International Journal of Emerging Engineering Research and Technology
Volume 3, Issue 8, August 2015, PP 27-36
ISSN 2349-4395 (Print) & ISSN 2349-4409 (Online)

An Automated Programming Tool for Archiving and Interfacing Web Contents from Static and Dynamic Sites

Richa Bhatt[1], Harshitha P[1], Kalyani Jha[1], Bhakthavathsalam R[2], Gowranga K H[2], Saqquaf S M[2]

[1]*Department of Computer Science and Engineering, M.S Engineering College, Bangalore, India*
[2]*Supercomputer Education and Research Centre, Indian Institute of Science, Bangalore, India*

ABSTRACT

As web technologies evolve, most of the web contents and web pages can be archived from websites. This paper mainly focuses on the interface which clones the website and the contents of a website can be mapped into other websites. It can also be used to manipulate website contents and viewed offline. It is an all-purpose, high speed website downloader to save data. It can download website files to your hard drive for offline browsing, extract website files of a certain size and type, like image, video, picture, movie and music, retrieve a large number of files as a download manager with resumption support, and mirror sites. Both Dynamic sites as well as the Static sites can be downloaded and archived.

Keywords: Archiving, Mapping, Cloning, Web server, Web client, Website.

INTRODUCTION

The web content archiving includes making an immutable copy of web records from a point in time. The copy can consist of a collection of related files and associated metadata. The software downloads the web page contents. It can be used to manipulate the website contents. Also the saved Website Template can be used to display the content user wishes to add.

It is the website download tool that can resume broken downloads from HTTP, HTTPS and FTP connections, access password-protected sites, support Web cookies, analyze scripts, update retrieved sites or files, and launch more than fifty retrieval threads. With capabilities to update, retry, edit, delete, browse, and copy each task file type, size, name, URL, exploration depth, and server, it is a fully configurable, automated and multi-threaded. It can download both Dynamic sites as well as the Static sites.

A dynamic website is a website that not only uses HTML and CSS, but includes website scripting as well. It's not a full-blown web app like Facebook or Google, but it does have interactive elements like contact forms and search boxes. This website also shares the same HTML code for the header, menu and sidebar between all pages of the site A static website is the simplest kind of website you can build. The archiving of websites encompasses both static and dynamic content. Current websites vary from static digital publications to interactive dynamic websites that are used for services or transactions. Dynamic websites may well contain a number of static pages. Static websites are written in HTML and CSS only, with no scripting. The only form of interactivity on a static website is hyperlinks. If you intend your website to be a small one, then a static website might be the easiest way to go. But if you want to share elements between pages, you'll have to duplicate the HTML on each page.

Address for correspondence:

bhaktha@serc.iisc.in

Static websites are easier to make than dynamic websites, because they require less Coding and technical knowledge. However, fully static websites are very uncommon these days, since there is so much that scripting can do. Hence, there are three ways to copy entire websites the first, server-side archiving is the hard one. The second choice is to have this done close to the server and record all transactions (transactions archiving). The last one is to automatically collect the delivered information directly from websites, as a regular browser would do (client-side archiving).

LITERATURE SURVEY

It descends from an earlier program named GetURL by the author D. Gibson the development of which commenced [1]. The name changed to Wget after the author became aware of an earlier Amiga program named GetURL, written by J. Alpert in AREXX [2]. A review of published surveys of digital preservation initiatives in general will help situate the results of this narrower study within a broader context [3], [4]. Also, we discussed the published reports of universities that are archiving Web sites in some capacity. Finally, the professional information seeking practices of librarians and archivists are explored [5].use to find information about Web site archiving, this study can be considered a subtopic within the body of work described here [6].

It filled a gap in the web-downloading software available in the mid-1990s. No single program could reliably download files via both HTTP and FTP. Existing programs either only supported FTP (such as NcFTP and dl) or were written in Perl, which was not yet ubiquitous [7], [8]. While It was inspired by features of some of the existing programs, it aimed to support both HTTP and FTP and to enable the users to build it using only the standard development tools found on every Unix system [9]. At that time many users struggled behind extremely slow university and dial-up Internet connections, leading to a growing need for a downloading agent that could deal with transient network failures without assistance from the human operator. In 2010 US Army intelligence analyst PFC Chelsea Manning used it to download the 250,000 U.S [10] diplomatic cables and 500,000 Army reports that came to be known as the Iraq War logs and Afghan War logs sent to Wiki leaks.

EXISTING SYSTEM

Wget is a free utility for non-interactive download of files from the Web. It supports HTTP, HTTPS, and FTP protocols, as well as retrieval through HTTP proxies. GNU Wget is a free network utility to retrieve files from the World Wide Web using HTTP and FTP, the two most widely used Internet protocols. It works non-interactively, thus enabling work in the background, after having logged off. The recursive retrieval of HTML pages, as well as FTP sites is supported -- you can use Wget to make mirrors of archives and home pages, or traverse the web like a WWW robot (Wget understands /robots.txt).

Wget works exceedingly well on slow or unstable connections, keeping getting the document until it is fully retrieved. Re-getting files from where it left off works on servers (both HTTP and FTP) that support it. Matching of wildcards and recursive mirroring of directories are available when retrieving via FTP. Both HTTP and FTP retrievals can be time-stamped, thus Wget can see if the remote file has changed since last retrieval and automatically retrieve the new version if it has. Wget supports proxy servers, which can lighten the network load, speed up retrieval and provide access behind firewalls.

Some of the Disadvantages are: It is hard to determine the boundaries of a website using Wget. Most programs offer the possibility to limit a snapshot to all files that can be found within the same start URL, but folders outside it will then not be stored. Another problem is caused by the 'redirects' that occur in many websites. Most programs ask to determine the number of levels (depth of the links) to be stored. Before an exact number is indicated here, the number of levels needs to be known as otherwise part of the website will not be cloned. The second layer of roll-over images, server-sided image maps, DTDs and XSL style sheets are not always stored. Current off-line browsers have great difficulty to put websites with Flash applications locally.

PROPOSED SYSTEM

The Downloading interface is conveniently designed to download Internet websites exactly the way you want them, The Programming used is c# and .NET. The program can download up to 100 files at a time, which saves you a huge amount of time compared to ordinary browsers. All data retrieved are stored in the directory you select. Major options include Stop the download at any time and Browse the offline website at any point of time even between downloads etc. This means you can choose to download only the first few pages of any given site, while weeding out or skipping over the stuff you don't need. Data can be mapped into Templates of other websites.

The Main task is copying tool architecture is the robot responsible for gathering data from online websites – HTML pages, images, style sheets, and in general any media file available on the server. A stack of URLs to be collected initially filled with one or more "root" addresses of HTML pages given by the user, is used by the robot which connects to respective servers, sending requests, and handling downloads. This robot can work in parallel of the parser to improve performances, while the parser is scanning pages; the crawler downloads data using multiple connections, dispatching ready files to the parser.

The crawler/parser interactions can be summarized in the following diagram: these two processes share a common bucket of links – the heap – filled by the parser as new links are being discovered, and emptied by the crawler as new links are being successfully downloaded (Figure 1). Link URLs are not the only information that will have to be passed back to the crawler: the "tag context", such as whether it is an "embedded" resource or not, will also be important to take the decision "take this link or not." Advantages are: Easy to use, Reliability, Extensibility, HTTP, HTTPS and FTP protocols support, Web Authentication Support, Web Cookies Support, Powerful Web page and download filters, File Manager.

Re-crawling exhaustively a website in the purpose of having an up-to-date copy is a very inefficient method: waste of bandwidth, waste of time, and waste of storage space when handling multiple site versions. A regular browser usually stores recently pages and associated files in a specific location called "cache," which can be used to avoid retransferring data that was previously consulted. The diagram below gives a rough idea of a caching mechanism (Figure 2). When fetching a link, the crawler first checks whether it is already known by the local cache. If not, the file is downloaded as usual. But if the cache already has a version, the crawler can either decide to ask the server for freshness, or to directly take the existing cached file – when recovering an interrupted or crashed or mirror session. The cache can store files such as images or binary data, or reference their location if they already exists on the mirror file tree: in this case, original HTML data needs to be stored "as is" anyway in the cache, because the existing files which were modified by the parser and can no longer be used effectively due to URL mangling.

Websites with Static Content

A web application's content can be of two types: dynamic or static. Dynamic content is anything that requires some type of processing to be generated (e.g. a PHP script or Java Server Page), whereas static content is something that will never or occasionally change (e.g. a JavaScript library or HTML file).

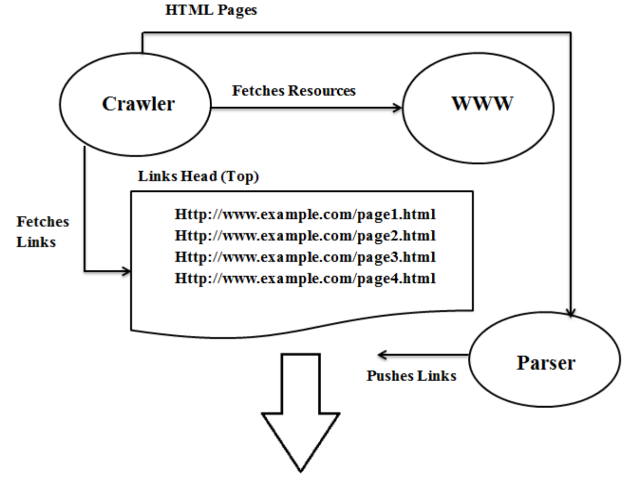

HTML Pages
Crawler
Fetches Resources
WWW
Links Head (Top)
Fetches Links
Http://www.example.com/page1.html
Http://www.example.com/page2.html
Http://www.example.com/page3.html
Http://www.example.com/page4.html
Parser
Pushes Links

Figure1. *Crawler/parser interaction*

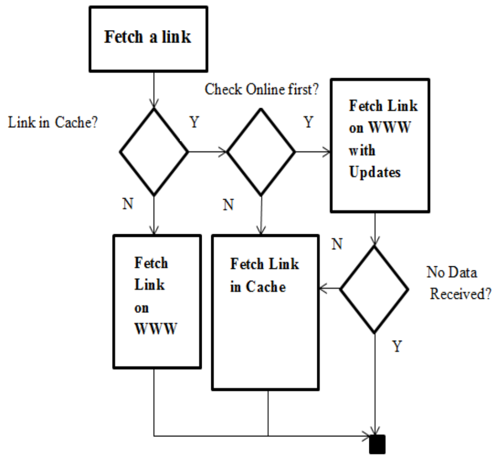


Figure2. *Flowchart for updating websites using Caching Mechanism.*

When a request is made for either type of content, a web server performs the execution needed to dispatch dynamic content or dispatch static content contained in a file. This process is illustrated in figure (Figure 3). Web server performs the execution needed to dispatch dynamic content or dispatch static content contained in a file.

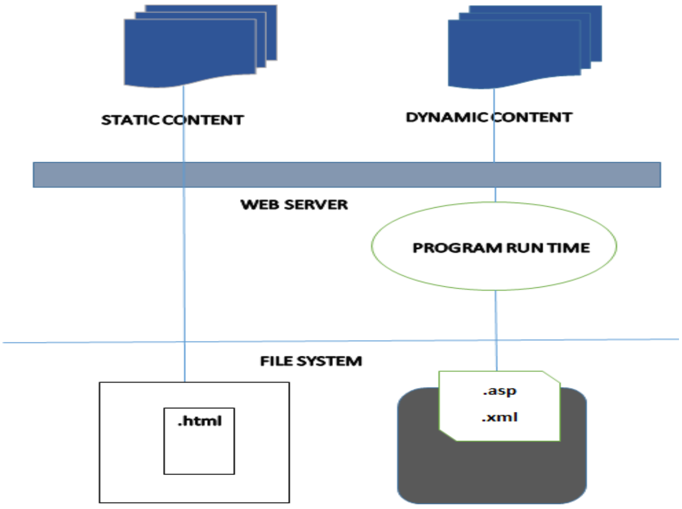


Figure3. *Dynamic and Static Content Requests.*

Type of static content refers to data like images, Java script libraries, and Cascading Style Sheets (CSS) or HTML snippets like copyrights or menus. In essence, a request made to an application page named index.jsp (dynamic), can in fact result in multiple requests for static content references inside the web page. The below diagram shows the static archiving which mainly involves the web server and the web browser for archiving the static contents (Figure 4).

Website with Dynamic Contents

The selection issue is more complex when archiving websites with dynamic content. The HTML pages here are only composed after the server receives a HTTP request or are being delivered via a linked application. The content of a web page can depend on the received query (for example consulting the timetable of the trains), the user profile or user preferences (for example via a cookie) or based on the information present in the linked document management system or the database. The below diagram (Figure 5) shows the dynamic content archiving which includes the web server with an application server and web browser to clone contents which are dynamic from the website.

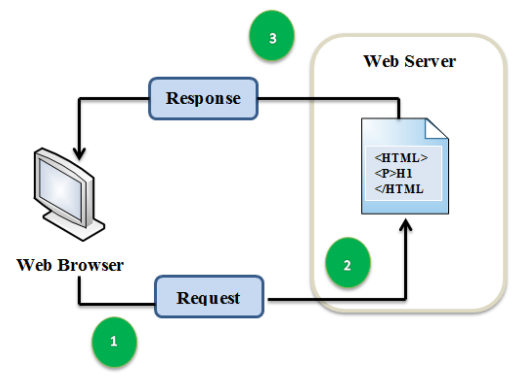


Figure4. *Archiving Static Contents from Website*

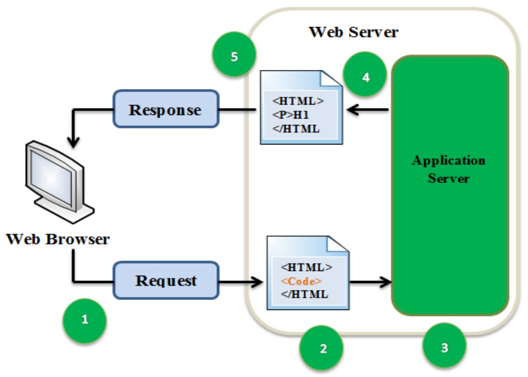


Figure5. *Archiving Dynamic Contents from Website*

IMPLEMENTATION

The flowchart given below (Figure 6) explains about parsing the html to collect links for downloading the required website. After parsing the html file, parsing the scripts like JavaScript which is found in the page. Authenticating session and redirecting it to connectivity which fetches data online. Only the HTML pages in the active window are stored locally. The result is that only one web page at a time is put off-line. To store a website that exists of multiple pages, each page has to be put off-line separately. Sometimes only the frame set is stored and not the web pages with the content. Log files with archival value will be kept as flat text files or stored in a database on the web/file server and will be cloned accordingly.

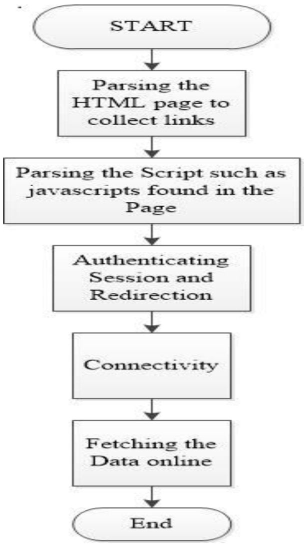

START
Parsing the HTML page to collect links
Parsing the Script such as javascripts found in the Page
Authenticating Session and Redirection
Connectivity
Fetching the Data online
End

Figure6. *Flowchart of Web Cloning*

A website's download files with archival value should preferably be transposed into a suited archiving format. Transposition causes the files to receive a new extension and that thus the links need to be adapted. Transposing the audio-visual streaming files into normal audio-visual files should be considered. These last file formats are usually more standardized and this could avoid another instance of software dependency.

Mapping the Contents of the Website

For archiving and mapping data by using an XML Schema a structured definition for an clone document that deals with file data and database data in separate sections an appropriate XML Schema was developed.

In addition to the required sections for storage of the different types of data on the cloned website, the storage of information about the source system at the time of archiving (system Metadata) may be useful. These Metadata may be helpful for discovering incompatibilities when it is necessary to recover the website after a long period of time. Figure 5.2 shows the three important sections of the XML Schema for the archiving process, which are: Metadata, Mapping of File Data and Mapping of Database Data. Since the procedure of archiving should ideally be usable on both static and database-driven websites, the sections containing the mapping of file and database data may, unlike the Metadata, be empty within the constructed clone document.

Clone Metadata

The Metadata section is required and contains processing and storage information. The processing information consists of details regarding the source system, e. g., the underlying operating system and the localization of the system.

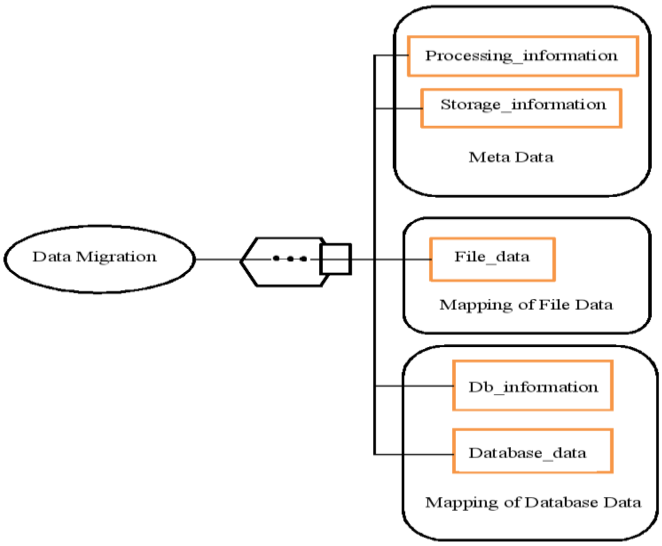


Figure7. *Mapping of Cloned Data*

This information is acquired at the beginning of the ingestion process and ensures that the most important properties of the host system are available, as they may be helpful for a recovery.

Mapping File Data

It is necessary to preserve the filename and the relative path to the file in order to recover a website's cloned file data. This information is stored as attributes of the file element located in the file data element shown in Figure 7. The file data are stored within the file element. Files may consist of printable and non-printable characters therefore mapping into a transfer format is required. Mappings like these are currently in use at many points, e.g., email attachments. BASE64 encoding is used to accomplish the mapping. During the mapping process at ingestion, each file is encoded into BASE64 data and put into the clone document.

Mapping Database Data

Since database-driven websites typically use relational databases, the procedure described here focuses on this type of database even though the mapping of other types is generally possible. As discussed earlier in Figure 7, the mapping of databases consists of the elements database information and database data. The element database information preserves basic vendor information from the database such as vendor name and version number. These data are preserved to keep the main properties of the database system at the time of the ingestion, but it is not necessary for recovering the database data.

The section database data contains the database data itself. These data's are arranged in several elements as shown in Figure 8. The element database provides information about the cloned database (size, amount of tables and table rows) which can be used for checking after recovering. The cloned database data are stored in multiple table elements which may consists of multiple row elements. Every row element may consist of multiple column elements.

The result of processing file and database data is an clone document that meets the requirements defined in the XML Schema. Since all data, files and database content are mapped with the BASE64 algorithm, the clone document consists only of printable characters.

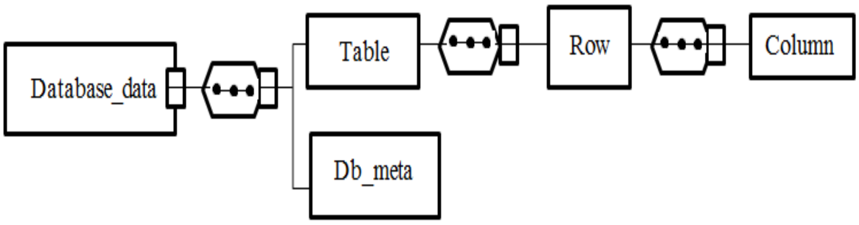


Figure8. *Storage of database data*

Since the recover procedure involves processing an XML document, it can be done using various freely available XML Parsers. The XML Parser reads the clone document line by line and executes different tasks based on the XML Element with its attributes. While processing file's data, the path to the file and its name is read followed by the encoded file content. The file content will be decoded and stored with the read filename at the read path.

During the processing of database data, tables and rows are processed as they are found in the clone document. Finding the 'Element-Start' of a table element would cause the creation of the required table with its columns and references. Embedded in the 'Element-Start' and 'Element-End' of the table elements containing the table are rows with the column data. The restore process of tables is done row by row.

RESULT AND ANALYSIS

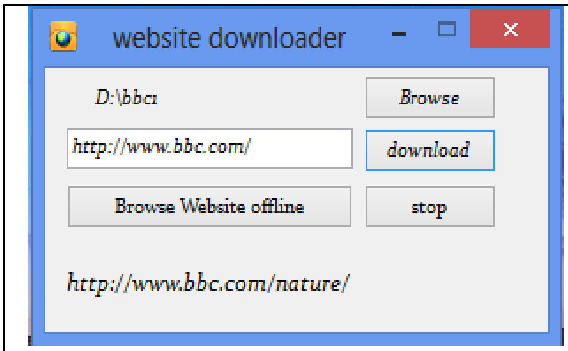
This allows you to download entire websites and download web pages to your local hard drive.it combines powerful features and a convenient interface. The Figures 9 and 10 show the downloader and the downloaded website. This permits you to quickly specify the website download settings. The downloading process can be paused or stopped any time and resumed later. It is a fast and convenient with easy navigation and simple interface. Capable of downloading files simultaneously, this can save a website to your hard drive completely or partially after the website has been downloaded, you can use to view it as offline downloaded web pages in it. HTTP and FTP Supports the main Internet protocols.

Downloads websites via the secure protocol HTTPS (SSL), supports connections via proxy servers (HTTP, FTP and Socks proxy) Supports HTTP and FTP authentication. It has been designed for robustness over slow or unstable network connections. If a download is not complete due to a network problem, it will automatically try to continue the download from where it left off, and repeat this until the whole file has been retrieved. It was one of the first clients to make use of the then-new Range

HTTP header to support this feature. It can optionally work like a web crawler by extracting resources linked from HTML pages and downloading them in sequence, repeating the process recursively until all the pages have been downloaded or a maximum recursion depth specified by the user has been reached.

The downloaded pages are saved in a directory structure resembling that on the remote server. This "recursive download" enables partial or complete mirroring of web sites via HTTP. Links in downloaded HTML pages can be adjusted to point to locally downloaded material for offline viewing. When performing this kind of automatic mirroring of web sites, It supports the Robots Exclusion Standard (unless the option -e robots=off is used). When downloading recursively over either HTTP or FTP, It can be instructed to inspect the timestamps of local and remote files, and download only the remote files newer than the corresponding local ones. This allows easy mirroring of HTTP and FTP sites, but is considered inefficient and more error-prone when compared to programs designed for mirroring. On the other hand, it doesn't require special server-side software for this task.

Written in a highly portable style of C with minimal dependencies on third-party libraries, It supports download through proxies, which are widely deployed to provide web access inside company firewalls and to cache and quickly deliver frequently accessed content. It makes use of persistent HTTP connections where available. IPv6 is supported on systems that include the appropriate interfaces. SSL/TLS is supported for encrypted downloads using the Open SSL or Gnu TLS library. Files larger than 2 GB are supported on 32-bit systems that include the appropriate interfaces. Download speed may be throttled to avoid using up all of the available bandwidth.



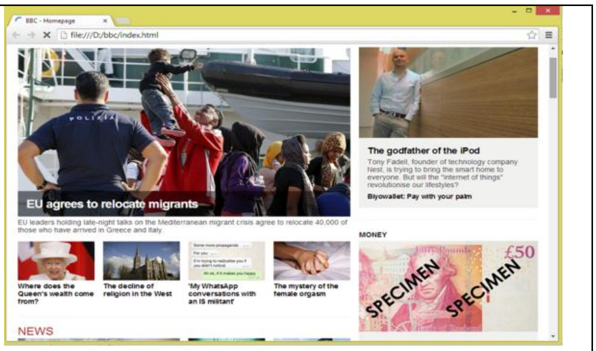
website downloader

D:\bbc1

http://www.bbc.com/

http://www.bbc.com/nature/

Figure9. *Downloader*



BBC - Homepage

file:///D:/bbc/index.html

EU agrees to relocate migrants

EU leaders holding late-night talks on the Mediterranean migrant crisis agree to relocate 40,000 of those who have arrived in Greece and Italy

The godfather of the iPod

Tony Fadell, founder of technology company Nest, is trying to bring the smart home to everyone. But will the "internet of things" revolutionise our lifestyles?

Biyowallet: Pay with your palm

MONEY

Where does the Queen's wealth come from?

The decline of religion in the West

'My WhatsApp conversations with an IS militant'

The mystery of the female orgasm

NEWS

Figure10. *Downloaded Website*

The Figure 11 shows the amount of completion of websites which is downloaded in terms of percentage. This chart mainly helps to know whether the downloading process of the website is complete or incomplete. There are three downloaded websites in the figure which is used for the comparison.

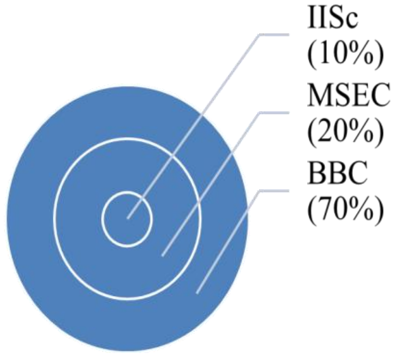


Figure11. *Chart shows the Percentage Outline of Downloaded Websites*

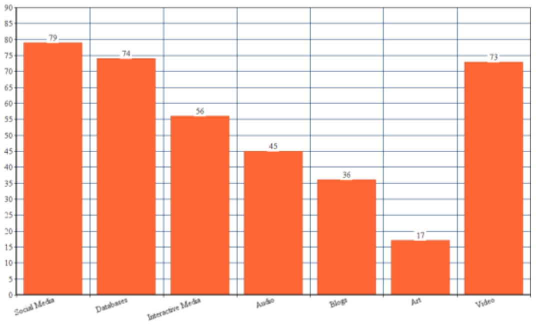


Figure12. *Bar Chart Shows the Content over Capacity*

There are varieties of web contents in each website which vary in the capacity. Based on the capacity of each web contents the downloading speed of the website may differ depending upon capacity. The diagram (Figure 12) shows that the content which is present in the cloned website and the capacity of such content varies from one to another.

Table1. *Result analysis of downloaded website status*

Sl. no	Website	Downloaded Status	Layer
1	http://www.bbc.com/	Complete	3
2	http://www.iisc.ernet.in/	Complete	2
3	http://www.msec.in/	Complete	5

The table (Table 1) also shows whether the downloading process of the website is complete or incomplete. It also shows how many layers of downloading process are completed which may help the user to know about the process. This table mainly focuses on the status which shows whether process of downloading website. There are the numbers of websites which can be cloned and can be used offline. The contents which are downloaded from the website may consist of the audio, video, social media and blogs .These contents may vary in the capacity and differ from each other. Each of the content may have higher capacity and occupy very large amount of memory and the lesser capacity may occupy less amount of memory.

CONCLUSION

Preserving, archiving websites and mapping the contents of the websites are a thrilling technical challenge which must be pursued, endlessly. Evolving technologies, evolving contents and evolving growth of the WWW means that no definitive archiving solution might ever be found. Existing solutions can still be improved as no perfect system was yet made. This continuous work in improving Web preservation techniques is done by passionate people around the world, in libraries, universities, companies and by individuals.

ACKNOWLEDGEMENT

The authors sincerely thank the authorities of Supercomputer Education and Research Center, Indian Institute of Science for the encouragement and support during the entire course of this work.

REFERENCES

[1] D. Gibson, K. Punera, and A. Tomkins, "The volume and evolution of Web page templates", WWW, 2014.J. Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[2] J. Alpert and N. Hajaj, "We knew the web was big", http://googleblog. blogspot. co.uk/ 2008/ 07/we-knew-web-was-big.html, 2008.M. Wegmuller, J. P. von der Weid, P. Oberson, and N. Gisin, "High resolution fiber distributed measurements with coherent OFDR," in Proc. ECOC'00, 2000, paper 11.3.4, p. 109.

[3] W.Cathro, C.Webband, J.Whiting, "ArchivingtheWeb:The PANDORA Clone at the National Library of Australia", Denmark, 2001.M. Shell. (2002) IEEEtran homepage on CTAN. [Online]. Available: http://www.ctan.org/tex-archive/macros/latex/contrib/supported/IEEEtran/

[4] S. Song and J. JaJa, "A New Technique for Archiving Temporal Web Information", UMIACS Technical Report- In Preparation, University of Maryland Institute for Advanced Computer Studies, 2008..

[5] J. Masanes, "Web Archiving: Issues and Methods, Web Archiving", Springer, Berlin, pp. 153, 2006..

[6] R. Bayer and E. M. McCreight, "Organization and Maintenance of Large Ordered Indexes", Acta informatica, 1, pp. 173-189, 1972.

[7] B. Becker, S. Gschwind, T. Ohler, B. Seeger and P. Widmayer, "An Asymptotically Optimal Multiversion B-tree", The VLDB Journal, 5 ,pp. 264-275,2006.Matlock, H., and Reese, L.C., 1960, Generalized solutions for laterally loaded piles., Journal of Soil Mechanics and Foundation, 86(5), 63–91.

[8] J. Hirai, S. Raghavan, A. Paepcke and H. Garcia-Molina, "WebBase : A repository of Web pages", The 9th International World Wide Web Conference (WWW9), Amsterdam, 2000.

[9] M. Faheem, P. Senellart, "Intelligent and adaptive crawling of Web applications for Web archiving", ICWE, 2013.

[10] Y. Diao, M. Altinel, M. J. Franklin, H. Zhang, and P. Fischer. "Path sharing and predicate evaluation for high-performance XML filtering", ACM TODS, 2003.

AUTHORS' BIOGRAPHY

Richa Bhatt received her B.E degree in Computer Science and Engineering from M S Engineering College, Visvesvaraya Technological University. Her research interests are Artificial Intelligence and Web Technologies.

Harshitha P obtained her B.E degree in Computer Science and Engineering from Visvesvaraya Technological University, India. Her Research interests include Sensor Networks and Robotics.

Kalyani Jha received her B.E degree in Electronic and Communication Engineering from Visvesvaraya Technological University, India. Her research interests are in the areas of Sensor Networks and Wireless Technologies.

Bhakthavathsalam R is presently working as a Principal Research Scientist in SERC, Indian Institute of Science, Bangalore, India. His areas of research interests are Pervasive Computing and Communication, Wireless Networks and Electromagnetics with a special reference to exterior differential forms. Author held the position of Fellow of Jawaharlal Nehru Centre for Advanced Scientific Research during 1993 - 1995.

Gowranga K H is currently working as Senior Scientific Assistant in Supercomputer Education and Research Center, Indian Institute of Science, Bangalore, India. He is actively involved in maintaining the Web and Mail servers in the campus. His research interests include Wireless Networks, Webmail Systems and Digital Communication.

Mr. Saqquaf S M graduated in Electronics and Communication Engineering and did his post graduate studies in Computer Science and Computer Management. Presently working as Senior Technical Officer, looking after overall Infrastructure and coordinating Computer System Admin Training program at SERC. His fields of interests are Power Management, Captive Power Generation, Power Auditing, UPS Systems and Campus Communication Network