I.J. Modern Education and Computer Science, **2014, 8, 59-65**
Published Online August 2014 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmecs.2014.08.07

Implementation of Parallel Web Crawler through .NET Technology

Md. Abu Kausar
Dept. of Computer & System Sciences, Jaipur National University, Jaipur, India
Email: kausar4u@gmail.com

V. S. Dhaka
Dept. of Computer & System Sciences, Jaipur National University, Jaipur, India
Email: vijaypal.dhaka@gmail.com

Sanjeev Kumar Singh
Dept. of Mathematics, Galgotias University, Gr. Noida, India
Email: sksingh8@gmail.com

***Abstract*—**The WWW is increasing at very fast rate and data or information present over web is changes very frequently. As the web is very dynamic, it becomes very difficult to get related and fresh information. In this paper we design and develop a program for web crawler which uses multiple HTTP for crawling the web. Here we use multiple threads for implementation of multiple HTTP connection. The whole downloading process can be reduced with the help of multiple threads. This paper deals with a system which is based on web crawler using .net technology. The proposed approach is implemented in VB.NET with multithread to crawl the web pages in parallel and crawled data is stored in central database (Sql Server). The duplicacy of record is checked through stored procedure which is pre complied & checks the result very fast. The proposed architecture is very fast and allows many crawlers to crawl the data in parallel.

***Index Terms*—**World Wide Web, Web Crawler, multiple HTTP connections, multi threading, URL, Database.

I. Introduction

Internet [1] is basically a worldwide collection of information which is logically connected by a worldwide unique address called as IP address. World Wide Web is a network of interconnected Hyperlinks on Internet which is spread over far and remote geographical places. Actually, the web size is very huge and the information present over web is very tedious to explore as and when needed.

Web crawler [2] used in Search engines to continually gathering of web pages from the web.

Web crawler is basically a software or program/script which is used for downloading the web pages connected with one or more given seed URLs, take outs any hyperlinks which is present in them and continually download the web pages recursively recognized by take outs links. The important component of search engine is crawler, which is used for gathering of web pages indexed by the various search engines. Moreover web crawlers are used in many other applications that process large numbers of web pages.

The rest of this paper is organized as follow. Sections two describe related work. Sections three describe the web crawling terminology Sections four describe the proposed work section five describes implementation details of proposed system and section six concludes the paper.

II. Related Work

The job of web crawler is not only to handle the index of receiving information which is extracted but it also handle other problems such as the allocation of web sites which is to be crawled, network usage and bringing the downloaded web pages together with previously downloaded data, duplication of information and integration of results.

Author [3] discusses different criteria and options for parallelization of crawling procedure. A web crawler is globally distributed or centrally handled. A web crawler which is designed to avoid overlap of web pages which are downloaded while taking care load of network. The authors describe the feature of a web crawler which downloads the most important web pages before others. This process is significant for parallel web crawler as all crawling process only concentrate on local data for making the web pages important. Authors also stated that distributed web crawlers are beneficial than multithreaded web crawlers because of effectiveness, scalability and throughput. If reduction of network load and distribution of network are completed then parallel web crawler can give up high quality web pages.

Mercator is a web crawler which is extensible and scalable and is currently revolved into the Altavista [4] search engine. [5] Talk about implementation concern which is recognized for parallel web crawler developing, which can goes down performance. They talk about merits and demerits of various coordination methods and assessment criteria. Authors review the work with great presentation information. In short, authors agree that the overhead of communication does not enhance as more web crawlers are inserted, when more nodes are added then system's throughput enhances and the feature of the system also, i.e. the capability to fetch significant web pages first.

Fetterly et. al. [6] explains a major experiment for measuring the web page changes rate over a large time period. They download about 151 millions web pages once every week over a period of 11 weeks. Author recorded most important information about all crawled web pages. Shkapenyuk and Suel [7] produce data which are related to produce by [4]. Their design uses one device for each modular job. To keep away from bottlenecks while improve, they had to install more devices for exact job. Hence, for appending additional crawler devices the number of devices amplified with a greater curve. The communication of network load raised drastically caused by raised coordination job among machines with same type of jobs.

Authors [8] applied a system for obtain web pages from web servers near to them with the use of globally distributed web crawlers. The system is flexible to breakdowns but it has long flexibility point and the characteristic is injurious for information. When web crawler fails, the web sites which were crawling are moved to other web crawlers for crawling the fails web site. Huge overlap of data occurs due to this result. For decide the next web crawler a serious heartbeat protocol is required. Furthermore, a web crawler closer to various servers may be overloaded where a web server may be sitting inactive at a little bit more distance.

Cho et. al. [9], explain the significance of URL reordering in the crawl frontier. If web pages present in the crawl frontier and is connected with a variety of web pages which are to be crawled, a little amount of web pages which are connected from it makes sense to visit it before others. PageRank is employed as a heavy metric for URL ordering and built three models to evaluate web crawlers. Authors conclude that PageRank is a high-quality metric and web pages with a high Page Rank or ones with various backlinks are required first.

Cho and Molina [10], studied about building an incremental web crawler. Authors explain a periodic crawler as the crawler who visits the website until the collection has reached to desirable amount of web pages, and stops visiting web pages. Then when it is essential to refresh the repository, the web crawler constructs a fresh collection using the similar procedure described above, and then swaps the old collection with this fresh one. Alternately, an incremental web crawler crawls and updates web pages after a desired number of web pages are crawled and saved in the collection incrementally. By these incremental update, the crawler refreshes existing web pages and swaps insignificant web pages with novel and more significant web pages. The authors describe Poisson process which is used in web pages for check the rate of changes. A Poisson process is frequently applied to model a sequence of random events that occur independently with fixed rate of time. The authors explain two methods to maintain the repository. In first one, group of various copies of web pages are stored in the collection in which they were found during crawling and in the second one those copies of web pages are saved which is most recent. For this purpose one has to keeps a record of when and how regularly the web page changed. In this paper the authors conclude that an incremental web crawler can yields brand new copies of web pages more rapidly and maintain the storage area fresher than a periodic crawler.

Kausar et. al. [11], [12] proposed a Model for Web Crawling based on Java Aglets. Web Crawling based on Mobile Agent will yield high quality pages. The crawling process will migrate to host or server to start downloading.

III. Crawling Terminology

The web crawler keeps URL list which is not visited called as crawl frontier [13]. The list is initiated by start URL given by a user or some different program. The crawl frontier is used for selection of next URL which is to be crawled for each crawling steps recursively, getting the web page equivalent to particular URL, the downloaded web page is parsed to take out the URLs and information which is application specific, and lastly add the URLs to the frontier which is not visited. The crawling process finished when a specific amount of web pages or all web pages have been crawled. The WWW is observed as a huge graph with links as its edges and web pages as its nodes. A web crawler is started with a few of the web pages and then follows the links to arrive at other web pages. The procedure of retrieving a web page and take out the links present in it is similar to expanding a node in graph search [14].

A. Frontier

The crawling method initialize with a seed URL, extracting links from it and adding them to an unvisited list of URLs, This list of URLs which is not visited is known as Crawl Frontier. The crawl frontier is basically an agenda of crawler that includes the URLs of pages that are not visited. The crawl frontier may be applied as a FIFO queue in that case breadth first crawler can used to search the web blindly. The link which is to be crawled next comes from the top of the list and the new crawled URLs are inserted into the bottom of the list.

B. Fetching

To obtain a Web page, client sends a HTTP request for a particular web page and reads the reply of web pages. There must have timeouts of particular we page or web server to insure that an unnecessary amount of time is not spent on web servers which are slow or reading web page which is large.

C. Parsing

When a web page is obtained, then content of web pages are parsed to take out information from there and possibly direct the prospect path of the web crawler. Parsing involves the URL extraction from HTML pages or it may engage the more difficult procedure of meshing up the web page content.

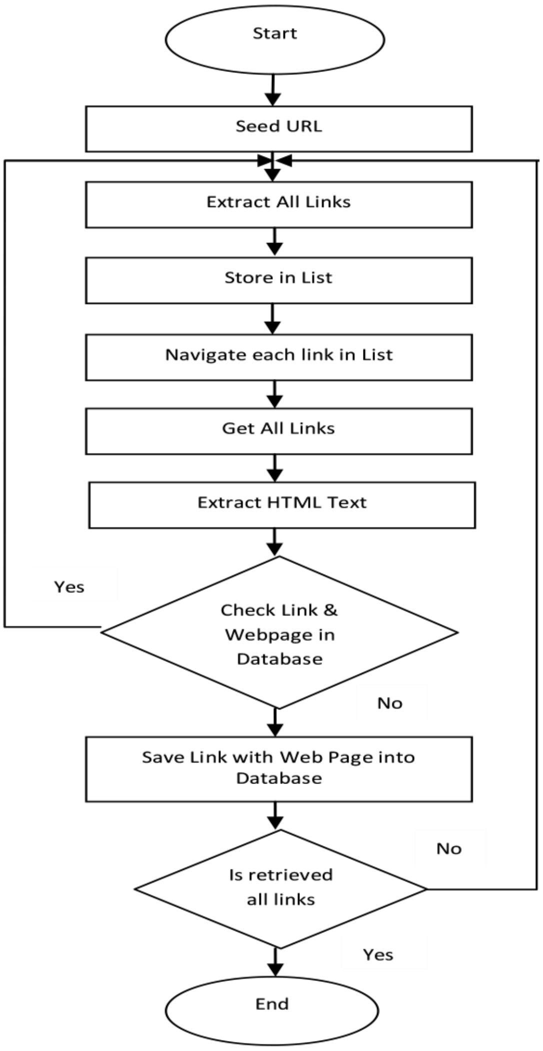


Fig. 1: Proposed Crawling Process

IV. PROPOSED WORK

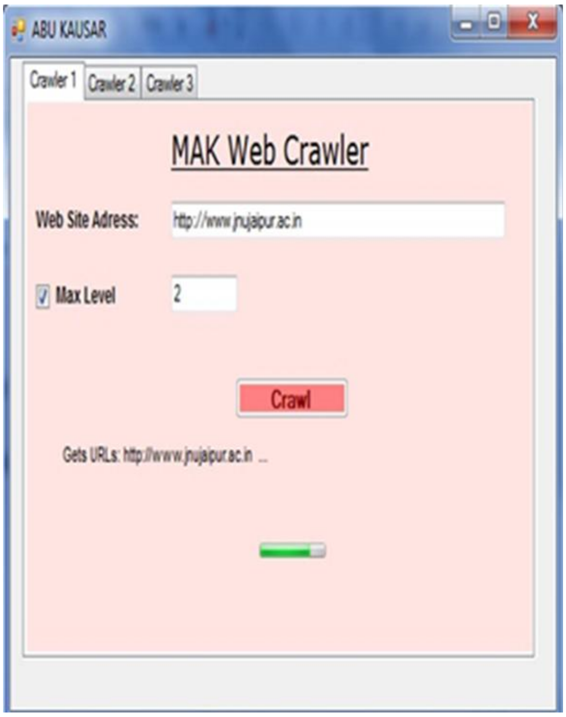
The functioning of Web crawler [15] is beginning with a set of URLs which is called as seed URLs. They download web pages with the help of seed URLs and take out new links which are present in the downloaded pages. The retrieved web pages are stored and well indexed on the storage area so that by the help of these indexes they can later be retrieved as and when required. The URLs which is extracted from the downloaded web pages are confirmed to know whether their associated documents have already been downloaded or not. If associated document are not downloaded, the URLs are again allocated to web crawlers for further downloading. The same process is repeated till no more URLs are missing for downloading. Millions of web pages are downloaded daily by a crawler to complete the target. Fig. 1 illustrates the proposed crawling processes.

V. IMPLEMENTATION DETAILS

Crawlers are used for crawling the web continuously to get up to date data for search purposes. Generally web crawler started by a list of URLs which is to be visited, this list is known as "seed urls". Since the web crawler visits these URLs, it gets the entire URLs in the visited web pages and inserted them to the list of URLs which is to be visit. These newly added hyperlinks are called crawl frontier.

We will build a parallel web crawler application. The user gives the initial URL in the provided interface. It initiates by a URL, get all the web site pages, and store each web page along with its URL to a backend database (Sql Server). To do this, our crawler application will work into two different stages. The first one is to take the web site URL, navigate to it, extract all the links present in the web page, store them in a list called crawl frontier, navigate to each link in the list and get the links present in it, and again store them to the same list (crawl frontier) till the whole web site is navigated. After creating this list of URLs, the second part of our application will start to get the HTML text of each link in the list and save it as a new record in the database. There is only one central database for storing all web pages.

Given below figure is the snapshot of the user interface of the Web Crawler application, which is designed in the VB.NET Windows Application, for crawling a website or any web application using this crawler internet connection must be required and as input use URL in a format as shown in figure. At every crawling step, the program selects the peak URL from the frontier and sends the information of web sites to a unit which will downloads the web pages from the Website. For this implementation we use multithreading for parallelization of crawling process so that we can download many web sites parallel.



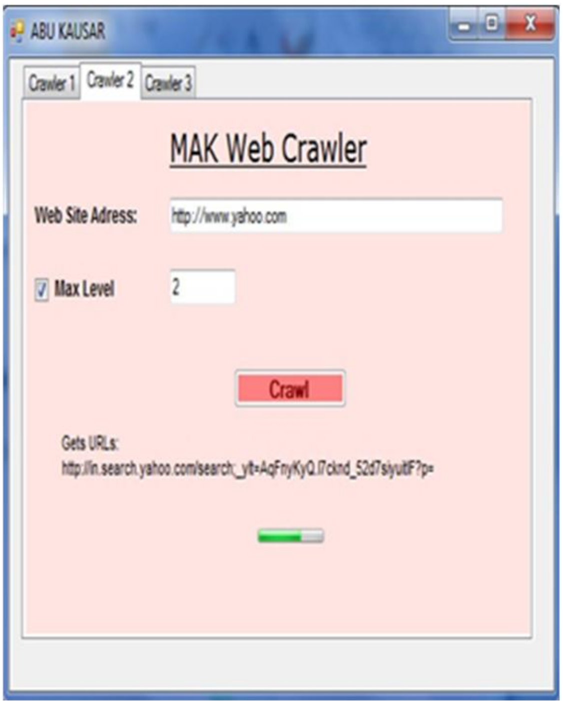
Web Site Adress: http://www.jnujaipur.ac.in

☑ Max Level 2

Gets URLs: http://www.jnujaipur.ac.in ...

Fig. 2: Crawler GUI

As we can see in the Fig. 2. We implement three crawlers in parallel; we can add as many crawlers as we need. Max Level is the maximum level of the website to crawl. When we clink on Crawler 2 tab, given Figure 3 appears which shows second crawl working.



Web Site Adress: http://www.yahoo.com

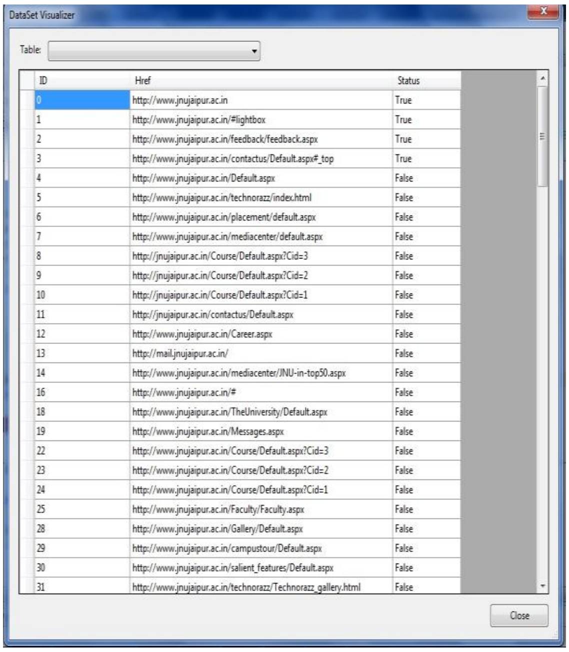
☑ Max Level 2

Gets URLs:
http://in.search.yahoo.com/search;_ylt=AqFnyKyQ.l7ckd_52d7siyuIF?p=

Fig. 3: Crawler 2 GUI

As you can see in the Fig. 3 they also show full progress of crawling.

The crawl frontier is implemented as in memory table, it initializes the in memory site URLs representation. It initiates the process for receiving all the hyperlinks in the given web site. It defines three columns ID, Href, and Status. The ID column is the primary key column for this table. The Href column is the column where the URL of the current hyperlink will be saved. The Status column is a Boolean (True or false) field shows whether the current link is visited or not. Finally a new row contains the current web site URL is added to the table. We can say this table as the crawler URLs list. The first inserted record to this table which is the row that holds the web site URL is known as the seed of the crawler. Other inserted URLs will consist of the frontier of the crawler. We use this memory structure to make advantage of the uniqueness check supported by this way. Any other duplicate row which is already stored in list can be discarded. Fig. 4. Shows the in memory representation of URLs called frontier.



ID	Href	Status
0	http://www.jnujaipur.ac.in	True
1	http://www.jnujaipur.ac.in/#lightbox	True
2	http://www.jnujaipur.ac.in/feedback/feedback.aspx	True
3	http://www.jnujaipur.ac.in/contactus/Default.aspx#_top	True
4	http://www.jnujaipur.ac.in/Default.aspx	False
5	http://www.jnujaipur.ac.in/technorazz/index.html	False
6	http://www.jnujaipur.ac.in/placement/default.aspx	False
7	http://www.jnujaipur.ac.in/mediacenter/default.aspx	False
8	http://jnujaipur.ac.in/Course/Default.aspx?Cid=3	False
9	http://jnujaipur.ac.in/Course/Default.aspx?Cid=2	False
10	http://jnujaipur.ac.in/Course/Default.aspx?Cid=1	False
11	http://jnujaipur.ac.in/contactus/Default.aspx	False
12	http://www.jnujaipur.ac.in/Career.aspx	False
13	http://mail.jnujaipur.ac.in/	False
14	http://www.jnujaipur.ac.in/mediacenter/JNU-in-top50.aspx	False
16	http://www.jnujaipur.ac.in/#	False
18	http://www.jnujaipur.ac.in/TheUniversity/Default.aspx	False
19	http://www.jnujaipur.ac.in/Messages.aspx	False
22	http://www.jnujaipur.ac.in/Course/Default.aspx?Cid=3	False
23	http://www.jnujaipur.ac.in/Course/Default.aspx?Cid=2	False
24	http://www.jnujaipur.ac.in/Course/Default.aspx?Cid=1	False
25	http://www.jnujaipur.ac.in/Faculty/Faculty.aspx	False
28	http://www.jnujaipur.ac.in/Gallery/Default.aspx	False
29	http://www.jnujaipur.ac.in/campustour/Default.aspx	False
30	http://www.jnujaipur.ac.in/salient_features/Default.aspx	False
31	http://www.jnujaipur.ac.in/technorazz/Technorazz_gallery.html	False

Fig. 4: Crawler Frontier

Figure 5. is the algorithm for saving of web pages

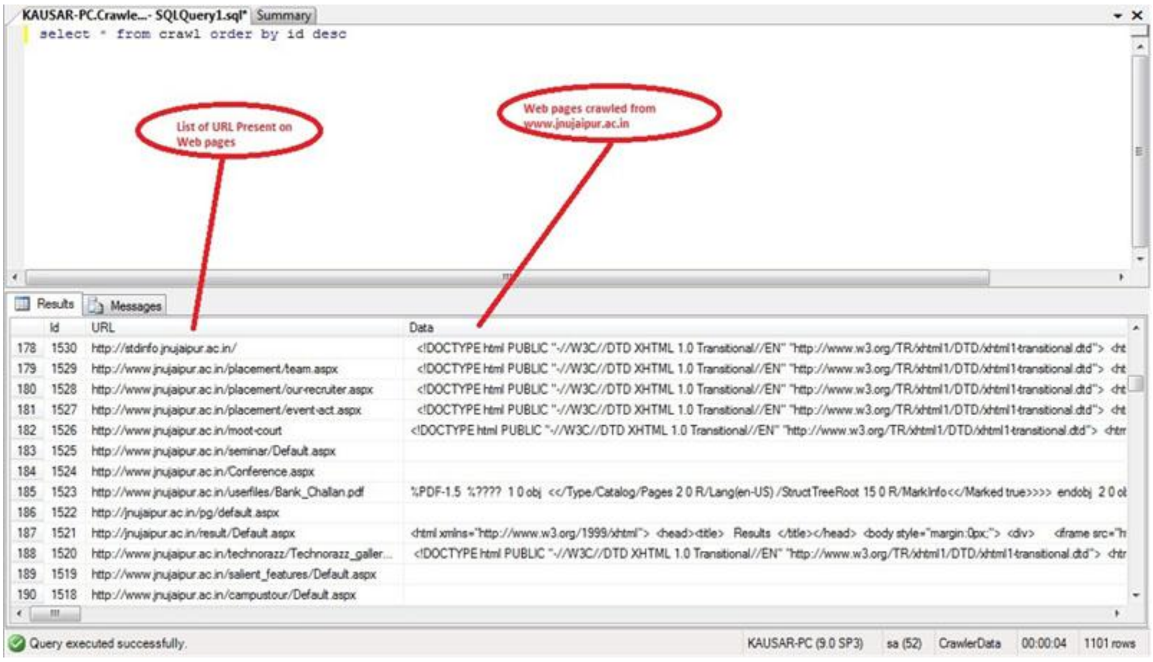
```
SaveWebSite ()
 {
      Assign k As Integer
      Assign stri As String
Start loop from k = 0 To UrlsT.Rows.Count - 1
         Invoke Progressbar()
         stri = UrlsT.Rows(i).Item(1)
Invoke procedure SaveWebPage(stri,
GetWebPage(stri))
      Repeat
  }
```

Fig. 5: Algorithm for Web page Save.

This subroutine traverse the in memory table i.e. URLs table and get the URL from it then get the HTML text from it and finally saves the URL, and the Page text to the data base(Sql Server).

There is only one central data base for storing all URLs & Web Pages. The data are stored in Sql Server database. Before putting the data into data base, data is checked for record duplication and avoid insertion in this case.

The Crawling result is stored in the form of table representing the result as row and columns the output of the Crawler is shown as a snapshot of Crawler1 URL http://www.jnujaipur.ac.in.

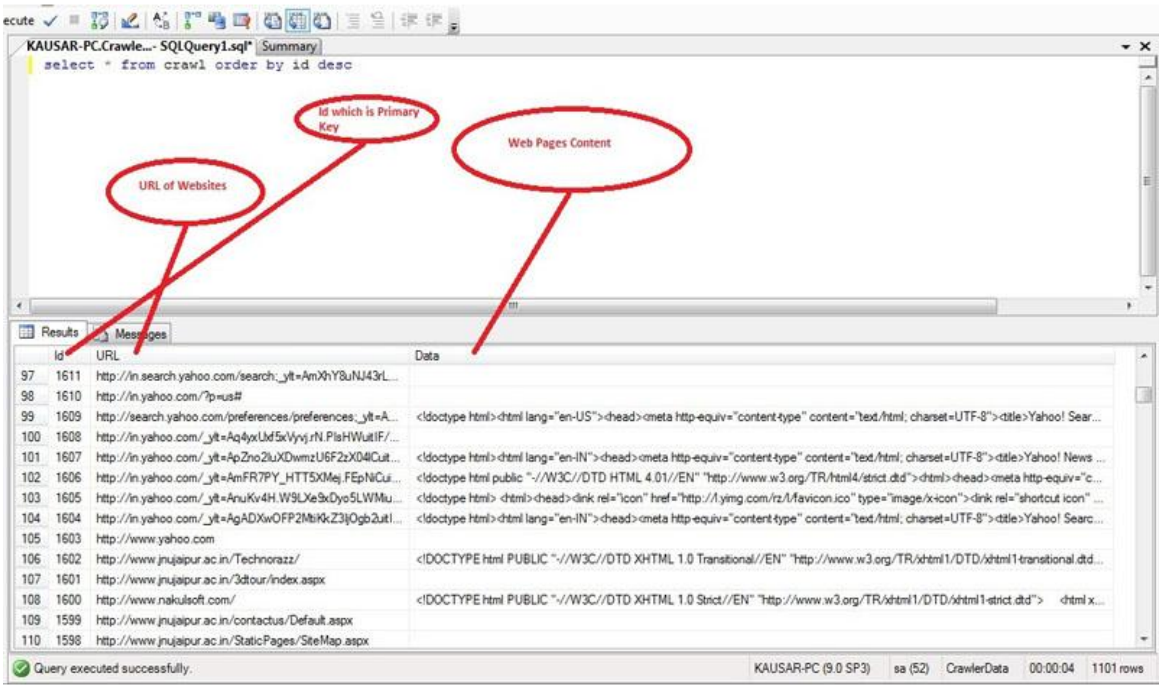

KAUSAR-PC.Crawle...- SQLQuery1.sql*
Summary
List of URL Present on Web pages
Web pages crawled from www.jnujaipur.ac.in
Results
Messages

```
select * from crawl order by id desc
```

	Id	URL	Data
178	1530	http://stdinfo.jnujaipur.ac.in/	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <ht
179	1529	http://www.jnujaipur.ac.in/placement/team.aspx	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <ht
180	1528	http://www.jnujaipur.ac.in/placement/our-recruiter.aspx	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <ht
181	1527	http://www.jnujaipur.ac.in/placement/event-act.aspx	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <ht
182	1526	http://www.jnujaipur.ac.in/moot-court	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <htm
183	1525	http://www.jnujaipur.ac.in/seminar/Default.aspx	
184	1524	http://www.jnujaipur.ac.in/Conference.aspx	
185	1523	http://www.jnujaipur.ac.in/userfiles/Bank_Challan.pdf	%PDF-1.5 %???? 1 0 obj <</Type/Catalog/Pages 2 0 R/Lang(en-US) /StructTreeRoot 15 0 R/MarkInfo<</Marked true>>>> endobj 2 0 ob
186	1522	http://jnujaipur.ac.in/pg/default.aspx	
187	1521	http://jnujaipur.ac.in/result/Default.aspx	<html xmlns="http://www.w3.org/1999/xhtml"> <head><title> Results </title></head> <body style="margin:0px;"> <div> <iframe src="h
188	1520	http://www.jnujaipur.ac.in/technorazz/Technorazz_galler...	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"> <htr
189	1519	http://www.jnujaipur.ac.in/salient_features/Default.aspx	
190	1518	http://www.jnujaipur.ac.in/campustour/Default.aspx	

Query executed successfully. KAUSAR-PC (9.0 SP3) sa (52) CrawlerData 00:00:04 1101 rows

Fig. 6: Crawler 1 stored data on Sql Server.



```
select * from crawl order by id desc
```

	Id	URL	Data
97	1611	http://in.search.yahoo.com/search;_ylt=AmXhY8uNJ43rL...	
98	1610	http://in.yahoo.com/?p=us#	
99	1609	http://search.yahoo.com/preferences/preferences;_ylt=A...	<!doctype html><html lang="en-US"><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><title>Yahoo! Sear...
100	1608	http://in.yahoo.com/_ylt=Aq4yxUxf5xVyvj.rN.PlsHWuitIF/...	
101	1607	http://in.yahoo.com/_ylt=ApZno2luXDwmzU6F2zX04ICuit...	<!doctype html><html lang="en-IN"><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><title>Yahoo! News ...
102	1606	http://in.yahoo.com/_ylt=AmFR7PY_HTT5XMej.FEpNiCui...	<!doctype html public "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd"><html><head><meta http-equiv="c...
103	1605	http://in.yahoo.com/_ylt=AnuKv4H.W9LXe9xDyo5LWMu...	<!doctype html> <html><head><link rel="icon" href="http://l.yimg.com/rz/l/favicon.ico" type="image/x-icon"><link rel="shortcut icon" ...
104	1604	http://in.yahoo.com/_ylt=AgADXwOFP2MtiKkZ3ljOgb2uitl...	<!doctype html><html lang="en-IN"><head><meta http-equiv="content-type" content="text/html; charset=UTF-8"><title>Yahoo! Searc...
105	1603	http://www.yahoo.com	
106	1602	http://www.jnujaipur.ac.in/Technorazz/	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd...
107	1601	http://www.jnujaipur.ac.in/3dtour/index.aspx	
108	1600	http://www.nakulsoft.com/	<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"> <html x...
109	1599	http://www.jnujaipur.ac.in/contactus/Default.aspx	
110	1598	http://www.jnujaipur.ac.in/StaticPages/SiteMap.aspx	

Query executed successfully. KAUSAR-PC (9.0 SP3) sa (52) CrawlerData 00:00:04 1101 rows

Fig. 7: Crawler 2 stored data on Sql Server.

Fig. 7 is the snapshot of crawler 2 URL http://www.yahoo.com

A. *Format of Result*

Data is downloaded page by page after crawling from the website http://www.jnujaipur.ac.in and store in the Sql Server database table’s Data Column field using the format shown below:

```
<html>
<head>
<title>  JNU Jaipur, top grade Life sciences, MCA,
Bachelor of Arts, Engg, Pharmacy, Biotech
University in India, Best Placement, University in
Jaipur </title>
<meta name="description" content="World best
university in India which offers best quality
education in the fields of computer Science,
engineering, management and life sciences. We had
known as a world-class institution who offered
degrees, UG courses, PG course or Ph.D programs in
mostly all fields through regular teaching." />
<meta name="keywords" content="Technical
Engineering Colleges, Top 10 Engineering
University In India, University with International
Exposure, Foreign Exchange Programmes,
Rajasthan, Engineering College In Jaipur,
Engineering and Technology, Bio-Informatics, M.
Tech, Best College in Rajasthan, Top Institute in
Rajasthan" />
……………………..
……………………..
……………………..
</div>
</body>
</html>
```

The format above shows the crawled data for the home page of the website jnujaipur.ac.in.

VI. Comparison Of Proposed Architecture With Existing Architecture

The performance of proposed architecture is compared with that of existing architecture [3]. The summary of the comparison is as shown in Table 1.

Table 1: Compression of Existing System with Proposed System

Existing	Proposed
There is no central server. It works in distributed environment.	It has central server.
It has a distributed environment so it may not be aware about others collection, hence the problem of duplicity occurs.	There is a central server which stores all data. Before insert data into database data is checked for duplicity.
Due to problem of duplicity more network bandwidth is consumed in this architecture.	Proposed approach reduces duplicity problem. So it reduces network bandwidth.

VII. Conclusion

We have described the design and development details of our parallel web crawling system, and also give some snapshots of GUI.

This paper deals with a system which is based on web crawler using .net technology. The proposed approach is implemented in VB.NET to crawl the web pages and crawled data is stored in central database (Sql Server). The duplicacy of record is checked through stored procedure which is pre complied & checks the result very fast. The proposed architecture is very fast and allows many crawlers to crawl the data in parallel.

Acknowledgment

The author especially thanks to mention the financial help provided by MANF SRF under University Grant Commission (UGC), New Delhi.

References

[1] Barry M. Leiner, Vinton G. Cerf, David D. Clark, Robert E. Kahn, Leonard, Kleinrock, Daniel C. Lynch, Jon Postel, Larry G. Roberts, Stephen Wolff, “A Brief History of the Internet”, www.isoc.org/internet/history.

[2] Arvind Arasu, Junghoo Cho, Hector Garcia-Molina, Andreas Paepcke, Sriram Raghavan "Searching the Web." ACM Transactions on Internet Technology, vol. 1, no. 1, pp. 2-43, 2001.

[3] J. Cho and H. Garcia-Molina, “Parallel crawlers.” In Proceedings of the Eleventh International World Wide Web Conference, 2002, pp. 124 - 135, 2002.

[4] Altavista, Mar. 2008. URL www.altavista.com.

[5] A. Heydon and M. Najork, “Mercator: A scalable, extensible web crawler”, World Wide Web, vol. 2, no. 4, pp. 219-229, 1999.

[6] D. Fetterly, M. Manasse, M. Najork, and J. Wiener. “A large-scale study of the evolution of web pages”, In Proceedings of the twelfth international conference on World Wide Web, Budapest, Hungary, pp. 669-678. ACM Press, 2003.

[7] V. Shkapenyuk and T. Suel, Design and implementation of a high-performance distributed Web crawler. In Proceedings of the 18th International Conference on Data Engineering (ICDE'02), San Jose, CA Feb. 26--March 1, pp. 357-368, 2002.

[8] O. Papapetrou and G. Samaras, “Minimizing the Network Distance in Distributed Web Crawling.” International Conference on Cooperative Information Systems, pp. 581-596, 2004.

[9] J Cho, H. G. Molina, Lawrence Page, “Efficient Crawling Through URL Ordering”, Computer Networks and ISDN Systems, vol. 30, no. (1-7), pp. 161-172, 1998.

[10] J. Cho and H. G. Molina, “The Evolution of the Web and Implications for an incremental Crawler”, In Proceedings of 26th International Conference on Very Large Databases (VLDB), pp. 200-209, September 2000.

[11] Md. Abu Kausar, V S Dhaka and Sanjeev Kumar Singh. “Web Crawler Based on Mobile Agent and Java Aglets” I.J. Information Technology and Computer Science, vol. 5, no. 10, pp. 85-91, 2013.

[12] Md. Abu Kausar, V S Dhaka and Sanjeev Kumar Singh. “An Effective Parallel Web Crawler based on Mobile Agent and Incremental Crawling” Journal of Industrial and Intelligent Information, vol. 1, no. 2, pp. 86-90, 2013.

[13] G. Pant, P. Srinivasan, and F. Menczer. “Crawling the Web.” In M. Levene and A. Poulovassilis, editors, Web Dynamics. Springer, 2004.

[14] Andrei Z. Broder, Marc Najork and Janet L. Wiener “Efficient URL Caching for World Wide Web Crawling”, WWW 2003 , May 20–24, 2003, Budapest, Hungary.

[15] Md. Abu Kausar, V S Dhaka and Sanjeev Kumar Singh. “Web Crawler: A Review.” International Journal of Computer Applications, vol. 63, no. 2, pp. 31-36, 2013.

Authors’ Profiles

Md. Abu Kausar received his BCA degree from T. M. Bhagalpur University, Bhagalpur in 2002, Master in Computer Science from G. B. Pant University of Agriculture & Technology, Pantnagar, Uttrakhand, India in 2006 and MBA (IT) from Symbiosis, Pune, India in 2012. He has received Microsoft Certified Technology Specialist (MCTS).

At present, he is pursuing Ph.D in Computer Science from Jaipur National University, Jaipur, India and he is receiving UGC MANF SRF Fellowship during Ph.D Programme.

He is having 8 years of experience in Software Development and research. His research interest includes Information Retrieval and Web Crawler**.**

Dr. V. S. Dhaka is a young and dynamic technocrat with 10 years of intensive experience in industry and academia. He is M.Tech and Ph.D in computer Science from Dr. B R Ambedkar University, Agra, India. With more than 32 publications in international journals and paper presentations in 27 conferences/seminars, he always strives to achieve academic excellence.

He has been awarded by the employers with "Employee of the Quarter Award", "Mentor of the year award" and with letters of appreciations for his commitment, advocacy and mentor-ship. He has organized several Conferences, Seminars and Workshops.

Dr. Sanjeev Kumar Singh is working as Assistant Professor in Department of Mathematics at Galgotias University, Gr. Noida, India. He earned his M. Sc. and Ph.D. degrees with major in Mathematics and minor in Computer Science from G.B. Pant University of Agriculture and Technology, Pantnagar, Uttrakhand, India. Before that he completed B.Sc. (Physics, Mathematics & Computer Science) from Lucknow University, Lucknow.

He is having more than nine years of teaching and research experience. Besides organizing three national conferences, he has published several research papers in various International/National journals of repute and several national and international conferences and workshops. His areas of interest include Mathematical Modeling, Differential Geometry, Computational Mathematics, data mining & Information Retrieval.