Computing Brokerage and Its Application in VLSI Design*

Youn-Long Lin
Department of Computer Science
Tsing Hua University
Hsin-Chu, Taiwan 30043, Republic of China
e-mail: ylin@cs.nthu.edu.tw

Abstract

With Internet access available to virtually every one in this community, it is interesting to investigate on how Internet will affect the future of VLSI design and CAD. We will describe an experimental WWW-based computing broker. Theoretically, the broker is capable of providing every user with access to any hardware platforms and any software over the Internet. It makes possible pay-per-use of both hardware and software resources. It also automatically manages multiple resources ranging from a few seats within an organization to thousands of seats anywhere with the Internet access. This new model of resource usage will have significant impact on the users, the software developers, and the computer vendors. Users no longer have to own nor maintain expensive computers and software tools before they can start their projects. They will have more flexibility in allocating resources to meet the project schedule. Also they will be able to access to the latest technology at lower overall cost. Tool developers and computer vendors will have broader customer base with very little marketing and field support effort. This new model will also provide a better chance for new tools and new platforms.

Key Words: Computing Brokerage; WWW; Internet; CAD; VLSI Design; Pay-per-Use;

1 Introduction

Internet is transforming the ways business operates and the ways people works. Many people are taking advantage of Internet to improve their personal productivity and business efficiency. Many more people are exploiting new business opportunity made possible by the Internet[1]. VLSI design is definitely no exception. Every party involving with this activity must be aware of the promise and peril of Internet.

A VLSI design project requires using computer hardware, CAD/CAE software, and design library. Presently, in order to design VLSIs, most companies or institutes purchase computer systems, software tools, and design library, hire supporting staffs, and pay for the operation and maintenance service. This business model was established well before the Internet is available. With Internet access available to virtually every VLSI designer and every CAD/CAE researcher/developer, it is interested to study how this community can be maximally benefitted.

In this talk, we will first identify bottlenecks existing in the present day business model. Then, we will speculate on whether Internet can help with alleviating these bottlenecks. Finally, we will describe a prototype WWW-based brokers for VLSI design, and the problems that may raise.

2 Models of Computing

A VLSI design project requires performing many computational tasks including various level of capturing, synthesis, simulation, and verification. Some tasks may take hours, even days, of CPU time to complete on a state of the art high-performance computer, while some others may need only very little CPU time. A computational task consists of three elements: computer system, application software, and data. Depending on where these elements locate, we can categorize computations as following:

1. Local Computing: All three elements are on the local computer. This is the traditional way of computing practiced before the age of Internet and, still in many place, today.

*Partially supported by the National Science Council of the Republic of China under contracts No. NSC86-2221-E-007-019 and No. NSC86-2221-E-007-049.

2. Remote Login: All three elements are on a same remote computer. The user reach the computer via a remote login procedure such as "rlogin" or "telnet". This model is essentially the same as the previous one despite the user has access to multiple machines.

3. Browsing: While both the computer and the software are local, the data is on a remote computer. A typical scenario is the browsing of a home-page over the World-Wide Web.

4. Software Pay-per-Use: While the data is on the local computer, the software is on a remote site. Invoking a JAVA applet is a typical example. We have seen development of a JAVA-based finite state machine design tool[2].

5. Computing Pay-per-Use: Unlike the software pay-per-use model, both the hardware and software are pay-per-use. Only the data is created and stored locally. Both the computing power and the software are on a remote site. The data is sent over to the remote site before the computation and sent back to the local after the computation.

Each model has its own advantages and disadvantages in terms of cost, performance, flexibility, security, ...etc. Different models may be suitable for different types of computational tasks.

3 CAD Computing Model

Presently, users purchase hardware from software vendors and software from software vendors. Their relationship can be depicted as Figure1.

Three groups of people are involved with a CAD tool. Researchers proposed new algorithms, and build prototypes to demonstrate the feasibility; Vendors developed CAD tools incorporating new algorithms; Designers use vendor-supplied tools. Each group of people use Internet for one purpose or another. Researchers use it to for exchanging research results in the form of technical papers or public-domain software. Vendors use it for software distribution or on-line documentation. Designers use it for data transfer.

To the users, the present day model has several disadvantages. First, they have to invest in capital. They have to purchase computers and software tools. Second, because the demand on computing resources varies over various phases of a design project, the computers and software are either under utilized or overloaded. Third, it is difficult, both financially and

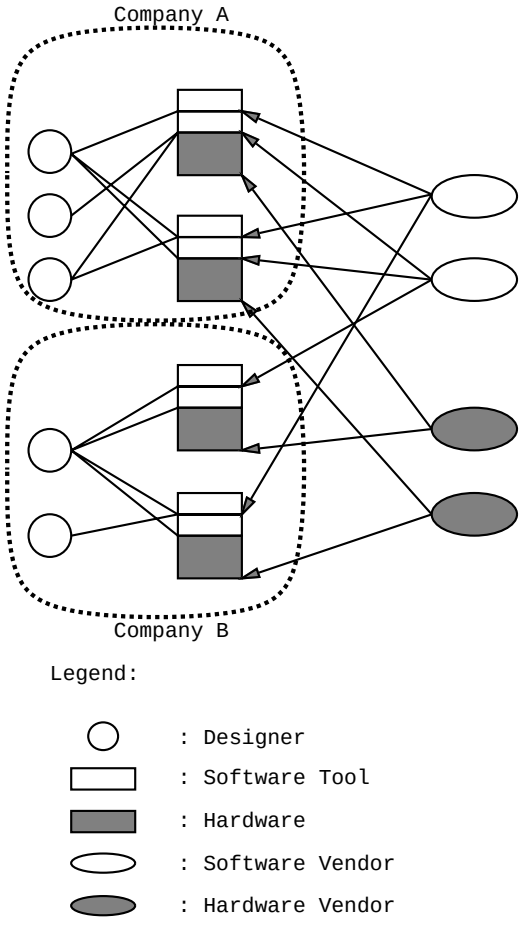

Figure 1: Present business model

technically, to switch to better hardware or software technologies when they become available shortly after the acquisition of the computing resources.

From a user's (the user of the computer and the software tools) point of view, he is not interested in owning any fast-depreciating properties such as computers and software tools. Owning them is costly because they consume capital, space, utilities, and man-power even when they are sitting idle. Instead, what he is interested in is paying a reasonable amount of money to get his job done on time. It does not matter where the computer and the software locate. We call this model pay-per-use computing. To make this model a reality, we need a computing broker.

In the early 70s, before the arrival of mini-computers, there was a pay-per-use business model as depicted in Figure2. A mainframe computer costed millions of dollars. A typical user who could not afford a mainframe had a terminal, a card reader, and a printer in his site while the vendor has mainframes in a remote site. The two sites were connected via a leased data line. To compute, the user issued commands via the terminal, sent data via the card reader, and got results from the printer. At the end of each month, the vendor billed the user according to the amount of CPU and disk usage.

The old pay-per-use business model disappeared with the arrival of mini-computers. Users purchased computers not only because of the affordability but also the controllability. They felt more comfortable working close to the computers. With PCs and workstation servers replacing mini-computers, little change occurs to the usage model except the shared file system.

In many areas such as VLSI design, heavy computation is performed using expensive software tools. Today's business model requires that users purchase both the software and hardware, pay for their operation and maintenance, and purchase more, bigger machines shall the computing power become a bottleneck of the design process. Usually, a company cannot afford to buy all software needed. This is especially true for a small design house. Even if a company does have a particular software tool. It will find out that both the tool and computer are either under-utilized or overloaded because a design project calls for different tools during different phases of its cycle. For example, a design rule checker (DRC) is needed only during the last phase of the project while a behavioral simulator is only used early in the design cycle. Worse yet, with all these capital investment, it is very difficult for the users to switch to some later, better hardware or software.

Under today's business model, it is difficult for a small tool developer to succeed because of lack of marketing power even if it poses superior technology over the existing big vendors.

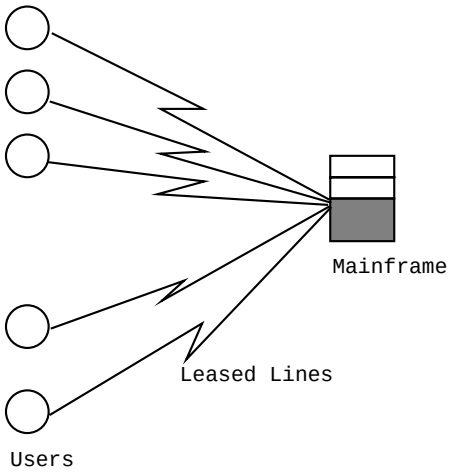

Figure 2: Pay-per-use of mainframe computers in the 70s.

4 Computing Brokerage

Over the Internet, many intelligent agents have been created for gathering useful information, finding good bargaining, ...etc. We will see more agents with more powerful capability in the future.

A computing broker is a software that manages a pool of hardware and software resources as depicted in Figure3. A resource owner can open up his hardware or software to the outside by adding to or deleting from the broker's database the information regarding his resource. A user requests a computing service from the broker by specifying tool name and pointers to the data files. He can also optionally specify the machine type, OS version, RAM/disk configuration, ...etc. The broker will automatically find a computer capable of serving the request. It will also handle the accounting and billing. Hence, it makes pay-per-use possible.

With the infrastructure in Internet and World-Wide Web (WWW), we implement a *software computing broker* called **CompBro**[3]. Based on WWW, CompBro takes computing requests from the users on the one hand, and manages a large number of hardware and software resources on the other. Any computers and associated software tools connected to the Internet can be managed by CompBro and made available to any users with Internet access.

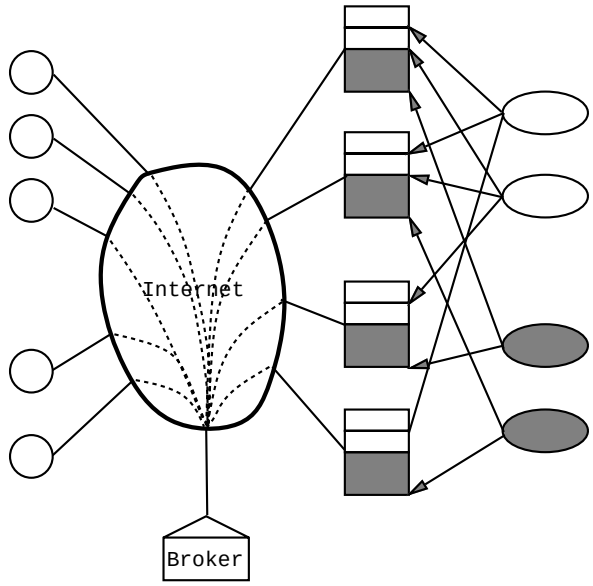

Figure 3: Computing model with a computing broker

An immediate application of the computing broker is as the resource manager within an organization. In a company or a school, many users run many CAD tools on many workstations. Some software licenses are locked while some are floating. To best utilize the resources while maximally satisfying every user's computing requirement, a centralized management policy is needed. The computing broker is very suitable for this task.

To run one or more long batch jobs (e.g., logic or circuit simulation for several hours), the user does not have to worry about which machines to use nor does he have to worry about possible conflict with other users. Instead, he just submit his jobs to the broker. The broker is the only interface to the user. It maintains a database of the availability of every hardware and software resource as well as every request from every user. It schedules the jobs and allocates the resources in a globally optimal way following a policy set by the administrator.

Apparently, this model of computing can be extended over organizational boundary. For example, many schools can share computing resources among one another if they let a single broker manage all the resources. It can even go beyond country boundary such that the users in one country can utilize the idle computing resources in some other countries due to difference in working hours or holidays.

Establishment of such a model will profoundly affect every user, software tool vendor, and hardware vendor.

The users do not need to own a tool before they can use it. Therefore, they spend less on capital investment. This is especially attractive to those small firms, who used to settle for less powerful tools. The users can easily switch to using different, more advanced tools shall there are ones available.

They users share their resources with others. Therefore, they will have more flexibility in allocating resources for their projects. For example, through advanced reservation, they can make sure that they will have enough computing resources for certain phases of the project. On the other hand, when their resources are idle, they can earn credit for later usage by letting someone else to use the resources through the broker.

The hardware vendors (i.e., makers of powerful computing servers) can bring in revenue without even shipping machines out of their factory. Through the broker, they will be able to make thousands of machines accessible to the users. There will be dramatic cost reduction in cabinating, shipping, installation, and field support.

The software developers could have their tools used by all users without much marketing effort. Of course, there must be well-established standards on input/output formats for each type of tools.

5 Future Challenges

Although Internet in general and computing brokerage in particular are very promising in supporting VLSI design, there are many technical challenges before we can make computing brokerage a reality.

1. There is confidential concern from the users over their design data (trade secrete) being stolen during its transmission to and from the remote site. There is research going on cryptography for such applications as electronic commerce. Its progress will help easing this concern.

2. The communication bandwidth available to date is inadequate for smooth interactive usage of remote resources. Communication technology is being pushed to meet high bandwidth demands such as video conferencing and video-on-demand. When it becomes matured, there will be no difference between interactive usage of remote resources local resources.

3. There may be not too many tools that require no on-site support. Presently, many CPU-intensive, matured tools such as SPICE circuit simulator

and Verilog logic simulator require virtually no on-site support. Therefore, they are suitable for the broker to manage. In the future, if the paradigm of computing indeed shift, more software tools will be developed in such a way that its usage requires very light or no on-site support. This requires strict definition of standards on input/output formats for various types of tools.

4. CAD vendors may feel that the higher the tool utilization, the smaller the number of seats they can sell. Therefore, they may restrict a tool to organizational use only.

In addition to computing service, a broker can be created for the exchanging of intellectual properties as well. With well-defined interface and protocol, any reusable design can be marketed through the broker, while a designer or integrater can better locate the most suitable components for his project.

6 Conclusion

Internet is changing the ways we work. It certainly will change the interaction among researchers, tool developers, and tool users. It will no doubt greatly increase our productivity. To fully utilize its power, we have to redesign the flow of knowledge, tools, and service among researchers, vendors, and users.

An experimental WWW-based computing broker has been implemented. It is useful for certain types of computations such as circuit and logic simulation.

Because it departs so much from the present day computing model, a lot of effort needs to be done before the brokerage-based computing model can be widely accepted.

7 Acknowledgment

Mr. D-Y. Liu and Mr. H-H. Chang implement the WWW-based computing broker, **CompBro**.

References

[1] Don Tapscott, "Digital Economy," *McGraw Hill Publishing Co.*, 1996.

[2] Design Technology Warehouse Home Page, University of California, Berkeley, (URL: http://www-cad.eecs.ucb.edu)

[3] The Computing Brokerage Home Page, Tsing Hua University, Hsin-Chu, Taiwan, (URL: http://theda28.cs.nthu.edu.tw)