A Method for Automatic Generation of Metadata
A Method for Automatic Generation of Metadata

Examensarbete inom information- och programvarusystem, grundnivå - kandidat.
Kurs II121X, 15hp
Degree Project in Information and Software Systems, First Level- Bachelor.
Course II121X, 15hp
Stockholm, Sweden 2014

Submission date:
9th October 2014

Author:
Menatalla Ashraf Fawzy Kamel mafk@kth.se

Supervisor and Examiner:
Anne Håkansson
Abstract

The thesis introduces a study about the different ways of generating metadata and implementing them in web pages. Metadata are often called data about data. In web pages, metadata holds the information that might include keywords, a description, author, and other information that helps the user to describe and explain an information resource in order to use, manage and retrieve data easily. Since web pages depend significantly on metadata to increase the traffic in search engines, studying the different methods of generation of metadata is an important issue. Generation of metadata can be made both manually and automatically. The aim of the research is to show the results of applying different methods including a new proposed method of generating automatic metadata using a qualitative study. The goal of the research is to show the enhancement achieved by applying the new proposed method of generating metadata automatically that are implemented in web pages.

Keywords: Metadata, generation, automatic, manual, method, algorithm.
Sammanfattning


Nyckelord: Metadata, generationen, automatisk, manuell, effektiv, metod, algoritm.
# Table of Contents

Abstract ........................................................................................................................................... 2  
Sammanfattning ................................................................................................................................. 3  
1. Introduction .................................................................................................................................. 6  
   1.1 Background .............................................................................................................................. 6  
   1.2 Problem statement .................................................................................................................. 7  
   1.3 Purpose .................................................................................................................................... 7  
   1.4 Goal .......................................................................................................................................... 7  
   1.5 Ethics ....................................................................................................................................... 8  
   1.6 Sustainability .......................................................................................................................... 8  
   1.7 Method .................................................................................................................................... 8  
   1.8 Delimitations .......................................................................................................................... 9  
   1.9 Outline of the thesis ................................................................................................................ 10  
2. Theory description of Metadata .................................................................................................. 11  
   2.1 Metadata and Meta-tags ....................................................................................................... 11  
   2.2 Semantic Web ....................................................................................................................... 12  
3. Method ......................................................................................................................................... 14  
   3.1 Methodology .......................................................................................................................... 14  
   3.2 Design Science ...................................................................................................................... 16  
4. Two ways of generation of Metadata ....................................................................................... 17  
   4.1 Manual generation .................................................................................................................. 17  
   4.2 Automatic generation ............................................................................................................ 17  
5. The need for a new automatic creation ...................................................................................... 19  
6. The proposed automatic method of metadata generation ....................................................... 21  
7. Results .......................................................................................................................................... 27  
   7.1 Webpage 1 ............................................................................................................................... 27  
      7.1.1 Discussion of webpage 1 .................................................................................................. 29  
   7.2 Webpage 2 ............................................................................................................................... 31  
      7.2.1 Discussion of webpage 2 .................................................................................................. 33  
   7.3 Evaluation ............................................................................................................................... 36  
8. Conclusion and future work ........................................................................................................... 37  
   8.1 Conclusion ............................................................................................................................... 37  
   8.2 Future work ............................................................................................................................. 38
List of Figures

Figure 1 - Metadata existing and added DC elements in the proposed methods ............20

Figure 2 - Shows the steps used in the algorithm to extract metadata from web pages.......23

List of Tables

Table 1 - Shows the metadata collected from the proposed method.......................28

Table 2 - Shows the difference in the metadata collected in the current web page, Klarity, DC.dot and the new method in the web page cprogramming.com.................................30

Table 3 - Shows the collected analysis of metadata using the proposed method..........32

Table 4 - Shows the difference in the metadata collected in the current web page, Klarity, DC.dot and the new method in the web page www.tutorialspoint.com/operating_system....35
1. Introduction

Metadata is data that describes other data. Metadata makes it possible to describe, locate and manage information resources [1].

The use of metadata is an important feature found in web pages in order to increase the chances of the search engines to find related pages, which is also known as Search Engine Optimization (SEO) [2]. A search engine can be software or a program that enables users to search the Internet using keywords. There are few steps needed in order to find the most relevant pages according to the words provided in the web search engine. First, the web search tool automatically visits websites using the metadata [2]. Secondly, the search engine automatically shows the websites related and stores them into a database that generates the results based on users’ search criteria [2]. Thirdly, indexing of web pages by which data is stored in a database to be accessed later in future queries [2]. The last step is searching, which makes the search engine look up the indexes that are stored in the database in order to find the best-matching web pages according to the based users’ criteria [2]. Since the correct use of metadata would result into a better result of finding the relevant pages according to the users’ use of keyword in the search engine, the study of the implementation of metadata is a necessary.

1.1 Background

The use of metadata is included in the World Wide Web Consortium’s (W3C) accessibility recommendation as W3C makes documents easier to find and use, which helps the user to describe, explain, and locate an information resource in order to use, manage and retrieve data easily [3]. The W3C is a set of standards, recommendations and protocols that ensures the long-term growth of the Web [3].

A clear description of the World Wide Web (WWW) is needed in order to further discuss the implementation of metadata and how metadata are applied in web pages. The WWW was invented in 1989 by Tim Berners-Lee [4]. Tim proposed three fundamental technologies that are the main foundations of the Web. One of these technologies is the HyperText Markup Language (HTML), which is the publishing format for the Web, including the ability to format documents and link to other documents and resources [4]. Metadata are usually found in the HTML headers of a webpage in the form of meta-tags [1]. The second technology is the Uniform Resource Identifier (URL), which is an address that is unique to each resource on the web [1]. The third technology Tim had proposed was the Hypertext Transfer Protocol
(HTTP), which allows retrieving linked resources across the web [1]. Metadata are stored in the form of meta-tags in web pages. Meta-tags are tags that are found between the open and the closing head tags in the HTML code of a webpage [48]. Storing meta-tags along with the object makes it easier to find it, eliminates problems with linking data with metadata. Also this storage ensures that both the data and metadata will be updated parallel together [1].

Metadata can be generated either manually or using an automatically information processor that will be discussed later [5].

1.2 Problem statement

Automatic generation of metadata must be done cautiously, since the generation by people who are not familiar how they are indexed and catalogued may cause quality problem [1]. The automatic generation of meta-tags by using editing tools can correspond to a less accurate and comprehensive descriptions of the document [6]. Another problem associated with metadata generation is that it does not always provide a readable form, there can be different formats used to express the same element (i.e. DC.Creator or Creator) according to the standard metadata follows [6]. The main problem that needs to be addressed is that metadata generation is sparse and therefore, a study of the different methods of automatic generation of metadata is needed to clarify which is the most enhancing generation way on web pages.

1.3 Purpose

The main purpose of the research is to present the various ways of generation of metadata and their way to retrieve metadata from web pages. The thesis presents a proposed method to automatically generate metadata using a simple algorithm that will be described. Also, the thesis investigates the differences between existing methods and the proposed method by analyzing the metadata collected from both methods.

1.4 Goal

The goal of the research is to show a proposed method for the automatic generation of metadata from web pages. Also, the goal is to clarify the importance of using the metadata
correctly in HTML code and understanding the usage of metadata in web pages. Moreover, the goal is to indicate an improvement of retrieval of automatic metadata using a more detailed extraction method. This proposed method will give an enhancement of the currently existing methods.

1.5 Ethics
Ethics is an important issue that needs to be taken into consideration when looking into generation of metadata. Creators of metadata can misuse the benefits of metadata and could enter improper words so that their website would appear in search engines, since metadata depends mainly on what the authors’ sets in the HTML code. Metadata must be treated properly to avoid misuse of it by youth by using improper words, which is against ethics. When creating a proposed method for the automatic generation of metadata, there will be ethical issues that need to be taken into account to how this generation will affect the involvement in the management of information found in the web pages. The proposed method needs to carefully use the information found in the web pages without intruding the creators’ privacy.

1.6 Sustainability
Consideration of economic and sustainability problems will be taken into account, since one of the main importance’s of metadata is that it helps websites appear in a better search ranking position [7] therefore a company can find a website they are searching for in a simpler way if the metadata are set correctly. Finding the web page easier will reduce both time and effort to the users. Using the proper metadata can help market the company internationally by using a language metadata that will be explained later.

1.7 Method
To perform the work, a literature review is needed to analysis the present situation to be performed. After comes an analysis of the collected information to be carried as well as a reflection on the problem area.

The two main basic research methods are quantitative research method and qualitative research method [8]. The quantitative research method uses experiments and large data sets to
reach a conclusion [8] while the qualitative research method uses investigations in an interpretative manner to create theories [8]. The thesis applies the qualitative method because the paper helps to understand how important the generation of metadata is. Also, it shows the importance of the relation between the generations of metadata in HTML code with the appearance of websites in the search engines. Qualitative research method answers to questions, such as why, how and what [8].

Research approaches are needed to draw conclusion and to be clear whether these conclusions are true or false [8]. There are several methods that can be used, such as inductive, deductive or abductive approach, which includes both the latter named methods [8]. Inductive approach is when the research resonate theories based on experiences and opinions [8]. While, deductive approach resonate theories to verify or falsify hypotheses [8]. Inductive approach is often used when data is collected with qualitative methods [8].

The thesis will apply inductive approach since there is no clear theory to verify or falsify hypotheses which applies in a deductive approach [8]. However, a deep analysis is needed to gain an understanding of the different methods of generation of metadata and how they are applied to extract metadata. Also, since the outcome is based on experiencing and on the behavior [8] of web pages appearance in search engines therefore an inductive approach is the most appropriate approach.

1.8 Delimitations

The work is limited to show the few different ways of automatic generation of metadata, due to presence of few methods of generation in web pages. Also, due to limited automatic methods of generation of metadata, therefore, the work will introduce a proposed method to evaluate these available methods.

The extraction of the *date* metadata was not an easy task, since some web pages include several dates; therefore it was hard to extract the date of the production of the web page. Also, some web pages include dates as historical significance for example, a web page showing when countries statues were made and therefore will cause confusion of the presence of several dates.
1.9 Outline of the thesis

Chapter 2 presents the reader a history and a wide background of metadata and meta-tags with a broad definition of the Dublin core element. Also, gives a detailed definition of the Semantic Web.

Chapter 3 presents the methods and methodology used in performing the work. Also, gives the reader a more insight of which methods are used in this research and why they are preferable methods. The chapter gives more details of how data is collected and how the analysis of metadata is performed.

Chapter 4 gives an introduction on the type of generation methods, both manually and automatic generation of metadata in web pages. Also, introduces the two methods of automatically generation of metadata.

Chapter 5 presents the need for a new automatic generation of metadata method.

Chapter 6 presents the proposed method in details with illustrations of an algorithm that will be used to extract metadata.

Chapter 7 presents the algorithm applied on web pages and showing the collected metadata using the proposed method. Also it shows the results and differences of the different methods of extraction. Then an evaluation of the proposed method is made.

Chapter 8 presents a conclusion of the importance of metadata and discussion. Also stating suggestions on future work will be provided.

Chapter 9 presents the references used in the performance of the work.
2. Theory description of Metadata

This chapter presents the theories used in the thesis and begins with the definition of Metadata and Meta-tags. It also follows a comparison between the manually and automatically generation of metadata and a definition of the Semantic Web.

2.1 Metadata and Meta-tags

Metadata is data that describes other data [5]. Filtering through metadata can make locating a particular document easier for a user searching for a specific related web page [5]. For example, author, date created and date modified and file size are examples of very basic document metadata [5].

Other than documents, metadata can be used in images, videos, spreadsheets and web pages [5]. The importance of metadata on web pages is significant. In web pages, metadata is expressed in a form called Meta-tags, which contains information about the webpage main contents. These meta-tags are evaluated by search engines to make the web pages that are most relevant to the users search visible in the search results, determining whether a user would enter a webpage or not. The accuracy of meta-tags keyword is important, since it determines the chances of appearance of the web pages in the search engines [5].

The tag <META/> carries information about the webpage without being visible in the browser [9]. This tag is accessible to search engines and other applications that read HTML code in order for web pages to appear in search results [9]. HTML element is used to carry metadata for a webpage [9]. As mentioned earlier, HTML is the publishing format for the Web, including the ability to format documents and link to other documents and resources [4].

One of the most famous metadata standards is The Dublin Core Metadata Element Set (DC), which is a vocabulary of fifteen properties for use in resource description [10]. It was developed to help describe web-based documents in a simpler and concise way [1]. The DC element set involves Title, Creator, Subject, Description, Publisher, Contributor, Date, Type, Format, Identifier, Source, Language, Relation, Coverage, and Rights [1]. A resource is any information that has an identity such as a document, webpage, image or spreadsheet.

Title is the name given to the resource, which is visible on the top of the documents and web pages [11]. The creator, the publisher and the contributor may include a person, an organization or a service. However, the creator is who is responsible for making the resource,
while the publisher is who made the resources available and the contributor is who contributed to the resource [11]. Description is what describes the resource; it might include an abstract, a table of contents, or a graphical representation of the resource [11]. Date is the period of time when the resource was created [11]. Type involves the nature or genre of the resource [11]. Format is what describes the file format, physical medium, or dimensions of the resource [11]. Examples of dimensions can include size and duration [11]. Identifier is what identifies the resource, for example, the International Standard Book Number (ISBN), which identifies a book [11]. Language states the language the resource is provided in [11]. Relation describes a related resource [11]. Rights include the information that contains property rights associated with the resource, which might include intellectual property rights [11]. Source is a related resource from which the described resource is derived [11]. Subject includes the information regarding the topic of the resource [11], which includes keywords or key phrases [11]. Coverage is related to how the temporary topic of the resource is applied and how relevant and covered the resource is compared to the overall topic [11].

There are three main types of metadata: [1] Descriptive Metadata, which describes resources for the purpose of identification and discovery such as title, abstract, authors and keywords [1]. Descriptive metadata are commonly used to create web pages, which would include information as keywords, description and language [12]. Secondly, Structural Metadata describes an object’s structure, which involves how the pages are ordered and how the objects are put together [1]. They are used to locate a resource by navigating using indexes [12]. Finally, Administrative metadata that shows information about the file, when it was created, file types and technical information [1]. They can be used to record both short and long-term information about data collections [12].

2.2 Semantic Web

To understand the Resource Description Framework (RDF), which is the infrastructure that makes it possible to encode and reuse structured Metadata [13], a brief description of Semantic Web must be given first:

According to Berners-Lee et al 2001 “The Semantic Web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation” [14].
The Semantic Web is a web of data [15]. It consists of dates, titles and any other data that can be imagined [15]. The collection of Semantic Web technologies are Resource Description Framework (RDF), Web Ontology Language (OWL), Simple Knowledge Organization System (SKOS), etcetera [15]. These technologies are needed to provide an environment and a model for data interchange [15].

RDF is a W3C recommendation designed to standardize the definition and use of metadata [16], since it is a language for representing information on the web [17] and adding semantics to a document [13].

On top of RDF is built OWL [18]. OWL is a Semantic Web Language designed that became a W3C recommendation in February 2004 [18], which represents rich and complex knowledge about things, groups of things, and relations between things [19]. Although OWL and RDF are almost the same, OWL is a stronger language since it includes a larger vocabulary and stronger syntax than RDF [18].

OWL is an ontology language [20]. Ontology is the exact description of things such as web information and relationships between web information [18]. The third technology, SKOS is also a W3C standard [20]. SKOS is an ontology created in OWL to provide a standard way to represent knowledge organization systems such as controlled vocabularies, taxonomies and thesauri [20]. Controlled vocabularies are the list of words that an organization agrees upon while the taxonomy is the controlled vocabulary but organized in a hierarchy and finally, thesauri are taxonomies but with an explanation about each concept [20]. SKOS is based on both semantic web technologies OWL and RDF [20].

Concluding, semantic web depends on the ontologies and technologies mentioned above, in order to structure the data in a comprehensive and transportable machine understanding [21].
3. Method

This chapter represents methods and the methodology of the research, data collection, selection, application and evaluation of the data collection method.

The work with this thesis can be divided into two parts; the first was reading literature regarding the Metadata, Semantic Web, and other related subjects. The literature also included reading articles and journals. This was needed to get a better understanding of the concept and importance of metadata. The second part was to deeply analyze the automated generation of metadata. It will also introduce the concept of design science needed to design the proposed method.

3.1 Methodology

The method for the work, presented in this thesis, is based on a qualitative method of research. Additionally, an inductive research approach was used.

Qualitative research method [8] allows better understanding of metadata creation and to find out more about the implementations and extraction of metadata unlike the quantitative method, which is based on numbers and statistics [8].

Research methods are applied to help the process of conducting research [8]. A conceptual research method is applied since; it is used in interpreting existing concepts [8]. Also, conceptual method concerns literature reviews and theory development that can be used to establish concepts in an area [8]. The thesis presents a study of the literature to analyze and interpret used concepts therefore the conceptual method is the most suitable for this study.

The inductive approach was preferred over the deductive approach in the thesis since; there was no hypothesis of the outcome of the proposed method to extract metadata automatically. Also the thesis was approached with a qualitative study to establish why metadata is important and to show the different methods used to generate metadata for web pages. Therefore an inductive approach is the most suitable to use in the thesis.

Since the research will study the different methods of extracting metadata, both manual and automatic. Therefore, a Qualitative Comparative Analysis (QCA) will be used. QCA offers a new, systematic way of studying configurations of cases [30]. QCA is normally used when using case-study research methods [30].
Data collection methods vary according to which research method is used and the reason behind collecting the data [8].

There are several data collection methods such as experiments, questionnaires, case studies, document reviews and observations that are usually used in quantitative research method [8]. Those data collection methods also exist in qualitative research method except collecting data through experiments which only exists in quantitative research method [8].

Qualitative research is done by studying language and text interpretations and collecting data from books [8].

The data collection method for carrying out the study was document review and case studies. Document review is a way to collect data by reviewing existing documents [31]. These documents can be found either as a soft copy or electronic [31]. The reason for choosing document review is because when analyzing metadata it is needed to gather background information by reviewing existing documents to help understand the concept and the application related to the topic [31]. Also it is used when information is needed to develop other opinions for evaluation [31]. This can be done by reviewing existing documents to better understand how the automatic generation methods are used. Case studies were used, since there was information that was gathered from multiple sources to prove a particular phenomenon [8]. Many documents and information were gathered in the research to study the different methods and there pros and cons.

The data analysis methods are used to analyze the collected material [8]. Since the thesis applies qualitative research, then the most commonly used data analysis method are Coding, Analytic Induction, Grounded Theory, Narrative Analysis, Hermeneutic, and Semiotic [8].

The most suited method that was used in the thesis was both analytic induction and grounded theory as they are iterative methods [8]. The iterations involve reading existing documents and trying to compare the different techniques used for metadata generation to validate a theory of the effectiveness of generating metadata. Other methods, such as coding are not used in the thesis as this approach is used when data is collected through interviews and surveys. Also, coding is used when the data is summarized by the notes taken while interviewing [8].

Quality assurance is the validation and verification of the research material [8]. Since the research applies a qualitative research, with an inductive approach, therefore it must apply and discuss dependability, validity, confirmability, transferability and ethics [32].
Dependability, which responds to reliability [8], is a process for judging how clear and concise the outcome is [8; 33]. This is applied in the research as the document review is described to make it easy for the reader to repeat the procedures in order to have a similar outcome given the same time parameters. One of the other features that must apply is validity [43]. Validity is a way to assure that the research has been conducted according to existing rules [8]. Validity is applied to the research, by enabling respondents to confirm the results of the method described by applying the same steps and getting the same results. Transferability is to create a rich and deep description of the research to enable others researchers’ to easily repeat the project [8]. Transferability is applied in the research, by writing a detailed chapter 5 of how the proposed method will be applied to extract metadata automatically, by following a simple algorithm.

The last feature that will be applied in the research is ethics, which describes the moral principles in planning and conducting results of a research [8]. Ethics also covers treating the material with confidentiality [8]. Ethics will be covered in the research, by using the automatic method to create metadata; confidentiality in the retrieval of metadata of web pages without manipulating the content of the web pages.

3.2 Design Science

Design science is a way to solve problems by introducing new artifacts into the environment. The need for these artifacts will be used by humans to help them solve practical problems [46]. Scientific methods help to structure the work to create new knowledge [47]. By using these methods, it is easier to openly and critically discuss the research findings [47]. Design Science can be used to produce an artifact such as a program, algorithm, method, architecture, system and process model [47]. Since design science tries to solve practical problems that surround the society, the method used to present the proposed method for automatic generation of metadata is design science [47]. The solutions needed to be solved are the problems in terms of time and effort wasted in the manual generation of metadata [28]. There are several steps in the design science process. **First step** is to identify the problem [47]. **Secondly** is to define the goal and requirement of your artifact [47]. **Thirdly** is to design the method [47]. **Fourthly** is to try the method [47]. **The fifth step** is to evaluate the artifact [47]. These five design science process steps were used to easily produce the proposed method and evaluate the outcome of the method and will be discussed in details in chapter 6.
4. Two ways of generation of Metadata

This chapter describes the two ways of generation of metadata. Metadata can be either created manually or automatically by automatic tools. Also it shows the two methods that are extracted automatically and use the Dublin core elements.

4.1 Manual generation

Manual creation is when the web designers inputs any information they feel is relevant to help describe the documents, which tends to be more accurate [5] but can be manipulated by the author of the web pages in order to increase the chances of appearance of the web page in the search results as the most appealing web page [28].

Another possible problem with authors creating their own meta-tags is the consistency and reliability of the keywords written as metadata keywords in HTML code [28]. Since they maybe are not professionally trained, they are unlikely aware of how to follow the standards in order to make the process generation easier [28].

This inefficiency of the manual generation in terms of time and expense [28] leads to the automatic metadata generation being more appealing.

4.2 Automatic generation

Automatic metadata generation is the other way for generating metadata in web pages.

It is a machine process of metadata extraction and metadata harvesting [22].

Metadata harvesting and extraction are two automatic techniques for the generation of metadata [23]. Metadata extraction involves extracting metadata from a resource’s content that is displayed on a web browser [23]. Extraction is specified to web pages, where structured metadata is extracted from the “body” section of the HTML code [23]. Metadata extraction method uses automatic indexing techniques to search and obtain resource content and produce structured metadata according to metadata standards, which proves to become less expensive and more effective than manual indexing [24].

Metadata harvesting is when meta-tags are automatically collected from the “header” source code of an HTML code [23; 22], in other words, they depend on previously defined metadata [25]. The harvesting process also depends on the metadata produced by humans or by
automatic processes [25]. Automatic indexing is preferred over manual indexing as the same algorithm is used on each document, and thus, produces unbiased results or consistency problems [20].

The automatic generator might be a better solution to avoid the human error of manually creating metadata [23].

Two examples of generators for automatic generation of metadata are the tools Klarity [26] and DC.dot [27; 23]. These are extraction tools that are used to generate Dublin Core metadata with the 15 elements named earlier [20].

The Klarity generator tool [26] uses mainly the extraction generation technique [23]. The method is generated by submitting a URL, and then a Dublin Core metadata is automatically generated [23]. Afterwards the metadata is converted into HTML meta-tags. Klarity generates metadata for five elements identifier, “title”, “concepts”, “keywords”, “description”. DC.dot is an automatic generation tool that uses mostly the harvesting techniques [23]. DC.dot generates the same metadata as Klarity with an addition to “type,” “format” and “date” metadata [23]. Both tools work by accepting a URL as input and produces metadata elements than can then be manually edited for greater accuracy [23].
5. The need for a new automatic creation

This chapter shows the need for a proposed method which will include an addition to language, rights, creator to the metadata. Also illustrates the metadata collected from the two methods stated previously Klarity and DC.dot.

There are two techniques, metadata extraction and metadata harvesting, which are vital to metadata generation functions, but the effectiveness of these techniques for creating Dublin Core metadata are questionable [23]. Klarity and DC.dot are two of the methods used to collect metadata and use both extraction and harvesting techniques in their concepts [23]. Both Klarity and DC.dot are methods, which generate metadata that include: identifier, title, concepts, keywords, description, type, format and date [23].

Google stated in September 2009 [34] that neither meta-descriptions nor meta-keywords are used in their ranking algorithms for web search [34]. Some web page authors believe that search engines use meta-keywords; therefore it is better to add them in the DC metadata [35].

The proposed method in the following chapter will include an addition to language, rights, creator to the metadata.

In this new method, almost all the Dublin Core is included in the metadata. DC is considered one of the best known standards for creating metadata [36]. Including most of DC elements are an advantage to make more accurate metadata output to the user of the web page. The authors of the web page can easily place the metadata generated in their HTML code.

Adding the author/creator, which includes the name of the person, organization or service responsible for creating the web page content [35] is crucial in the metadata. Since a verified author of a document makes the document more reliable and increases the credibility of the document when compared to documents or articles without an author [37]. Therefore, the author of the web pages will be added in the proposed method.

Since the internet is a worldwide platform, where it is easy for unscrupulous people to steal the documents on the internet and write it on their own web pages [37], which is a violation of copyrights [37]. To avoid stealing from happening, an addition of copyrights is needed in the metadata. This addition will be added in the proposed method.
The reason for adding language as metadata, which states the language of the content [35] is to guide the user of which documents are in the language he/she is looking for, making the search process easier for him/her [38].

Each of the added elements that are shown in figure 1 is an important factor in the successfulness of generating metadata. Therefore a need for a new method is proposed.

Figure 1 represents in the red circle, the existing metadata that are collected with Klarity and DC.dot methods. The metadata found in the existing methods were Title, Keywords, Description, Date, Type, Format, and Identifier. While the green circle represents the addition of three DC elements (Creator, Rights and Language) to the new proposed method. The details of the how the metadata added will be collected will be shown in the next chapter.

Figure 1 - Metadata existing and added DC elements in the proposed method.
6. The proposed automatic method of metadata generation

This chapter illustrates in more details the design science used in the production of the proposed methods and the steps used. Also the details of how the algorithm works following a flow chart to ease the understanding.

Design Science was used to ease the process of creation of metadata. First step was to identify the problem [47], which was the time and effort spent in creating manually metadata in web pages. Secondly is to define the goal and requirement of your artifact [47], which is to indicate an improvement of retrieval of automatic metadata. The requirement of the method is to make it easy to use for all web page creators. Thirdly is to design the method [47], which is how it will work and the algorithm that will be associated to extract the metadata for the authors automatically. Fourthly is to try the method [47], which was done in the chapter 7, applying the proposed method on two web pages. The fifth step is to evaluate the artifact [47], which was done by discussing the difference between all the methods shown in the thesis and their outcomes. These five design science process steps were used to easily produce the proposed method and evaluate the outcome of the method.

The proposed automatic method of metadata generation is composed of an algorithm that helps in the extraction of most of the DC 15 set elements [11]. The DC elements that will be extracted from new method are Keywords, Creator, Rights, Language and Description. The method is proposed by the author of this thesis to show that the added elements can help in a reliable better result in generation of metadata by adding Creator, Rights and Language when compared to the previous stated methods Klarity and DC.dot.

The method can be applied to two possible scenarios. The first scenario is when the webpage is already on the internet, but does not include DC metadata in its HTML code. The second scenario is when the authors have written the HTML in their own computer and are ready to put the documents up on the internet.

In the first scenario, the web page is already on the internet. Therefore the method uses an algorithm that requires extracting the HTML code from the web page by sending a HTTP request, which is a request-response protocol between a client and server [39]. The web browser may be the client and the computer that hosts the web site may be the server [39]. The received HTML page is saved.
In the second scenario, involves having the authors HTML code for their website in their own computer and are ready to put the documents up on the internet. Since, the web pages are already in the computer stored, there is no need for the HTTP request as in the first scenario. For both scenarios, the algorithm will continue by using a HTML parser. The parser is used when header fields and values are needed to be extracted from HTML messages [40]. Also, this HTML parser is defined usually as part of a compiler that receives inputs in the form of markup tags and breaks them up into parts of attributes which is needed in this algorithm [40]. After starting the parser and receiving the markup tags. The algorithm starts by searches for the “lang” attribute and takes the word after it, stating which language the document is written in [41]. Then the search continues for the author, the algorithm goes through the HTML code looking for the word author or rel=author. Rel=author helps identify the author of the article in the internet, therefore helps search engines to provide a better search result for their users [37]. This attribute can also help the author find their own content [37]. Then, the algorithm searches for the word rights or copyrights and saves the data that follows it or if there is a link to copyright page. All the tags such as </head>, </script>, <div class = "sidebar-box">, <script type="text/javascript"> and all the other similar tags are deleted [42].

For the paragraph tags which start with <p> and end with </p> [42]. The generation of the description metadata begins by searching for words that exist in the title of the page. All sentences that contain the words in the title are stored in an array. Then all the sentences are put together to represent the description metadata. After creating the description metadata, the creation of keywords metadata is done by taking all the words between <p> and </p> and begins by deleting all the trivial words such as the, a, is, where, how, etcetera. Then, add all the words that are between these tags in an array called WordsArray. More details of how the WordsArray look like are shown in appendix A. Secondly, make another array called WordsCountArray, which increases by one when the word is repeated. WordsCountArray is used as an index to count the number of times the words appear. WordsArray need to be arranged in descending order according to the WordsCountArray that shows the amount of repetition of a word in the web page. The descending order shows the most repeated word at the top of the array and the least repeated word at the end. Then, the top five words that are repeated are added as metadata-keywords, which are supposed to best describe the webpage. After collecting all the valuable information into a file, the Dublin Core metadata is created and pasted between the HTML headers of the documents [11].
Figure 2 - Shows the steps used in the algorithm to extract metadata from web pages.
Figure 2 shows the detailed steps in the algorithm, to extract the metadata that will be stored in the HTML code using the DC set elements. The figure shows the

1. **START**: To start the HTML parser to extract all the tags, attributes and values
2. **SEACRH**: The HTML parser searches for the different tags.
   - Title tag, and then return value.
   - Lang attribute, then return value
   - (author or ref=author) attribute, then return value.
   - Copyrights or rights attribute, and then return value.
3. **LOOK FOR**: The words found in between paragraph tag <p> and </p> or <h> and </h>
4. **CAPTURE**: Words in title in all sentences and capture all sentences and store as description metadata.
5. **LOOK FOR**: The words found in between paragraph tag <p> and </p> or <h> and </h>
   - **Ignore**: All trivial words e.g.: a, the, or, and, where, who, what, how ... etcetera. and unwanted tags and attributes </head>, </script>, <div class = "sidebar-box">, <script type="text/javascript">
   - **Store**: All other words add them in the wordsArray if they do not already exist.
   - **Increase**: value for the index in the wordsCountArray.
   - **Order**: wordsArray with respect to wordsCountArray with the most repeated word to the least repeated word. The top five words that are repeated the most are stored as keywords metadata.
The algorithm presented starts by loading the HTML source code for any webpage. Then the HTML parser is initialized, which is used when header files and values are needed to be extracted from HTML messages. The HTML starts by taking input of markup tags and break them up into parts of attributes which is needed in algorithm.

/* Load HTML source code for a webpage */

/* Initialize a Parser instance and send it the HTML source code */

/* Start parser */

Then searching the language attribute, if found then add to the metadata dictionary. Then searches for the author or ref=author attribute, if found then add to the metadata dictionary. Else if the word rights or copyrights were found, then add the value to the metadata dictionary.

if /* lang tag was found */{
    if /* attribute value was found */
        /* Add the attribute value to the metadata dictionary */
    }else if /* author OR ref = author was found */
        /* Add the attribute value to the metadata dictionary */
    }else if /* rights OR copyrights was found */
        /* Add value to the metadata dictionary */

Then the HTML parser searches all the words between any paragraph tags such as <a></a> and <p></p>. For every word in the title, add the sentence to an array. Then the parser keeps searching and adds any relevant sentence to the array to gather all the sentences that will represent the description and will be stored in the metadata dictionary.

}else if /* paragraph tag <p> was found */

for every word in the title

    { /*Add all sentences in between that contain the words in the title*/

}
Afterwards the HTML parser searches again the paragraph tags. It has a choice. If it is trivial word e.g.: a, the, or, and, where, who, what, how, etcetera and if there is unwanted tags, such as </head>, </script>, <div class = "sidebar-box">, <script type="text/javascript"> ignore them and do nothing. Else if a word doesn’t exist in the wordsArray, add that word and index the wordsCountArray with 1. Else if the word already exists, increase the wordsCountArray.

Finally, the words are ordered in descending order according to the wordsCountArray, from the most repeated word to the less repeated word. Then the choice of the top five repeated words are considered the keyword of the webpage and added to the metadata dictionary.

```java
else if /* paragraph tag <p> was found */

for word in paragraph  // Scan the paragraph process each word

  { if /* word is a trivial word e.g.: a, the, or, and, where, who, what, how OR </head>, </script>, <div class = "sidebar-box">, <script type="text/javascript">... etc */

    /* Do nothing with the word */

  else if /* word doesn't not exist in the wordsArray */

    /* Add the word to the wordsArray */

    /* At the same index initialize the wordsCountArray with the value 1 */

  else /* word already exists in wordsArray */

    /* Get index of the word from the wordsArray */

    /* Increase value for that index in the wordsCountArray */

  }

  /* Order both wordsArray and wordsCountArray in descending order by how many times the word appeared in the paragraph */
```
7. Results

The method is aimed to be applied on web pages that will be uploaded to the World Wide Web. The aim of the method is to help authors who have created a webpage using HTML code to use an automatic tool to handle metadata in an easier way; the analysis was performed on a web page already available and compared with the existing keywords to prove the efficiency of the proposed method. The results were analyzed using two web pages and the steps are analyzed in details in the following two section. Random web pages were selected for the analysis, including a discussion on the slight improvement. Also, an evaluation of the proposed method is conducted.

7.1 Webpage 1

The analysis started by examining the webpage www.cprogramming.com [44]. At the beginning the HTML code was about 13 pages. First the search for the attribute “lang” in the HTML code was found and extracted the word after the “=” sign, which was “en”. Then the search for rel=author and author was done, but no result was found. Finally the search for the words copyrights and rights, the content and the link after it was reserved.

Then the biggest process of searching the keywords of the webpage was started by searching for all the paragraph tags <p>. Then the algorithm started examining all the paragraph tags and removing the trivial words from the list of words found in the array such as the, a, with, can, look, at, etcetera. The most repeated words detected were C++ that appeared around 22 times, programming appeared 15 times, tutorials appeared 14 times, C appeared 13 times, Cprogramming appeared 6 times. Other words such as Ebook, compiler, book, games, graphics, practice and data structures appeared between 2 and 4 times. The top five repeated words were selected to generate the metadata keywords which included C++, programming, tutorials, C, Cprogramming were set to be put as metadata keywords. A more detailed searching with all the words found from the webpage after removing all the unused tags and words can be found in appendix A. Table 1 below shows the conclusion of the metadata collected in order for it to be processed and turned in Dublin Core metadata that can be added between the headers in the HTML code.
Table 1 - Shows the metadata collected from the proposed method.

<table>
<thead>
<tr>
<th>Site</th>
<th>lang, copyrights, creator.</th>
<th>Keywords (times found in HTML code)</th>
<th>Keywords that were found inserted in the webpage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cprogramming.com</td>
<td>1. lang=en</td>
<td>1. C++ (~ 22 times)</td>
<td>C, C++, Programming, C++ tutorials, Cprogramming, source code, Programming quiz</td>
</tr>
<tr>
<td></td>
<td>3. Author: Not found</td>
<td>3. tutorials (~14)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. C (~13times)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. Cprogramming (~6)</td>
<td></td>
</tr>
</tbody>
</table>

Then the metadata description was detected by searching for all the sentences that include the titles’ name. That means any sentence that includes “cprogramming” will be added.

The outcome of the analysis of sentences was:

Welcome! Cprogramming.com is your source for everything C and C++! C Programming and C++ Programming.

Want to become a C++ programmer?

The Cprogramming.com ebook, jumping into C++, will walk you through it, step-by-step.

What's new in at Cprogramming.com??

Site
7.1.1 Discussion of webpage 1

The new method extracted and created language, rights, keywords, description metadata that will be placed in the head section of the HTML code of webpage. This will give a better result in comparison with that existed. Table 2 shows the metadata that already existed in the website cprogramming.com [44] versus the data collected from both methods Klarity and DC.dot and the data that is added using the proposed method that uses a simple algorithm.

If compared with the two other methods Klarity and DC.dot, there are three extra metadata extracted rights, language and creator which the latter methods do not extract in their methods. Concerning the keywords collected using the proposed method, the difference between the collected words and the words that was found in the HTML code was in the words source code and programming quiz. These words were not found as often as the rest and would not harm the document if not written as metadata keywords. This proves the ability of the method to capture the same keyword metadata found in the HTML code. Also, an improvement is the addition of more attributes that would be added as metadata. Attributes that were added was the language attribute showing in which language the documents were written for example <meta name="DC.Language" content="English">. A copyright word was found and the algorithm took what follows “Copyright © 1997-2011 Cprogramming.com”, then added as a Dublin core metadata <meta name="DC.Rights" content="Copyright © 1997-2011 Cprogramming.com ">. Unfortunately there was no author/creator tag found. However, the addition of language, copyrights metadata to keywords and descriptions metadata gave a more detailed metadata. This proves a small improvement in the proposed methods automatically generating metadata.

Therefore the new proposed method gives a better result when compared with both the latter methods and the metadata that already existed in the webpage.
<table>
<thead>
<tr>
<th></th>
<th>Existing Metadata</th>
<th>Klarity/DC.dot</th>
<th>New Metadata Added</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Keywords:</strong></td>
<td>&lt;meta name=&quot;keywords&quot; CONTENT=&quot;C, C++, programming, C++ tutorials, C++ programming, C programming, source code, programming quiz&quot;&gt;</td>
<td>Will extract</td>
<td>&lt;meta name=&quot;DC.subject&quot; CONTENT=&quot;C, C++, programming, C++ tutorials, C++ programming, C programming&quot;&gt;</td>
</tr>
<tr>
<td><strong>Descriptions:</strong></td>
<td>&lt;meta name=&quot;Description&quot; CONTENT=&quot;A website designed to help you learn C or C++. Understandable C and C++ programming tutorials, compiler reviews, source code, tips and tricks.&quot;&gt;</td>
<td>Will extract</td>
<td>&lt;meta name=&quot;DC.description&quot; CONTENT=&quot;Welcome! Cprogramming.com is your source for everything C and C++. Want to become a C++ programmer? The Cprogramming.com ebook, Jumping into C++, will walk you through it, step-by-step. What's new in at Cprogramming.com? &quot;&gt;</td>
</tr>
<tr>
<td><strong>Type:</strong></td>
<td>&lt;meta http-equiv=&quot;Content-Type&quot; content=&quot;text/html; charset=iso-8859-1&quot;&gt;</td>
<td>Will extract</td>
<td>Does not extract</td>
</tr>
<tr>
<td><strong>Language:</strong></td>
<td>Not listed</td>
<td>Does not extract</td>
<td>&lt;META NAME=&quot;DC.Language&quot; content=&quot;English&quot;&gt;</td>
</tr>
<tr>
<td><strong>Rights:</strong></td>
<td>Not listed</td>
<td>Does not extract</td>
<td>&lt;META NAME=&quot;DC.rights&quot; content=&quot;Copyright © 1997-2011 Cprogramming.com &quot;&gt;</td>
</tr>
<tr>
<td><strong>Creator</strong></td>
<td>Not listed</td>
<td>Does not extract</td>
<td>Not found using the method</td>
</tr>
</tbody>
</table>

Table 2 - Shows the difference in the metadata collected in the current webpage, Klarity, DC.dot and the new method in the webpage cprogramming.com [44].
Clarification of the words in the table:

**Not listed:** This means that it is not listed in the webpage metadata between the head section of the HTML code.

**Does not extract:** This means that this type of method does not extract this type of metadata.

**Will extract:** This means that the method will extract that specific metadata, but the way it will extract is unknown

**Not Found:** This means that after using a specific method, the metadata specified is not found using the algorithm

### 7.2 Webpage 2

A second analysis was made by examining the webpage [www.tutorialspoint.com/operating_system/](http://www.tutorialspoint.com/operating_system/) [45]. Initially, the webpage consisted of 6 pages and after removing all the unused words and tags, the result was 2 pages. However, the outcome of this analysis was different from webpage 1. The metadata keywords that were detected with the algorithm were not the same as the metadata found in the HTML code. The new keywords detected were **OS (~18 times)**, **Operating Systems (~17 times)**, **Tutorials (~12 times)**, **Computer (~6)** and **UNIX (~4)**. A detailed analysis of the keyword can be found in appendix B. These words were detected as the top 5 repeated words in the webpage. Although the suggested Meta keywords from the method used were not matching the Meta keywords found in the HTML code, but they were more related to what the webpage included. Since the webpage existing metadata does not describe the metadata found in the HTML code, therefore it could be misleading to some search users. The new suggested method is more precise of what the webpage includes.
Table 3 - Shows the collected analysis of metadata using the proposed method.

Also, the metadata description was detected by searching for all the sentences that include the titles’ name. That means any sentence that includes tutorial, Operating systems was included.

The outcome of the sentences analysis was:

Tutorials Point - Simply Easy Learning, Operating System Tutorial. This tutorial will take you through step by step approach while learning Operating System concepts. An operating system (OS) is a collection of software that manages computer hardware resources and provides common services for computer programs. The operating system is a vital component of the system software in a computer system. Before you start proceeding with this tutorial, I'm making an assumption that you are already aware about basic computer concepts like what are keyboard, mouse, monitor, input, output, primary memory, secondary memory and

<table>
<thead>
<tr>
<th>Site</th>
<th>Title, lang, copyrights, creator.</th>
<th>Keywords (times found in HTML code)</th>
<th>Keywords that were found inserted in the webpage</th>
</tr>
</thead>
</table>
| www.tutorialspoint.com/operating_system/ | 1. &lt;title&gt;Operating System Tutorial&lt;/title&gt;  
2. lang="en"  
3. 2014 by tutorialspoint  
4. Author: Not found | 1. OS (~ 18 times)  
2. Operating Systems (~17 times)  
3. Tutorials (~12times)  
4. Computer (~6)  
This reference has been prepared for the computer science graduates to help them understand the basic to advanced concepts related to *Operating System*. If you are not well aware of these concepts then I will suggest to go through our short tutorial on *Operating System* Quick Guide. A quick *Operating System* reference guide for *Operating System* Programmers. A short tutorial on Unix *Operating System*. Download a quick *Operating System* tutorial in PDF format.

### 7.2.1 Discussion of webpage 2

The proposed method extracted and created the same metadata elements like the previous webpage 1 language, rights, keywords and description metadata that will be placed in the head section of the HTML code of webpage. This will give a better result in comparison with that existed. Table 4 shows the metadata that already existed in the website [www.tutorialspoint.com/operating_system/](http://www.tutorialspoint.com/operating_system/) versus the data collected from both methods Klarity and DC.dot and the data that can be added using the proposed method that uses a simple algorithm. The new method extracted and created *language, rights, keywords* and *description* metadata that will be placed in the head section of the HTML code of webpage. This will give a better result in comparison with that existed. Also, if compared with the two other methods Klarity and DC.dot, there are three extra metadata extracted *rights, language* and *creator* which the latter method do not extract in their methods. Although the creator metadata was not found in this specific website, there were two extra metadata found *rights* and *language*. The algorithm detected the language as English “en” and the copyrights to “2014 by tutorialspoint”. Therefore the new proposed method gives a better result when compared with both the latter methods and metadata that already existed in the webpage.
<table>
<thead>
<tr>
<th>Existing Metadata</th>
<th>Klarity /DC.dot</th>
<th>New Metadata Added</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Keywords:</strong></td>
<td>&lt;meta name=&quot;Keywords&quot; content=&quot;Operating System, Tutorials, Learning, Beginners, Basics, Definition, Functions, Conceptual View, Program Execution, Program Execution, Communication, Error Handling, User Account Management, Multitasking, Real Time System, Process, Program, Memory Management, Security&quot; /&gt;</td>
<td>&lt;meta name=&quot;DC.Keywords&quot; content=&quot;OS , Operating Systems, Tutorials, Computer , Unix &quot; /&gt;</td>
</tr>
<tr>
<td><strong>Descriptions:</strong></td>
<td>&lt;meta name=&quot;description&quot; content=&quot;Operating System Tutorial for Beginners - Learning operating system concepts in simple and easy steps : A beginner's tutorial containing complete knowledge about an operating system starting from its Definition, Functions, Conceptual View, Program Execution, Program Execution, Communication, Error Handling, User Account Management, Multitasking, Real Time System, Process, Program, Memory Management, Security&quot; /&gt;</td>
<td>&lt;meta name=&quot;DC.description&quot; content=&quot;Tutorials Point - Simply Easy Learning, Operating System Tutorial .This tutorial will take you through step by step approach while learning Operating System concepts. An operating system (OS) is a collection of software that manages computer hardware resources and provides common services for computer programs. The operating system is a vital component of the system software in a computer system. Before you start proceeding with this tutorial, I'm making an assumption that you are already aware about basic&quot; /&gt;</td>
</tr>
</tbody>
</table>
Real Time System, Process, Program, Memory Management, and Security."

computer concepts like what is keyboard, mouse, monitor, input, output, primary memory and secondary memory etc. This reference has been prepared for the computer science graduates to help them understand the basic to advanced concepts related to Operating System. If you are not well aware of these concepts then I will suggest to go through our short tutorial on Operating System Quick Guide. A quick Operating System reference guide for Operating System Programmers. A short tutorial on Unix Operating System. Download a quick Operating System tutorial in PDF format.

<table>
<thead>
<tr>
<th>Language:</th>
<th>Not listed</th>
<th>Does not extract</th>
<th>&lt;meta name=&quot;DC.Language&quot; content=&quot;English&quot;/&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rights:</td>
<td>Not listed</td>
<td>Does not extract</td>
<td>&lt;meta name=&quot;DC.rights&quot; content=&quot;&quot; 2014 by tutorialspoint &quot;&quot;/&gt;</td>
</tr>
</tbody>
</table>

Table 4 - Shows the difference in the metadata collected in the current webpage, Klarity, DC.dot and the new method in the webpage www.tutorialspoint.com/operating_system/ [45].

Clarification of the words in the table:

**Not listed:** This means that it is not listed in the webpage metadata between the head section of the HTML code.

**Does not extract:** This means that this type of method does not extract this type of metadata.

**Will extract:** This means that the method will extract that specific metadata, but the way it will extract is unknown

**Not Found:** This means that after using a specific method, the metadata specified is not found using the algorithm.
7.3 Evaluation

The proposed method has an advantage of generation of more metadata when compared to the existing method such as Klarity and DC.dot, which is useful for different causes. If a webpage is moving internationally, then a language metadata is useful to declare the main language of the webpage. Also, the rights/copyrights metadata is important as unscrupulous people can steal the content of the webpage. Lastly, the creator metadata is also an addition to the proposed method. This metadata is important to increase the creditability and reliability of a webpage. Moreover, the keywords collected using the algorithm implemented in the method were the same collected as that found in the webpage 1 with a difference of two words, which were not stated in the whole webpage except twice. While in webpage 2, the keywords collected were the same in only two words and the rest was not found in the current webpage metadata, which proves that the method produces keywords that are really found in the webpage. Furthermore, the description that was found with the algorithm was explaining the websites content, since the algorithm extracts sentences found in the webpage, not sentences created by the author, which is more reliable and unbiased.

Considering the efficiency, the automatic proposed method is cost effective as a machine will generate the metadata automatically and no need for a web developer to insert the metadata manually. Also, the proposed method is timesaving, which is valuable to the author of the webpage to speed up the process of website creation. So overall the proposed method is simple, cost effective, and timesaving if compared with the manual methods. Moreover, there is noticed a slight improvement of the retrieval process in terms of time, expense, consistency and unbiased results.

Any method contains some drawbacks that need to be considered. One of the drawbacks is that the proposed method does not extract the date of creation of the webpage. When looking at webpage that has technical, facts and scientific content then it is important to know the date of creation of the webpage whether it is updated or old content. Another drawback is if the webpage does not contain an author attribute, then the method cannot extract anything to be put as an author metadata.
8. Conclusion and future work

This chapter provides a summary of the results achieved and the benefits of using automatic metadata. Also illustrating the goals reached and possible future work in this area.

8.1 Conclusion

Metadata, data about data, is an important feature in websites to rank web pages in search engines. The significance of the relation of correctly writing metadata and increasing visitors to the webpage is huge. After analyzing both the manual and automatic generation in terms of their benefits and drawbacks, a conclusion is made that manually creating metadata is both time consuming and is more expensive than automatic generation. Also, manual creation can cause biased results and consistency problems. Using the design science process made it an easier way to detect the problem associated with the existing methods and finding a solution for the practical problems caused by the manual generation of metadata. Time, expense, biased results and consistency problems, where the main focus to solve when creating a proposed method to automatically extract metadata for the authors of the web pages. Therefore, a study of applying different methods including a new proposed method of generating automatic metadata using a qualitative study was needed to be done.

The study of the automatic generation was made on existing web pages as it was impossible to get HTML code from authors who have not uploaded there pages yet on the internet. The purpose of the thesis was to present the various ways to generate metadata from web pages. By this way, the benefits and withdraws of each method were represented in tables 2 and 4 of comparison. The results of the study were that automatic generation is proceeding to be a better way to generate metadata in terms of time consuming and expense. However, challenges that face metadata are the quality and reliability of the creation. Therefore, the study aimed to analyze the proposed method on two web pages and create extra metadata such as language, rights and author using the Dublin Core elements. Then the comparison was made between the existing metadata description, keywords with the metadata collected using the method proposed. There was a slight difference in the keywords found using the proposed method with the keywords that was found in the HTML code in the webpage. However, the new keywords created were a better description of the webpage content which would prove reliability and quality that was missing in the existing keywords metadata. While, the description metadata was totally different from that found in the webpage. The
description metadata collected using a technique that searches for sentences using the titles’ words making it more reliable as that is what the page describes and contains, instead of what the authors would enter manually. The goal of the research was achieved by illustrating the proposed method for the automatic generation of metadata from web pages. Also, the research indicated the slight improvement showed in the proposed method and the difference between the two methods Klarity and DC.dot with the new method showing a slight enhancement in extracting three metadata. To conclude, metadata are a broad subject that needs further study and work. There is always space to get new information and updates about the metadata since they exist in web pages, which are in constant change and development. Metadata generation automatically is of large importance to the authors of the page and should be taken more into account and considered more in web pages, in order to get a higher rank, be recognized internationally and prevented from being stolen of unscrupulous people.

8.2 Future work
The field of metadata is very broad, there is always space to add more metadata to webpages such as date metadata, but since some web pages have several dates, to find the specific date of creation of the webpage was a difficult task. This problem can be solved with a new algorithm in the method performing a comparison with the dates in the webpage. Since the complexity of the issue to extra date metadata correctly, therefore it could be included in the future work. Additional work can be done by adding the date and type metadata and testing the benefits of having all metadata used in web pages. Testing more web pages with different functionalities and contents is needed to make a better conclusion of the benefits of metadata and their existence in web pages. Metadata has many forms and is present not only in web pages, but also in documents, images, videos, etcetera. By using all forms of metadata correctly, the search of a certain webpage, video, image and documents will be much easier.
9. References


http://scholar.google.com/scholar?q=eLearning+based+on+the+Semantic+Web&btnG=&hl=en&as_sdt=0%2C5&as_vis=1

[14] Berners-Lee et al., Available from: http://www.w3.org/RDF/Metalog/docs/sw-easy


[18] Introduction to OWL. Available from: http://www.w3schools.com/webservices/ws_rdf_owl.asp


[25] AsaneeKawtrakul and Chaiyakorn Yingsaeree, A UNIFIED FRAMEWORK FOR AUTOMATIC METADATA EXTRACTION FROM ELECTRONIC DOCUMENT,
Department of Computer Engineering, Faculty of Engineering, Kasetsart University, Bangkok, Thailand. Available from: papercut.googlecode.com/hg-history/.../success/metadata_extraction.pdf


[40] Margaret Rouse, Parser. Available from: http://searchsoa.techtarget.com/definition/parser


Appendix A:

Web page: www.cprogramming.com

Existing Metadata Keywords:
&lt;META NAME="keywords" CONTENT="C, C++, programming, C++ tutorials, C++ programming, C programming, source code, programming quiz"&gt;
&lt;META NAME="Description" CONTENT="A website designed to help you learn C or C++ Understandable C and C++ programming tutorials, compiler reviews, source code, tips and tricks."&gt;

Existing Metadata Description:

&lt;META NAME="Description" CONTENT="A website designed to help you learn C or C++ Understandable C and C++ programming tutorials, compiler reviews, source code, tips and tricks."&gt;

Title: C programming.com - Learn C and C++ Programming - Cprogramming.com

Rights: Copyright © 1997-2011 Cprogramming.com. All rights reserved.

Language: lang="en"

New Metadata Keywords:

<table>
<thead>
<tr>
<th>wordsArray</th>
<th>wordsArrayCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>C++</td>
<td>22</td>
</tr>
<tr>
<td>Programming</td>
<td>15</td>
</tr>
<tr>
<td>Tutorials</td>
<td>14</td>
</tr>
<tr>
<td>C</td>
<td>13</td>
</tr>
<tr>
<td>Cprogramming</td>
<td>6</td>
</tr>
<tr>
<td>Game</td>
<td>5</td>
</tr>
<tr>
<td>Learn</td>
<td>4</td>
</tr>
<tr>
<td>Ebook</td>
<td>4</td>
</tr>
<tr>
<td>practice</td>
<td>4</td>
</tr>
<tr>
<td>Problem</td>
<td>4</td>
</tr>
<tr>
<td>common</td>
<td>4</td>
</tr>
<tr>
<td>Expert</td>
<td>3</td>
</tr>
<tr>
<td>Topic</td>
<td>Frequency</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------</td>
</tr>
<tr>
<td>Graphics</td>
<td>3</td>
</tr>
<tr>
<td>sensors</td>
<td>2</td>
</tr>
<tr>
<td>ways</td>
<td>2</td>
</tr>
<tr>
<td>Algorithms</td>
<td>2</td>
</tr>
<tr>
<td>Data</td>
<td>2</td>
</tr>
<tr>
<td>Structures</td>
<td>2</td>
</tr>
<tr>
<td>Debug basics</td>
<td>2</td>
</tr>
<tr>
<td>mistakes</td>
<td>2</td>
</tr>
<tr>
<td>Quiz</td>
<td>2</td>
</tr>
<tr>
<td>Start source</td>
<td>2</td>
</tr>
<tr>
<td>Compiler</td>
<td>2</td>
</tr>
<tr>
<td>Code face</td>
<td>2</td>
</tr>
<tr>
<td>Tips</td>
<td>1</td>
</tr>
<tr>
<td>syntax</td>
<td>1</td>
</tr>
<tr>
<td>links</td>
<td>1</td>
</tr>
<tr>
<td>Recommendations</td>
<td>1</td>
</tr>
<tr>
<td>Java course</td>
<td>1</td>
</tr>
<tr>
<td>Resource</td>
<td>1</td>
</tr>
<tr>
<td>Cross-platform</td>
<td>1</td>
</tr>
<tr>
<td>Windows</td>
<td>1</td>
</tr>
</tbody>
</table>
Appendix B:

Webpage: http://www.tutorialspoint.com/operating_system/

Existing Metadata Keywords:
<meta name="Keywords" content="Operating System, Tutorials, Learning, Beginners, Basics, Definition, Functions, Conceptual View, Program Execution, Program Execution, Communication, Error Handling, User Account Management, Multitasking, Real Time System, Process, Program, Memory Management, Security" />

Existing Metadata Description:
<meta name="description" content="Operating System Tutorial for Beginners - Learning operating system concepts in simple and easy steps : A beginner's tutorial containing complete knowledge about an operating system starting from its Definition, Functions, Conceptual View, Program Execution, Program Execution, Communication, Error Handling, User Account Management, Multitasking, Real Time System, Process, Program, Memory Management, and Security." />

Title: Operating System Tutorial

Rights: Copyright © 2014 by tutorialspoint. All Rights Reserved

Language: lang="en"

New Metadata Keywords:

<table>
<thead>
<tr>
<th>wordsArray</th>
<th>wordsArrayCount</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td>18</td>
</tr>
<tr>
<td>Operating Systems</td>
<td>17</td>
</tr>
<tr>
<td>Tutorials</td>
<td>12</td>
</tr>
<tr>
<td>Computer</td>
<td>6</td>
</tr>
<tr>
<td>UNIX</td>
<td>4</td>
</tr>
<tr>
<td>Concepts</td>
<td>3</td>
</tr>
<tr>
<td>Learning</td>
<td>2</td>
</tr>
<tr>
<td>----------------</td>
<td>---</td>
</tr>
<tr>
<td>Home</td>
<td>2</td>
</tr>
<tr>
<td>Scheduling</td>
<td>2</td>
</tr>
<tr>
<td>Processes</td>
<td>2</td>
</tr>
<tr>
<td>MySQL</td>
<td>2</td>
</tr>
<tr>
<td>basic</td>
<td>2</td>
</tr>
<tr>
<td>Software</td>
<td>2</td>
</tr>
<tr>
<td>HTML</td>
<td>2</td>
</tr>
<tr>
<td>CSS</td>
<td>2</td>
</tr>
<tr>
<td>Javascript</td>
<td>1</td>
</tr>
<tr>
<td>C++</td>
<td>1</td>
</tr>
<tr>
<td>Easy</td>
<td>1</td>
</tr>
<tr>
<td>Simply</td>
<td>1</td>
</tr>
<tr>
<td>Overview</td>
<td>1</td>
</tr>
<tr>
<td>Types</td>
<td>1</td>
</tr>
<tr>
<td>Services</td>
<td>1</td>
</tr>
<tr>
<td>Properties</td>
<td>1</td>
</tr>
<tr>
<td>Algorithms</td>
<td>1</td>
</tr>
<tr>
<td>Multi-threading</td>
<td>1</td>
</tr>
<tr>
<td>Memory Management</td>
<td>1</td>
</tr>
<tr>
<td>Virtual Memory</td>
<td>1</td>
</tr>
<tr>
<td>I/O Hardware</td>
<td>1</td>
</tr>
<tr>
<td>I/O Software</td>
<td>1</td>
</tr>
<tr>
<td>File System</td>
<td>1</td>
</tr>
<tr>
<td>Security</td>
<td>1</td>
</tr>
<tr>
<td>Linux</td>
<td>1</td>
</tr>
<tr>
<td>Perl</td>
<td>1</td>
</tr>
<tr>
<td>Ruby</td>
<td>1</td>
</tr>
<tr>
<td>Hardware</td>
<td>1</td>
</tr>
<tr>
<td>resources</td>
<td>1</td>
</tr>
<tr>
<td>programs</td>
<td>1</td>
</tr>
<tr>
<td>component</td>
<td>1</td>
</tr>
<tr>
<td>Term</td>
<td>Count</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>JAVA</td>
<td>1</td>
</tr>
<tr>
<td>PHP</td>
<td>1</td>
</tr>
<tr>
<td>Phyton</td>
<td>1</td>
</tr>
<tr>
<td>audience</td>
<td>1</td>
</tr>
<tr>
<td>science</td>
<td>1</td>
</tr>
<tr>
<td>Keyboard</td>
<td>1</td>
</tr>
<tr>
<td>mouse</td>
<td>1</td>
</tr>
<tr>
<td>monitor</td>
<td>1</td>
</tr>
<tr>
<td>Input</td>
<td>1</td>
</tr>
<tr>
<td>output</td>
<td>1</td>
</tr>
<tr>
<td>primary memory</td>
<td>1</td>
</tr>
<tr>
<td>secondary memory</td>
<td>1</td>
</tr>
<tr>
<td>programmers</td>
<td>1</td>
</tr>
<tr>
<td>sites</td>
<td>1</td>
</tr>
<tr>
<td>books</td>
<td>1</td>
</tr>
<tr>
<td>Articles</td>
<td>1</td>
</tr>
<tr>
<td>Unix</td>
<td>1</td>
</tr>
<tr>
<td>ASP.NET</td>
<td>1</td>
</tr>
<tr>
<td>AJAX</td>
<td>1</td>
</tr>
<tr>
<td>jQuery</td>
<td>1</td>
</tr>
<tr>
<td>ANT</td>
<td>1</td>
</tr>
<tr>
<td>JSP</td>
<td>1</td>
</tr>
<tr>
<td>Struts</td>
<td>1</td>
</tr>
<tr>
<td>SQL</td>
<td>1</td>
</tr>
</tbody>
</table>