CHAPTER-1
INTRODUCTION

1.1 OVERVIEW

The Internet is a network of networks, joining many government, educational institutions, private computers etc. together for providing an infrastructure for the use of E-mail, bulletin boards, file archives, hypertext documents, databases and other computational resources. The vast collection of computer networks which form and act as a single huge network for transport of data and text messages across distances which can be anywhere within the office or office to anywhere in the world. It is a global information system that is logically linked together by a globally unique address space based on a protocol known as the Internet Protocol (or its subsequent extensions). It is the biggest repository of online knowledge wherein the end-user employs the tool for searching and sharing information. The Internet has revolutionized the computer and communications world like never before. Within a span of few years, it has changed the way of business and communication. Today, it has become a world-wide broadcasting capability, a mechanism for information dissemination, and a medium for collaboration and interaction between individuals and their computers irrespective of geographic location.

Today use of Internet has become an integral part of human life. The current population of the world is approximately 7.02 billion out of which approximately 2.41 billion (34.3%) use Internet [1]. In fact from .36 billion in 2000, the number of Internet users has increased to 2.41 billion in 2012 i.e. an increase of 566.4% over a span of 12 years. Same growth rate is expected in future too. In Asia alone, around 1.07 billion people use Internet that is approximately 44.8% of worldwide Internet users. In India approximately 0.137 billion (out of 1.2 billion) people use Internet. Thus it is not far away when one will start feeling that life is incomplete without Internet.

1.2 THE WORLD WIDE WEB

The World Wide Web (WWW or web) [2] is a web of hyperlinked [3] repository of trillions of hypertext documents lying in different websites, distributed over far end distant geographical locations. In fact, the size of web is so enormous that it is frustrating
and tedious task to search the right information at the right time. With a web browser a user views web pages containing text, images, videos, multimedia documents etc. and navigates between them using hyperlinks. Viewing a web page normally begins either by typing the Uniform Resource Locator (URL) of the page into a web browser, or by following a hyperlink to that page or resource. The web browser then initiates a series of communication messages, behind the scene, in order to fetch and display them. With such a huge web repository, finding the right information at the right time is a challenging task. Hence, there is a need for a very effective way of retrieving information from the web.

### 1.3 SEARCH ENGINES

Web search engines [4, 5] act as a bridge between web users and web pages. Without search engines, the unlimited source of information stored in web pages remain hidden for us. A search engine is a coordinated set of programs that is able to read every searchable page on the web, create an index of the information it finds, compare that information to a user's search request (i.e. query), and finally return the results back to the user. The invention of this technology has granted users quick and easy access to the knowledge they seek; by essentially categorizing web pages according to their relevancy in regards to a request, or query. A general web search engine has three parts a crawler, indexer and query engine as shown in Figure 1.1.

![Figure 1.1: Architecture of a General Search Engine](image-url)
Web crawlers continuously collect web pages from the web, index and store them in a database. This continuous updation of database renders a search engine more reliable source of right and updated information. The need of maintaining the up-to-date pages in the collection causes a crawler to revisit the websites again and again. Due to this, the resources like CPU cycles, disk space, and network bandwidth, etc., become overloaded and sometime a website may crash due to such overloads on resources.

Study [6, 7] reports that the current web crawlers have indexed billion of pages and about 40% of current Internet traffic and bandwidth consumption is due to the web crawlers. The maximum web coverage of any popular search engine is not more than 16% of the current web size. It is very difficult to search relevant information from such huge collection of web documents as the web pages are neither organized as books on shelves in a library, nor completely catalogued at one central location. It is not guaranteed that users will be able to retrieve information even after knowing where to look for information by knowing its URLs as web is constantly changing. Therefore, there is a need to develop efficient information retrieval tools to search the required information from the web as quickly as possible.

1.4 MOTIVATION

In order to download the huge number of web documents from the web, a highly scalable and efficient crawling system is needed. The traditional centralized crawling techniques are unable to cope up with constantly growing web. Some of the issues to be kept in mind while designing a scalable and efficient crawling system are as follows:

- **What to download?** The web is not exactly a distributed hypertext. It is impossible to organize and add consistency to the data and the hyperlinks. Web pages are not well structured and semantic redundancy can also increase traffic. Moreover, a lot of web pages do not involve any editorial process which means that data can be false, inaccurate, outdated or poorly written. Data is distributed widely in the web world, and is located at different sites and platforms. The communication links between computers vary widely, and there is no topology of data organization. Documents can be added or removed easily in the web and changes to these documents remain unnoticed by others. The growth of data is exponential that causes scaling issues and are difficult to cope with. Data on the
web are heterogeneous in nature i.e. they are written in different formats, media types, and natural languages.

- **How to download?** Due to enormous size of the web, it is often imperative to run the crawling process at several locations. A single-process crawler simply cannot achieve the required download rate. In place of having single and centralized crawling process, an ecology of the crawl workers be implemented to work in parallel, so that it may complete the crawling process in time. It would be scalable in nature and can even work with growing web size.

- **How to disperse Network-load?** Multiple crawling processes of a distributed crawler may run at geographically distant locations, each downloading “geographically-adjacent” pages. For example, a process in United Kingdom may download all European pages, while another one in India crawls all Asian pages. In this way, one can disperse the network load to multiple regions. In particular, this dispersion might be necessary when a single network cannot handle the heavy load from a large-scale crawling process.

- **How to regulate the crawling frequency to maximize freshness of collection?** To keep the local collection fresh, it is necessary that the process of revisiting [8, 9, 10] of a website be managed with a view to maintain fairly fresh documents at the search engine site. The computation of refresh time dynamically [11, 12] helps in improving the effectiveness of the crawling system by efficiently managing the revisiting frequency of a website; and giving appropriate chance to each type of website to be crawled at a fast rate.

- **How to check overlapping of web documents?** The problem occurs when multiple crawlers run at far distant geographic locations in distributed manner. As one crawler running at one location may not be aware about pages downloaded by another crawler. It may also happen that multiple servers contain same copy of a document. In such cases the crawlers should avoid having duplicate copies of same document in the collection.

- **How to improve quality of the local collection and identify obsolete pages?** The incremental crawler provides quality information to the user by replacing less important pages with more important pages. It needs to employ appropriate methods to measure priority [13] of a page so that quality of local collection is
maintained. To make the local collection up-to-date, the obsolete pages i.e. pages that do not match with pages on the web.

- **How to secure migrating crawlers?** Migrating crawlers have limitations, primarily in the area of security. Two different points of view are that from the platform perspective there is a requirement to protect the host from malicious mobile agents such as viruses and Trojan horses that are visiting it and consuming its resources and the other from the mobile agent point of view it needs to protect the agent from malicious hosts. Therefore, security is a fundamental precondition for the acceptance of mobile agent applications. The system should have a program that actively protects itself against execution environment that possibly may divert the intended execution towards a malicious goal.

### 1.5 THE MOBILE AGENTS

In general, agent [14] is an autonomous entity that acts on behalf of others in an autonomous fashion, performs its actions in some level of pro-activity and reactivity, and exhibits some levels of the key attributes of learning, co-operation and mobility [15, 16]. Nwana [17] identifies seven type of agents i.e. collaborative agents, interface agents, mobile agents, information agents, reactive agents, hybrid agents and smart agents. Mobile (or migrating) agents are computational software processes capable of roaming wide area networks such as the web, interacting with foreign hosts, gathering information on behalf of its owner and reverts back having performed the duties set by its master (see Figure 1.2).

![Figure 1.2: Agent System Model](image_url)
In other words, mobile code gives users the time and flexibility to provide their users with more useful applications, each with more useful features. The key characteristics of mobile agents are migration, data acquisition, route determination and communication. Advantages of mobile agents are bandwidth, latency, asynchronous task execution, fault tolerance and peer-to-peer communication.

1.6 MIGRATING CRAWLING AGENTS

In traditional crawling the pages from all over the web are brought to the search engine site and then processed. When a page is brought to the search engine site and analysed, many a times it is found that it was not needed. In such cases the efforts made to send request to the web server and bringing the page to the search engine site seems to be useless. Moreover this results a lot of network traffic.

In the distributed crawling with migrating agents approach, agents allow packaging a conversation and dispatching it to a destination host where the interactions can take place locally. Migrating agents are also useful when it comes to reducing the flow of raw data in the network. When very large volumes of data are stored at remote hosts, these data should be processed in the locality of the data rather transferring over to the network. The main concern is to move the computations to the data rather than the data to the computations. By migrating to the location of the resource, a mobile agent can interact with the resource much faster than from across the network. This process reduces network traffic also.

1.7 SECURITY IN MIGRATING CRAWLING AGENTS

Wayne [18, 19] identifies four threat categories which are, threats stemming from an agent attacking an agent platform, an agent platform attacking an agent, an agent attacking another agent on the agent platform, and other entities attacking the agent system. The last category covers the cases of an agent attacking an agent on another agent platform, and of an agent platform attacking another platform, since these attacks are primarily focused on the communications capability of the platform to exploit potential vulnerabilities. It also includes more conventional attacks against the underlying operating system of the agent platform.
Mobile agent based crawler has some limitations in the area of security. Recent research efforts in the area of mobile agent security adopt two different points of view. Firstly, from the platform perspective, there is a requirement to protect the host from malicious mobile agents such as viruses and Trojan horses that are visiting it and consuming its resources. Secondly, from the mobile agent point of view it needs to protect the agent from malicious hosts. Therefore, security [20, 21] is a fundamental precondition for the acceptance of mobile agent applications. The system should have a program that actively protects itself against execution environment that possibly may divert the intended execution towards a malicious goal.

Many approaches aim at protecting mobile agents but there are some problems, which have to be solved before these approaches can be used. The particular attacks regarding malicious host or malicious agent can be summarized as unauthorized access, masquerading, denial of execution, annoyance attack, eavesdropping, alteration, confidentiality, integrity and availability.

1.8 CHALLENGES OF THE WEB CRAWLING SYSTEM

A critical reading to the available literature indicates that the following issues have to be addressed while designing a secured frequency regulated migrating web crawler:-

- **Network-load reduction:** Web crawlers revisit the websites again and again to download web pages for search engines to create and maintain the web indices. Due to this, the resources like CPU cycles, disk space, network bandwidth, etc., become overloaded.

  *Solution:* Migrating crawlers (mobile processes) are capable of roaming the web, interacting with web servers that host web pages, gathering information on behalf of its owner and coming back having performed the duties set by its user. Distributed migrating crawlers minimize network utilization and also keep up with document changes.

- **Managing Volatile web contents:** Due to the dynamic nature of the web, it becomes very difficult for a search engine to provide fresh information to the user.
An incremental crawler downloads modified contents only from the web for a search engine which helps in reducing the network load.

**Solution:** The migrants migrate to the web server for the purpose of downloading, filtering and compressing the documents before transferring them to the search engine side. In this work, a migrating crawling approach has been proposed wherein migrants after moving to the web servers downloads the .TVI (table of variable information) file(s) only for maintaining the freshness of search engine repository.

- **Regulating revisit frequency to maximize freshness of collection:** Since the web contains highly dynamic data, it is essential to estimate after how long contents of a page gets changed and how often a crawler should revisit the website to maintain its collection up to date. It helps in improving the effectiveness of the crawling system by efficiently managing the revisiting frequency of a website; and appropriate chance to each type of web site to be crawled at appropriate rate.

  **Solution:** The proposed architecture adjusts the frequency of revisit by dynamically assigning a priority of revisiting to a site by computing the priority based on previous experience that how many times the crawler founds changes in content in ‘n’ visits and the interest of the users shown in the websites.

- **Security in Migrating crawlers:** Since a migrating crawling agent roams around the web world and executes on a foreign platform, the security has become hindrance for development and maintenance of migrant technology. There is a need to develop secured migrants and to fix issues like maintaining security and integrity of the agent, data it carries and the remote platform on which it executes.

  **Solution:** The proposed approach presents a remote platform oriented reliability based security system approach that is helpful in maintaining security and integrity of remote platform and migrant as well as data it carries. It proposes to provide a restricted secured environment to the migrating agent and its restriction be decreased in incremental manner as reliability of the agent increases.
• **Perspective term based index construction**: Keyword searching is the most common form of text search on the web. The present search engines are unable to search for the keywords based on sense behind the words.

  **Solution**: *An improvement in the index construction is proposed, in which related terms based on perception (viewpoint) of the user are being taken into account. The index that stores keywords with their synonyms, acronyms and alternate spellings, and searches the related terms based on user perception i.e. the word for which he is intended to search. This will help the search engines to provide better results to the user based on his mental vision.*

1.9 **ORGANIZATION OF THE THESIS**

An outline of the contents of the thesis is as follows:-

**Chapter-1**: Provides general introduction of the Internet, World Wide Web, Search Engines, crawling based on migrating agents, security issues in migrating crawling agents, motivation behind designing an efficient crawling system, problem identification and a view to the proposed solution.

**Chapter-2**: This chapter presents general architecture of Internet and World Wide Web followed by naming conventions. It discusses the need of search engines to retrieve information from the web and general architecture of a search engine. It also presents design issues of a search engine and several different types of search engines. Thereafter, it presents role of web crawlers followed by various web crawling techniques.

**Chapter-3**: Provides a view to software agents and its various types. It also presents a view on using mobile agents in different area including web crawling, and various issues related with security of migrating crawling agents.

**Chapter-4**: Presents problem related with frequency of migrating the crawling agents based on user’s interest. It presents a method to dynamically compute the revisit frequency based on user’s interest.

**Chapter-5**: This chapter presents the use of migrating agents in web crawling. It also presents an architecture to crawl only the volatile web contents using Table of Volatile
Information (.TVI) and reduces the network load significantly. Thereafter it presents the need of finding the keywords based on the mental vision of the user and proposes a new approach to construct index based on perspective terms.

**Chapter-6:** This chapter presents a remote platform based reliability oriented security system that computes reliability dynamically and provides a secured environment for migrating agents, their platform and data they carry.

**Chapter-7:** This chapter covers conclusion and scope of future work in the area of research.

This follows the Bibliography of publications in this area and, the list of papers published towards the thesis.

The next chapter presents a survey on existing web search engines and crawling techniques.