Var % | Volume | Last close |
|---|---|---|---|---|
| 370.00 | 0.00 | +0.00% | - | 370.00 on 01-M |

Figure 2.4: Visual Web Ripper User Interface [Vis15]



Export target: Microsoft Excel (Excel XML)

File path:

D:\Work\WebScra
\Londonstockexcl

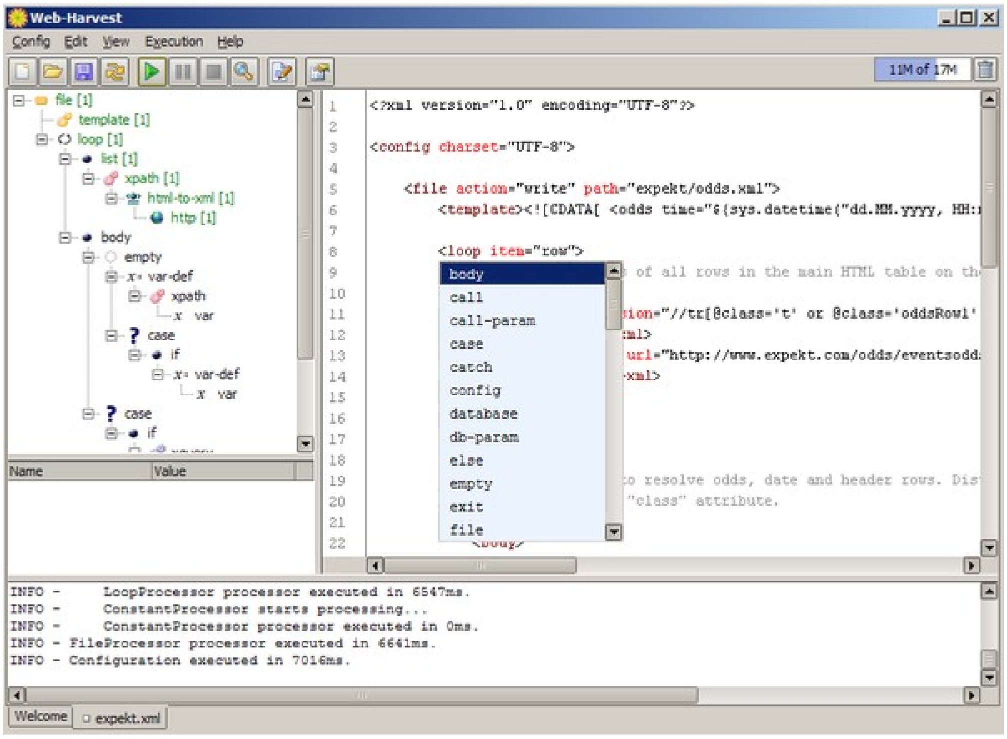
□ Use default file

Downloaded files

D:\Work\WebScra

□ Use default folder

Figure 2.5: Visual Web Ripper Output Formats [Vis15]



```
<?xml version="1.0" encoding="UTF-8"?>

<config charset="UTF-8">

    <file action="write" path="expekt/odds.xml">
        <template><![CDATA[ <odds time="${sys.datetime("dd.MM.yyyy, HH:

        <loop item="row">
                                 of all rows in the main HTML table on the

                                sion="//tr[@class='t' or @class='oddsRow1'
                                ml>
                                url="http://www.expekt.com/odds/eventsodd
                                -xml>




                                to resolve odds, date and header rows. Dis
                                "class" attribute.


```

```
INFO -     LoopProcessor processor executed in 6547ms.
INFO -     ConstantProcessor starts processing...
INFO -     ConstantProcessor processor executed in 0ms.
INFO - FileProcessor processor executed in 6641ms.
INFO - Configuration executed in 7016ms.
```

Figure 2.6: Web harvest tool [Web15]

file in xml. One of the disadvantages for this tool is when you extract content from Website with complex data structure the xml file becomes unreadable because the file expands quickly due to lot of coding. The filtering of the resulting data is a complex task, also adding context variables from Java code does not work. Web Harvest is an open source tool and actual version is beta which contains a bunch of bugs and the last version of the tool is pretty old having in mind actual growing property of technology. Although it contains bugs this tool is useful for some simple Websites with simple structures.

Kimono is a tool which takes unstructured data and it converts it into a structured format. It is coded in a bookmarklet form which you add to your browser. With Kimono you can extract data on demand or as a scheduled job. The output format can be in the form of a file or what is a more useful feature is to have it as an API. That API can be invoked in some program in order to get extracted data. In order to call an API you need to have API key which you get with Kimono account. That key is used for tracking user activity.[Kim15]

Kimono is coded in a bookmarklet form which you just add to your browser. The user interface is shown in Figure 2.7

When u run the bookmarklet you then select the individual elements you want to extract. This is done simply by clicking on the desired elements on the page. Kimono then tries to identify similar elements on the page; it will highlight some suggested ones and you can confirm a suggestion with a checkmark. When u are done with selection of data for extraction u can list all your data when u click on Data button in the bookmarklet. When u click on Model button all fields are displayed. As a final step of the extraction process u create an API and u define how often will data be extracted and u limit the number of pages for the extraction.

The great feature which Kimono offers is that it actually converts Websites into APIs. You make an API once and when u need to extract data from that Website u just call an API in your program.

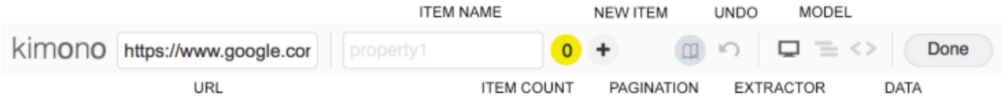
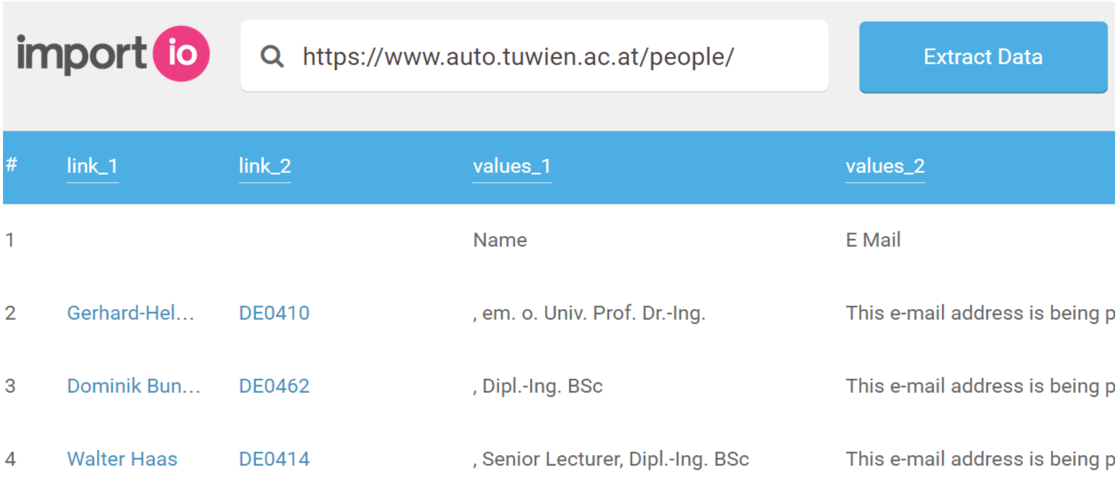


Figure 2.7: Kimono User Interface [Kim15]

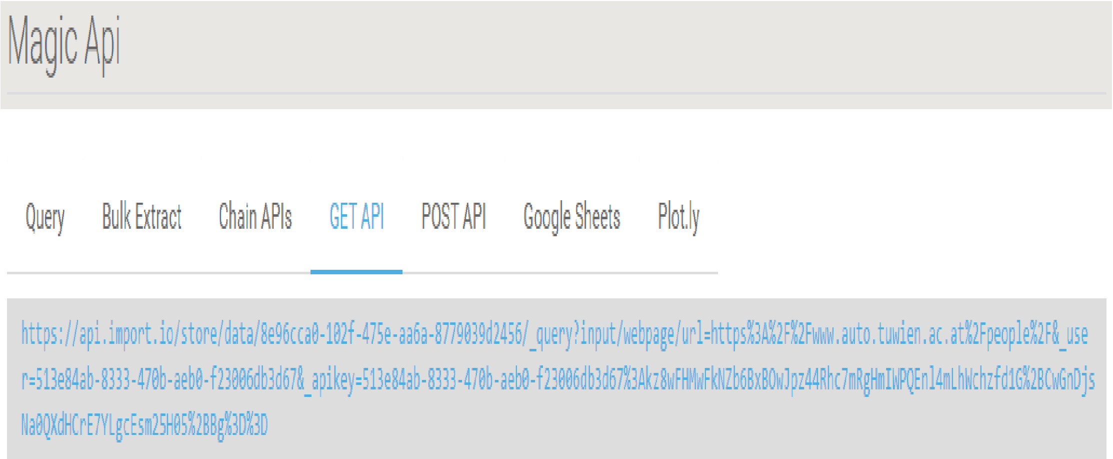
Import IO is a data extraction tool which allows most of the users to get structured data from Websites because of its ease of use. This tool allows to structure the data into rows and columns. Collected data is stored in the cloud. As this tool creates an API it is simple to get the data and to integrate live data into applications.[Imp15]

The output format can be in a form of a csv file which is very straightforward to obtain and the other output format can be in the form of an API. This tool has its own browser where u enter url of a page from which u want to extract the data. Then the browser opens a page and puts data into rows and columns in order to have a structured data. You can export that data into csv file or you can create an API which u can query by making a GET or POST requests. GET request is shown in Figure 2.9. This request url is composed of an API key which you get with your account. This is a very nice feature because u can get a data in a program from an API. The drawback for this API is that it only accepts one parameter which is an url of the page from which u want to extract the data. In this setting it extracts the data from the whole page which could be performance costly in some cases.



| # | link_1 | link_2 | values_1 | values_2 |
|---|---|---|---|---|
| 1 | | | Name | E Mail |
| 2 | Gerhard-Hel… | DE0410 | , em. o. Univ. Prof. Dr.-Ing. | This e-mail address is being p |
| 3 | Dominik Bun… | DE0462 | , Dipl.-Ing. BSc | This e-mail address is being p |
| 4 | Walter Haas | DE0414 | , Senior Lecturer, Dipl.-Ing. BSc | This e-mail address is being p |

Figure 2.8: ImportIO User Interface



```
https://api.import.io/store/data/8e96cca0-102f-475e-aa6a-8779039d2456/_query?input/webpage/url=https%3A%2F%2Fwww.auto.tuwien.ac.at%2Fpeople%2F&_use
r=513e84ab-8333-470b-aeb0-f23006db3d67&_apikey=513e84ab-8333-470b-aeb0-f23006db3d67%3Akz8wFHMwFkNZb6BxBOwJpz44Rhc7mRgHmIWPQEnl4mLhWchzfd1G%2BCwGnDjs
Na0QXdHCrE7YLgcEsm25H05%2BBg%3D%3D
```

Figure 2.9: ImportIO GET Request

Table 2.1: Commercial data extraction tools

| Tool | Price | Details |
|---|---|---|
| Mozenda | Individual account:$99 per month<br>Professional account: $199 per month or $1,990 per year | Process up to 25,000 pages per month<br>1 GB of storage<br>1 User<br>Up to 10 agents<br>Email & Phone Support<br>Free online training |
| Visual Web Ripper | Single user license: $349<br>2 Seat License: $558<br>5 and 10 Seat License: $1395 and $2090 | User friendly visual project designer<br>Extract complete data structures such as product catalogues<br>Repeatedly submit forms for all possible input values<br>Extract data from highly dynamic websites including AJAX websites<br>Web data extraction scheduler with e-mail notifications and logging<br>Custom post-processing and comprehensive API<br>With single user license 6 months maintenance is included |
| Kimono | N/A | Free edition : fetch 20,000,000 pages, support: none<br>Pro edition: fetch 100,000,000 pages support: 24hr<br>Enterprise edition: custom page fetching, support: custom<br>Unlimited public APIs for all editions, Private APIs related to user account in Pro and enterprise edition |
| Import IO | Free | Data storage in cloudCreate APIs for latter use in other applications<br>Free desktop application |

Table 2.1 lists some available Web data extraction tools, their prices and what they offer.

## 2.3 Automation in Web data extraction

Commercial Web data extraction systems are able to extract data from limited number of Websites. Those tools offers nice user interfaces where most of the users can easily add some additional sites for data extraction process. But this process still needs human interaction and nowadays there is a need to automate the whole process. Currently there is no such fully automatic system available but there is some research in this field.

One of the most promising projects is a DIADEM project which is developed at Oxford University. This project tends to automatically extract Web data on a large scale. This project has a chance to become one of the next generation Web data extraction systems by integrating currently available technology with reasoning using expert knowledge. Without human involvement this system locate and analyze Websites of some domain and with automatically generated wrappers data is extracted. Websites are parameterized by using the domain knowledge what enables the system to function without human involvement. Currently, Web data extraction tools require human involvement because they select Web pages and record navigation steps for extraction process. DIADEM project is limited to Websites written in English language. That is a limitation for this project because there are many users which are interested to extract data from non-English Websites. Also, the domain knowledge creation process could be simplified in the future. [GT13]

One research which is done at Yahoo deals with Web-scale information extraction. For any Website which contains semi-structured indormation about a set of schemas, this approach shows how to populate objects in schema by automatically extracting information from Website. This approach is *domain-centric* which means that there is a human involvement where humans defines some rules regarding the domain. Researchers working on this project believe that this *domain-centric* approach could be very important principle in the area of Web-scale information extraction. [GT13]

The challenges in the field of Web-scale extraction remains even with this approach. There are some aspects which help in solving these issues like discovering and identifying websites that contatin information of interest, analyzing those Websites to get cluster of pages to extract from, automatic learning of extraction rules, data linking. [BDF+12]

In some cases design and implementation of Web data extraction systems is based on scientific methods related to disciplines like Logic, Natural Language Processing, Machine Learning, etc ... For the design of those systems some factors needs to be taken into consideration where some of them are independent of the domain in which Web data extraction is performed and some of them rely on domain which implies that some technological solutions which are effective in some context could not effective in others. [FMFB14]

The key challenges in the design of Web data extraction systems could be described as:

- The first challenge is to provide a *high degree of automation* to reduce human involvement. On the other hand, human feedback can contribute to accuracy achieved by Web data extraction system. There is a trade-off between a need for highly automated systems and accuracy of those systems.

- The second challenge is to have a system which will extract a large amount of data in short time. This challenge is closely related to *Competitive Intelligence* because in that field the data is required for analysis of competitors activity in markets.

- Personal data also needs to be taken into consideration. Some potential, unintentional attempts to violate user privacy need to be identified.

- In some cases Web data source can evolve over time. Some Websites can change their structure in the future so there is a need to maintain a Web data extraction system to deal with that issue if that system cannot automatically adapt to changes. [FMFB14]

Current automatic wrappers do not use semantic properties of data records in their design. It was a case before that automatic wrappers extracted data records by checking patterns using the DOM structure. MDR checks HTML tags in order to locate data records. When we compare those wrappers with the ones which are visually assisted we get that they are not very accurate. The state-of-the-art automatic wrappers are visually assisted and they are more reliable than wrappers which relies on HTML tags. [Hon11]

ViNT is a tool for automatically producing the wrappers for any given search engine. ViNT uses content lines for data extraction. A group of content lines may form a content block, which constitutes a data record. [ZMW$^+$05]

ViPER is a fully automatic tool for information extraction. It is based on an assumption that a Web page containts at least two consecutive data records which builds a data region which have some structural and visible similarity. ViPER is able to extract and discriminate the relevance of different repetitive information contents with respect to users visual perception of a Web page. [SL05]

## 2.4 Research in Ontology Area

In this subsection some information about ontologies will be presented since this thesis aims to extract University staff competencies based on an ontology.

We can think about ontologies as a conceptual schema in database systems. A conceptual schema provides a logical description of shared data and it defines relations on data. On the other side, an ontology contains terms which are used to represent knowledge. An ontology defines a vocabulary used to compose complex expressions. [Gru93]

There are already proposed formal ontologies for management of competencies. Nonetheless, more work is required in the clarification of the concept of competency and also in providing integrative schemas for competencies.[Sic06]

From the scientific papers it can be realized that competences are mainly structured in hierarchies or are at least based on hierarchies. In general, the proper size of an ontology depends on its purpose. As long as the competencies can be determined we can assume that given ontology has a sufficient amount of concepts and relations. For the ontology with competence concepts it seems that when there are more concepts defined the more accurate user expertise can be described. An ontology which represents a given domain on a trivial level is built quickly and it is easy to handle but the downside is that it may not be able to provide enough concepts to gain a meaningful statement describing competence of someone. [Hoc12]

There are also wrappers which use ontology domain. Most of those wrappers are domain specific except ODE wrapper which is fully automatic which means that all data extraction steps can be performed automatically without manual labeling and human interaction. ODE workflow is show on Figure 2.10 and discussed in more detail in [SWL09].

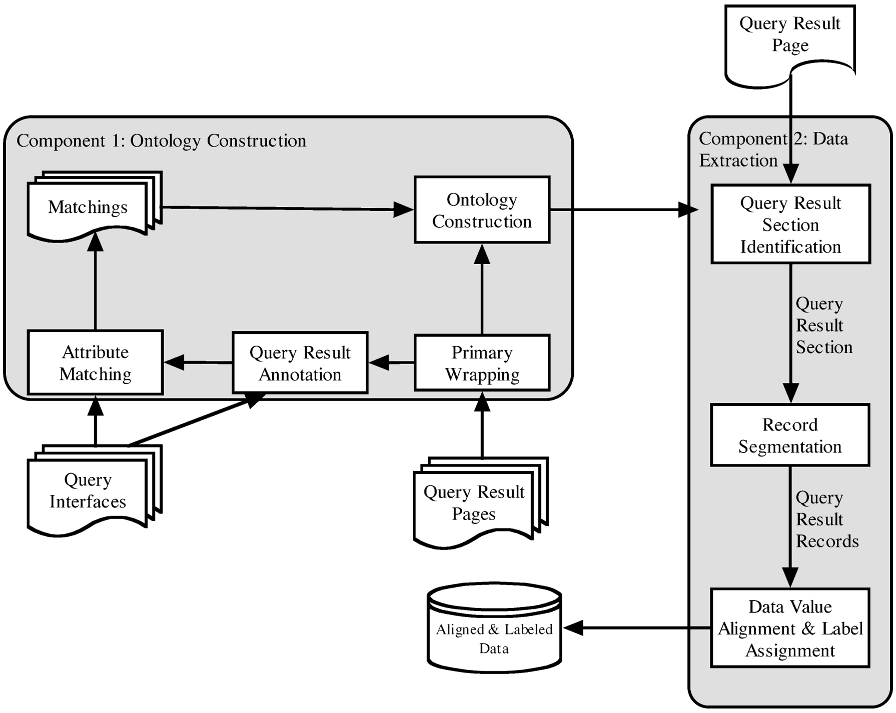


Figure 2.10: Ontology-assisted Data Extraction (ODE) workflow [SWL09]

Ontology assisted wrappers were proposed in [?]. They describe data relationships, lexical appearance and keywords. Those wrappers parses the predefiend ontology and they construct a database schema for extracting data from unstructured sources. [?]

A wrapper was proposed in [SMN06] which extracts data in three phases:

1. Parse and convert a HTML page into XML format
2. Use ontology to construct a data model
3. Provide mapping between XML elements and an ontology
4. Use ontology to construct a data model

Research in ontology area aims to reduce the effort and time for building an ontology. The goal is to semi-automatically or automatically build ontologies from some information like documents or Web pages with limited human involvement. [HER09] The work done in [HER09] presents a system to automate a process for creating an ontology using semi-structured domain specific Web documents. This system takes a set of concepts which represent the main concepts in a domain and a set of documents from which the ontology should be formed as an input. Then the system extracts HTML headings from input source, analyzes their structure, extracts concepts from them and builds in the end a taxonomical ontology. The system is shown in Figure 2.11.

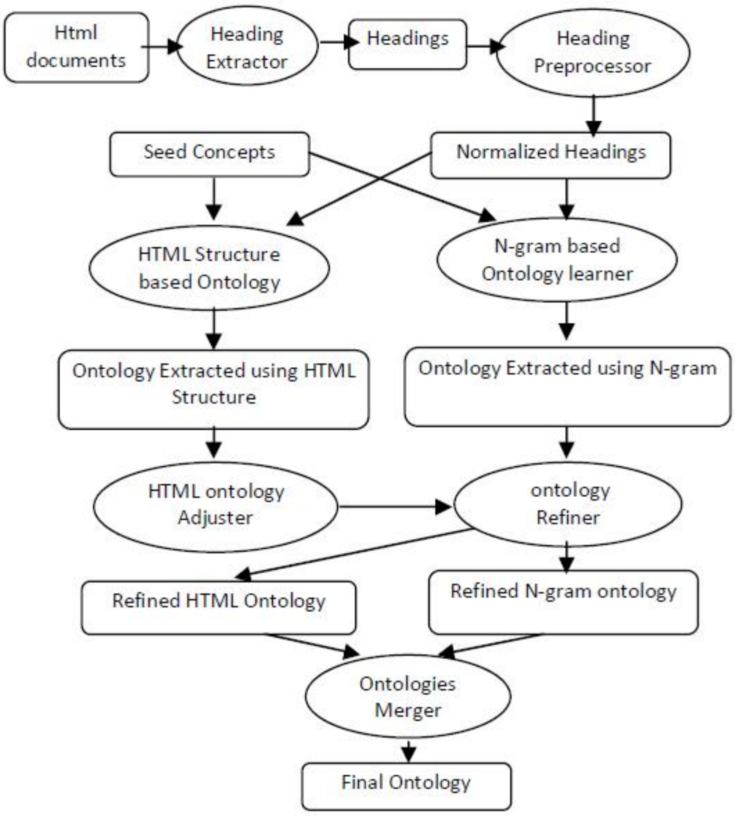


Figure 2.11: The ontology learning process [HER09]

## 2.5 Competence models

The important method for identifying competences is to construct a competence model. Competence model is the combination of different factors in order to achieve a certain competence structure. Competence model could be compared to a iceberg floating in the water where the ice parts presents a basic competence including knowledge and skills and the part under the ice which presents some social skills. [hZW11]

According to Dr McClelland's study the competence model can be constructed by using the steps shown in Figure 2.12.

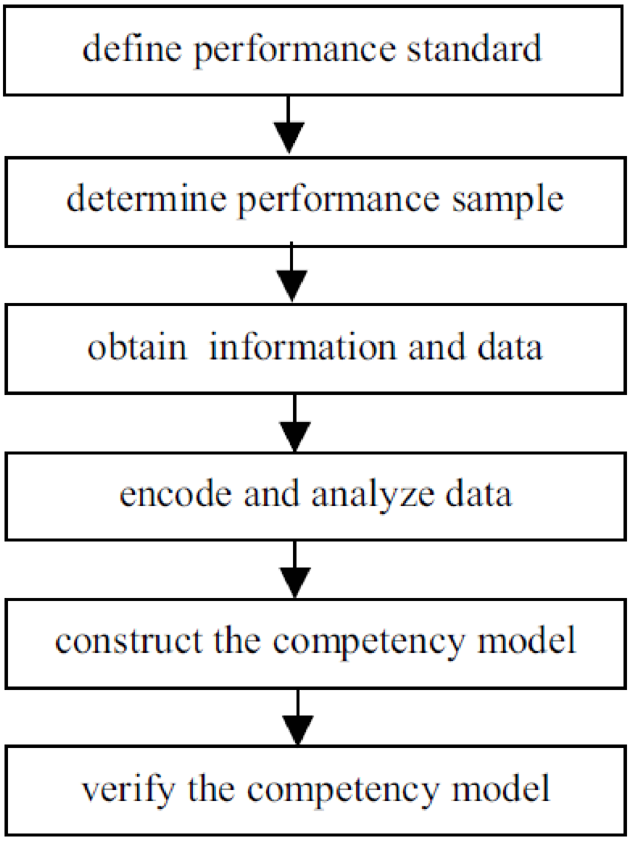

define performance standard
determine performance sample
obtain information and data
encode and analyze data
construct the competency model
verify the competency model

Figure 2.12: The construction steps of the competency model [hZW11]

Due to the requirements of today's economy, learning must be adapted to fulfill just-in-time and just-enough learning needs. One way is adoption of competence-based learning approach. This approach enables to exploit competence models for instructional units to help the learner master targeted competencies. Competence-based learning models constitute a meaningful structure for just-in-time and just-enough learning. [ZNF07]

In [ZNF07] the ontology-based competence model is proposed that allows the on-the-fly generation of Learning Knowledge Objects. This process relies on knowledge objects and ontologies created through text mining and natural language processing. In this model, a competence is a set of skills linked to domain concepts. The skilsl are declared according to the Bloom Taxonomy, a model which describes six levels of knowledge mastery. Each level is related to a set of verbs that defines the skills that can be mastered at this level.

Text mining deals with finding particular knowledge in learning objects through document parsing. It is applied on learning objects and domain documents to extract a domain ontology which is linked to the competence model through semantic concepts. The domain ontology takes the form of a concept map composed of semantic concepts and their relationships. Maching learning and linguistic analysis methods are used to determine domain concepts used in the competence-based model. Those methods help identify siginificant semistructured information such as key phrases from documents. The resulting concept maps are stored in an OWL ontology that is used to index the learning objects and to support the competence model. The importance of modeling the relation between competencies and instructional roles through semantic rules instead of modeling it in the ontology relies on the ability to change them. A designer may decide that defining a concept requires an introduction, a definition and an example about the concept whereas another one may be satisfied with a sole definition. [ZNF07] . The competence-based ontological model is shown on Figure 2.13.

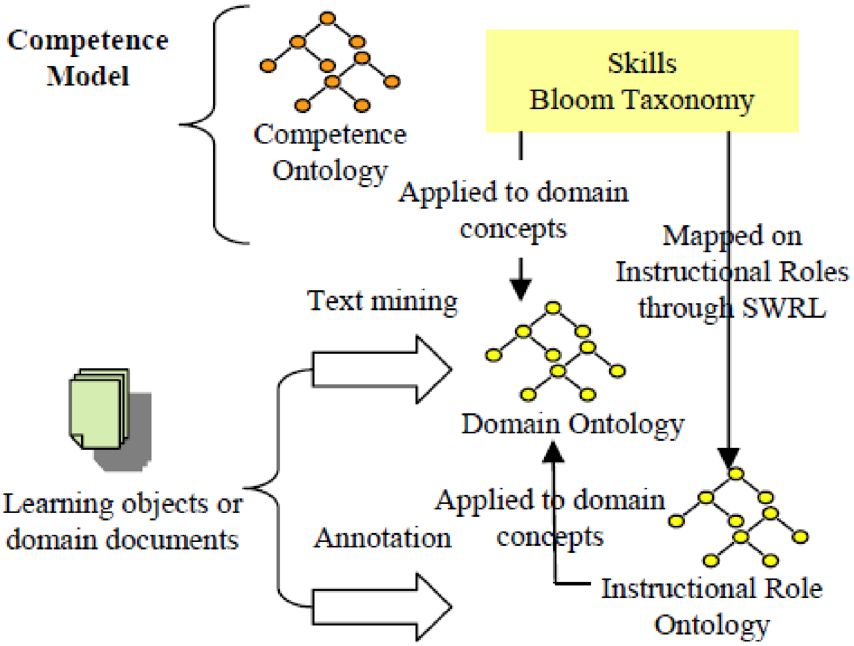

Competence Model
Competence Ontology
Skills
Bloom Taxonomy
Applied to domain concepts
Mapped on Instructional Roles through SWRL
Text mining
Domain Ontology
Learning objects or domain documents
Applied to domain concepts
Annotation
Instructional Role Ontology

Figure 2.13: The competence-based ontological model[ZNF07]

The paper [TSH06] presents the use of ontologies to formally represent competence models describing individual and enterprise competences. The competence models are defined with a meta-model, the latter being an ontology language. Competence profiles are instances of models and they describe individuals and enterprises. This approach to competence modeling is based on a representation of relevant competences for a given task in a formalized notation. It uses meta-model which defines language or formalism for representing the competence model types. Ontology language is used as a meta-model to represent competences. Ontology language is used to describe ontology. An ontology describes a problem domain by using concepts, attributes, contraints and relations between concepts. Ontology can be built by ontology editor that allows exporting developed ontologies to OWL or RDF languages. By using ontology each competence item can be represented with a concept, competence measurement or description with a property and competence sub-items with relations to other concepts.

### 2.5.1 Competence modeling considerations

Modeling complex systems is a challenging process for designers due to their complexity and requirements. The same situation is in the case of modeling competences. Designing a competence model should include some intelligent behaviour. Knowledge management offers designers a real base for developing intelligent features of a system. For every domain, the knowledge must be understood and clearly represented in the model. Competences can be described as reausable domain knowledge. Any model representing competences describes what a competence is and how it is composed of sub-competences. Competences must be represented and described in order to:

- determine how a competence may be achieved(acquiring some sub-competences, etc...)
- determine on which level each competence should be acquired
- determine whether sub-competences must be all achieved or subset of them
- determine whether sub-competences must be acquired in a specific order

Another significant problem in competence modeling is the capability of a model to represent aggregate and alternative structures of the competence. The aggregation allows that the competence is composed from several sub-competences all of them required. Alternative structure is a set of competences and possibility to specify by a number interval what is the number of alternatives that must be acquired. In terms of reusing the model it is important that the relationships between the competences are clear and that it is well structured in order to be easily understood by many users. [VI08]

### 2.5.2 Building competence ontology

The powerful inference mechanisms coming with ontologies allow improving the effectiveness of complex competences and knowledge management processes. If an ontology must be formal then three formalizations are taken into consideration:

- Terms definition - In this case an ontology corresponds to a taxonomy
- Definition of concepts and their relationships
- Definition of axioms concerning concepts and their relationships [Har04]

Ontology formalism kinds are shown on 2.14.

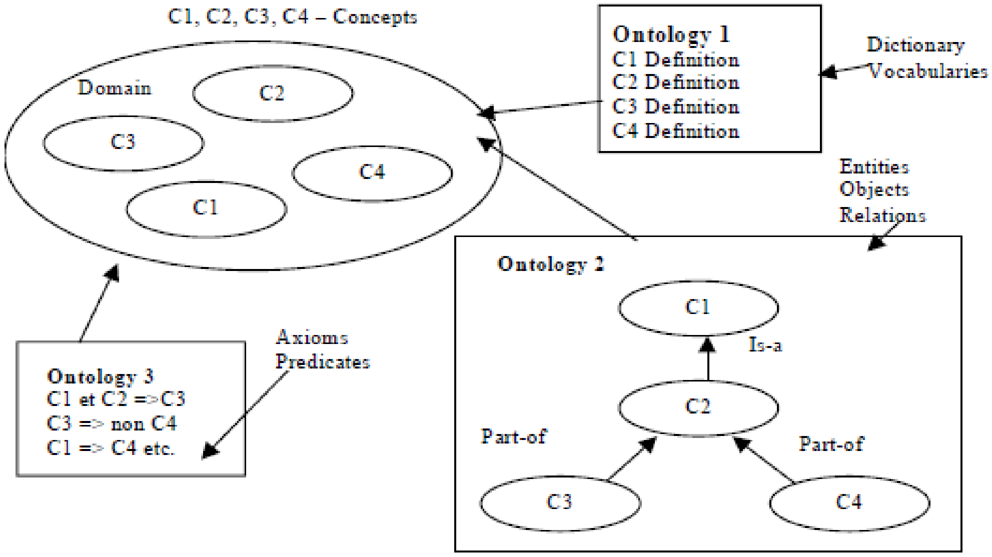


Figure 2.14: Ontology formalism kinds[Har04]

There are three methods of building ontologies:

- Manual methods where experts of a specific domain build a new ontology or extend an existing one

- Automated methods where an ontology is built by using techniques of text mining: concepts and their relationships are extracted and they are verified by inference mechanisms to define a consistent ontology

- Mixed methods where an ontology is built by automated techniques, but it can be manually extended [Har04]

Competence ontology concerns the competencies related to a specific domain. This ontology is composed of competencies related between them. There are two considerations related to competence ontology relations:

- The kind of a relationships between competencies. Relationships ***Is-a*** or ***Part-of*** explaining aggreagation of competencies or the relationship explaining that an individual should have a given competency to acquire new one can be used.

- What concepts would be on a competence ontology? Is is the case that ontology includes only competencies or is it the case where ontology includes some other concepts like studied domain and its knowledge. [Har04]

The structure of the ontology is conceived so that description logics can be used to represent the concept definitions of the application domain in a structured and formally well-understood way. The ontology also contains the rules for integrity validation and inference. The inference rules allows describing knowledge about the competences. In the knowledge acquisiton step the ontology is enriched by competence requirements. [CN09]

### 2.5.3 Competence Management

Competence management deals with identification and management of experts and their knowledge in certain competence areas. As knowledge became a crucial factor in achieving commercial success, competence management became a growing area in research. Every organization tends to keep up-to-date with the competences among its work force. Competence coverage and structure change rapidly over time so some kind of automatic identification of competence coverage and structure, e.g. from publications, is therefore of increasing importance because this allows for a dynamic and time-efficient approach to competency management. [PB08]

In order to support competence management many companies use competence management systems which are implemented on different abstraction levels. Competence management systems have to fulfil following requirements:

- content - how detailed the competence model is
- technical implementation - which data is kept when and how data is kept up to date
- organisational implementation - who implements the system, how are people motivated to use the system and to keep data up to date [DH09]

One of the existing competence management tools is eCompetence management tool which enables users to co-manage the system ontology via a graphical user interface. The competencies are identified through text extraction methods. Each competence is represented by a collection of skills. The evaluation algorithm(applied recursively in an ontology) is used to measure a percentage value that represents the degree of a certain competence. [DH09]

The paper [PB08] proposed an approach to the extraction of competences in a knowledge-research organization (scientific topics) from publicly available scientific publications. The assumption in that approach is that such topics does not occur randomly across documents, but they occur only in specific contexts that can be used as patterns for topic extraction of scientific publications. Competence management has been focused mostly on the delevopment of methods for identification, modeling and analysis of skills and their gaps. Initial step is to identify skills and knowledge of interest which is done mostly through interviews, surveys and manual analysis of existing competence models. Ontology-based approaches have been proposed that aim at modeling the domain model of some organization. The use of formal ontologies for competence management is very important, but there is a need to make construction and dynamic maintenance of ontologies to be more automatic process. For automated and dynamic support of competence management a richer analysis od competences and their relations is needed.

The approach in paper [PB08] towards the automatic construction and dynamic maintenance of ontologies for competence management is based on the extraction of relevant competences and semantic relations between them. It is based on the domain-specific linguistic patterns for the extraction of competences from publicly available scientific publications.

### Competence Measurement

The approach in paper [Sha10] for measuring competences is based on a Assessment Triangle shown on Figure 2.15.

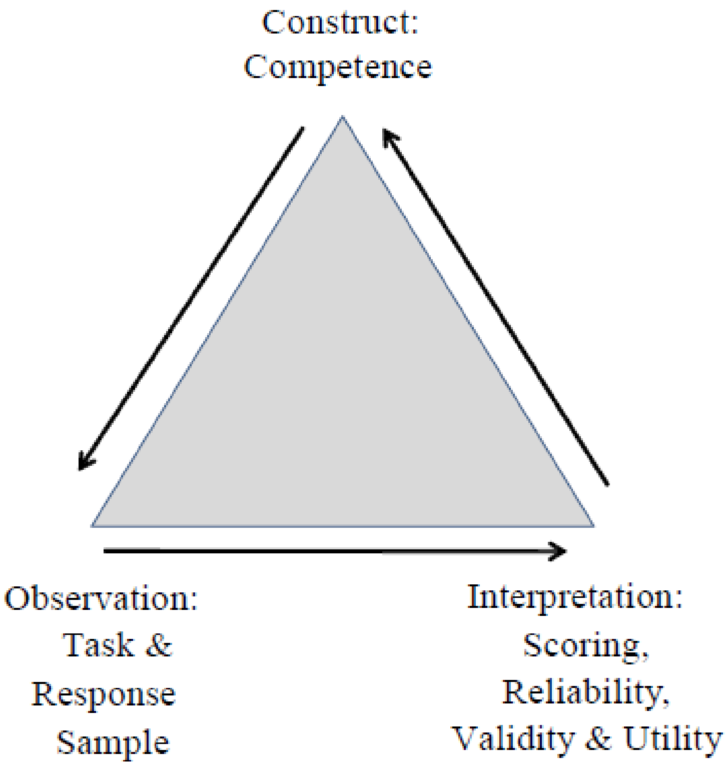


Figure 2.15: The Assessment Triangle[Sha10]

This triangle has three concepts: Construct, Observation and Interpretation. Competences may be simple or complex. Underlying performance and competence are a complex set of abilities. Those abilities are grouped together when a person attempts to meet task and response demands. Competence measurement focuses on real-world tasks and responses to them recognizing a multitude of abilities. A task is invoking the construct which is a competence. By engaging in the task a person's behavior can be observed. The presence or absence of the construct and person's level of performance can be observed. From the definition of the construct the universe of possible tasks and responses for observing competence perfomance can be constructed. The competence measurement os driven by an idea, definition or mini-theory of competence. The competence definition guides the selection or sampling of tasks and responses that go into an assessment. From a sample of responses (person behavior) an inference is drawn about the degree of that person's competence in some domain.

The definition of competence should identify a domain or universe od tasks that might be sampled for assessment. For the observation phase from the Assessment Triangle the following sampling approach is needed:

- The domain in which competence is to be assessed should be specified.
- Domain should be analyzed. Potential tasks and responses should be included and others excluded.
- The sample should be chosen.

Next step in competence assessment is an interpretation. In this phase the focus is on methods and tools used to score perfomance and draw inferences from a sample to an interpretation that reflects a competency level of a person [Sha10].

In paper [DPST08] two types of competencies which are functional and behavioral are distinguished and they both have different values, knowledge and experience. Regarding the measurement of functional and behavioral competencies the difference between these two is that the functional ones are easier to measure. For example, it is easire to measure some technical skills like programming, designing a system, etc... than measuring the decision-making skills. Another important fact is the self-assessment of the target person whose skills are measured. It is easier when person whose skills are measured can more easily give some of their own assessment than having just assessment of the experts. That is the reason why functional competencies are easier to measure because it is more difficult for persons to measure their own behavioral competencies.

In [DP07] it is stated it must be agreed on for the evaluation of strength of skills. It is not easy to determine and measure whether a person has some technical skills like programming. It is even more difficult to measure whether a person has decision-making skills, leadership, etc... There are also interdependencies between skills. For example if a person has skills in .NET programming language we can say that the person has also skills in object-oriented programming languages. There can be strong and soft dependencies between the skills.

The method in this thesis extracts technical competences of University staff members. In order to measure more accuratelly those technical competences there are some aspects that need to be considered:

- Technical competences can be measured by using competence strength and trust levels

- The fact is that technical competences can change over time
- Competence improvement should be considered in a measurement phase
- The dependencies between competences could be considered

The measurement of technical competence strength and trust levels in proposed method in this thesis is based on publications data. The data which is extracted from publications (title, keywords, abstract) is used to get technical competences. The competence trust level is measured by determining if a person is a main author of a publication or if a person is a sub author and the number of other authors is measured. Three competence trust levels which are low, medium and high are distinguished in the measurement process. For the measurement of technical competence strength level the number of occurrences of that technical competence in the database related to that author is used. Three technical competence strength levels which are beginner, intermediate and expert are distinguished.

The measurement process of technical competences need to consider the fact that technical competences can change over time. This means if some person has technical competence today in JAVA programming language that does not mean that it will last forever. Nowadays, technology rapidly changes, especially there are many changes in Computer Science. New technologies, concepts, programming langauges are introduced which have an influence on technical competences of some persons. If those persons do not follow and learn new technologies then there might be the case that they are not competent in some area anymore. Due to this fact the measurement process of technical competences must be adapted to technology changes and it must run continuosly in order to have up-to-date data.

Competence improvement should also be considered in the measurement phase. There might be some factors which influence the improvement of some technical competence like education, training, working on many projects, doing research is that area and publishing papers, etc... That is also one of the reasons that competence measurement should be a continuous process.

In the method proposed in this thesis the Ontology is used to present technical competence concepts related to some domain. The Ontology structure provide the ability to measure dependencies between competences. For example there might be a technical competence concept which is a Software development in the Ontology tree which has sub concepts like C# programming language, JAVA, Visual Basic, C, Python, etc... If some technical competence is obtained like JAVA programming and it is located somewhere in the tree, the Ontology provides the ability to measure the dependencies which must be determined. Those dependencies might be determined by setting some value which shows how deeply we consider concepts in the tree, how many other technical competences in the same category must person have in order to be competent in that field. In this case if a person has a technical competence in JAVA programming we could say that it also has competence in software development. But, in the other case we can say that a person should have programming skills also in C, Visual Basic, Python and C# in order to be competent in software development.

CHAPTER 3

# Competence Extraction

## 3.1 Background information

We have seen from presented literature in Chapter 2 that it is very hard to create a Web data extraction software which will be able to deal with all the challenges which are present in Web data extraction field. Some of the challenges are :

- Extract data from many sources which have different structures
- Ability to link unstructured data which is located in different Web elements.
- Design a software which will have a minimal human interaction with the software
- Make software as a full automatic process which will adapt to structural changes on Web sources without any human assistance
- Make an automatic Web data extraction software and try to get an accurate system
- Identify and measure competences

Existing Web data extraction tools have some common functionalities and some of them are more powerfull than the others in the sense that they can extract data from Websites with a more complicated structures.

This thesis presents a method which will extract a list of University staff members and query digital libraries in order to get their publication data(title, keywords, abstract) and to map that data to an existing ontology which will be an input to this software in order to get professional competences of those faculty members. One of the main requirements of this software is that it works for all Universities. The fact is that there a many Universities in the world and some of them have similar Web page structures and some have pretty much different structure.

The method presented in this work is composed of two parts. First part of the software will gather the data in the form of a bookmarklet which is shown in Figure 3.1 and the second part is a program coded in C# programming language which takes data from a bookmarklet as its input.

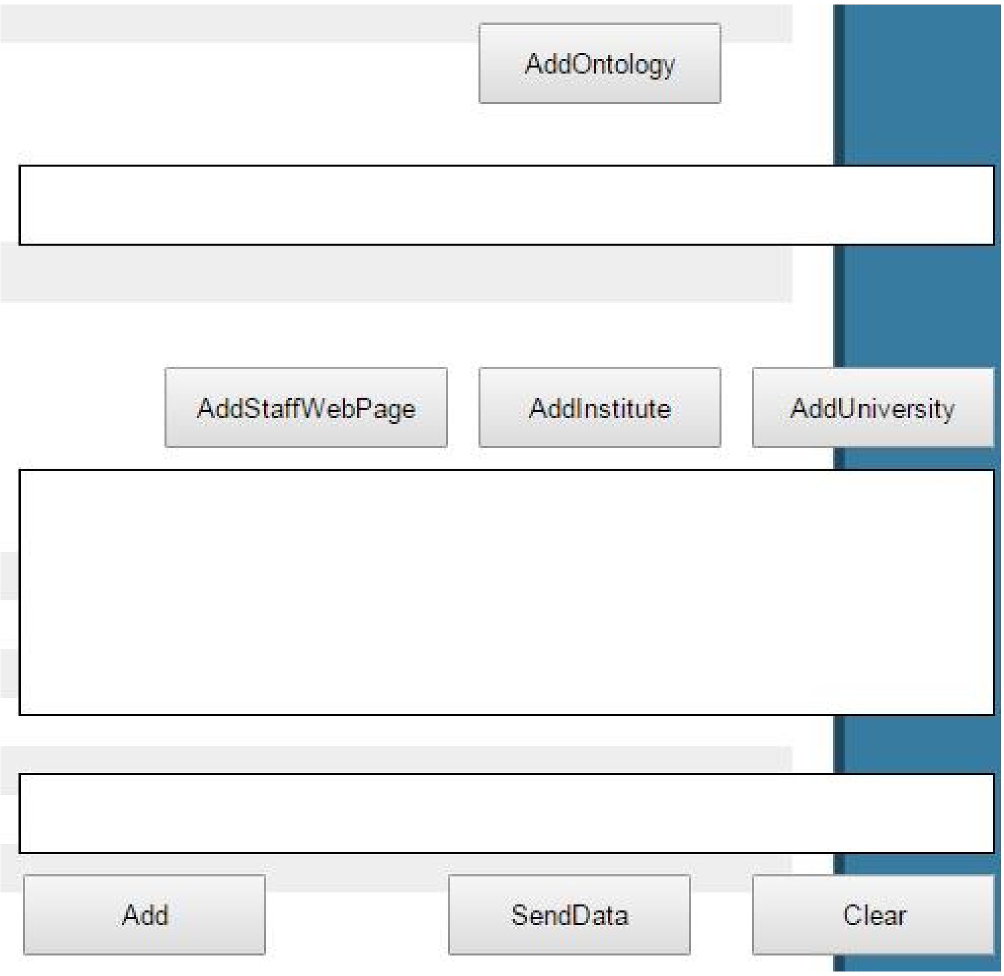


Figure 3.1: Bookmarklet

## 3.2 Why this approach?

This approach to make a Web data extraction software in a form of a bookmarklet and an additional separate program is chosen due to advantages which bookmarklets provide. Bookmarklet is a program which is coded in a bookmark form by using JavaScript and it is stored in a Web browser. By using bookmarklets you can select elements on a Web page, highlight some text, get some data, modify existing elements etc. . . You can also append HTML of a Web page by adding some input fields and buttons. With bookmarklets you can also add or read cookies from a Web browser. This cookie feature is very useful because you can save data from a home page which could be latter loaded and used as an input on some subpages or it can be just used to store some data from a page and its subpages.

One of the reasons why bookmarklet is used in this software is also due to its simplicity and ease of use. Most of the users can very easily use bookmarklets by simply loading it from bookmarks menu.

As we already know, Web pages change their structures. Also, the requirement of this software is to extract data from all Universities. When we consider these two important challenges we get the main reason why this approach is chosen to solve these issues. Bookmarklet can handle these challenges since it is not dependent on a Web page structure and it can be loaded on every Web page.

The second part of the Web data extraction software is a program written in C# which gets data from a bookmarklet through a Web service and extracts data according to the input provided by a bookmarklet. Data extraction is done by using a Selenium framework. Selenium acts as a Web browser and provides a possibility to read and extract data from a Website. As Selenium can locate elements on a Web page it gets a CSS selector of elements where some data is located as an input from a bookmarklet and it extracts data according to those CSS selectors. An example program which uses Selenium is shown in Listing 3.1.

Listing 3.1: A method in a program which uses Selenium Firefox WebDriver

```
public void extractstaffdata()
{
firefoxdriver.Navigate().GoToUrl(staffwebpage);
firefoxdriver.Manage().Window.Maximize();
firefoxdriver.Manage().Timeouts().ImplicitlyWait(TimeSpan.FromSeconds(1));

IList<IWebElement> lstofPeople =
firefoxdriver.FindElements(By.CssSelector(stafflistlocation));

foreach (IWebElement people in lstofPeople)
{

   if (!String.IsNullOrEmpty(people.Text))
   {
         peoplelist.Add(people.Text);

   }

}
```

## 3.3 Software requirements

Table 3.1 shows the list of user requirements.

Table 3.1: User requirements

| No. | Requirement Description |
|---|---|
| 1. | User should be able to have a possibility to choose from which University Website to extract professional competences |
| 1.1 | User should be able to run competence extractor on chosen Website |
| 2. | User should be able to add an ontology as an input to the extractor |
| 2.1 | Ontology source should be a file location or URL |
| 3. | User should be able to select a CSS-selector of list of University staff members using SelectorGadget bookmarklet |
| 3.1 | There should be a possibility to add a CSS-selector to the main bookmarklet as an input |
| 4. | User should be also able to add University name, Institute name and University Website as an input to the extractor |
| 5. | User should be able to send input data from bookmarklet to the Web service which processes data and extracts professional competences from digital libraries |

Table  3.2 shows the list of system requirements.

Table 3.2: System requirements

| No. | Requirement Description |
|---|---|
| 1. | **Get user request.** The user's request is received and processed by the software. |
| 1.1 | Handle user request. User request with input data is handled by the software. |
| 2. | **Extract list of University staff members.** |
| 2.1 | Software extracts list of University staff members from the target Website. |
| 2.2 | Store list of University staff members in a database. |
| 3. | **Query digital libraries.** Digital libraries are queried in order to get publications data for extracted list done in step 2.1 |
| 3.1 | Publications data which is extracted are keywords, title and abstract. |
| 3.2 | Store publications data in a database. |
| 4. | **Get professional competences of University staff members.** |
| 4.1 | Professional competences are obtained by mapping publications data with the competence concepts in an ontology. |
| 5. | **Store professional competences in a database.** |
| 6. | **Calculate professional competence strength level.** This is done by executing a stored procedure in a database. |

## 3.4 Professional competence model

Model for obtaining professional competences of Uiversity staff members is based on following concepts:

- It is based on an existing ontology which already contains some competence concepts and types like personal competences, problem solving competence, professional competences...
- The list of professional competence concepts in ontology is extended to cover the domain of an use case
- The research is done in order to get knowledge about use case in order to more accurately define competences and its subcompetences in an ontology

- In this professional competence model ontology the relation ***subClassOf*** is used to get parent competence
- Publications data is mapped to concepts in the model in order to get professional competences and subcompetences
- Ontology language OWL is used to describe ontology and SPARQL queries are used to query the ontology of the professional competence model

The professional competence model is shown on Figure 3.2.

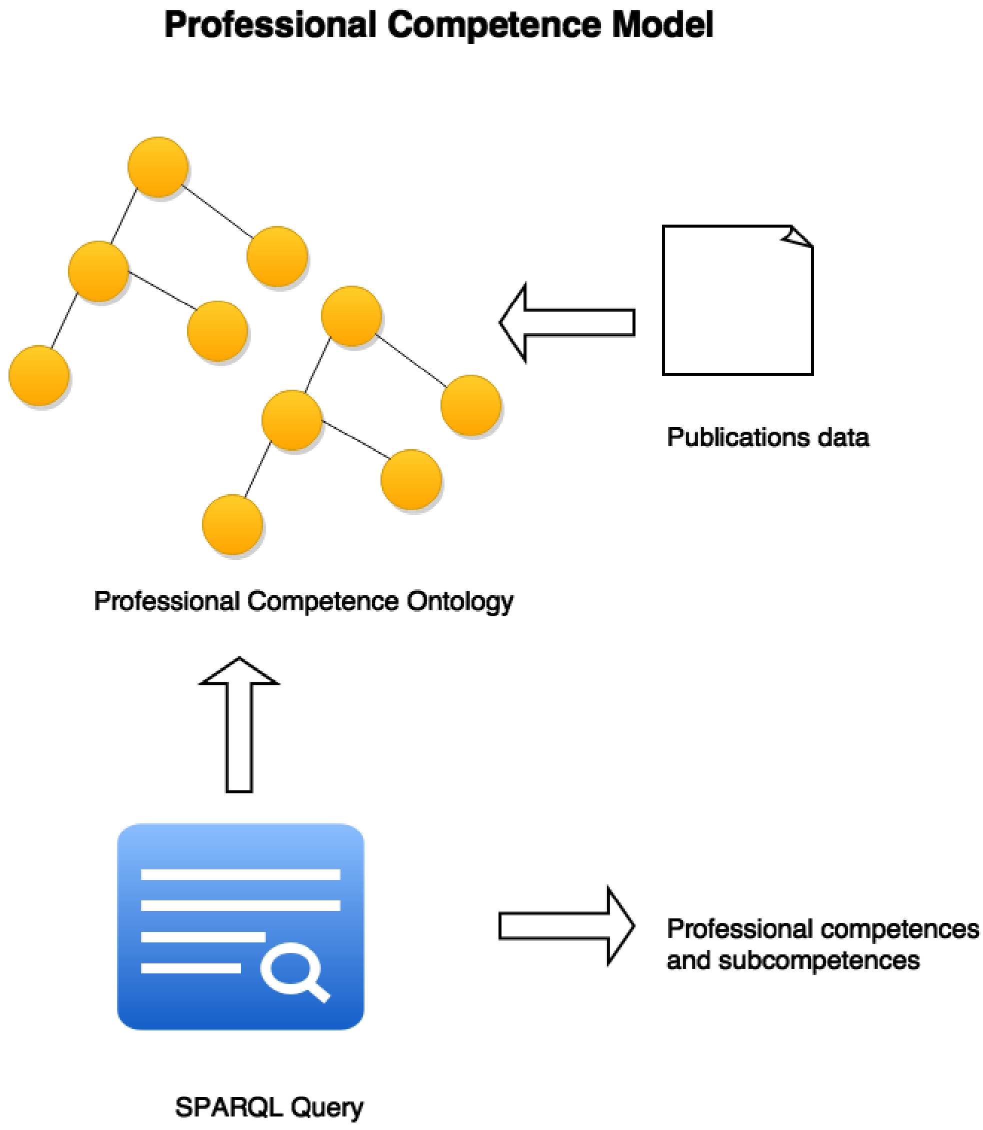


Figure 3.2: Professional competence model

CHAPTER 4

# Use Case

## 4.1 Institute of Computer Aided Automation

The use case chosen for the program in this thesis is the Institute of Computer Aided Automation at the Vienna University of Technology. This institute has two research areas. First research area is Automation Systems Group which is doing a research in the following areas:

- Distributed automation systems
- Program analysis for real-time programs
- Design of fault-tolerant systems
- Formal methods for control systems
- Networks security [Aut15]

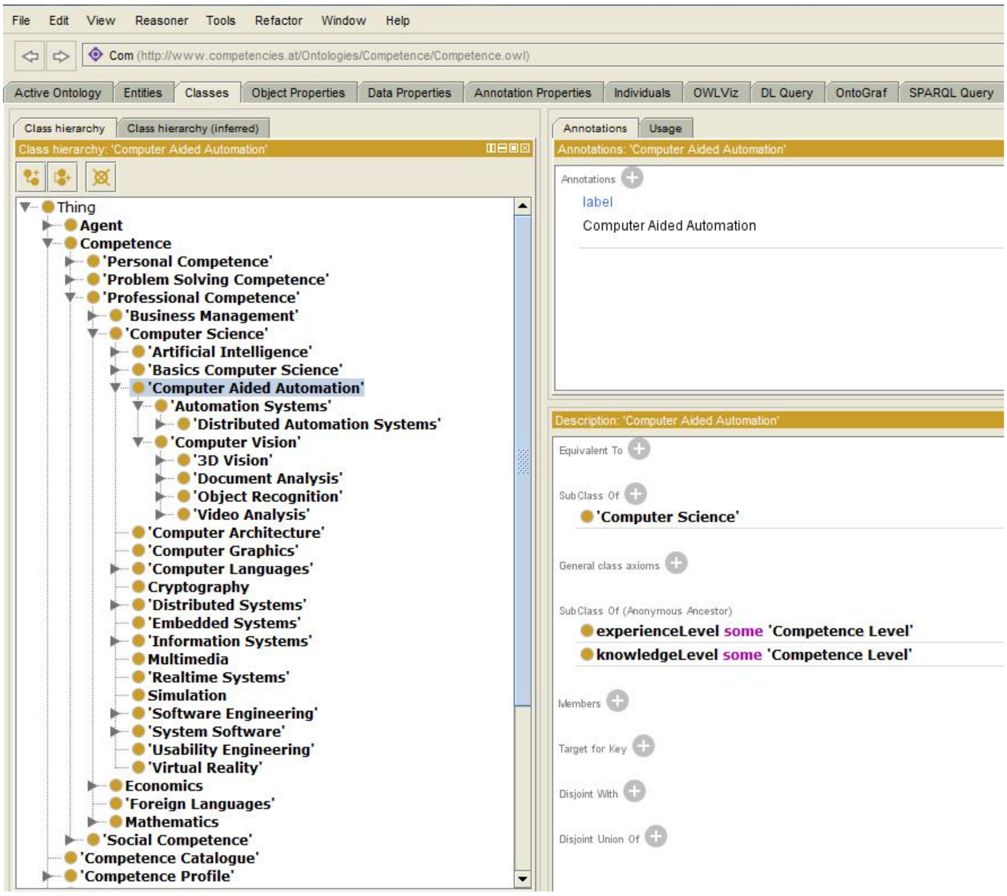
Second research area is a Computer Vision Lab whose research is done in:

- Object Recognition
  - Robust Local Video and Image Features
  - Large Scale Object Recognition
  - Efficient Replica Detection
- Document Analysis
  - Text Classification and Writer Identification
  - Segmentation

  - Layout Analysis
  - Multispectral Imaging
- Video Analysis
  - Motion detection
  - Object tracking
  - Scene Analysis
- 3D Vision
  - Calibration
  - Stereo Vision
  - Structured Light [Com15]

The program needs to extract a list of staff members from this institute and then to query digital libraries in order to find their publications. The next step is to extract publications data(keywords, title and abstract) and to use that data in a program to get professional competences of those University members.

In order to get professional competences there is a need to have an ontology where publications data will be mapped. For this use case an existing ontology is used which is extended based on a research done at that institute in order to cover the domain of an use case. Ontology is extended using Protege editor and OWL Web Ontology Language and shown in Figure 4.1.



Class hierarchy: 'Computer Aided Automation'

```
Thing
  Agent
  Competence
    'Personal Competence'
    'Problem Solving Competence'
    'Professional Competence'
      'Business Management'
      'Computer Science'
        'Artificial Intelligence'
        'Basics Computer Science'
        'Computer Aided Automation'
          'Automation Systems'
            'Distributed Automation Systems'
          'Computer Vision'
            '3D Vision'
            'Document Analysis'
            'Object Recognition'
            'Video Analysis'
        'Computer Architecture'
        'Computer Graphics'
        'Computer Languages'
        Cryptography
        'Distributed Systems'
        'Embedded Systems'
        'Information Systems'
        Multimedia
        'Realtime Systems'
        Simulation
        'Software Engineering'
        'System Software'
        'Usability Engineering'
        'Virtual Reality'
      Economics
      'Foreign Languages'
      Mathematics
    'Social Competence'
  'Competence Catalogue'
  'Competence Profile'
```

Annotations: 'Computer Aided Automation'

Annotations
label
Computer Aided Automation

Description: 'Computer Aided Automation'

Equivalent To

SubClass Of
'Computer Science'

General class axioms

SubClass Of (Anonymous Ancestor)
experienceLevel some 'Competence Level'
knowledgeLevel some 'Competence Level'

Members

Target for Key

Disjoint With

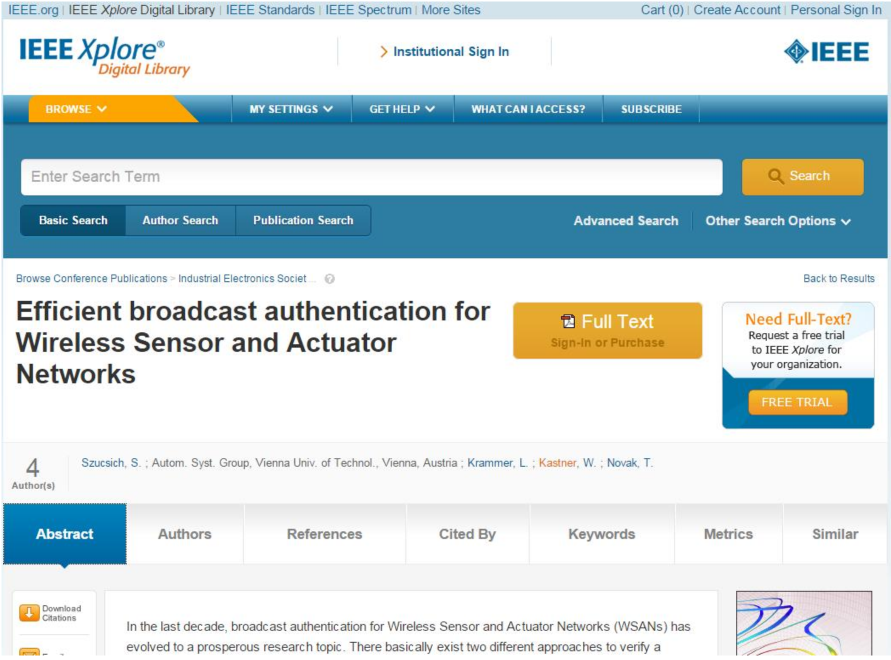
Disjoint Union Of

Figure 4.1: Extended ontology with a domain of an use case

## 4.2 Digital libraries for use case

Digital libraries which are used for extracting publications data are IEEE Xplore and ACM. They are used because they contain publications in Informatics field and the domain of an use case is also in this field.

As we can see on Figures 4.2 and 4.3 both digital libraries provide a nice user interface where user can search for publications of particular author. When we open some publication we get a well organized structured data where we can easily locate publication title, abstract and keywords which we want to extract.



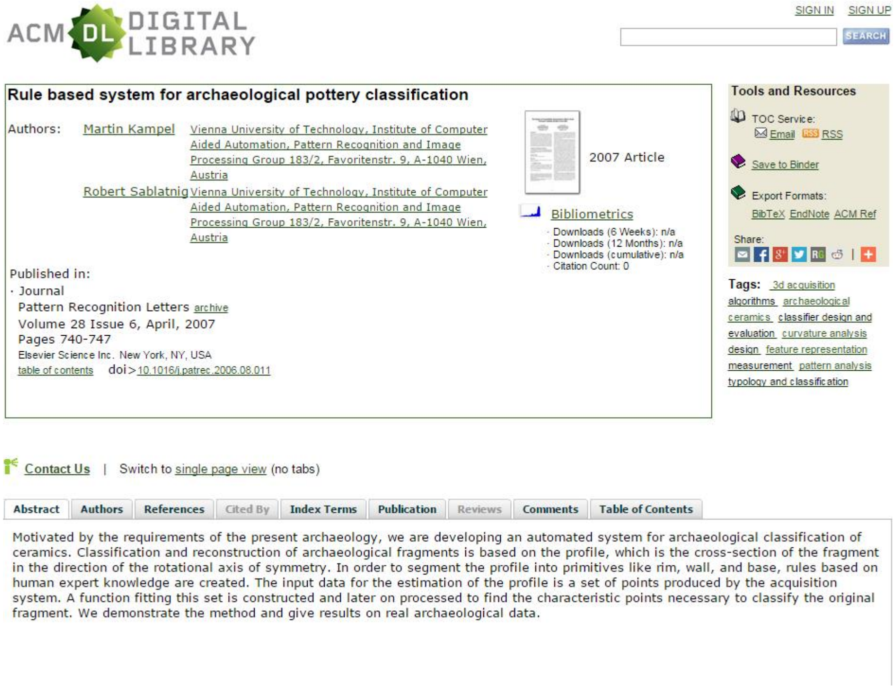
Efficient broadcast authentication for Wireless Sensor and Actuator Networks

4 Author(s)

Szucsich, S. ; Autom. Syst. Group, Vienna Univ. of Technol., Vienna, Austria ; Krammer, L. ; Kastner, W. ; Novak, T.

In the last decade, broadcast authentication for Wireless Sensor and Actuator Networks (WSANs) has evolved to a prosperous research topic. There basically exist two different approaches to verify a

Figure 4.2: IEEE Xplore digital library



Rule based system for archaeological pottery classification

Authors: Martin Kampel — Vienna University of Technology, Institute of Computer Aided Automation, Pattern Recognition and Image Processing Group 183/2, Favoritenstr. 9, A-1040 Wien, Austria

Robert Sablatnig — Vienna University of Technology, Institute of Computer Aided Automation, Pattern Recognition and Image Processing Group 183/2, Favoritenstr. 9, A-1040 Wien, Austria

2007 Article

Bibliometrics
- Downloads (6 Weeks): n/a
- Downloads (12 Months): n/a
- Downloads (cumulative): n/a
- Citation Count: 0

Published in:
- Journal
  Pattern Recognition Letters archive
  Volume 28 Issue 6, April, 2007
  Pages 740-747
  Elsevier Science Inc. New York, NY, USA
  table of contents doi>10.1016/j.patrec.2006.08.011

Tags: 3d acquisition algorithms archaeological ceramics classifier design and evaluation curvature analysis design feature representation measurement pattern analysis typology and classification

Motivated by the requirements of the present archaeology, we are developing an automated system for archaeological classification of ceramics. Classification and reconstruction of archaeological fragments is based on the profile, which is the cross-section of the fragment in the direction of the rotational axis of symmetry. In order to segment the profile into primitives like rim, wall, and base, rules based on human expert knowledge are created. The input data for the estimation of the profile is a set of points produced by the acquisition system. A function fitting this set is constructed and later on processed to find the characteristic points necessary to classify the original fragment. We demonstrate the method and give results on real archaeological data.

Figure 4.3: ACM digital library

CHAPTER 5

# Implementation

The Web data extraction software which extracts University staff professional competences is implemented by using the following technologies :

- SelectorGadget bookmarklet
- JavaScript
- Selenium framework
- Microsoft Visual Studio and C# programming language
- Web service
- Protege editor and OWL Web Ontology Language
- MySQL database

This software has two phases. The first phase is about gathering required data for the software.

That data is an input for the second phase which deals with extracting publication data from digital libraries and mapping it to concepts in an ontology in order to get professional competences of University staff for which publications data is extracted.

Technology overview and data flow of the implemented software are shown in Figure 5.1.

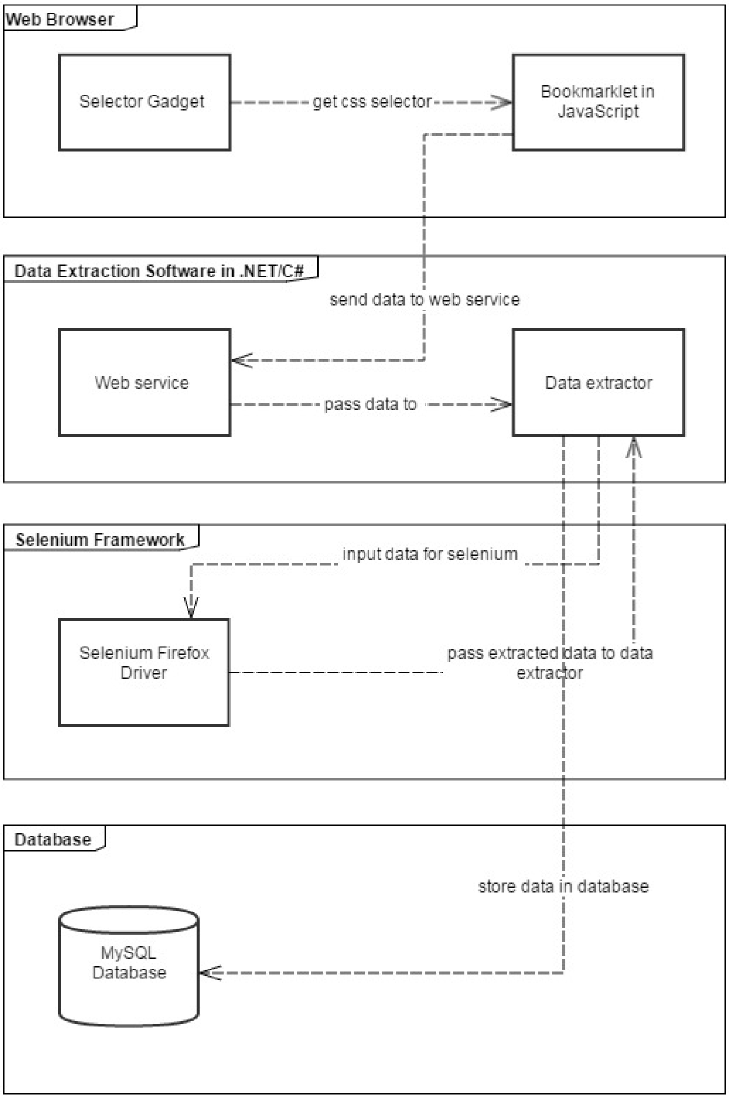


Figure 5.1: Technology overview and data flow

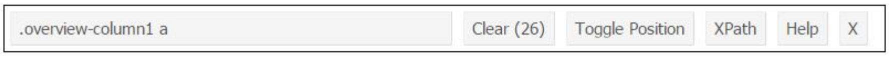
## 5.1 SelectorGadget

The first part of the software uses SelectorGadget bookmarklet. It is an open source tool which makes CSS selector generation and discovery on complicated sites as an easy process. First step is to add this bookmarklet to bookmarks bar in a browser and when we navigate to a target page we launch it. By clicking on this bookmarklet the dialog box will open in the right bottom of the page which is shown on Figure 5.2. When the SelectorGadget is loaded then user needs to select desired elements which contain some data for which he wants to get CSS selector. When user clicks on some element it turns green and the SelectorGadget automatically tries to match all related elements and they are highlighted by yellow color. If SelectorGadget selects some undesired additional elements there is a possibility to easily deselect them by clicking on some undesired highlighted element. That element becomes highlighted with a red color. Through this process of selection and rejection, this bookmarklet provides a very nice and efficient user interface for users and it provides an ease of use where majority of users can use it without much technical knowledge. [Sel15]

The common usage of this bookmarklet is:

- For providing CSS selectors to Web data extraction systems which they use to locate and extract data
- To generate jQuery selectors for dynamic sites
- As a tool to examine JavaScript-generated DOM structures
- For Selenium framework tests [Sel15]

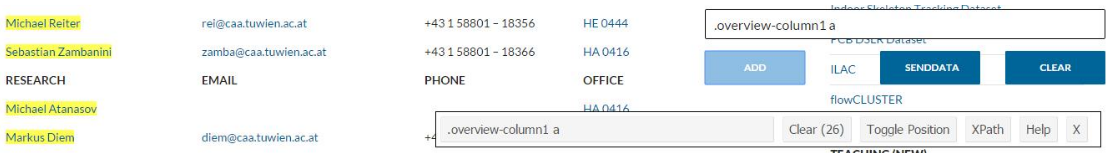
SelectorGadget also provides a possibility to obtain an XPath of the selected elements on a page. XPath button is shown on Figure 5.2. When user clicks this button a pop-up dialog opens and generates an XPath of the elements. This is also an usefull option since some Web data extraction systems use XPath to locate and extract data rather than CSS selector.



.overview-column1 a

Figure 5.2: SelectorGadget

For this software SelectorGadget is used to get CSS selector of the element which contains a list of University staff. This CSS selector is passed to another bookmarklet which is used to store all required data for the second phase in one field. This process of adding a CSS selector is shown in Figure 5.3. On this figure the elements which contain a list of University staff are highlighted and the CSS selector is added to the bookmarklet.



| | | | |
|---|---|---|---|
| Michael Reiter | rei@caa.tuwien.ac.at | +43 1 58801 - 18356 | HE 0444 |
| Sebastian Zambanini | zamba@caa.tuwien.ac.at | +43 1 58801 - 18366 | HA 0416 |
| RESEARCH | EMAIL | PHONE | OFFICE |
| Michael Atanasov | | | HA 0416 |
| Markus Diem | diem@caa.tuwien.ac.at | +4 | |

.overview-column1 a

.overview-column1 a

Figure 5.3: Adding CSS selector to bookmarklet

## 5.2 Bookmarklet

The complete bookmarklet is shown in Figure 5.4. The first input field is used to add ontology source since this software takes as an input ontology in order to provide a possibility to extend it to some domains which are target for Web data extraction process. The ontology can be loaded from a file or from some url. The next buttons provided by this bookmarklet are to add University, Institute and Web page from which list of University staff is extracted. This is added manually by the user for the accuracy reasons. There is a one main input field where all data is added and the data is added according to the pattern :

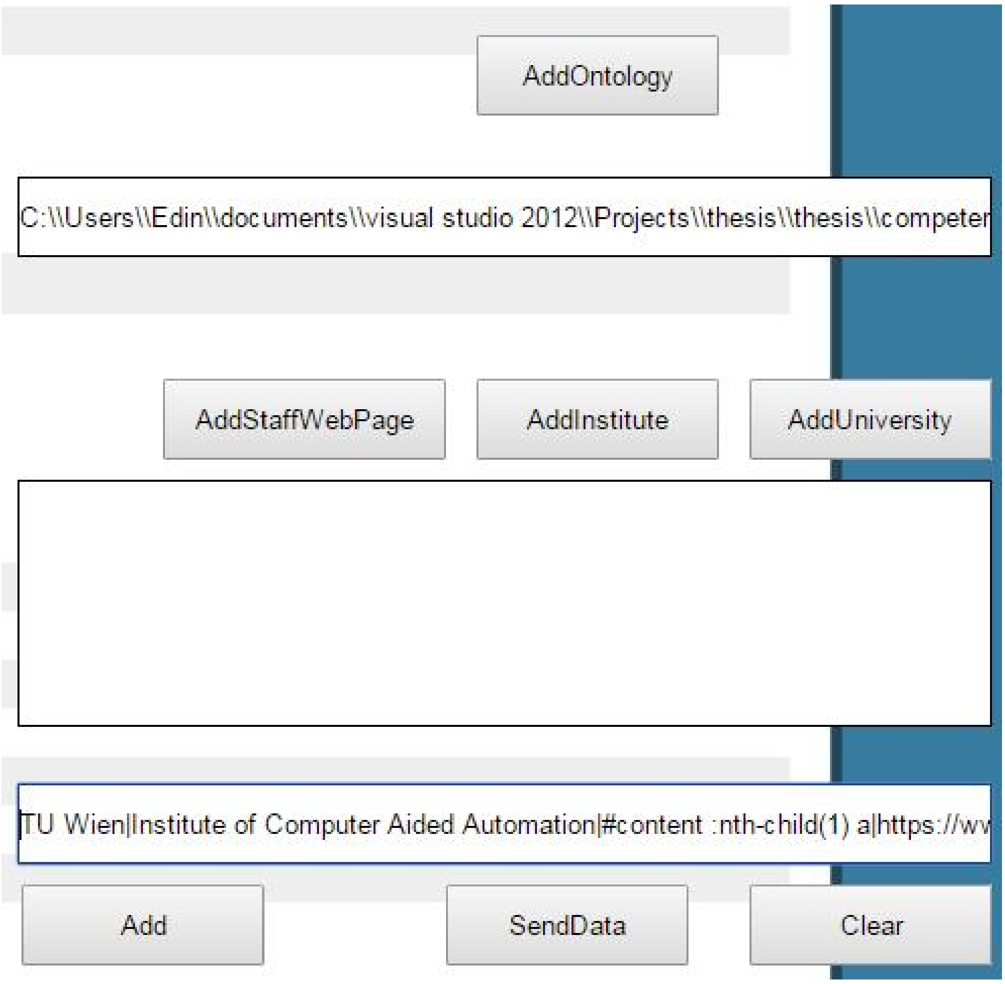
University|Institute|CSS selector of elements which containt list of University staff|Web page where staff is located|Ontology location

The data for use case following this pattern is shown in Figure 5.4 in the bottom input field of a bookmarklet.

The cookies are also used in this bookmarklet in order to store data. This is due to a reason when we locate some University data on some subpage and we need to store it in a bookmarklet and to load it latter. This is done by storing it in a cookie and when we get to the target page where University staff list is located we load a bookmarklet and data from a cookie and we populate input field with that data. One more reason for storing a data in a cookie is when we run a bookmarklet and after that we run SelectorGadget bookmarklet and we want to click to a button on a bookmarklet in order to get CSS selector of some element which we selected by SelectorGadget. This cannot be done because we already had an open bookmarklet and SelectorGadget was loaded after and it just selects that button as an element when we click on it. To overcome this issue data is stored in a cookie and we load again bookmarklet after a SelectorGadget is loaded and in that case we overcome this issue. A code which sets a cookie with some data is shown in 5.1.

Listing 5.1: A code in JavaScript for adding data to a cookie

```
var universitybtn = document.createElement('button');

universitybtn.value = 'AddUniversity';
universitybtn.style.width = '120px';
universitybtn.style.height = '40px';
universitybtn.style.position = 'fixed';
universitybtn.style.top = '400px';
universitybtn.style.right = '5px';
universitybtn.onclick = function () {

        var universityname = getSelection();
        if(inputelement.value)
        {

        }
        else
        {
                inputelement.value= universityname;

        }
        var newv =  inputelement.value;

        setCookie('dataselector', newv,1);

    };
universitybtn.appendChild(document.createTextNode('AddUniversity'));
document.getElementsByTagName('body')[0].appendChild(universitybtn);
```



C:\\Users\\Edin\\documents\\visual studio 2012\\Projects\\thesis\\thesis\\competen

TU Wien|Institute of Computer Aided Automation|#content :nth-child(1) a|https://wv

Figure 5.4: Bookmarklet with added data

When the all required data which is already described by the mentioned pattern is added to an input field on a bookmarklet, the next step is to send that data to a Web service. The Web service HTTP GET request and response is shown in 5.2.

Listing 5.2: A HTTP GET request and response for a Web service

```
GET /GetDataWebService.asmx/calldataextractor?extractdata=string HTTP/1.1
Host: localhost

HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
```

```
<string xmlns="http://tempuri.org/">string</string>
```

This bookmarklet also uses the HTML tag iframe which is an inline frame which provides possibility to insert some HTML from another source within the current HTML of a page. It is shown on a Figure 5.4 as a blank rectangle. The reason why it is used is to get a response from a Web service to which data is sent. That response can be used for testing purpose. The iframe src property is used to set an URL of some source and to display it in a iframe. This property is used to send data to a Web service. The first step is loading data from a cookie and the next step is to use a function encodeURIComponent() in order to escape some characters in a URL. The data is sent to a Web service by setting a URL of a Web service which takes one parameter as a string which is in this case encoded cookie data. This whole process is shown in Listing 5.3.

Listing 5.3: A code in a bookmarklet which sends data to a Web service

```
var senddatabtn = document.createElement('button');
senddatabtn.value = 'SendData';
senddatabtn.style.width = '120px';
senddatabtn.style.height = '40px';
senddatabtn.style.position = 'fixed';
senddatabtn.style.top = '650px';
senddatabtn.style.right = '155px';

senddatabtn.appendChild(document.createTextNode('SendData'));

document.getElementsByTagName('body')[0].appendChild(senddatabtn);

senddatabtn.onclick = function ()
{
var datasel = getCookie('dataselector');
document.getElementById('edin').src =
'http://localhost:55742/GetDataWebService.asmx
/calldataextractor?extractdata='+encodeURIComponent(datasel) + ''; };
```

## 5.3 Selenium framework

When Web service receives data sent from the bookmarklet it passes it to a Selenium WebDriver. Selenium is a set of different software tools each with a different approach to supporting test automation. Those tools are:

1. Selenium 2(Selenium WebDriver) which makes direct calls to the browser using each browsers native support for automation.

2. Selenium 1 (Selenium RC or Remote Control)

3. Selenium IDE (Integrated Development Environment) is a prototyping tool for building test scripts. It is a Firefox plugin and provides an easy-to-use interface for developing automated tests. Selenium IDE has a recording feature, which records user actions as they are performed and then exports them as a reusable script in one of many programming languages that can be later executed. [?]

Selenium WebDriver goal is to supply a well-designed object-oriented API that provides improved support for modern advanced Web application testing problems. Selenium WebDriver is a tool for automating Web application testing, and in particular to verify that they work as expected. [?]

Selenium Firefox WebDriver is used for the software in this thesis. When the data is retrieved which is sent by the bookmarklet, then this WebDriver executes the following:

1. Navigating to a University Web page
2. Locating WebElements which are list of University staff members by using CSS selector
3. Extracting a list of University staff
4. Navigating to a first digital library(IEEE Xplore)
5. Locating all required WebElements of a digital library by XPath
6. Submitting a list of University staff which was previously extracted to the search box and browsing for their publications
7. Locating and extracting publications data(title, keywords, abstract)
8. Navigating to a second digital library(ACM)
9. Executing steps 5,6,7

Extracted data which is a list of University staff and a list of their publications is saved in a database. Also, during the program execution the publications data(title, keywords, abstract) is mapped to concepts in an ontology by comparing strings using Levensthein distance.

## 5.4 Ontology

Ontology which is used in this software is an existing ontology created in [Pic08] and extended in [Hoc12] and it is extended by using the ontology editor Protégé and the Web Ontology Language (OWL) in order to cover domain of an use case. Use case is already described in Chapter 4. Publications data is mapped to the concepts in an ontology in order to get professional competences of University staff and those competences are stored in a database. SPARQL Protocol and RDF Query Language (SPARQL) is used to query concepts in an ontology.

SPARQL is a standard query language for Resource Description Framework (RDF) data that are commonly used to represent and store Semantic Web data. Although it greatly helps us query semantic data with ontologies, their diversity of ontologies make it difficult to query the data without understanding of their target ontologies. By using SPARQL queries we can specify concepts based on ontologies which are defined by RDFS or OWL in its query and the result. To realize a query processing which is not based on the ontology used at each endpoint, ontology mappings should be prepared. Ontology mappings define correspondences or semantic relations between the concepts in the ontologies. However, these mapping techniques are still emerging and they require further improvements. [FF12]

SPARQL query which queries an ontology to find all classes in an ontology which will be used for mapping with publications data is shown on Listing 5.4.

Listing 5.4: SPARQL query which lists all ontology classes which have subclasses

```
string sparqlQuery1 = "PREFIX owl: <http://www.w3.org/2002/07/owl#>"
+ "PREFIX swrl: <http://www.w3.org/2003/11/swrl#>"
+ "PREFIX swrlb: <http://www.w3.org/2003/11/swrlb#>"
+ "PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>"
+ "PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>"
+ "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"
+ "PREFIX protege: <http://protege.stanford.edu/plugins/owl/protege#>"
+ "PREFIX xsp: <http://www.owl-ontologies.com/2005/08/07/xsp.owl#>"

+ "SELECT  ?allclasses "
+ "WHERE{" ?allclasses a owl:Class .
+ "FILTER  EXISTS { ?allclasses rdfs:subClassOf ?somesubclass .}}";
```

## 5.5 Professional competence strength and trust levels

In order to have more accurate resulting professional competences we determine the trust and the strength of those extracted competences.

The trust has three levels: low, medium and high. The professional competence trust is determined by considering following cases:

- If a person is a main author of the publication and the list of all authors is greater than two then we assume that we get a medium competence trust level

- If a person is a main author of the publication and the list of all authors is less than two then we assume that we get a high competence trust level

- If a person is not a main author of the publication and the list of all authors is higher than two then we assume that we get a low competence level

The professional competence strength has three levels: beginner, intermediate and expert. We distinguish the following cases:

- If a number of occurences of some extracted professional competence for some author is less than five then we assume that we get a begginer strength level of that professional competence
- If a number of occurences of some extracted professional competence for some author is from five to ten then we assume that we get a intermediate strength level of that professional competence
- If a number of occurences of some extracted professional competence for some author is greater than ten then we assume that we get a expert strength level of that professional competence

The stored procedure in the database which calculates professional competence strength level of extracted data is shown in Listing 5.5.

Listing 5.5: Stored procedure which calculates professional competence strenght level

```
CREATE PROCEDURE ‘calculatecompetencystrenght‘()
BEGIN

DECLARE done BOOLEAN DEFAULT FALSE;

DECLARE inputname VARCHAR(50);
DECLARE cur CURSOR FOR SELECT distinct(name) FROM competencies;

DECLARE CONTINUE HANDLER FOR NOT FOUND SET done := TRUE;

OPEN cur;

selectLoop: LOOP

FETCH cur INTO inputname;

IF done THEN
      LEAVE selectLoop;
END IF;

CREATE TABLE temp AS SELECT count(*) AS numofoccurences, name, competency,
subcompetency FROM competencies WHERE name = inputname
GROUP BY competency
HAVING COUNT(*) > 1;
```

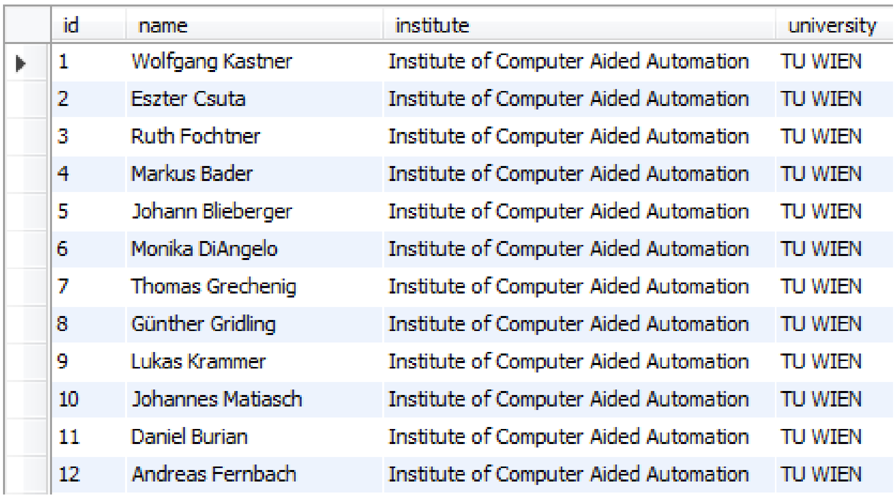
```
CREATE TABLE temp2 AS SELECT numofoccurences,name, competency,
subcompetency,
case  when numofoccurences < 5 then 'beginner'
when numofoccurences between 5 and 10 then 'intermediate'
when numofoccurences >10 then 'expert'
end as strenghtlevel
from
temp;

UPDATE  competencies SET strengthcomp =
(select strenghtlevel from temp2  where temp2.name = competencies.name and
temp2.competency = competencies.competency) where name = inputname ;

DROP TABLE temp;
DROP TABLE temp2;

END LOOP selectLoop;

CLOSE cur;


END
```

CHAPTER 6

# Results and Evaluation

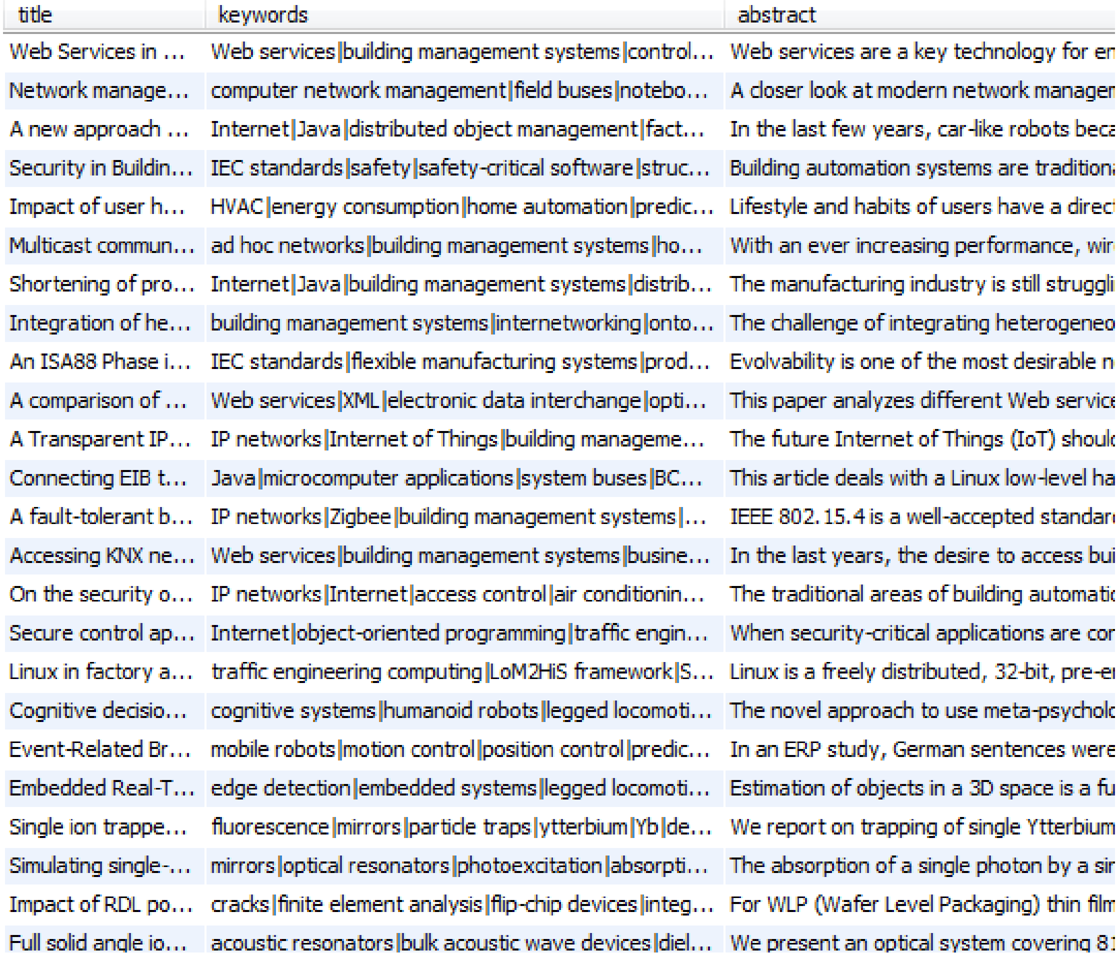
## 6.1 Extracted data

The results obtained from the software proposed in this thesis are shown in Figures 6.1 6.2 6.3. Figure 6.1 shows some of the extracted University staff members, Figure 6.2 shows some extracted data about publications which is a publication title, list of keywords and publication abstract which was used to get professional competencies of staff members which are shown in Figure 6.3. The number of extracted University staff members is 54, while the number of their extracted publications from digital libraries is 500 and the number of calculated professional competencies is 1463. This data is shown on Bar chart ———.



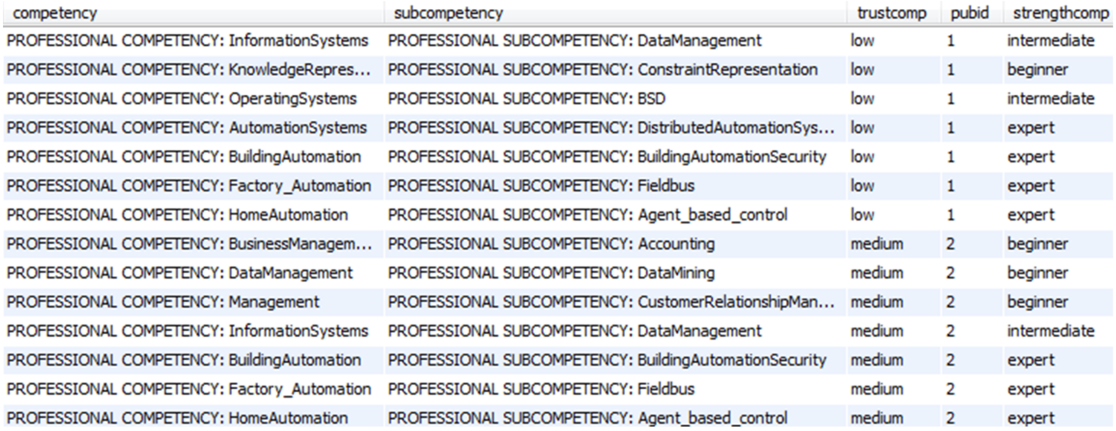
| id | name | institute | university |
|---|---|---|---|
| 1 | Wolfgang Kastner | Institute of Computer Aided Automation | TU WIEN |
| 2 | Eszter Csuta | Institute of Computer Aided Automation | TU WIEN |
| 3 | Ruth Fochtner | Institute of Computer Aided Automation | TU WIEN |
| 4 | Markus Bader | Institute of Computer Aided Automation | TU WIEN |
| 5 | Johann Blieberger | Institute of Computer Aided Automation | TU WIEN |
| 6 | Monika DiAngelo | Institute of Computer Aided Automation | TU WIEN |
| 7 | Thomas Grechenig | Institute of Computer Aided Automation | TU WIEN |
| 8 | Günther Gridling | Institute of Computer Aided Automation | TU WIEN |
| 9 | Lukas Krammer | Institute of Computer Aided Automation | TU WIEN |
| 10 | Johannes Matiasch | Institute of Computer Aided Automation | TU WIEN |
| 11 | Daniel Burian | Institute of Computer Aided Automation | TU WIEN |
| 12 | Andreas Fernbach | Institute of Computer Aided Automation | TU WIEN |

Figure 6.1: Extracted University staff list



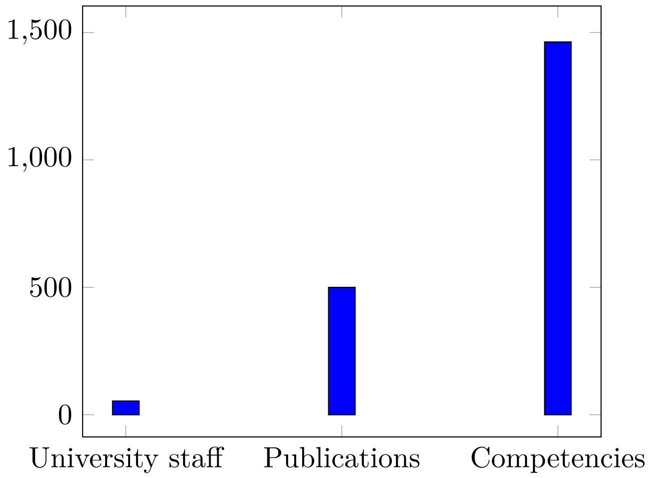
| title | keywords | abstract |
|---|---|---|
| Web Services in ... | Web services\|building management systems\|control... | Web services are a key technology for en |
| Network manage... | computer network management\|field buses\|notebo... | A closer look at modern network managen |
| A new approach ... | Internet\|Java\|distributed object management\|fact... | In the last few years, car-like robots beca |
| Security in Buildin... | IEC standards\|safety\|safety-critical software\|struc... | Building automation systems are tradition |
| Impact of user h... | HVAC\|energy consumption\|home automation\|predic... | Lifestyle and habits of users have a direct |
| Multicast commun... | ad hoc networks\|building management systems\|ho... | With an ever increasing performance, wir |
| Shortening of pro... | Internet\|Java\|building management systems\|distrib... | The manufacturing industry is still struggli |
| Integration of he... | building management systems\|internetworking\|onto... | The challenge of integrating heterogeneo |
| An ISA88 Phase i... | IEC standards\|flexible manufacturing systems\|prod... | Evolvability is one of the most desirable n |
| A comparison of ... | Web services\|XML\|electronic data interchange\|opti... | This paper analyzes different Web service |
| A Transparent IP... | IP networks\|Internet of Things\|building manageme... | The future Internet of Things (IoT) should |
| Connecting EIB t... | Java\|microcomputer applications\|system buses\|BC... | This article deals with a Linux low-level ha |
| A fault-tolerant b... | IP networks\|Zigbee\|building management systems\|... | IEEE 802.15.4 is a well-accepted standard |
| Accessing KNX ne... | Web services\|building management systems\|busine... | In the last years, the desire to access bui |
| On the security o... | IP networks\|Internet\|access control\|air conditionin... | The traditional areas of building automatic |
| Secure control ap... | Internet\|object-oriented programming\|traffic engin... | When security-critical applications are cor |
| Linux in factory a... | traffic engineering computing\|LoM2HiS framework\|S... | Linux is a freely distributed, 32-bit, pre-er |
| Cognitive decisio... | cognitive systems\|humanoid robots\|legged locomoti... | The novel approach to use meta-psycholo |
| Event-Related Br... | mobile robots\|motion control\|position control\|predic... | In an ERP study, German sentences were |
| Embedded Real-T... | edge detection\|embedded systems\|legged locomoti... | Estimation of objects in a 3D space is a fu |
| Single ion trappe... | fluorescence\|mirrors\|particle traps\|ytterbium\|Yb\|de... | We report on trapping of single Ytterbium |
| Simulating single-... | mirrors\|optical resonators\|photoexcitation\|absorpti... | The absorption of a single photon by a sir |
| Impact of RDL po... | cracks\|finite element analysis\|flip-chip devices\|integ... | For WLP (Wafer Level Packaging) thin film |
| Full solid angle io... | acoustic resonators\|bulk acoustic wave devices\|diel... | We present an optical system covering 81 |

Figure 6.2: Extracted publications data



| competency | subcompetency | trustcomp | pubid | strengthcomp |
|---|---|---|---|---|
| PROFESSIONAL COMPETENCY: InformationSystems | PROFESSIONAL SUBCOMPETENCY: DataManagement | low | 1 | intermediate |
| PROFESSIONAL COMPETENCY: KnowledgeRepres... | PROFESSIONAL SUBCOMPETENCY: ConstraintRepresentation | low | 1 | beginner |
| PROFESSIONAL COMPETENCY: OperatingSystems | PROFESSIONAL SUBCOMPETENCY: BSD | low | 1 | intermediate |
| PROFESSIONAL COMPETENCY: AutomationSystems | PROFESSIONAL SUBCOMPETENCY: DistributedAutomationSys... | low | 1 | expert |
| PROFESSIONAL COMPETENCY: BuildingAutomation | PROFESSIONAL SUBCOMPETENCY: BuildingAutomationSecurity | low | 1 | expert |
| PROFESSIONAL COMPETENCY: Factory_Automation | PROFESSIONAL SUBCOMPETENCY: Fieldbus | low | 1 | expert |
| PROFESSIONAL COMPETENCY: HomeAutomation | PROFESSIONAL SUBCOMPETENCY: Agent_based_control | low | 1 | expert |
| PROFESSIONAL COMPETENCY: BusinessManagem... | PROFESSIONAL SUBCOMPETENCY: Accounting | medium | 2 | beginner |
| PROFESSIONAL COMPETENCY: DataManagement | PROFESSIONAL SUBCOMPETENCY: DataMining | medium | 2 | beginner |
| PROFESSIONAL COMPETENCY: Management | PROFESSIONAL SUBCOMPETENCY: CustomerRelationshipMan... | medium | 2 | beginner |
| PROFESSIONAL COMPETENCY: InformationSystems | PROFESSIONAL SUBCOMPETENCY: DataManagement | medium | 2 | intermediate |
| PROFESSIONAL COMPETENCY: BuildingAutomation | PROFESSIONAL SUBCOMPETENCY: BuildingAutomationSecurity | medium | 2 | expert |
| PROFESSIONAL COMPETENCY: Factory_Automation | PROFESSIONAL SUBCOMPETENCY: Fieldbus | medium | 2 | expert |
| PROFESSIONAL COMPETENCY: HomeAutomation | PROFESSIONAL SUBCOMPETENCY: Agent_based_control | medium | 2 | expert |

Figure 6.3: Calculated competencies



From the data shown above we can conclude that there is a significant amount of extracted professional competencies of University staff members. There are some factors which had an impact on results:

- Some people do not publish all of their publications on digital libraries which causes number of publications and competencies to decrease.
- Some people have same name and surname and search on digital libraries lists all publications related to that name even in the case when those are two different persons. This causes the software to have some wrong publications data in database.
- If some people lists publications on their Website and when we take that data into consideration together with the publications data from digital libraries then we could increase the resulting number of publications and competencies.

## 6.2 Evaluation of Results

### 6.2.1 Survey

The evaluation is done by comparing results with data obtained from the survey. The survey was created using Google forms and it contained just one question where participants were asked if they have professional competences and subcompetences in given fields(if yes, they were asked to select all competences which they believe to posses). The participants were contacted by e-mail where they were given a link to the Google form which contained a survey. At the beginning of the survey the participants were given a short introduction about what is survey about and how is data obtained which is presented in the survey and what was the use case so that they know why they participate in this survey. Provided answers were in the multiple choice form where participants could select one or many answers. The answer contained professional competency in some field and its subcompetency. The provided answers were the extracted competencies stored in the database.

| name | competency | subcompetency |
|---|---|---|
| Wolfgang Kastner | Automation Systems | Distributed Automation Systems |
| Wolfgang Kastner | Building Automation | Building Automation Security |
| Wolfgang Kastner | Computer Aided Automation | Automation Systems |
| Wolfgang Kastner | Distributed Automation Systems | Factory Automation |
| Wolfgang Kastner | Home Automation | Agent Based Control |

Table 6.1: First participant survey results

| name | competency | subcompetency | trust | strength |
|---|---|---|---|---|
| Wolfgang Kastner | Automation Systems | Distributed Automation Systems | low | expert |
| Wolfgang Kastner | Building Automation | Building Automation Security | low | expert |
| Wolfgang Kastner | Computer Aided Automation | Automation Systems | medium | intermediate |
| Wolfgang Kastner | Distributed Automation Systems | Factory Automation | medium | intermediate |
| Wolfgang Kastner | Home Automation | Agent Based Control | medium | expert |
| Wolfgang Kastner | Computer Vision | Document Analysis | low | beginner |
| Wolfgang Kastner | Smart home control | Clustering algorithms | low | beginner |

Table 6.2: Extracted competencies of the first participant from the database with competencies trust and strength levels

| name | competency | subcompetency |
|---|---|---|
| Robert Sablatnig | Computer Vision | Document Analysis |
| Robert Sablatnig | Document Analysis | Document Reconstruction |
| Robert Sablatnig | Image Processing | Image Color Analysis |
| Robert Sablatnig | Text Recognition | Handwriting Recognition |
| Robert Sablatnig | Layout Analysis | Determining Regions |
| Robert Sablatnig | Stereo Vision | Stereo Evaluation |
| Robert Sablatnig | Segmentation | Binarization |
| Robert Sablatnig | Video Analysis | Motion Detection |
| Robert Sablatnig | Writer Identification | Writer Classification |

Table 6.3: Second participant survey results

| name | competency | subcompetency | trust | strength |
|---|---|---|---|---|
| Robert Sablatnig | Computer Vision | Document Analysis | medium | intermediate |
| Robert Sablatnig | Document Analysis | Document Reconstruction | low | intermediate |
| Robert Sablatnig | Image Processing | Image Color Analysis | low | expert |
| Robert Sablatnig | Text Recognition | Handwriting Recognition | low | expert |
| Robert Sablatnig | Layout Analysis | Determining Regions | low | beginner |
| Robert Sablatnig | Stereo Vision | Stereo Evaluation | low | beginner |
| Robert Sablatnig | Segmentation | Binarization | low | intermediate |

Table 6.4: Extracted competencies of the second participant from the database with competencies trust and strength levels

### 6.2.2 Discussion

Table 6.1 contains a list of competencies and subcompetencies obtained from the survey for the first participant. Table 6.2 contains a list of extracted competencies and subcompetencies with their strength and trust levels for the first participant of the survey. When we compare competencies from Tables 6.1 and 6.2 we can see that we have a 100% match of the competencies obtained in the survey with the extracted competencies from the database. We can see in Table 6.2 strength and trust levels of those competencies and they all have intermediate or expert strength levels. Competence trust levels in this case are not so relevant because they are related to particular publication and they measure if a person is a main author of the publication or not and they measure number of authors for that publication. Competence trust levels are useful when we consider just one publication and extracted competences for that particular publication.

The last two rows in Table 6.2 contains competencies with low trust and beginner strength levels which we cannot find in Table 6.1. Those competencies are taken from database in order to show that there is a possibility to get some competencies which are not so relevant for some particular author because competencies are determined by mapping publication title, keywords and abstract with the concepts in an ontology. Ontology has many concepts so it is very likely that we have a match which is not so relevant to domain in which we are interested. For that reason we have competence strength and trust levels in order to filter extracted competences.

For the second participant of the survey there is a 77% match of the competencies obtained in the survey with the extracted competencies from the database. For the six out of nine competencies shown in Table 6.3 there were intermediate and expert strength levels and the just seventh competence had a beginner strength level which means that participant published very few publications in that area. The last two competencies in Table 6.4 did not match with the extracted competencies for this participant from the software.

Tables 6.2 and 6.4 contains some of the extracted competencies(there are many more) from the database which are relevant for the comparison of the survey results.

## 6.3 Reuse of Competence Extractor

The Institute chosen for the software reuse is Institute of Software Technology and Interactive Systems at the Vienna University of Technology. The research group chosen from that Institute is Electronic Commerce Group. The Electronic Commerce Group performs research in Business-2-Business(B2B) and Business-2-Consumer(B2C) domains. In the B2B domain, the group focuses on:

- business (process) modeling, definition, specification and implementation of e-business systems, also contributing to semantic Web research and service oriented architectures

- ontology engineering and information integration

- "Web science" focusing on network analysis and content, also text mining

In the B2C domain, the group focuses on:

- research of visual interaction paradigms
- research on mobile applications

Overall, the Electronic Commerce Group focuses on development and reserach in the area of e-tourism and mobile applications [Ele15].

The software reuse process is done in the following steps:

- Research is done at the target Institute in order to define concepts for the ontology
- Ontology is extended with the concepts from the target Institute
- Software is executed with the ontology as an input

Software execution process is done following these steps:

1. First step is to open the target Institute Website and to run a bookmarklet which is used for the data extraction
2. Second step is to fill in the input data for the bookmarklet such as University name, Institute name
3. Third step is to run the SelectorGadget bookmarklet and select list of University staff members in order to obtain CSS selector of those elements
4. Then, there is a need to run the main bookmarklet again since there is no an option to click on a button to add the input data to the main bookmarklet since SelectorGadget is running and it selects that button on a click as an element. Since main bookmarklet is stored in a cookie, on the second run data is populated from the cookie
5. Then, Web page URL and ontology location are added
6. After all input data is filled in, Web service is called with the input parameter
7. Web service passes data to the program which opens a target Web page, then extracts the University staff list, then queries IEEEXplore and ACM digital libraries in order to extract publications data(title, keywords and abstract) of those University staff members
8. That publications data is mapped to the concepts in an ontology in order to get professional competences of University staff members

9. University staff list, publications data and professional competences are stored in the database and when all data is stored then the stored procedure is executed in order to calculate professional competence strength level

For 42 University staff members 224 publications were extracted and 1222 professional competencies were obtained.

## 6.4 Software limitations

Some of the software limitations are:

- Currently it is limited to two digital libraries: IEEEXplore and ACM from where the publications data is extracted
- The amount of data is limited to a number of publications which authors published on the mentioned digital libraries
- Some authors do not publish all their work on digital libraries so this certainly presents a limitation in this software
- This software does not extract data for former members of some target Institute
- It is limited to publications data on digital libraries, but there are cases when some authors have their own Web pages where they publish some of their work or they have a list of their publications on the Institute where they work

Overall, the data source is limited which could be extended in the sense to cover some other sources of publications data in order to have more accurate results. Former staff members could also be considered, but on some Web pages there is a list of former staff members and on some pages they are not listed which creates a challenge in order to find that data and relate it to the target Institute.

CHAPTER 7

# Conclusion and Future Work

## 7.1 Summary

This thesis presented an approach on how to extract data from Web pages of the Universities and digital libraries in order to get professional competences of University staff members.

Before the design and implementation phase of the software in this thesis the analysis and testing was done of already existing Web data extraction tools. This analysis was done in order to get an overview of the functionalites which they have and to see what are the challenges for those tools. The result of the analysis of those tools and data from scientific publications contributed to the design of the software in this thesis. The software was created in the form of a bookmarklet which was used to gather input data for the other part of the software. That input data was XPath of some elements on the Website and other data such as Ontology source. The other part of the software was dealing with extracting pulblication data from digital libraries and mapping that data to the concepts in an ontology in order to get professional competencies on University staff members which was a goal of this thesis.

In Chapter 4 the use case for this software was introduced and it presented some research areas with which that Institute is dealing with. The software was tested on this use case and the survey was made in order to check if the extracted professional competencies are correct. The results were pretty much accurate as it is shown in Chapter 6.

Although this software provided accurate results and was able to extract data without many issues it has some drawbacks like performance, some unnecessary data and the whole process of Web data extraction was not fully automatic.

## 7.2 Future work

The performance of this software could be improved by switching Firefox Web driver with Selenium HtmlUnit Web driver. It is significantly faster than Firefox Web driver because it is simulates a Web browser without a Graphical User Interface.

Extraction of some unnecessary data happens in the cases when there are some authors with the same name and surname on digital libraries. In that case when there is a search by name and surname at the digital libraries all publications related to that name will be listed even in the case when they are two different persons. What could be taken into consideration is a list of publications which in some cases can be found on the authors Web page. In that case the search on digital libraries can be done by publication title other than author name which leads us to more accurate results but the fact remains that not all authors publish their full list of publications on digital libraries.

Regarding the automatic process of Web data extraction the following cases need to be considered:

- When there is a case to make the software as a fully automatic process then there must be very accurate settings of that kind of software since there is a requirement to predict all factors which could influence the correct functioning of the software
- In the field of Web data extraction it is known that the challenge of unstructured, dynamic data remains so it is very challenging to make the whole process of Web data extraction fully automatic

Regarding the process of identification and measurement of competences this process could be improved. As it is already pointed out in this thesis there is a need to continuosly measure competences in order to have up-to-date data. For the identification process of competences there is also a requirement to have up-to-date data. Identification process of technical competences proposed in this thesis is based on the concepts in the Ontology for some domain. Over some period of time new research fields emerge in some domain so some new concepts need to be added in the Ontology in order to have a complete and up-to-date structure. As Ontology is an input to the software proposed in this thesis this can be manually done by adding new concepts in the Ontology. But some more efficient and automated process could be created in future work in order to decrease human involvement in the whole process.

# Listings

# List of Tables

# List of Figures

# Glossary

**Deep Web** content hidden behind the Web forms. 11

**Web wrappers** A procedure, that might implement one or many different classes of algorithms, which seeks and finds data required by a human user, extracting them from unstructured (or semi-structured) Web sources, and transforming them into structured data, merging and unifying this information for further processing, in a semi-automatic or fully automatic way.. 3

**content lines** a rectangular box enclosing a HTML text node. 21

**Levensthein distance** is a string metric for measuring the difference between two sequences. 40

# Acronyms

**CSS** Cascading Style Sheet. 8

**OWL** Web Ontology Language. 8

**AJAX** Asynchronous JavaScript Request. 11

**DIADEM** Domain-centric, Intelligent, and Automated Data Extraction Methodology. 20

**HTML** HyperText Markup Language. 1

**DOM** Document Object Model. 2

**XPath** XML Path Language. 2

**W4F** WysiWyg Web Wrapper Factory. 4

**SGWRAM** Schema-Guided WRApper Maintenance. 4

**DTD** Document Type Definition. 5

**MDR** Mining data region. 21

**ViNT** Visual information aNd Tag structure. 21

**ViPER** Visual Perception-based Extraction of Records. 22

**RDF** Resource Description Framework. 41

**SPARQL** SPARQL Protocol and RDF Query Language. 40

# Bibliography

[Aut15] Institute of computer aided automation, automation systems group. https://www.auto.tuwien.ac.at/, 2015. Accessed: 2015-03-09.

[BDF$^+$12] Philip Bohannon, Nilesh Dalvi, Yuval Filmus, Nori Jacoby, Sathiya Keerthi, and Alok Kirpal. Automatic web-scale information extraction. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, SIGMOD '12, pages 609–612, New York, NY, USA, 2012. ACM.

[CN09] Stefan TRAUSAN-MATU Cristina NICULESCU. An ontology-centered approach for designing an interactive competence management system for it companies. *Informatica Economica*, 13(4):159–167, 2009.

[Com15] Research areas, computer vision lab. http://www.caa.tuwien.ac.at/cvl/research-areas/, 2015. Accessed: 2015-03-17.

[DH09] Jürgen Dorn and Martin Hochmeister. Techscreen: Mining competencies in social software. In *The 13th World Multi-Conference on Systemics, Cybernetics and Informatics*, pages 115–126, Orlando, FLA, 2009. International Institute of Informatics and Systemics.

[DP07] Jürgen Dorn and Markus Pichlmair. A competence management system for universities. In Hubert Österle, Joachim Schelp, and Robert Winter, editors, *Proceedings of European Conference on Information Systems*, pages 759–770. Proceedings of European Conference on Information Systems, 2007. Vortrag: 15th European Conference on Information Systems (ECIS 2007), St. Gallen; 2007-06-07 – 2007-06-09.

[DPST08] Jürgen Dorn, Markus Pichlmair, K Schimper, and Hilda Tellioglu. Supporting competence management in software projects. In *Proceedings of International Conference on Concurrent Enterprising*, pages 451–458. ICE Proceedings, 2008.

[Ele15] Electronic commerce group. http://www.ec.tuwien.ac.at/ec/frontpage, 2015. Accessed: 2015-07-22.

[FF12] T. Fujino and N. Fukuta. A sparql query rewriting approach on heterogeneous ontologies with mapping reliability. In *Advanced Applied Informatics (IIAIAAI), 2012 IIAI International Conference on*, pages 230–235, Sept 2012.

[FMFB14] Emilio Ferrara, Pasquale De Meo, Giacomo Fiumara, and Robert Baumgartner. Web data extraction, applications and techniques: A survey. *Knowledge-Based Systems*, 70(0):301 – 323, 2014.

[Gru93] Thomas R. Gruber. A translation approach to portable ontology specifications. *Knowl. Acquis.*, 5(2):199–220, June 1993.

[GT13] Cenys A. Grigalis T. State-of-the-art web data extraction systems for online business intelligence. *Informacijos mokslai 2013 64*, 2013.

[Har04] M. Harzallah. Ontology of enterprise competencies. In *CAiSE'04 Workshops in connection with The 16th Conference on Advanced Information Systems Engineering, Riga, Latvia, 7-11 June, 2004, Knowledge and Model Driven Information Systems Engineering for Networked Organisations, Proceedings, Vol. 3*, pages 288–292, 2004.

[HER09] Maryam Hazman, Samhaa R. El&#45;Beltagy, and Ahmed Rafea. Ontology learning from domain specific web documents. *Int. J. Metadata Semant. Ontologies*, 4(1/2):24–33, May 2009.

[Hoc12] Martin Hochmeister. *Measuring User Expertise in Online Communities.* PhD thesis, Vienna University of Technology : Austria, 2012.

[Hon11] Jer Lang Hong. Data extraction for deep web using wordnet. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 41(6):854–868, Nov 2011.

[hZW11] Shi hai Zhu and Li Wang. Research on software undergraduates training countermeasures based on the competency model. In *Computer Science Education (ICCSE), 2011 6th International Conference on*, pages 804–807, Aug 2011.

[Imp15] What is import io? http://support.import.io/knowledgebase/articles/251955-what-is-import-io, 2015.

[Kim15] Using kimono labs to scrape the web. http://moz.com/blog/web-scraping-with-kimono-labs. http://moz.com/blog/web-scraping-with-kimono-labs, 2015. Accessed: 2015-03-08.

[Moz15a] Screen scraping, data scraping, data extraction software | mozenda. http://mozenda.com/Tour02-Web-Site-Data-Mining-User-Friendly-Interface/, 2015. Accessed: 2015-03-08.

[Moz15b] Screen scraping, data scraping, data extraction software | mozenda. http://mozenda.com/, 2015. Accessed: 2015-03-07.

[Nar13] N. Narwal. Improving web data extraction by noise removal. In *Communication and Computing (ARTCom 2013), Fifth International Conference on Advances in Recent Technologies in*, pages 388–395, Sept 2013.

[PB08] Thomas Eigner Paul Buitelaar. Topic extraction from scientific literature for competency management. In *Personal Identification and Collaborations: Knowledge Mediation and Extraction (PICKME2008)*, October 2008.

[Pic08] Markus Pichlmair. *Universitäres Kompetenzmanagementsystem - Entwicklung und Evaluierung eines Prototypen.* PhD thesis, Vienna University of Technology : Austria, 2008.

[Sel15] Selectorgadget:point and click css selectors. http://selectorgadget.com/, 2015. Accessed: 2015-03-09.

[Sha10] Richard J. Shavelson. On the measurement of competency. *Empirical Research in Vocational Education and Training*, 2(1):41–63, 2010.

[Sic06] Miguel-Angel Sicilia. Ontology-based competency management: infrastructures for the knowledge intensive learning organization. In *Ontology-based competency management: infrastructures for the knowledge intensive learning organization*, Intelligent learning infrastructure for knowledge intensive organizations : a semantic web perspective, ISBN:1591405033, pages 302–324. Information Science Publ., 2006.

[SL05] Kai Simon and Georg Lausen. Viper: Augmenting automatic information extraction with visual perceptions. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, pages 381–388, New York, NY, USA, 2005. ACM.

[SMN06] Hicham Snoussi, Laurent Magnin, and Jian-Yun Nie. Heterogeneous web data extraction using ontology. In *Agent-Oriented Information Systems*, 2006.

[SWL09] Weifeng Su, Jiying Wang, and Frederick H. Lochovsky. Ode: Ontology-assisted data extraction. *ACM Trans. Database Syst.*, 34(2):12:1–12:35, July 2009.

[TSH06] V. Tarassov, K. Sandkuhl, and B. Henoch. Using ontologies for representation of individual and enterprise competence models. In *Research, Innovation and Vision for the Future, 2006 International Conference on*, pages 206–213, Feb 2006.

[VI08] Alexandru Cicortas Victoria Iordan. Ontologies used for competence management. *Acta Polytechnica Hungarica*, 5(2), 2008.

[Vis15] Visual web ripper review. http://scraping.pro/visual-web-ripper-review/, 2015. Accessed: 2015-03-07.

[Web10] Webharvest: Easy web scraping from java. http://masochismtango.com/2010/02/15/webharvest-web-scraping-from-java/, 2010. Accessed: 2015-03-07.

[Web15] Web-harvest project home page. http://web-harvest.sourceforge.net/, 2015. Accessed: 2015-03-07.

[ZMW+05] Hongkun Zhao, Weiyi Meng, Zonghuan Wu, Vijay Raghavan, and Clement Yu. Fully automatic wrapper generation for search engines. In *Proceedings of the 14th International Conference on World Wide Web*, WWW '05, pages 66–75, New York, NY, USA, 2005. ACM.

[ZNF07] A. Zouaq, R. Nkambou, and C. Frasson. Using a competence model to aggregate learning knowledge objects. In *Advanced Learning Technologies, 2007. ICALT 2007. Seventh IEEE International Conference on*, pages 836–840, July 2007.