2 Need For Semantic Web

Objective:

This chapter covers the following topics:

- Working of the current web.
- Web Crawlers.
- Benefits of Semantic Web.

2.1 Introduction

'Why do we need Semantic Web?' This is another question that would strike anyone who reads about Semantic Web. When people are comfortable with the current web then what's the need to switch to a new platform. The answer to this is explained in the following chapter.

The World Wide Web is not a static container of information, but it's an ever expanding ocean of facts. Every year, on an average, 51 million websites get added to the web. And this figure has meager chances of getting deprecated in the future. It's a proven fact that it will increase in the years about to come. Nowadays, almost all the organizations support open data and make their data available over the web. There was a time when innovation was confined to the four doors of innovation labs. Now is the time when the doors are open to all via open source data. No doubt that more and more information getting added to the web will make it more resourceful, but this will also pose a serious problem too in the near future. We may not know which one is the correct data to be used when we have too much information in front of us.

2.2 Simple Activity

Open Google and type the following two words-

- Joy (press 'Enter' and view the result)
- Delight (press 'Enter' and view the result)

You will get different sets of results in both the cases, though both the words mean the same. This is because the web can't understand that both the words mean the same. At the most basic level, the web pages are considered as strings of words and processed.

Now assume, I am submitting the following two queries to any of the current web's search engines-

Query 1: Domino serves well under heavy load also.

Query 2: Domino serves well under high-demand also.

Query1 is about software called as Domino (An IBM server application platform used for enterprise e-mail, messaging, scheduling and collaboration) that is capable of working well under heavy load as well whereas Query 2 is about a pizza outlet that serves well even under high demand. When submitted for processing, the search engine could not provide an appropriate result for either of the queries. The reason behind this is that the system is not smart. Semantic Web provides this smartness.

2.3 Web 2.0 approach

Semantic Web being a very complex topic, the best way to understand it is to compare and study with the applications and technologies that we are familiar with. So let's study the topic of Semantic Web with the help of our favorite search engine i.e. Google. Before moving on to the algorithm and technic used by Google, let's try to understand some of the basic concepts related to it.

2.3.1 Web Crawler

The main aim of designing a web crawler is to enable the retrieval of web pages and to add their representation to a local repository. A crawler is a program that visits web sites and reads their pages and other information in order to create entries for a search engine index. Web Crawlers are also called as a 'spider'.

Role of a Web Crawler:

- Web Crawlers roam the web with the aim of automating specific tasks related to the web.
- They are responsible for collecting the web-content.

Basic algorithm followed by Web Crawlers:

- Begin with the 'seed' page.
- Create a row/queue for the related pages.
- Retrieve the seed page and process it.
- Extract the URLs they point to.
- Create an entry in the repository.
- Place the extracted URLs in a queue.
- Retrieve each URL from the queue one by one.
- For each of the retrieved URL repeat the above step.

Types of Crawlers:

- **Batch Crawler**
  This type of crawlers crawl a snapshot of their crawl space till they reach a certain size or time limit.

- **Incremental Crawler**
  This type of crawlers keep crawling their crawl space continuously, revisiting the URL so as to ensure freshness.

- **Focused Crawler**
  This type of crawlers, as the name suggests, are quite focused on the pages they crawl. They attempt to crawl pages pertaining to a specific topic and minimize the number of pages that are out of topic and are being collected.

- **Distributed Crawler**
  Distributed web crawling is a distributed computing technique. Many crawlers work together to distribute in the process of web crawling, in order to cover maximum part of the web. A central server manages the communication and synchronization of the nodes, as it is geographically distributed. It basically uses Page rank algorithm for its increased efficiency and quality search. The benefit of a distributed web crawler is that it is robust against system crashes and other events, and can be adapted to various crawling applications.

- **Parallel Crawler**
  Multiple crawlers running in parallel are referred as Parallel Crawlers. A parallel crawler consists of multiple crawling processes called as C-procs which can run on a network of workstations. The Parallel crawlers depend on Page freshness and Page Selection. A Parallel crawler can be on a local network or be distributed at geographically distant locations. Parallelization of crawling system is very vital from the point of view of downloading documents in a reasonable amount of time.

URL Normalization:

Crawlers generally perform some type of URL normalization in order to avoid crawling the same resource again and again. It basically aims at modifying and standardizing a URL in a consistent manner.

Examples of crawlers:

- Scooter
- WebRACE
- RBSE
- Google Crawler
- Www Worm
- Web Fountain

2.3.2 Page Rank

Google uses PageRank to determine the importance of a web page. It is one of the many factors used to determine what all pages must appear in a search result. PageRank measures a web page's importance. Page and Brin's theory is that the most important pages on the Internet are the pages with the maximum number of links leading to them. PageRank considers links as votes, where a page linking to another page is casting a vote. This makes sense, because people do tend to link to relevant content, and pages with more links to them are usually better resources than pages that nobody links. PageRank doesn't stop there. It also looks at the importance of the page that contains the link. Pages with higher PageRank have more weight in "voting" with their links than pages with lower PageRank. It also looks at the number of links on the page casting the "vote." Pages with more links have less weight.

This also makes a certain amount of sense. Pages that are important are probably better authorities in leading web surfers to better sources, and pages that have more links are likely to be less discriminating about where they're linking. PageRank is measured on a scale of one to ten and assigned to individual pages within a website, not the entire website. To find the PageRank of a page, use Google Toolbar. Very few pages have a PageRank of 10, especially as the number of pages on the Internet increases.

2.3.3 How does search engine work?

Step 1:

Crawlers crawl through the web in order to gather the contents about a site that has been changed or a new web-site that has been added, periodically. As mentioned above this work is done periodically and not for each query submitted. The truth is no search engine works in real time.

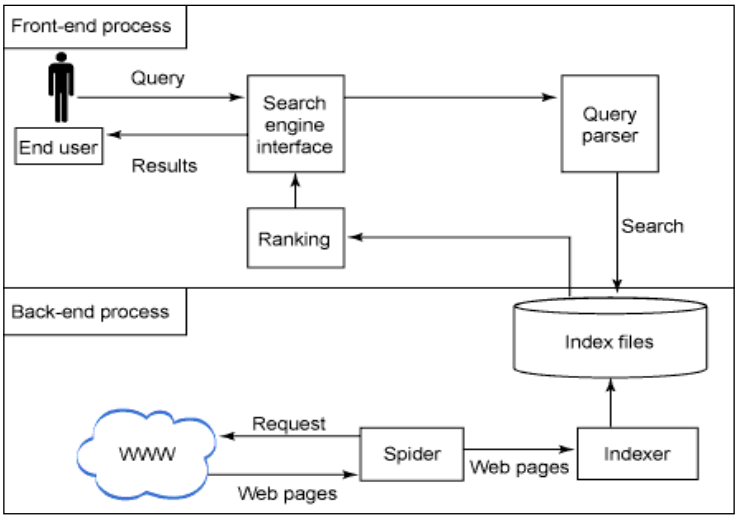

Figure 2.1: Working of Search Engine.

Note:
The crawlers do not cover the entire web. The part of the web that's not covered is called the invisible web.

Step 2:

After gathering the information, the next step is to organize the information. The pages gathered during the crawl process are organized by creating an index, so that we know how to search for it when needed again. The index may include information about words and their locations. When you search, the search terms are searched in the index to find the appropriate pages.

Step 3:

Whenever you submit a query the search engine goes back to its mammoth index library to fetch the required information. Since the search engine finds millions of matching information, it uses an algorithm to decide the order in which the result must be displayed.

2.3.4 How does Google work?

Google runs on a distributed network of thousands of computers and can therefore carry out fast parallel processing. Parallel processing is a method of computation in which many calculations can be performed simultaneously, significantly speeding up data processing. Google works in three parts:

- Googlebot, a web crawler that finds and fetches web pages.
- The indexer that sorts every word on every page and stores the resulting index of words in a huge database.
- The query processor, which compares your search query to the index and recommends the documents that it considers most relevant.

Googlebot

Googlebot is Google's web crawling robot which finds and retrieves pages on the web and hands them off to the Google indexer. It's easy to imagine Googlebot as a little spider scurrying across the strands of cyberspace, but in reality Googlebot doesn't traverse the web at all. It functions much like your web browser, by sending a request to a web server for a web page, downloading the entire page and then handing it off to Google's indexer. Googlebot consists of many computers requesting and fetching pages much more quickly than you can, with your web browser. In fact, Googlebot can request thousands of different pages simultaneously. To avoid overwhelming web servers, Googlebot deliberately makes requests to each individual web server more slowly than it's capable of doing.

When Googlebot fetches a page, it pulls all the links appearing on the page and adds them to a queue for subsequent crawling. Googlebot tends to encounter little spam because most of the web authors link only to what they believe is high-quality pages. By harvesting links from every page it encounters, Googlebot can quickly build a list of links that can cover most part of the web. This technique, known as deep crawling, also allows Googlebot to probe deep within individual sites. Because of their massive scale, deep crawls can reach almost every page in the web. Because the web is vast, this can take some time, so some pages may be crawled only once a month.

Although its function is simple, Googlebot must be programmed to handle several challenges. First, since Googlebot sends out simultaneous requests for thousands of pages, the queue of 'visit soon' URLs must be constantly examined and compared with URLs already in Google's index. Duplicates in the queue must be eliminated to prevent Googlebot from fetching the same page again. Googlebot must determine how often to revisit a page. On one hand, it's a waste of resources to re-index an unchanged page. On the other hand, Google wants to re-index changed pages to deliver up-to-date results.

Google's Indexer

Googlebot gives the indexer the full text of the pages it finds. These pages are stored in Google's index database. This index is sorted alphabetically by search term, with each index entry storing a list of documents in which the term appears and the location within the text where it occurs. This data structure allows rapid access to documents that contain user query terms.

To improve search performance, Google ignores (doesn't index) common words called stop words (such as the, is, on, or, of, how, why, as well as certain single digits and single letters). Stop words are so common that they do little to narrow a search, and therefore they can safely be discarded. The indexer also ignores some punctuation and multiple spaces, as well as converting all letters to lowercase, to improve Google's performance.

Google's Query Processor

The query processor has several parts, including the user interface (search box), the 'engine' that evaluates queries and matches them to relevant documents and the results formatter. PageRank is Google's system for ranking web pages. A page with a higher PageRank is deemed more important and is more likely to be listed above a page with a lower PageRank. Google considers over a hundred factors in computing a PageRank and determining which documents are most relevant to a query, including the popularity of the page, the position and size of the search terms within the page, and the proximity of the search terms to one another on the page. A patent application discusses other factors that Google considers when ranking a page. Visit SEOmoz.org's report for an interpretation of the concepts and the practical applications contained in Google's patent application.

Google also applies machine-learning techniques to improve its performance automatically by learning relationships and associations within the stored data. Google closely guards the formulae it uses to calculate relevance; they're tweaked to improve quality and performance, and to outwit the latest devious techniques used by spammers. Indexing the full text of the web allows Google to go beyond simply matching single search terms. Google gives more priority to pages that have search terms near each other and in the same order as the query. Google can also match multi-word phrases and sentences. Since Google indexes HTML code in addition to the text on the page, users can restrict searches on the basis of where query words appear, e.g., in the title, in the URL, in the body, and in links to the page, options offered by Google's Advanced Search Form and using Search Operators (Advanced Operators).

2.4 Semantic Web's approach

- Knowledge will be organized in conceptual spaces according to its meaning.
- Automated tools will support maintenance by checking for inconsistencies and extracting new knowledge.
- Keyword-based search will be replaced by query answering, i.e. requested knowledge will be retrieved, extracted, and presented in a human friendly way.
- Query answering over several documents will be supported.
- Defining who may view certain parts of information (even parts of documents) will be possible.

In addition to retrieving the results from a search, the way a computer does now (that is to say, systematically, taking one question and pairing it with keywords to get the user millions of answers), a Semantic Web would carry out a more human-like way of solving problems. It would connect not only from A to Z, but also from A to B to C and so on, until it reaches Z.

A Semantic Web reorganizes the vast amount of information that is accessible to us on the internet in a way similar to that of our mind. It would be like training the internet to understand the context surrounding whatever word or phrase being searched through tags the searcher attaches to the subject. The Semantic Web would serve as a connection between human and computer by making the computer think more like a human while still allowing the humans to do the real thinking. This is exhilarating and terrifying at many levels.

For example, let's say that you wanted to have lunch with your friend, Hydie. Then you might have a series of conversation with Hydie. For instance, consider the following set of conversations:

"I have a meeting in my office so cannot go tomorrow afternoon, but after 3 p.m. I am free."

"That will do. Let's go to Meluha?"

"I'm a vegetarian, so that doesn't work for me." (And so on…)

If Semantic Web technology were used in this transaction, Hydie would have an 'agent' that would have access to all kinds of information about her, including her calendar, any food preferences or allergies she might have, and restaurant ratings she's given. Your own agent would have access to similar information about you. These two agents would communicate with one another and then automatically suggest something that makes sense for both of you. They could even make the reservation for you!

More practically, researchers are using Semantic Web technologies to enable machines to infer new facts from existing facts and data. That is, Semantic Web technologies enable computers not only to store and retrieve information, but also to come up with entirely new information on their own.

2.5 Benefits of Semantic Web

- Computers can operate automatically. Since computers can make decisions like people do, they can complete work automatically, thus saving a lot of energy and money.
- Computers can also customize business systems, and companies can run the business more economically requiring less human effort.
- We can use a standardized way to store and query information efficiently.
- Data sharing can be done more easily with the Semantic Web because data warehousing can be distributed. Proper information can help people make instant and correct decisions.
- Facilitates the exchange of content and learning objects.
- Allows learners search learning resources based on semantics, thus making it easier to search their targeted knowledge.
- Improves the context-aware semantic e-learning environments by providing semantic models for context modeling.
- Scalable, reusable, sharable course content.
- The ability to find and move entire course.
- Assemble content to meet the learner's needs.