High Ended Web Search Engine Taxonomy Using Concurrent Processing

Bala Dhandayuthapani Veerasamy
Mekelle University, Mekelle, Ethiopia

dhanssoft@gmail.com

Abstract

Everyone around the world is searching for information. In the Internet world people use web search engine to look for certain topics. Basically, search engines are more helpful for everyone to search their needs such as Google, Yahoo, and AltaVista. Sometimes, searching same topic on different search engine will produce different results. Thus, we have a novel on high ended search engine taxonomy to meet more and clear information. This paper proposes to exercise one or more search key and search engines on single web search.

Keywords: Concurrent, Multithread, Process, Search engine, Search key, Task.

Introduction

Processor, program and process are three vital elements of any computer execution. Processor is a hardware device, which executes machine instructions. Program is a sequence of instructions awaiting execution. It is a completely static description of what would happen on computer. Process is a program in execution on a processor. It is an active entity in the system, which executes the algorithm defined by a program.

Typically, Tasks are logically discrete section of computational work. A task is typically a program or program-like set of instructions that is executed by a processor. Concurrent programming is just a computer program (Veerasamy, 2010a, 2010b), which involves more than one process. Two events are said to be concurrent if they occur within the same time interval. Two or more tasks executing over the same time interval are said to execute concurrently. Tasks that exist at the same time and perform them in a same time period are concurrent.

Concurrent tasks can execute in a single or multiprocessing environment. In a single processing environment, concurrent tasks exist at the same time and execute within the same time period by context switching. In a multiprocessor environment, if enough processors are free, concurrent tasks may execute at the same instant over the same time period.

Concurrency techniques are used to allow a computer program to do more work over the same time period or time interval. Rather than designing the program to do one task at a time, the program is broke down into tasks that can be executed concurrently. In some situations, doing more work over the same time period is not the goal, rather simplifying the programming solution is the goal. Sometimes, it makes more sense of thinking for the solution through concurrently executed tasks.

In the early days of computing, computers were single tasking that is, they ran a single job at a time. The big lumbering machine would start one job, run that job to completion and then start the next job, and so on. When engineers became overly frustrated with these batch oriented systems, they rewrote the programs that ran the machines and thus was born the modern multitasking operating system.

Multitasking refers to a computer's ability to perform multiple jobs concurrently. For the most part, modern desktop operating systems like Windows 95 or OS/2 have the ability to run two or more programs at the same time. While you are using Netscape to download a big file, you can be playing solitaire in a different window; both programs are running at the same time.

Multithreading is an extension of the multitasking paradigm. But, rather than multiple programs, multithreading involves multiple threads of control within a single program. Not only the operating systems run multiple programs, but also each program can run multiple threads of control within the program. For example, using a web browser, you can print one web page, download another, and fill out a form in a third all at the same time. Usually, a thread is a single sequence of execution within a program.

Basically, search engine can search matching details from its own database and the results displayed on web pages. It is inadequate also irrelevant for any users. Hence, high ended web search taxonomy proposes to serve full-fledged search results to the internet users. Search key is the topics, which we about to search on the web. The proposed high ended search engine taxonomy is described in bellow.

Methodology

There are different ways to classify parallel computers. One of the more widely used classifications, in use since 1966, is called Flynn's Taxonomy. Flynn's taxonomy distinguishes multiprocessor computer architectures according to how they can be classified along the two independent dimensions of instruction and data. Each of these dimensions can have only one of two possible states, are single and multiple. This dimension and states organize four possible classifications (Veerasamy, 2010a, 2010b) according to Flynn’s and is shown in Figure 1.

SISD Single Instruction, Single Data	SIMD Single Instruction, Multiple Data
MISD Multiple Instruction, Single Data	MIMD Multiple Instruction, Multiple Data

Figure 1: Flynn's Taxonomy

According to dimensions and state of Flynn's taxonomy, we take instruction as Search key and data as search engine. Each of these dimensions can have only one of two possible states, are single and multiple. This dimension and states organize four possible classifications and is shown in Figure 2. These classifications proposed as “High Ended Web Search Engine Taxonomy”.

SSSS Single Search key Single Search engine	SSMS Single Search key Multiple Search engine
MSSS Multiple Search key Single Search engine	MSMS Multiple Search key Multiple Search engine

Figure 2: High Ended Web Search Engine Taxonomy

SSSS- Single Search key Single Search engine

Everyday world, everybody is using this Single Search key Single Search engine (SSSS) taxonomy. In this taxonomy, we are using one Search key that is assigned to one search engine. SSSS will be displaying lot of details as we search, some time these pages may be relevant or may not be relevant to our target. Sometimes, it may waste our time. The following Figure 3 shows the taxonomy of Single Search key Single Search engine.

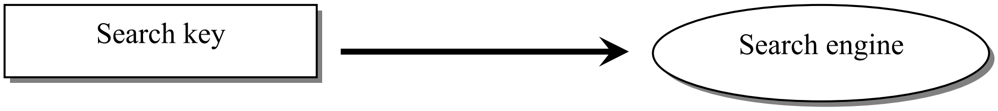
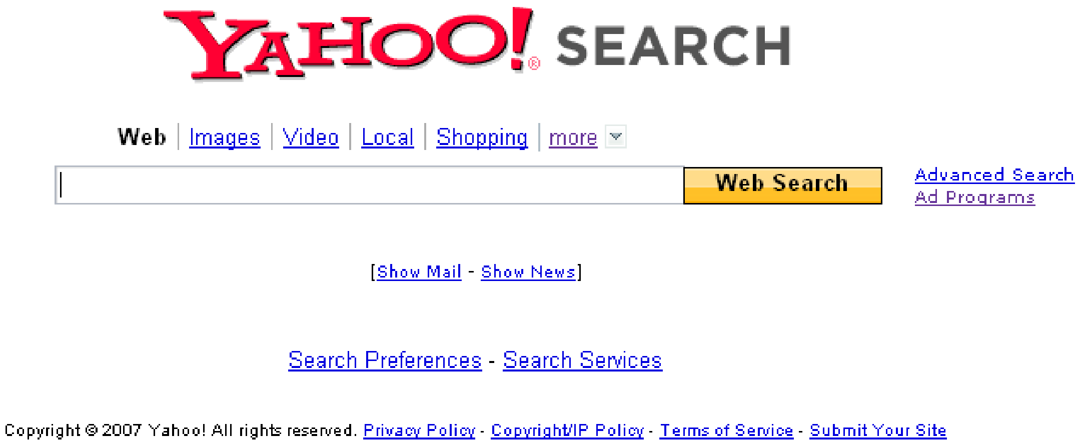


Figure 3: Single Search key Single Search engine Taxonomy

Single Search key Single Search engine illustrates usual search on the web. For instance, Yahoo search engine asks Single Search key from internet users and carry relevant topics from Yahoo's server. The following Figure 4 shows Yahoo search engine.


YAHOO! SEARCH
Web | Images | Video | Local | Shopping | more
Web Search
Advanced Search
Ad Programs
[Show Mail - Show News]
Search Preferences - Search Services

Copyright © 2007 Yahoo! All rights reserved. Privacy Policy - Copyright/IP Policy - Terms of Service - Submit Your Site

Figure 4: Yahoo Search Engine

SSMS- Single Search key Multiple Search engines

Single Search key Multiple Search engines (SSMS) taxonomy uses one Search key that is assigned to more than search engine. SSMS proposed to get details from many servers' database. This taxonomy is more powerful than SSSS. Here, increasing number of search engine will directly proportional to increasing the target details. The following Figure 5 shows the taxonomy of Single Search key Multiple Search engines.

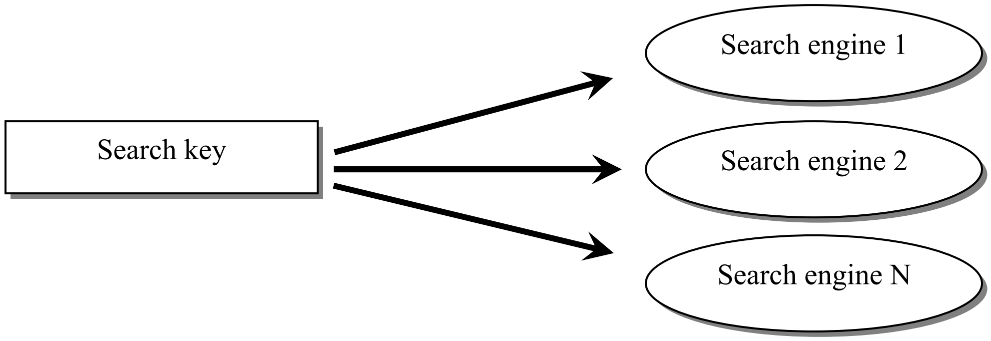


Figure 5: Single Search key Multiple Search engines Taxonomy

For example, Figure 6 illustrates conducting the same search key using different search engines simultaneously. Thus, we are passing same Search key to AltaVista (http://www.altavista.com), Yahoo (http://www.search.yahoo.com), and Google (http://www.google.com). As a result, we will be obtaining more search results on single web search.

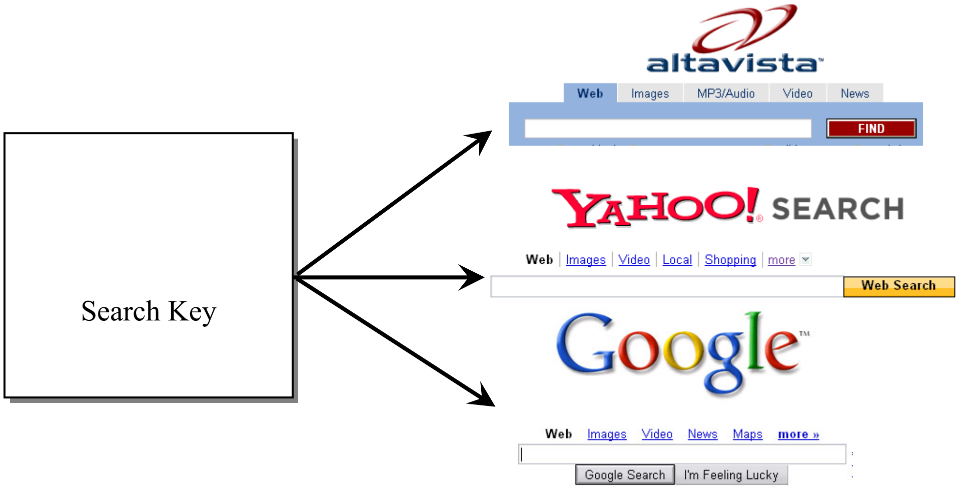


Figure 6: Example for SSMS

MSSS- Multiple Search keys Single Search engine

Mostly computer technologists are at presently using the taxonomy of Multiple Search keys Single Search engine (MSSS) for advanced web search. In this taxonomy, we will use more than one Search keys that is assigned to a search engine. MSSS will be displaying lot of details as we search, but these details are from one search engine and will display them. The following Figure 7 shows the taxonomy of Multiple Search keys Single Search engine.

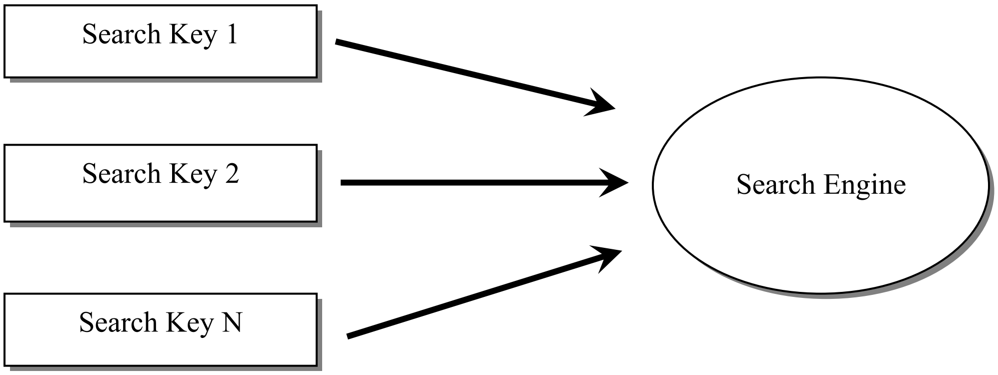
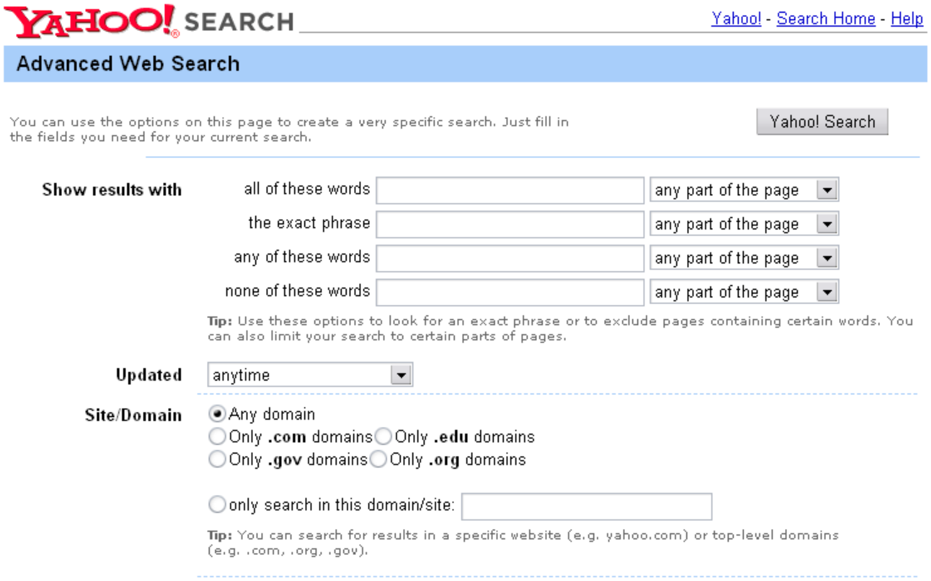


Figure 7: Multiple Search keys Single Search engine Taxonomy

This Multiple Search keys Single Search engine (MSSS) illustrates an advanced search on the web search engines. For instance, Yahoo search engine contain an advanced web search, which will ask Multiple Search Keys from users to search relevant topics and is shown in Figure 8.



Advanced Web Search

You can use the options on this page to create a very specific search. Just fill in the fields you need for your current search.

Show results with

all of these words		any part of the page
the exact phrase		any part of the page
any of these words		any part of the page
none of these words		any part of the page

Tip: Use these options to look for an exact phrase or to exclude pages containing certain words. You can also limit your search to certain parts of pages.

Updated anytime

Site/Domain
● Any domain
○ Only **.com** domains ○ Only **.edu** domains
○ Only **.gov** domains ○ Only **.org** domains

○ only search in this domain/site:

Tip: You can search for results in a specific website (e.g. yahoo.com) or top-level domains (e.g. .com, .org, .gov).

Figure 8: Advanced Yahoo Search Engine

MSMS- Multiple Search keys Multiple Search engines

Multiple Search keys Multiple Search engines (MSMS) taxonomy illustrates an advanced search on the web. Here, we use more than one Search keys that is assigned to more than one search engines. Really, the entire search details will generate from many databases and display them. Some details may come from one search engine and some another details come from another search engines. It is more powerful than Multiple Search key Single Search engine. Here, increasing number of Search key and increasing numbers of search engines will lead to increasing the target details. The following Figure 9 shows the taxonomy Multiple Search keys Multiple Search engines.

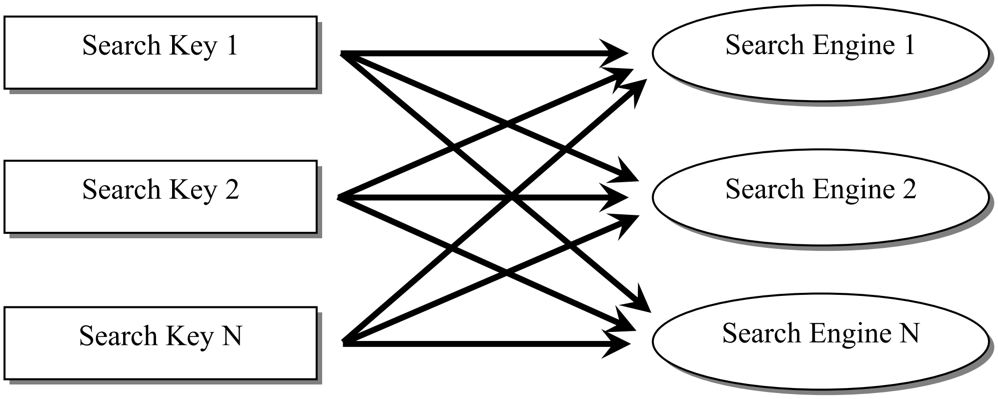
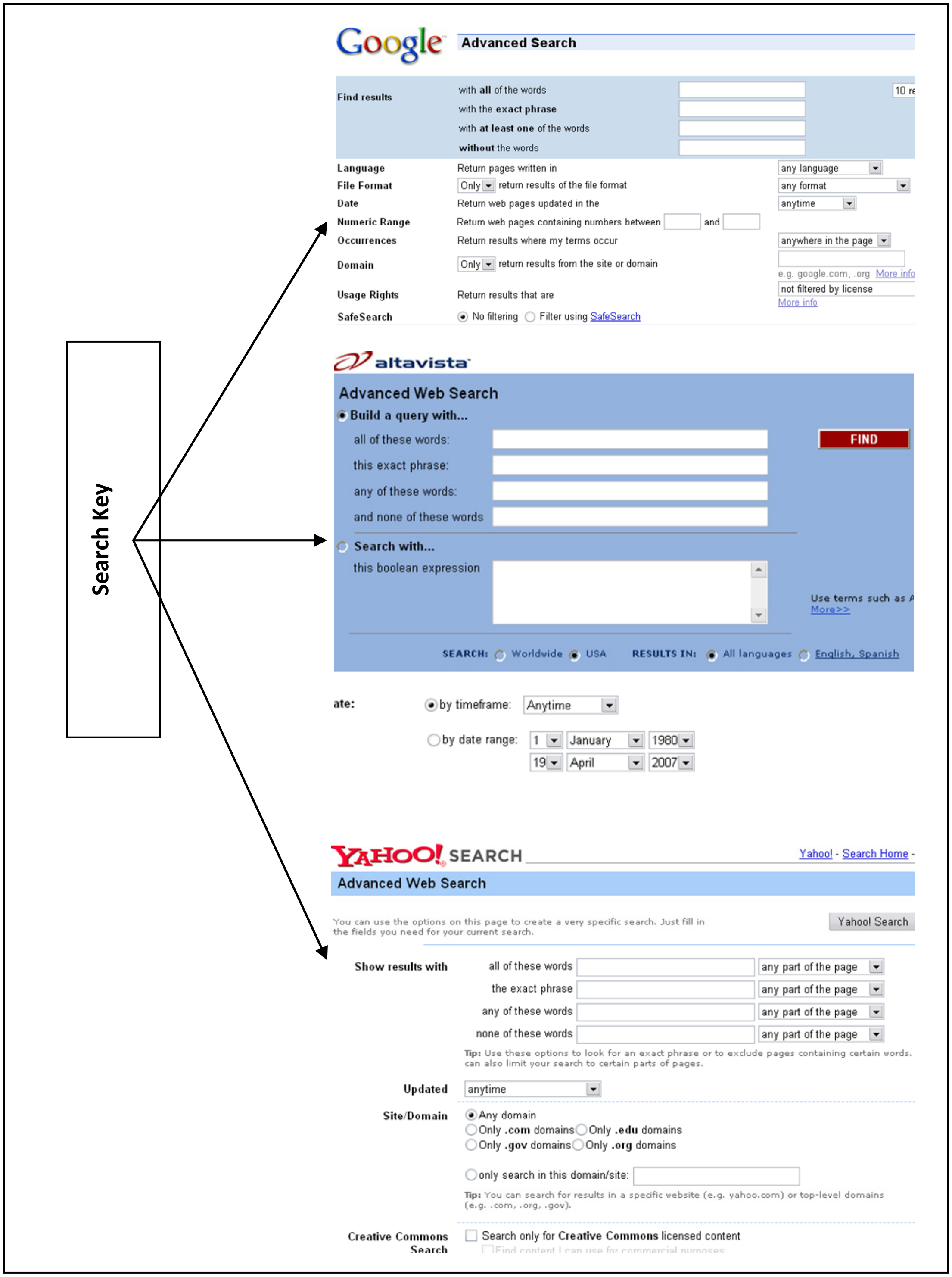


Figure 9: Multiple Search keys Multiple Search engines Taxonomy

For example, the following Figure 10 illustrates an advanced search on different search engine at a same time. Here, we pass multiple Search keys on AltaVista, Yahoo, and Google. As a result, we will search out exact details about a particular topic. Hence, we never face any failure in searching.



Google Advanced Search

Find results
with **all** of the words
with the **exact phrase**
with **at least one** of the words
without the words
10 re

Language Return pages written in: any language
File Format Only return results of the file format: any format
Date Return web pages updated in the: anytime
Numeric Range Return web pages containing numbers between and
Occurrences Return results where my terms occur: anywhere in the page
Domain Only return results from the site or domain
e.g. google.com, .org More info
Usage Rights Return results that are: not filtered by license
More info
SafeSearch ● No filtering ○ Filter using SafeSearch

altavista

Advanced Web Search
● Build a query with...
all of these words:
this exact phrase:
any of these words:
and none of these words
FIND
○ Search with...
this boolean expression
Use terms such as A
More>>
SEARCH: ○ Worldwide ● USA RESULTS IN: ● All languages ○ English, Spanish

ate: ● by timeframe: Anytime
○ by date range: 1 January 1980
19 April 2007

YAHOO! SEARCH Yahoo! - Search Home -

Advanced Web Search

You can use the options on this page to create a very specific search. Just fill in the fields you need for your current search.
Yahoo! Search

Show results with
all of these words: any part of the page
the exact phrase: any part of the page
any of these words: any part of the page
none of these words: any part of the page
Tip: Use these options to look for an exact phrase or to exclude pages containing certain words. can also limit your search to certain parts of pages.

Updated anytime

Site/Domain
● Any domain
○ Only **.com** domains ○ Only **.edu** domains
○ Only **.gov** domains ○ Only **.org** domains

○ only search in this domain/site:
Tip: You can search for results in a specific website (e.g. yahoo.com) or top-level domains (e.g. .com, .org, .gov).

Creative Commons Search □ Search only for **Creative Commons** licensed content
□ Find content I can use for commercial purposes

Figure 10: Example for MSMS

Implementations

Single Search key Single Search engine (SSSS) and Multiple Search keys Single Search engine (MSSS) are usually held on most of search engines. But, we recently have Single Search key Multiple Search engines (SSMS) and Multiple Search keys Multiple Search engines (MSMS). These two taxonomies are implemented in the following algorithms

1. SSMS algorithm
2. MSMS algorithm

SSMS algorithm is written for Single Search key Multiple Search engines taxonomy and MSMS algorithm is written for Multiple Search keys Multiple Search engines.

SSMS Algorithm

```
Step 1: Get Search key
Step 2: Declare number of search engine, in which you are going to search
Step 3: Create threads, based on the number of search engine. Thread, which includes retrieving process from each search engine
Step 4: Assign Search key to each thread
Step 5: Run threads concurrently, while running task
Step 6: While fetch data from relevant search engines do usual search on all
number of search engines and display on the page.
Step 7: Until complete the searching, continue step 5, step 6
Step 8: After finishing search, destroy the thread
```

MSMS Algorithm

```
Step 1: Get Search key 1, Search key 2… Search key N
Step 2: Declare search engine, in which you are going to search
Step 3: Create threads, based on the number of search engine. . Thread, which includes
retrieving process from each search engine
Step 4: Based on the number of Search key, assign each Search key to all number
of search engine
Step 5: Run thread concurrently, while running thread
Step 6: While fetch data from relevant search engines do advanced search on all
number of search engines and display on the page.
Step 7: Until complete the searching, continue step 5, step 6
Step 8: After finishing search, destroy the thread
```

Conclusion

Concurrency technique allows a computer program to do more work over the same time period. This kind of computer programs really involves more than one process. It makes more sense of executing tasks concurrently. Web search engines are more supportive to everyone to search topics. The actual idea is, to create search engine program by using concurrent programming with the help of "High Ended Web Search Engine Taxonomy". This taxonomy can be real word web search engines, which recently have Single Search key Multiple Search engines (SSMS) and Multiple Search keys Multiple Search engines (MSMS) with SSMS and MSMS algorithms. These algorithms are suggested to create any newly proposed web search engines. SSMS algorithm is somewhat moderated and MSMS algorithm can be an ultimate web search engine, because it has advanced search with multiple search engines. Hence, MSMS algorithm provides a high ended taxonomy for web search engine using concurrent programming.

References

Veerasamy, B. D. (2010a). Concurrent approach to Flynn's MPSD classification through Java. *International Journal of Computer Science and Network Security, 10*(1), 126-129.

Veerasamy. B. D. (2010b). Concurrent approach to Flynn's MPMD classification through Java. *International Journal of Computer Science and Network Security, 10*(2), 164-167.

Biography

Bala Dhandayuthapani Veerasamy was born in Tamil Nadu, India in the year 1979. The author was awarded his first masters degree M.S in Information Technology from Bharathidasan University in 2002 and his second masters degree M.Tech in Information Technology from Allahabad Agricultural Institute of Deemed University in 2005. He has published more than fifteen peer reviewed technical papers on various international journals and conferences. He has managed as technical chairperson of an international conference. He has an active participation as a program committee member as well as an editorial review board member in international conferences. He is also a member of an editorial review board in international journals. He has offered courses to Computer Science and Engineering, Information Systems and Technology, since 8 years in the academic field.

His academic career started in reputed engineering colleges in India. At present, he is working as a Lecturer in the Department of Computing, College of Engineering, Mekelle University, Ethiopia. His teaching interest focuses on Parallel and Distributed Computing, Object Oriented Programming, Web Technologies and Multimedia Systems. His research interest includes Parallel and Distributed Computing, Multimedia and Wireless Computing. He has prepared teaching material for various courses that he has handled. At present, his textbook on “An Introduction to Parallel and Distributed Computing through java” is under review and is expected to be published shortly. He has the life membership of ISTE (Indian Society of Technical Education).