International Journal of Scientific Engineering and Applied Science (IJSEAS) - Volume-1, Issue-3, June 2015
ISSN: 2395-3470
www.ijseas.com

# More Computing With the Existing Infrastructure: LAN Computing

**Anil Gupta**
**Department of Computer Science and Engineering**
**MBM Engineering College, J.N.V. University, Jodhpur**
**anilgupta@jnvu.edu.in**

Abstract:

*The speed and population of the computers are increasing regularly along with the data size. Computers are available at very low cost in huge quantity in all big organization or technical institutions, well connected by LAN, WAN with internet connectivity. These computers remain only ten percentage utilization in terms of CPU cycle. The rest ninety percentage idle cycles of these computers can collectively be used as a virtual computer with increased computational power to solve the huge and complex problems. In this paper it is proposed to make use of this extra computing facility either by sharing the computing needs of the society or making some hardware or software solution to use it only at the required level. Thus, the ninety percentage unutilized computing efficiency of the computer can be used for the non prime data processing involving heavy database or to solve the Engineering / Scientific problems without affecting it's normal working.*

## 1. Introduction:

LAN computing promise to change the way we tackle complex problems. They will enable large-scale aggregation and sharing of computational, data and other resources across institutional boundaries. And harnessing these new technologies effectively will transform the available of maximum computing in originate.

Fifty years of innovation have increased the raw speed of individual computers by a factor of around one million, they are still far too slow for many challenging scientific problems. For example, detectors at the Large Hadron Collider at CERN, the European Laboratory for Particle Physics, by 2005 will be producing several petabytes of data per year -- a million times the storage capacity of the average desktop computer. Performing the most rudimentary analysis of these data will probably require the sustained application of some 20 teraflops (floating-point operations) per second of computing power. Compare this with the 3 teraflops per second produced by the fastest contemporary supercomputer, and it is clear that more sophisticated analyses will need orders of magnitude more power.

The Personal Computers installed in the organizations/ Institutions and the computers at home segment remain idle or work with five to ten percentage of their total computing cycle resulting ninety percentage of the computing cycle remain idle consuming electricity and decaying the life of electronic components without an output. In the new scenario, the size of data is increasing and needs a fast speed computer to complete the process in time is also emerging at lower level, apart from this it is also required to solve the Engineering or Scientific Problems dealing with heavy Database. Now a days, most of the Management and Technology institute are equipped with hundreds of computers and these remain idle most of the time, all such computing facility are also well connected either by LAN or WAN including Internet.

The collective idle cycles of these computers are required number of years to execute these idle cycles on the single computer. The cycles are distributed among the computer as the idle cycle and if the data is distributed on the computer to use the idle cycle of the computer then the process completed on the computers might takes years to complete the same process on the single computer.

## 2. LAN Computing:

Rapid improvements in communications technologies are leading many to consider more decentralized approaches to the problem of computing power. There are over 400 million PCs around the world, many as powerful as an early 1990s supercomputer. And most are idle much of the time. Every large institution has hundreds or thousands of such systems. LAN computing seeks to exploit otherwise idle workstations and PCs to create powerful distributed computing systems with institutional reach and supercomputer capabilities.

Alternatively the power, component life can be optimally utilized by providing operating system based software solution or hardware based variant integrated knob.

It is therefore, suggested with following solution

1. An Application is to be developed for network OS to make use of idle cycle of Computer.
2. Design a network operating system with embedded solution to use the idle cycle of Computer.
3. An integrated hardware enhancement to control the CPU utilization as per varied Computing needs.

In this paper, our approach is to solve the problem on first solution as follows:

Since, the application needs to work in a collaboration manner. Only network OS can be used and few of the operating system can be used for this purpose are shown in table 1.

Table 1: **Network OS**

| | |
|---|---|
| 1 | AIX 5.2 |
| 2 | Dedian 4 |
| 3 | FreeBSD 6 |
| 4 | HP/UX 11.00 |
| 5 | MacOS/X 10.4 |
| 6 | Red Hat Linux |
| 7 | SLES 9 |
| 8 | Solaris 2.9 |
| 9 | Win NT 5.0 |
| 10 | YDL 5 |

There shall be two application program develop in JAVA. The Java is a platform independent programming language and having verity of library class available to develop network based program. The program will be:

1. Server Application
2. Client Application

The working flowcharts of the program are shown in the figure 1-6.

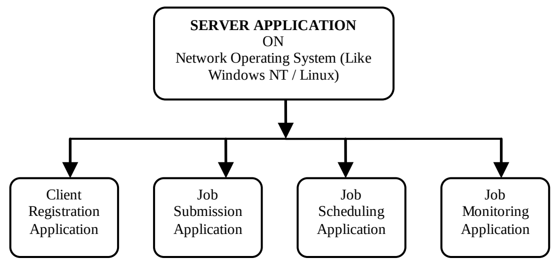



Fig. 1

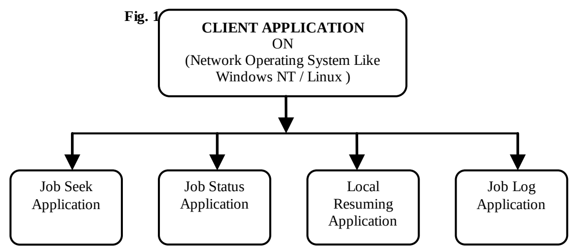



Fig. 2 Client Application: Block Diagram

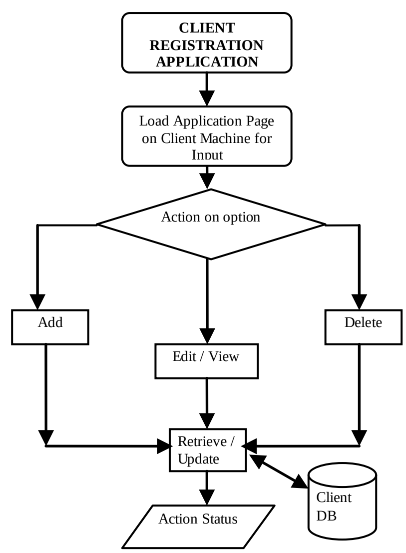



Fig. 3 Client Registration Process

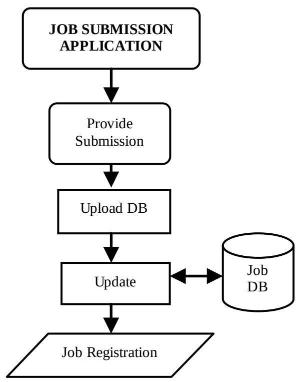



Fig. 4 Job Registration

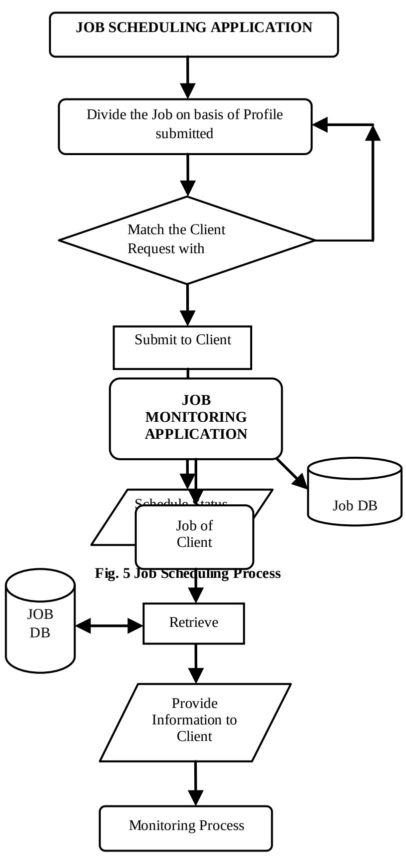
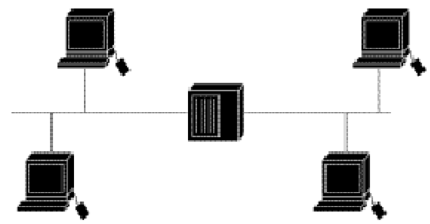



Fig. 5 Job Scheduling Process

Fig. 6 Job



Typically operation for the program shall be as given bellow:

1. Either the server will search for idle work station or alternatively workstation shall submit its probable idle time with the help of network agent floated on network by save the application application.
2. In parallel, server shall keep on collecting job request from either of the client. Along with job request it shall also collect data and preferable computing environment needs depending upon weather the application is data prone or calculation prone.
3. The server shall manage these job requests along with its other attributes in database and shall maintain it's queue.
4. The server shall also enroll, maintain a database and deliver the client application for its installation to the clients willing to make use of their idle computing cycle in a collaborative approach.
5. Depending upon the available idle client requests and the queued job base, server shall accordingly part the jobs and will deliver the client after making a consideration to job specific computing environment needs.

6. Server shall regularly update itself with the progress of all distributed jobs and will also reschedule the job in case of partly finished ones.
7. Finally on the completion request from client, the server shall collect and compile back the parted jobs till the whole job is over.

For the other two solution suggested in this paper, same needs to be handled by operating system companies or the hardware companies. It is also learned that Intel I7 processor lunched recently has already being designed to save its power cycle idle time, means that hardware companies have already started working in this direction. However, an OS level solution to save the idle time has not been attended properly so far.

The solution discussed in this paper may also be enhanced to make itself as OS service on both server and clients resulting in an operating system level embedding of this application, where a group of computers can any time look for available computing facilities in the LAN and can make use of it.

**Other Similar Technology:**

**Internet Computing:**

The opportunity represented by idle computers has been recognized for some time. In 1985, Miron Livny showed that most workstations are often idle, and proposed a system to harness those idle cycles for useful work. Exploiting the multitasking possible on the popular Unix operating system and the connectivity provided by the Internet, Livny's Condor system is now used extensively within academia to harness idle processors in workgroups or departments. It is used regularly for routine data analysis as well as for solving open problems in mathematics. At the University of Wisconsin, for example, Condor regularly delivers 400 CPU days per day of essentially "free" computing to academics at the university and elsewhere: more than many supercomputer centres.

**Examples of Internet Computing:**

1. Condor, one of the project in foreign University is effective on a small scale, true mass production of Internet computing cycles had to wait a little longer for the arrival of more powerful PCs, the spread of the Internet, and problems (and marketing campaigns) compelling enough to enlist the help of the masses. In 1997, Scott Kurowski established the Entropia network to apply idle computers worldwide to problems of scientific interest. In just two years, this network grew to encompass 30,000 computers with an aggregate speed of over one teraflop per second. Among its several scientific achievements is the identification of the largest known prime number.
2. SETI@home project. This enlisted personal computers to work on the problem of analysing data from the Arecibo radio telescope for signatures that might indicate extraterrestrial intelligence. Demonstrating the curious mix of popular appeal and good technology required for effective Internet computing, SETI@home is now running on half-a-million PCs and delivering 1,000 CPU years per day -- the fastest (admittedly special-purpose) computer in the world.

The Internet computing now has access to a tremendous new computing resource. All they have to do is cast their problem in a form suitable for execution on home computers and then persuade the public (or an Internet computing company) that their problem is important enough to justify the expenditure of "free" cycles.

**3. Grid Computing:**

The Grid But the real significance is broader. Internet computing is just a special case of something much more powerful -- the ability for communities to share resources as they tackle common goals. Science today is increasingly collaborative and multidisciplinary, and it is not unusual for teams to span institutions, states, countries and continents.

Grid computing can be thought of as distributed and large-scale cluster computing and as a form of network-distributed parallel processing. It can be confined to the network of computer workstations within a corporation or it can be a public collaboration (in which case it is also sometimes known as a form of peer-to-peer computing).

Grid computing concepts were first explored in the 1995 I-WAY experiment, in which high-speed networks were used to connect, for a short time, high-end resources at 17 sites across North America.

## 6. The Future

Although Internet and Grid computing are new technologies, they have already proven themselves useful and their future looks promising. As technologies, networks and business models mature, we expect that it will become commonplace for small and large communities of scientists to create "Science Grids" linking their various resources to support human communication, data access and computation. we also expect to see a variety of contracting arrangements between scientists and Internet computing companies providing low-cost, high-capacity cycles. The result will be integrated Grids in which problems of different types can be routed to the most appropriate resource: dedicated supercomputers for specialized problems that require tightly coupled processors and idle workstations for more latency tolerant, data analysis problems.

In India the internet has not attained the age of very fast communication channels so far. We, therefore, propose to move with in LAN computing with in institutional boundaries, so that same can latter be implemented in internet / Grid computing world wide to make use of whole world ‘s shared computing power a major recourses community as whole.

## 6. References

1. Mutka, M. & Livny, M. The Available Capacity of a Privately Owned Workstation Environment. Performance Evaluation 12, 269-284 (1991).
2. Foster, I. & Kesselman, C. (Eds). The Grid: Blueprint for a New Computing Infrastructure. Morgan-Kaufmann (1999).
3. Johnston, W., Gannon, D. & Nitzberg, W. Grids as Production Computing Environments: The Engineering Aspects of NASA's Information Power Grid, Proc. 8th Symposium on High Performance Distributed Computing, IEEE Computer Society Press (1999).
4. Foster, I. & Kesselman, C. Globus: A Toolkit-Based Grid Architecture. In ref. 2, pages 259-278. Morgan Kaufmann Publishers (1999).
5. Butler, R., Engert, D., Foster, I., Kesselman, C., Tuecke, S., Volmer, J., & Welch, V. Design and Deployment of a National-Scale Authentication Infrastructure. IEEE Computer (in press).
6. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. Foster, I., Kesselman, C., Tuecke, S. www.globus.org/research/papers/anatomy.pdf