PI CACULATING BASED ON GRID COMPUTING

WONG AI WA

A report submitted in fulfillment of the requirement for the award of the degree of
Bachelor of Computer Science (Computer Systems and Network)

Faculty of Computer Systems and Software Engineering
University College of Engineering & Technology Malaysia

NOVEMBER, 2005

ABSTRACT

Nowadays computer technology is moving towards aggregating numerous compute resources (from mainframes to desktops) and abstracting them into a single, unified resource that provides a pool of compute power. Grids computing apply this concept to form a virtual computer resource to compute the task of complex scientific problem such as calculating the value of Pi in Mathematic in a short respond time. This project of calculating the value of Pi in Mathematic is developed using a .NET based grid computing application middleware which is Alchemi and is written using C# programming language. The result of this development is to compute the value of Pi in Mathematic in a very short respond time where meet the goal of grid computing to move response times from days to hours/minutes/seconds.

ABSTRAK

Teknologi komputer masa kini sedang bergerak ke arah mengumpulkan pelbagai sumber komputer (dari kerangka utama hingga ke komputer peribadi) dan mengabstrakkan sumber komputer itu menjadi satu sumber kuasa komputer untuk digunakan. Pada masa yang sama, Grid Komputan telah mengaplikasikan konsep ini untuk mewujudkan satu sumber kuasa komputer yang maya untuk menyelesaikan masalah saintifik yang kompleks seperti mengira nilai Pai dalam Matematik dalam masa yang singkat. Projek ini bertujuan untuk membangunkan satu sistem untuk mengira nilai Pai dalam Matematik dengan menggunakan perisian Alchemi yang menggunakan aplikasi Grid Komputan .NET dan menggunakan bahasa pengaturcaraan C#. Hasil pembangunan sistem ini adalah mengira nilai Pai itu dalam masa yang singkat iaitu dari sejam ke seminit atau beberapa saat sahaja dah ini memenuhi tujuan aplikasi Grid Komputan.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

API	-	Application Program Interface
CLR	-	Common Language Runtime
CPU	-	Central Processor Unit
GB	-	Gigabytes
IDE	-	Integrated Development Environment
IT	-	Information Technology
MSDE	-	Microsoft SQL Server 2000 Desktop Engine
OS	-	Operation System
SDLC	-	System Development Life Cycle
r	-	Radius
UI	-	User Interface
VB	-	Visual Basic

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Since the starting of the innovation of computer, human are trying to increase the raw speed of individual computers, yet they are still far too slow for many challenging scientific problems. These problems need greater requirements for compute power to solve. Scientists cannot take advantage of existing Information Technology (IT) capacity, even though woefully underutilized. Unfortunately, today's Information Technology (IT) environments are simply not responsive or flexible enough to meet these challenges.

Most environments comprise vastly underutilized hardware dedicated to specific applications, services or complex scientific jobs. These tightly coupled systems have limited scalability and are typically tied to expensive, proprietary hardware or increasingly maintenance-intensive distributed infrastructure that cannot scale effectively.

As a result of that human now are taking the next step toward to gather compute resources to solve scientific problems. Grid computing technology has met this point where it uses the recourse of many separate computers connected by local area network or internet to solve large-scale computation problems. This also meets the need of calculating complex Mathematical value such as Pi where Pi value is the base value for many scientific calculations.

1.2 Problem Statement

π may be defined either as the ratio of a circle's circumference to its diameter, or as the ratio of a circle's area to the area of a square whose side is the radius. Nowadays for ordinary calculation that relate to Pi just need to use the decimal value till two (2) decimal points but it is different for the scientist such as astronomies and engineers where they need to get the most precise value of the Pi to calculate the area and the distance of a place. Getting the most precise Pi value task has been carry out over the last few centuries has been put into computing more digits and investigating the number's properties.

Getting the precise value of Pi in Mathematic need lots of humans' brain and energy to calculate. It may be improve using calculator yet it still need sometimes to complete. Although there are many software that are available in market that used to calculate Pi value, yet still some are not accurate from the side of floating point of the value and can not calculate it in a short period of time. It is suggested that getting 100 decimal floating point value of Pi is the base for scientific calculation.

Theses suggest that grid computing system can solve the problem. Grids computing apply the concept of resource virtualization by aggregating numerous compute resources and abstracting them into a single to compute the task of calculating the value of pi in Mathematic in 100 decimal floating points. It can divide the Pi calculating task into several parts and carry out by numerous compute resources that are connected.

1.3 Objectives

There are a few objectives that need to fulfill in developing the system. The objectives are as below:

i. To develop a system to calculate Pi value in 100 decimal floating points using Grid Computing compute the each task parallel.

ii. To allow the system to fully utilize the compute resources that are connected and available to compute the task in order to reduce the processing time.

iii. To implement Grid Computing concept in this Pi calculating application to process Pi value calculation quickly and easily through the time reduction and the simple interface for user to use.

1.4 Scope

The scope of this project is to develop application base on Grid Computing system to calculate Pi value in Mathematic. The Pi value should be calculated in 100 decimal floating points. This application is work within 2 or more connected workstations in a local area network.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

At a basic level, grid can be view as an aggregation of multiple small unit of parallel execution process in a CPU and up to a high level of abstraction of a "virtual machine" with multiple CPUs. Many software developers have been exploring into grid computing application especially developing Windows based grid computing. "Luther,A.,Rajakumaran,B.,Rajiv,R. and Srikumar (2002) Alchemi has implement Microsoft's .NET framework in developing Windows based grid computing framework. Alchemi grid computing framework makes grid construction and development of grid software as easy as possible without sacrificing flexibility, scalability, reliability and extensibility".

2.2 Grid Computing

In the computing world there are many areas where technical development can't keep up with the demand for computational resources. Sometimes, workarounds used to overcome such deficiencies gain a life on their own and become the basis for new developments. As an example, modern particle physics experiments, such as the upcoming LHC experiments at CERN/Switzerland or the BaBar experiment at SLAC (Stanford, USA) will, over the years, produce more data than can be realistically processed and stored in one location, even when using sophisticated cluster architectures. Predictions for the data production of the four LHC experiments are in the range of one Petabyte per experiment per year, or altogether a data rate of 40 GBit/s. What's more, as the experiments evolve and particle accelerators become more sophisticated, the predicted growth in data production over the years far exceeds the predicted growth of computing power.

The latter is described by Moore's Law, according to which the processing power doubles every 18 months. So a local cluster of a given size won't be able to keep up with processing this data, even if it is constantly being upgraded to the newest technology. In such a situation, one has but two choices:

i. One can try to find additional monetary resources to frequently increase the computing and storage power in the location where the data originates.

ii. One can try to use distributed computing- and storage resources already available in participating institutions - particle physics experiments are international by design.

The lack of suitable, standardized solutions for distributed, large-scale computation has sparked a new research discipline, called GRID computing. The vision behind this new approach was first put forward by Ian Foster and Carl Kesselman in their book "The GRID - Blueprint for a new computing infrastructure". In short, it could be described as "Computing Power from a Plug in the wall". In the end, one shouldn't need to care for the location where data is being processed. Really important are speed, safety, and reproducibility of results. It is this obvious analogy with the electrical power grid that has also given the name to GRID computing

A grid is a type of parallel and distributed system that enables the sharing, selection, and aggregation of geographically distributed "autonomous" resources dynamically at runtime depending on their availability, capability, performance, cost, and users' quality-of-service requirements. Computational grids that couple geographically distributed resources are becoming the de-facto computing platform for solving large-scale problems in science, engineering, and commerce. Computational Grids enable the sharing, selection, and aggregation of a wide variety of geographically distributed computational resources (such as supercomputers, compute clusters, storage systems, data sources, instruments, people) and presents them as a single, unified resource for solving large-scale compute and data intensive computing applications (such as molecular modeling for drug design, brain activity analysis, and high energy physics).

Grid computing harnesses a diverse array of machines and other resources to rapidly process and solve problems beyond an organization's available capacity. Academic and government researchers have used it for several years to solve large-scale problems, and the private sector is increasingly adopting the technology to create innovative products and services, reduce time to market, and enhance business processes.

The term grid, however, may mean different things to different people. To some users, a grid is any network of machines, including personal or desktop computers within an organization. To others, grids are networks that include computer clusters, clusters of clusters, or special data sources. Both of these definitions reflect a desire to take advantage of vastly powerful but inexpensive networked resources. Commonly human are focusing on the use of grids to perform computations as opposed to accessing data, another important area known as data grid research.

"Rüdiger Berlich and Dr. Marcel Kunze (2002) current grid uses such as SETI@home which taps personal computers on an as-available basis to analyze data obtained in a search for evidence of intelligent life elsewhere in the universe that allow the spreading of a complex calculation over hundreds, thousands, or even millions of machines using a local area network (LAN) or the Internet". Although the computational problems solved today by grid computing are often highly sophisticated, the software available to manage these problems cannot handle connected parallel applications. As it turns out, creating a parallel application to run on a grid is even more difficult than creating a large monolithic custom application for a dedicated supercomputer or computer cluster.

Grid computing need the help of specialized middleware where software glue that connects an application to the "plumbing" needed to make it run that effectively hides the complexity of creating and deploying parallel grid applications. Such user-friendly middleware for connected parallel processing does not yet exist, but its development should automate the process and make it possible for people to run connected parallel problems without detailed knowledge of the grid infrastructure.

Some industrial applications are important enough to warrant the use of dedicated high-end computers such as supercomputers or clusters of computers. There are much larger body of scientific and engineering applications stands to benefit from grid computing, including weather forecasting, financial and mechanical modeling, immunology, circuit simulation, aircraft design, fluid mechanics, and almost any problem that is mathematically equivalent to a flow.

Grid computing is becoming a critical component of science, business, and industry. Making grids easy to use could lead to advances in fields ranging from industrial design to systems biology to financial management. Grids could allow the analysis of huge investment portfolios in minutes instead of hours, significantly accelerate drug development, and reduce design times and defects. With computing cycles plentiful and inexpensive, practical grid computing would open the door to new models for compute utilities, a service similar to an electric utility in which a user buys computing time on-demand from a provider.

2.2.1 Requirement of Grid Computing

Imagine when someone wants to submit a compute job to the GRID. There are certain parameters needs to be sure of. First of all, one wants to know that the job is submitted to a computer that fits the requirements of the program. Such requirements may include the processor type, local storage capacity for temporary files, amount of RAM and various other hardware parameters. Still, the idea behind GRID computing is that we do not need to know where our program is actually executed. So instead of choosing a machine from a list, one need to describe the requirements of the program in a way that can be understood by some GRID component responsible for choosing the target machine.

If handling sensitive data, one needs to know that no unauthorized party can gain access to it. Likewise, the owner of the machine used to do the computation needs to know that Grid program are using his hardware only in the way it is intended to. In short, there must be a trust relationship between the person who submits the job and the owner of the target machine. The complicated part is that these two people do not know each other and indeed should not need to have to interact in any way in order to allow the job submission.

Before program execution starts, any data needed by the program in order to do its job must be accessible to it from the target machine. Usually this means copying some data set over the network before transferring the program code. Alternatively, one could bring the program to the data rather than vice versa. During and after the calculation the Grid program need to get access to the output created by the program, so this information needs to be transported back to the Grid program.

Most of these requirements of Grid computing could probably be satisfied using existing tools. E.g., With a Virtual Private Network (VPN) and batch submission systems such as PBS it would be possible to submit jobs on remote machines in a secure and reliable way. But while many tools for distributed computing are available, they do not form – and, more importantly, do not intend to form a homogeneous approach. So the task at hand now is the creation of a standardized software infrastructure suitable to the requirements of Grid computing.

2.2.2 Grid Computing technology

"T. Silvestre, E. Nugues, G. Perrière, M. Gouy & L. Duret (1918) DATAGRID is an European research project that endeavor to set up an international grid between widely distributed scientific communities". A production grid has been set up between 5 countries (Netherlands, Italy, United Kingdom, Germany and France) regrouping up to 11 sites. Each site is composed of PC farms that may vary from tens of processors to several hundreds.

2.2.2.1 Hardware infrastructure for Grid Computing

It shouldn't come as a surprise that some of the main initiatives related to GRID computing deal with the formation of highspeed networks and the provision of large clusters. Here are two of the more well-known efforts:

i. Geant is a four year project, set up by a consortium of 27 European national research and education networks, to form a fast, pan-European network infrastructure. It incorporates nine circuits operating at speeds of 10 Gbit/s plus eleven others running at 2.5 Gbit/s.

ii. The TeraGrid is an effort to build and deploy the world's largest and fastest distributed infrastructure for open scientific research. When completed, the TeraGrid will include 13.6 teraflops of Linux Cluster computing power distributed at the four TeraGrid sites, facilities capable of managing and storing more than 450 terabytes of data, high-resolution visualization environments, and toolkits for GRID computing. These components will be tightly integrated and connected through a network that will initially operate at 40 gigabits per second and later be upgraded to 50-80 gigabits/second.

2.2.2.2 Software infrastructure for Grid Computing

In GRID computing, the link between applications and the physical GRID infrastructure is provided by middleware. It is the middleware's task to addresses most of the requirements of Grid computing. Due to the huge variety of projects, the following list can only be a subset of the middleware packages used in science:

i. The most common software component in GRID computing today is the Globus toolkit. It is an open source software toolkit used for building Grid systems and applications. It is being developed by the Globus Alliance and many others all over the world.

ii. Alchemi is an open source software framework that allows developer to painlessly aggregate the computing power of networked machines into a virtual supercomputer and to develop applications to run on the grid. It is suitable for complex calculation and image processing.

iii. Cactus is a higher-level middleware targeted more at computing GRID projects than data GRIDs.

iv. Legion falls into the same category as Globus, but aims more at generating the illusion of a single, distributed computer.

v. UNICORE (Uniform Interface to Computing Resources) offers a ready-to-run Grid system including client and server software. UNICORE makes distributed computing and data resources available in a seamless and secure way in intranets and the internet. Unicore allows for seamless interoperation of supercomputers over WAN.

vi. The Sun Grid Engine is a commercial GRID middleware, incorporating many of the features of a load leveler. It orchestrates the delivery of computational power based upon enterprise resource policies set by the organization's technical and management staff. The Sun Grid Engine system uses these policies to examine the available computational resources within the Campus grid, gathers these resources, and then allocates and delivers them automatically in a way that optimizes usage across the Campus grid.

2.2.3 Benefit of Grid computing technology

In the parallelization of 200 jobs on the grid, a five fold acceleration factor has been raised compared to a standalone computer.These results seem not to be so drastic in regards to what we could obtain in theory with a highly parallel mainframe machine. However, one has to consider that these data are reflecting real use conditions, with a lot of user submitting jobs at the same time.

Moreover resources availability is far much better compared to a batch system. In an average out of 100 job submissions, 10 jobs were directly running across the different sites as showed in Figure 2.1. Grids will be perhaps the most appropriate solution in regards to the different field specific needs. Grids of individual workstation may be more suitable for an intranet use in a company or for small jobs over the internet whereas grids of computing center are probably more efficient for highly intensive computational jobs and fulfill stability and security requirements.

Grid computing help in cost savings, the initiative will provide an unprecedented amount of computing resources and create a new research computing capability that will help any organization or scientists in advancing its research. Larger and more complex research problem can be solves more accurately with increased temporal and spatial resolution.

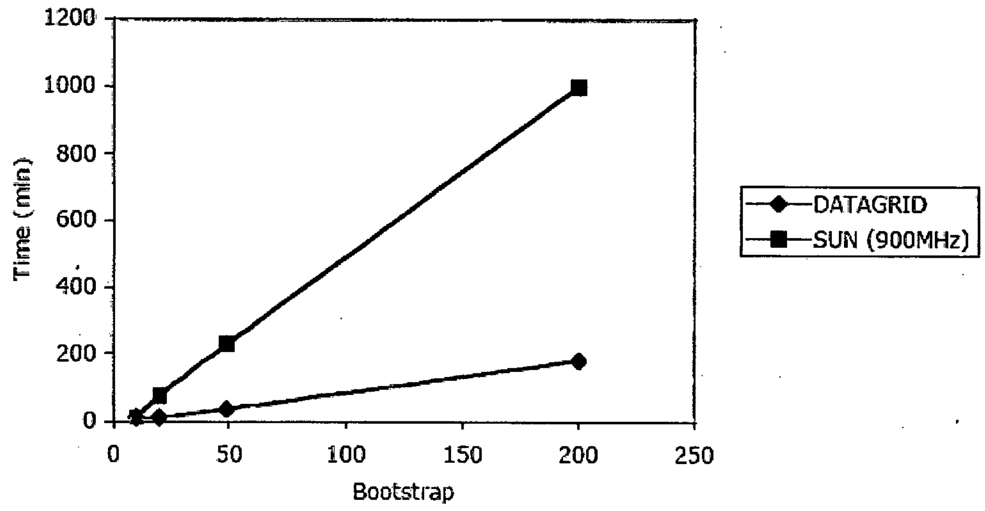

Figure 2.1: CPU time estimation of bootstrap calculation in DATAGRID environment compared to a standalone computer.

2.2.4 Issues in Grid Computing

There are numerous issues being discussed in Grid Computing as the major issue will be on the performance of Grid Computing. As in Grid Computing it involve the usage of network it will relate to the speed of a network include the bandwidth (the number of bits you receive per second on one end of the network) and the latency (the amount of time it has taken these bits to travel from the source to the recipient. But today one can scale the bandwidth of a network connection to virtually any level yet there are physical limits to its latency. Obviously, data cannot travel faster than the speed of light. So there is a lower limit to the amount of time needed to transfer the data, no matter how sophisticated network hardware is. But since this data will have to pass repeaters and routers along the way, the actual latency will be much higher than the physical limit. So the performance of the Grid Computing is still depend on the network bandwidth and latency. Issues of security, resource management and load balancing are also fundamental to be explore in Grid Computing

A key concept in Grid Computing is that of 'ownership'. A grid is a 'federation' of resources which can be accessed in a transparent way by authenticated grid users. This raises the fundamental question of accounting for resource usage, whether it is CPU time, disk, memory, license for software or preserved data. This perhaps needs to be considered in implementing a national grid environment.

"R.J. Allan(Daresbury), J.M. Brooke and F. Costen (MRCCS) and M. Westhead (EPCC) (1999)Grid computing describes the linking together of distributed computational resources to provide flexible access and a common interface for the user. This is sometimes referred to as an eServices environment in commercial applications". Meta-computing extends this concept to enable distributed systems to compete with supercomputers in order to try to overcome the limitations of a single computing system. To achieve these goals software systems must be provided which use Internet technology, now common in eCommerce, for the benefit of the computational science community. This has recently been referred to as eScience..

2.3 Pi Calculation

"Sangiorgi D. and Walker D (2003) Pi, Greek letter (Pi) used in mathematics as the symbol for the ratio of the circumference of a circle to its diameter. Its value is approximately 22/7; the approximate value of Pi to five decimal places is 3.14159. Pi is an infinite decimal. Unlike numbers such as 3, 9.876, and 4.5, which have finitely many nonzero numbers to the right of the decimal place, pi has infinitely many numbers to the right of the decimal point". The formula for the area of a circle, A = Pi r^2 (r,. is the radius), uses the constant. Various approximations of the numerical value of the ratio were used in biblical times and later. With computers, the value has been figured to more than 100 million decimal places, although this has no practical value. The ratio is actually an irrational number, so the decimal places go on infinitely.

Pi goes on forever, and can't be calculated to perfect precision. 3.1415926535897932384626433832795028841971693993751.... The basic idea is to throw random darts into the square (-1,-1) to (1,1). This is a 2x2 square, and thus has an area of 4. The circle inscribed in this square will have a diameter of 2, and thus a radius of 1. The circle's area is just *Pi * Radius^2*, which is just Pi, since the radius is 1. So, if every point in the square is equally likely, then the ratio of the number of darts that fall into the circle to the total number of darts thrown should just be the area of the circle divided by the total area -- which is just Pi/4. So, by counting the number of darts inside the circle and the total number of darts thrown, and multiplying this by 4, we should get the value of Pi. As we throw more and more darts, our estimation of Pi should get closer and closer to the real value of Pi.

The ancient Egyptians already start to know the value of pi. Those people who measure the dimensions of pyramids claim tremendous accuracy, much more accuracy than the rough exterior of a pyramid warrants. So, when they say that the Great Pyramid exhibits the value pi, to many decimal places, a person should be skeptical. But, it is also not very remarkable that the ancient Egyptians might have used pi in this pyramid. It seems that the base (much of it is missing) of the pyramid is within a few inches of being square, and is just a few minutes of arc from being aligned with true north, very accurate indeed.

The ancient Egyptians seem to have sometimes used a value of 22/7 (3.142857 . . .) for pi. There is also evidence that they estimated the area of a circle with a square with a side that is 8/9 the size of the circle's diameter. This gives a value of pi of 3.16049382716 . . .