Virtual Internet Personal Computer (VIPC)

Tilan Ukwatta (tilan.ukwatta@ifs.lk), Dr. Damitha Karunaratna (damithak.UOC@mail.cmb.ac.lk)

School of Computing
University of Colombo

ABSTRACT

The ability to access and process large volumes of data distributed over multiple sites on the Internet by using a variety of user communication devices such as mobile phones, PDAs or Laptops, is a contributing factor to the success of e-business. Due to the physical limitations of computer hardware it is logical to conclude that the resources and the processing power that a single CPU system can provide will reach a maximum in the near future. In this context new software architectures based on distributed computer systems must be explored to meet the future information demands. In this paper the concept of a "Virtual Internet Personal Computer (VIPC)" is proposed to exploit the existing computer systems and various user data communication devices to meet this future demand. The architecture presented in the paper is evaluated by developing a prototype system. This prototype system is implemented in Java as an application running on a dedicated Java virtual machine. Through the implemented prototype it has been able to demonstrate that the proposed architecture is possible to implement and achieves the intended objectives.

1. INTRODUCTION

Personal Computers are rapidly becoming the most widely used computing and communication medium of the world. However, due to the physical limitations, such as speed of light, it is logical to conclude that the amount of resources a single CPU system can provide will reach a maximum in the near future. Also the need for accessing distributed computing resources on the Internet from personal computers is increasing. In this context new architectures based on multiprocessor or multi-computer computing systems must be explored to meet PC users' future demands. In addition, there is a rising demand of personal computer users to access their personal computers at any time from anywhere in the world. However computers, even laptops, are bulky to carry physically and whenever these expensive devices are carried around there is a risk of breakage and data destruction. Thus to meet future demands, a mechanism should be provided to access information stored at different computers dispersed over Internet, timely and easily, by using a variety of inexpensive mobile devices such as mobile phones or Personal Digital Assistant (PDA).

VIPC is a virtually centralized computing system. To user its looks like single computer, but physically it may be distributed over hundreds of physical computers similar to a distributed system. The VIPC achieves advantages of distributed systems while keeping the concept of personal computers. Distributed systems have many advantages over centralized systems consisting of a single CPU [1]. For example distributed systems offer a better price/performance ratio than centralized systems.

VIPC concept emphasis on a virtual personal computer for each user than, a personal account on a distributed operating system. This kind of separation of the virtual computer from the underling-distributed operating system enables us to implement the VIPCs as set of mobile virtual computers. That is VIPC is a single object that can be moved from one computing system to another. Also VIPC concept let us eliminate more than one computer per user, which causes lot of data synchronization and data accessibility problems.

Since VIPC resides on the Internet it enables users to access their virtual computer from anywhere in the world. Also users can access their virtual computer using many forms of computing devices such as PCs, Laptops, Personal Digital Assistants (PDA), mobile phones and Thin Client Devices.

A Thin Client Device [2] is a desktop device connected over a network or serial connection to a central computing system or server that is operating in multi-user mode. While the desktop device may look like a PC, it is actually much simpler, if equally sophisticated device. All

application processing and storage is done on the server. The thin client is simply a device for input, output and display. Thin client is used only to transmit keystrokes and mouse clicks to the server, and to display on a desktop monitor what the server is doing. With thin-client/server computing, the desktop device is only a window in to the activity on the server, allowing it to be a simpler device. It does not require a hard drive, a floppy drive, or the latest CPU, and the thin-client chip only has to drive the display, and transfer input/output bit streams. Thin client device uses less power than a PC, and because it has no moving parts, less RAM, and a CPU that generate less heat, it does not require a fan. It has, therefore, no moving parts that can break. Thin client device is an ideal device to access and work with a VIPC.

VIPC can also work as an agent of the user to enable users to run their computers on the Internet on behalf of them and do various tasks when users are not connected. Another important feature of VIPC concept is; to release users from the burden of constant need to upgrade and maintain their hardware. According to the VIPC concept upgrading and maintaining tasks are handled by companies who are hosting virtual computers (i.e. VIPCs).

2. BACKGROUND

There are many computing models for computer networks. In the following section a number of popular network computing models are summarized and describes how the VIPC concept, in other words "VIPC Computing Model" differs from these existing models.

There is "Download and Run Computing Model" of Java applets and ActiveX components, where the client dynamically downloads components from the network into the client device for execution. This model is not network bandwidth friendly as the bandwidth required for an application increases with the size of the application. The model also requires the client-computing device to be equipped with a considerable amount of computing resources to access the network and to execute the downloaded applications. Furthermore the model limits the variety of client-computing devices that can be used to access applications and documents. But according to VIPC Computing Model there is no application execution takes place at client devices.

Another popular model is "X Windows Computing Model", which allows a user in one machine to use the other available machines in a network to run his applications remotely. The desktop at the user' machine shows interface screens of several different applications running on different machines on the network [3]. Unlike in X Windows Network Computing Model, VIPC Computing Model provides a virtually centralized personal computer, where distributed processing and storage is transparent to users. Users never need to worry about the distributed nature of the system.

Another model that is more recently immerged is "Remote Control Computing Model", where remote access software layer is used to extend the input/output capabilities of a particular machine in a computer network. This remote access software layer enables computers at different geographic locations to communicate with each other through phone lines, Internet connections, or through network and to run applications on a remote computer. In essence, a remote access software layer allows a user to view a computing 'desktop' environment of a particular computer on a different computer with different machine architecture. A large number of remote access solutions are available on the market today, for example Carbon Copy, CoSession Remote, LapLink Professional, VNC [4], GoToMyPC [5], pcAnywhere [6].

Remote Control Computing Model concentrates only in allowing users to use and administer remote computers through the network. Here in most cases exclusively only one person can use a given computer at a given time. But in the case of VIPC Computing Model one powerful computing system can host lot of virtual computers and serve many people at once. This powerful computing system can be a cluster of computers or a supercomputer. The VIPC Computing Model can use techniques used in Remote Control Computing Model to display desktop of each VIPC on each client device.

"Server based Computing Model" is another network computing model, which can be considered as a modern version of the old mainframe based centralized computing concept. In this model applications are deployed, managed, supported and executed completely on a server [7]. The client devices have instant access to the snap-shots of the applications via the server. Three main features characterize the model. The first is a multi-user operating system that enables multiple concurrent users to log on and run applications in separate, protected sessions on a single server. The second is a mechanism that separates the application's logic from its user interface so that only keystrokes, mouse clicks and screen updates travel through the network. As a result application performance is bandwidth independent. The third feature is the centralized application and client management.

On the first glance VIPC Computing Model looks very similar to Server based Computing Model combined with Remote Control Computing Model. But VIPC Computing Model emphasis on a virtual computer, a separate entity that can be moved if required. Also since VIPC runs on a network of computers, it can provide a much larger amount of resources and processing power than a single server based system.

"Distributed System Computing Model" unifies computers in an entire computer network logically to provide a user with a powerful virtual computer. As computer networks are getting larger, faster, and more powerful, opportunities for distributed systems are increasing. Gigabit networks [8], connecting powerful high-performance machines and workstations, have created enormously powerful infrastructures that can be utilized to solve complex problems and to store huge amounts of information. Linked together, these connected resources make up a single, worldwide, distributed system.

One of the major problems in implementing distributed systems successfully is the lack of software. Easy-to-use software that can manage a complex physical system and the support for large degree of parallelism is needed so that a distributed system becomes a reliable, efficient, and real opportunity for a wide variety of users. A number of research projects are currently underway to build reliable software for distributed systems. For example the project "Legion" at the University of Virginia [9]. Its goal is to design a system capable of handing millions of hosts and trillions of objects tied together with high-speed links.

The VIPC Computing Model is more or less different from other computing models for computer networks. It is the concept of a virtual computer, which sits on the Internet; to be accessed by its owner from anywhere in the world using any computing device, running any client operating system, makes it unique from others. Users can run multiple programs in this virtual computer and do whatever tasks they used to do in normal personal computer.

3. ARCHITECTURE

VIPC system's architecture comprises of four layers, as shown in the figure 1. The bottom layer is the hardware layer. It consists of multiple computer systems connected via a network. On the top of hardware layer lays the VIPC management system and distributed operating system layer. This layer facilitates the construction and destruction of VIPCs, on the third layer, on user demand. It also handles load balancing and various other management tasks, provided by a typical distributed operating system. The third layer consists of multiple VIPCs created for different users. Each VIPC on this layer emulates a fully functional personal computer with an operating system that controls it. Forth and the final layer consist of various user client devices that enable access to each VIPC. Typically client devices in this layer are inexpensive and easy to carry devices such as mobile phones, PDAs or Thin Clients.

Virtual Internet Personal Computer has two functioning modes. These two modes are graphics mode and text mode. In the text mode user enters commands at the client side and those commands are executed in the appropriate VIPC and results are send to the client. In this mode only text-based applications can be executed.

In the graphics mode communication between the two sides are done by, Virtual Internet Personal Computer sending only the screen shots to the client and client sending keyboard characters and mouse events back to the VIPC. Here the role of the client is to paint the screen shots sent by the VIPC on the client's screen. It also captures mouse events and keystrokes and sends them to the VIPC. This method of communication is bandwidth friendly.

Exact method of communication of client with the VIPC management system to get a VIPC depends on the implementation. But roughly it is possible to say that the user acquires a VIPC (in the layer 03 in figure 1) by requesting it from the VIPC management system (in the layer 02), through a client device (in the layer 04). The VIPC management system responds to such requests by activating a particular VIPC that belongs to the user and then sending details of this activated VIPC to the client device. Once a VIPC is activated, the user may use this connection details to get logon to his VIPC. After login user can work in text mode or in graphics mode depending on his requirements. After user finishes whatever work he was doing he can shutdown the VIPC or leave it as it is and reconnect when required. Exact details of the procedure heavily depend on the implementation. But for the user it should be just typing the user name and the password.

4. PROTOTYPE

The concept presented in this paper is tested using a prototype VIPC system. In this prototype system, each VIPC is implemented as an application running on a dedicated Java Virtual Machine (JVM). JVM provides the separate virtual computer that is required by each VIPC.

The application, which runs on JVM, behaves like a simple operating system, providing facilities for file management and program execution. In this prototype implementation, users can access their VIPC using Java client application, using a Web browser and using a WAP enabled mobile phone. The Current implementation has demonstrated that the proposed architecture is useful and can be fully implemented.

Figure 2 shows the basic design of the prototype. This system is based on client server architecture. According to this design there are two ways a client can initiate a connection with a VIPC. One way is client directly communicates with the VIPC manager and asks for a particular VIPC. Then the VIPC manager creates a VIPC for the client and sends details about this newly created VIPC to the client. Then using those details client makes a connection with the VIPC. Also note that in the case where client is previously connected but did not shutdown the VIPC, VIPC manager does not create a new VIPC but sends the details of the existing VIPC that belongs to this particular client and then the client uses this information to establish a connection. When the connection between a client and a VIPC is first established, the VIPC begins by requesting authentication from the client using a challenge-response scheme, which results in the user being prompted for a password at the client end.

In the second way client do not directly communicate with VIPC manager. Here the web browser or WAP browser in the client device makes a connection with particular web server. A servlet running on this web server dynamically create a login page and present it to the client. Then client sends login details to the servlet using this page. Now the servlet initiate a connection with the VIPC manager and ask for a VIPC. Then VIPC manager sends details of the activated VIPC to the servlet and servlet make connection with the VIPC. Here the role of the servlet is to get requests from the client send it to the VIPC and then get the results from the VIPC, create a html page or wml page containing those results dynamically and send it to the client. For more information about the implementation of the prototype refer to [10].

5. CONCLUSION

VIPC concept opens up an entire new dimension for Personal Computers. This new breed of PCs consists of a thin client device, connection to the Internet and a VIPC, hosted by a VIPC hosting company. These computers may cost only fraction of the price of a physical personal computer today. The processing power, storage capacity or any other resource can be increased simply by informing the new requirements to the VIPC hosting company and paying the required rental, which will be much less than the cost of procuring and maintaining a conventional PC.

The implementation of the VIPC Computing Model proposed by this paper has many consequences. Firstly, a new breed of companies will immerge for hosting VIPCs. Secondly, modified versions of thin client devices, which is capable of connecting to a VIPC will come to the market. Thirdly, computer users prefer to buy less expensive VIPC from a hosting company and a portable light weight inexpensive thin client device to access their VIPC from anywhere in the world, rather than buying a traditional personal computer.

REFERENCES

[1] Andrew S. Tanenbaum, Modern Operating Systems, Prentice Hall, Inc., NJ, 1992, pp 362 – 366.

[2] CITRIX White Paper. “Thin-Client Computing for the Government Sector,” 1999. http://www.citrix.com

[3] Tim Love, “Overview of X Windows,” 1996. http://www-h.eng.cam.ac.uk/help/tpl/graphics/X/X11R5/node4.html

[4] AT&T Laboratories Cambridge, Virtual Network Computing, 2001. http://www.uk.research.att.com/vnc

[5] Expertcity, Inc. White Paper. “Remote-Access Technologies: A Comparison of GoToMyPC™ and pcAnywhere™,” 2001. http://www.gotomypc.com

[6] Symantec White Paper. “Secure Remote Control for IT Support Organizations,” 2001. http://www.symantec.com

[7] CITRIX White Paper. “Server Based Computing,” 1999. http://www.citrix.com

[8] Andrew S. Tanenbaum, Computer Networks, Prentice Hall, Inc., NJ, Third Edition, 1996, pp 54 – 56, 568 – 572.

[9] Legion, A Worldwide Virtual Computer, 2002. http://legion.virginia.edu

[10] Tilan Ukwatta, “Virtual Internet Computer,” Final Year Project Dissertation, University of Colombo, December 2001.

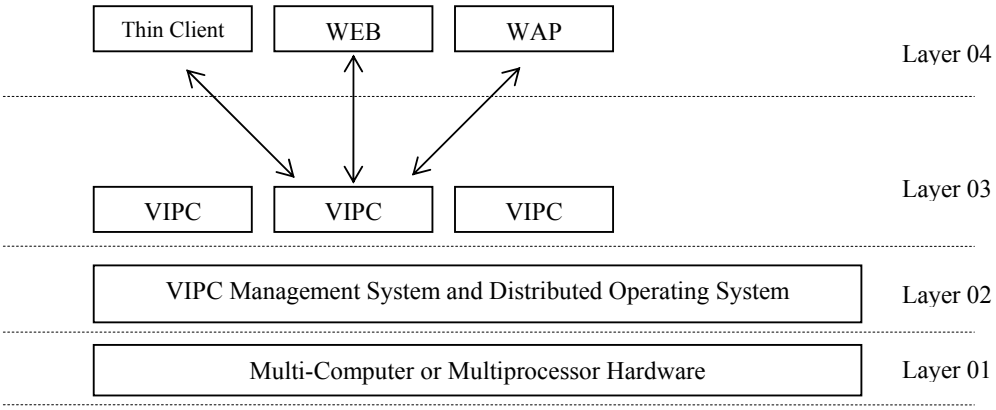

Figure 1: Architecture of the VIPC Computing Model

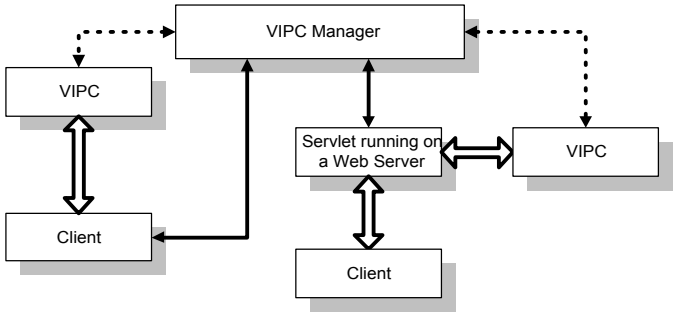

Figure 2: Architecture of the Prototype VIPC System