J. of university of anbar for pure science : Vol.3:NO.3 : 2009

ISSN: 1991-8941

Design and Implementation of Virtual Grid-Based Parallel Computer (VGBPC)

Rabah Nory Farhan
University Of Anbar - College of Computers,

Abstract : Many Areas in Science and industrial applications requires huge computation power that is must achieve the desired level of computation. The field of distributed computing covers all aspects of computing and information access across multiple processing elements connected by any form of communication network. In this paper we investigate the design and implementation of a grid based system depending on an Intranet based on Windows operating system. The proposed Virtual Grid-Based Parallel Computer (VGBPC) system was written using C# and .NET 3.5 framework. The system uses the remote computer as a node to accomplish the execution task. The Grid Manger system was built to be initiated on the server of the Intranet which responsible of partitioning the task and assigning each partition to Grid thread that is transmitted to the Grid Executer. The Grid Executer implemented in this work responsible of receiving the threads from the Grid manger, executing them and resending them back to the Grid Manager. The proposed system was used to accomplish parallel Prime Number Checker to investigate the (VGBPC) capabilities.

Keywords: Distributed Computing, Parallel Computer, .Net, Intranet, Grid, Thread Manager

Introduction

Since the advent of the Internet in the 1970s, there has been a steady growth of new applications requiring distributed processing. This has been enabled by advances in networking and hardware technology, the falling cost of hardware, and greater end-user awareness. These factors have contributed to making distributed computing a cost-effective, high-performance, and fault tolerant reality [2] .

The Grid is a heterogeneous collaboration of resources and thus will contain a diverse range of data resources. Heterogeneity in a data Grid can be due to the data model, the transaction model, storage systems, or data types. Data Grids provide seamless access to geographically distributed data sources storing terabytes to petabytes of data with proper authentication and security services[4].

The development of a Grid infrastructure was necessary for large-scale computing and data-intensive scientific applications. A Grid enables the sharing, selection, and aggregation of a wide variety of geographically distributed resources including supercomputers, storage systems, data sources, and specialized devices owned by different organizations for solving large-scale resource-intensive problems in science, engineering, and commerce. One important aspect is that the resources—computing and data—are owned by different organizations. Thus the design and evolution of individual resources are autonomous and independent of each other and are mostly heterogeneous.

A Grid can be defined as a set of dedicated and independent machines, connected by means of an internal network, and managed by a system that takes advantage of the existence of several computational elements. A Grid is expected to provide high performance, high availability, load balancing and scalability.

Evolution of grid computing

- In the early of 1990s, the first generation efforts were marked by the emergence of metacomputing projects, which aimed to link supercomputing sites to provide access to computational resources. Two representative projects of the first generation are FAFNER and I-WAY[4].
- Portable Batch System (PBS). The PBS project is a flexible batch queuing and

workload management system originally developed by Veridian Systems for NASA. The primary purpose of PBS is to provide controls for initiating and scheduling the execution of batch jobs.

- Maui scheduler. The Maui scheduler, developed principally by David Jackson for the Maui High Performance Computer Center, is an advanced batch job scheduler with a large feature set, well suited for high performance computing (HPC) platforms.
- Load Sharing Facility (LSF) LSF is a commercial resource manager for clusters from Platform Computing Corporation2. It is currently the most widely used commercial job management system. LSF focuses on the management of a broad range of job types such as batch, parallel, distributed, and interactive. The key features of LSF include system supports for automatic and manual checkpoints, migrations, automatic job dependencies and job re-scheduling.
- Grid Portal Development Kit (GPDK) The GPDK is a widely used toolkit for building non portals. The GPDK is collaboration between NCSA, SDSC and NASA IPG, which aims to provide generic user and application portal capabilities. It facilitates the development of grid portals and allows various portals to inter-operate by supporting a common set of components and utilities for accessing various grid services using the Globus infrastructure[3,9].

Parallel Computer Architecture.

A parallel computer(PC) {see Figure (1)} is consisting of two or more identical CPUs connected to a single shared main memory via a common bus (see Figure). The cost of accessing the shared memory is the same for all CPUs, for which reason PC are also called Uniform Memory Access (UMA) architectures. Each CPU, however, may have its local cache to exploit data locality and reduce the crude memory access latencies. The PC architecture supports the Single Instruction Multiple Data (SIMD) and Multiple Instructions Multiple Data (MIMD) programming models in Flynn's taxonomy. The PC machine uses a single operating system and all CPUs share the same input and output resources. SMP computers allow any CPU work on any task, no matter where the data for that task is located in memory. With proper operating system support, PC can easily move processes between CPUs to balance the workload effectively[2,10].

Virtual Grid-Based Parallel Computer (VGBPC)

Introduction:

The (VGBPC) system was written in C#.Net 2008 and works over an intranet works on Windows Platform (windows 2003 Server){see Figure(2)}. The main purpose of the system is to install a special application called the "Grid Manager (GM)" in the sever of the Intranet that is responsible for managing another type of application called "Grid Executer (GE)" that is responsible for executing the sub of intended task. The partition process occurred in the GM on the server Node as seen in the figure(2). The application needed to be implemented on the (VGBPC) system processed by the (GM) and partitioned into subtasks that is assigned to the special threads. The GM send these threads remotely to local and global CPUs in the Intranet. Each node in the intranet executes the received thread and return the result back to the GM. The GM responsible of synchronization of the implementation of these threads and cumulate the output for final result. It is assumed that if we have in the Intranet (n) node, then the parallelism produced is of the factor (n).

Multithreading in VGBPC):

The notion of Multithreading means assigning a special context for the resources of the computer system that is shared for one part of the application. It would be nice if we could perform one action at a time and perform it well, but that is usually difficult to do. The human body performs a great variety of operations in parallel or, concurrently. Respiration, blood circulation and digestion, for example, can occur concurrently. All the senses sight, touch, smell, taste and hearing can be employed at once. Computers, too, perform operations concurrently. It is common for your computer to be compiling a program, sending a file to a printer and receiving electronic mail messages over a network concurrently. The .NET Framework Class Library provides concurrency primitives. You specify that applications contain "threads of execution," each of which designates a portion of a program that may execute concurrently with other threads this capability is called multithreading. Multithreading is available to all .NET programming languages, including C#, Visual Basic and Visual C++. The .NET Framework Class Library includes multithreading capabilities in namespace `System.Threading`.A `Thread` object begins its life cycle in the Unstarted state when the program creates the object and passes a ThreadStart delegate to the object's constructor.

The `THReadStart` delegate, which specifies the actions the thread will perform during its life cycle, must be initialized with a method that returns `void` and takes no arguments. The thread remains in the *Unstarted* state until the program calls the *`Thread`'s `Start`* method, which places the thread in the Running state and immediately returns control to the part of the program that called `Start`. Then the newly Running thread and any other threads in the program can execute concurrently on a multiprocessor system or share the processor on a system with a single processor[1,5].

Grid Manger Architecture:

The GM may be regarded as the main part of VGBPC) system. GM architecture represented in the Figure (3). The main functions of GM parts is as follows:

- Initialize GM. In this stage, all the parameter for the GM is initialized and the GUI is rendered for the user.
- Initialize Task Pool. The Task Pool is the stage where the User prepare the application or task to be implemented in the proposed system.
- Establish Connections with GE Pool. This stage check the connections to the GE, that is running remotely on the other nodes in the intranet. If there exist such GE, the connection will be established using TCP protocol. The .Net framework supported with very rich tools and classes to deal with network sockets and tools.
- Update GM_Execuer_List. If the connections exist with remote GE, the GM_Executer_List will updated to make ensure that these GEs appended to the list.
- Get Task from Task_Pool. The Task_Pool contains the list of tasks needed to be executed in the system. It must contains at least one task to be executed concurrently.
- Initialize_Thread_Manager. In this stage, the task is assigned for the thread generated in this stage and mark it as Gob to be sent to the remote Node via Gob_Transmitter.
- Gob_Transmitter. In job_Transmitter, the Communication channel prepared for transmission the thread as XML file to the specified node. The remote Internet Protocol (IP) is reserved during the third stage. The protocol used is TCP protocol for authentic connection. An XML parser was used to encode and decode the transmitted Gob. The results obtained is translated and transmitted back using Gob_Receiver with same mechanism.

Grid Executer Architecture(GE):

The GE is the executer of the transmitted gob over the network. The GE identical to the Processor Element in the architecture of the real parallel computer. The GE is initialize at startup to be in the Listen State for any coming connection from the GE. If the connection, the Gob_reciever at GE extract the gob and set local thread for Gob execution and send the result to the GE Transmitter for sending the result to the GM as seen in the figure (4).

(VGBPC) implementation

The proposed system was setup and initializing on a simple case study intranet consisting of three computers. First computer is a laptop that is installing with (Windows Server 2003 and SQL Server 2008) and other computers are installed with (Windows XP SP2). The laptop is configured to be the Intranet Server and installed with the GM. While each other computers are installed with GE and connected with the server through UTP cables.

Graphical User Interface of (VGBPC).

The proposed system Graphical User interface permit for adjusting the type of connection that is exist between the GM and GE and many other configurations. Figure (5.A) shows the GUI of GM and Figure (5.B) shows the GUI of GE.

As seen in the figure (5.A), the GM interface provides us with the necessary tools for configuration of GM. We uses " خادم البيانات " to determine the location of the GM as IP address. The option " اسم مستخدم قاعدة البيانات " determine the name of the user of the SQL Server 2008 user account. This account used by the system to enforce authentication for the VGBPC system. When all the necessary configuration has been setting, the option of "ابدأ" will be initiated to transform the GM into action. The GM will accept any coming connection from GEs. The GUI of GE shown in figure (5.B) contains the name of the user and the password besides the IP and the port of the GM to connect. The user press "توصيل" to connect this GE to the GM. When connected the GE now can execute the coming gobs form the GM.

Executing Parallel Prime Number Finder on VGBPC:

The VGBPC was initiated and configured to be run on our case study intranet. The purpose of this experiment was to implement simple Prime Number Finder algorithm in parallel on our system. the GM was initiated on the Server and

two GE was initiated on the other computer on the intranet. To check for Prime number we set the application to test all the numbers smaller than specified limit. The Test operation simply was to implement Mode Operation on the all numbers less than the tested one against (2). If the result is no more (2) the number is Prime. The following algorithms show the details:

- ***Testing Prime Number in VGBPC Algorithm***
  - ***Input: the Maximum Limit***
  - ***Output: all the Prime numbers less than limit.***
- ***Start***

1) ***Initialize VGBPC***
   - ***Configure the intranet***
   - ***Initiate GM on the Server***
   - ***Initiate GE on the two other PCs***
   - ***Establish the Connections between GM and GEs***
2) ***Configure the Thread Manager to assign the task as Prime Checker Class***
3) ***The Task Partitioner split the overall job into (3) threads***
4) ***Gob Transmitter send the two threads remotely to the other two GEs while executing one thread locally***
5) ***The Result from each GE are collected***
6) ***The VGBPC report the final results to the user as been implemented locally.***

- ***END***

The splitting process occurs periodically, in which the test process for each thread assigned to each node independently performed at node. For example, if we have the threads labeled by the variable H= {t1, t2, t3, t4, t5, t6, t7, t8}. Step (3) of the algorithm assigned (t1) thread into GM, (t2) into GE1, (t3) into GE2, (t4) into (t1) and so on. Figure (6) shows the Prime Checker Class while Figure (7) shows the output from execution of the proposed system.

Conclusion and future works

We have implemented a .NET-based grid computing framework that provides the runtime machinery and object-oriented programming environment to easily construct desktop grids and develop grid applications. Its integration into the global cross-platform grid has been made possible via support for execution of grid jobs via a web services. The developed Grid Manager shows nice flexibility to serve and manage the threads dedicated to be executed on the Grid Executer.

We plan to extend the proposed system in a number of areas. Firstly, support for inter-thread communication and routing algorithms is planned. Secondly, we plan to investigate the multi-clustering with peer-to-peer communication between Managers. Thirdly, we plan to support utility-based resource allocation policies and security protocols driven by economic, quality of services, and service-level agreements.

References

[1] Andrew Troelsen, ""Pro C# 2008 and the .NET 3.5 Platform", APRESS, 2007.

[2] Ajay D. Kshemkalyani, "Distributed Computing Principles, Algorithms, and Systems" , Cambridge University Press 2008.

[3] Choi-Hong Lai, "Grid Resource Management Toward Virtual and Services Compliant Grid Computing",CRC Press, 2009

[4]David Taniar, "High-Performance Parallel Database Processing and Grid Databases", A John Wiley & Sons, Inc., Publication, 2008.

[5] Fiach Reid, "Network Programming in .NET With C# and Visual Basic .NET", Elsevier Digital Press,2004.

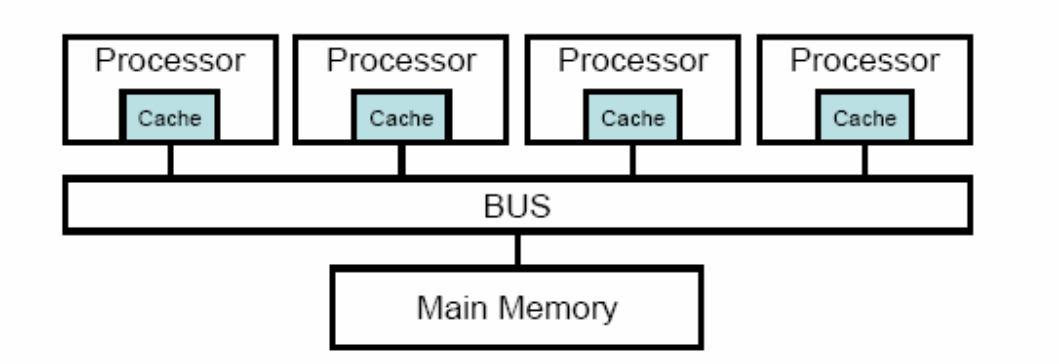
[6] H. Bal, F. Kaashoek and A. Tanenbaum, Orca: a language for parallel programming of distributed systems, IEEE Transactions on Software Engineering, 18(3), 1992, 180–205.

[7] J. M. Arfman and P. Roden, Project Athena: Supporting distributed computing at MIT, IBM Systems Journal, 31(3), 1992, 550–563.

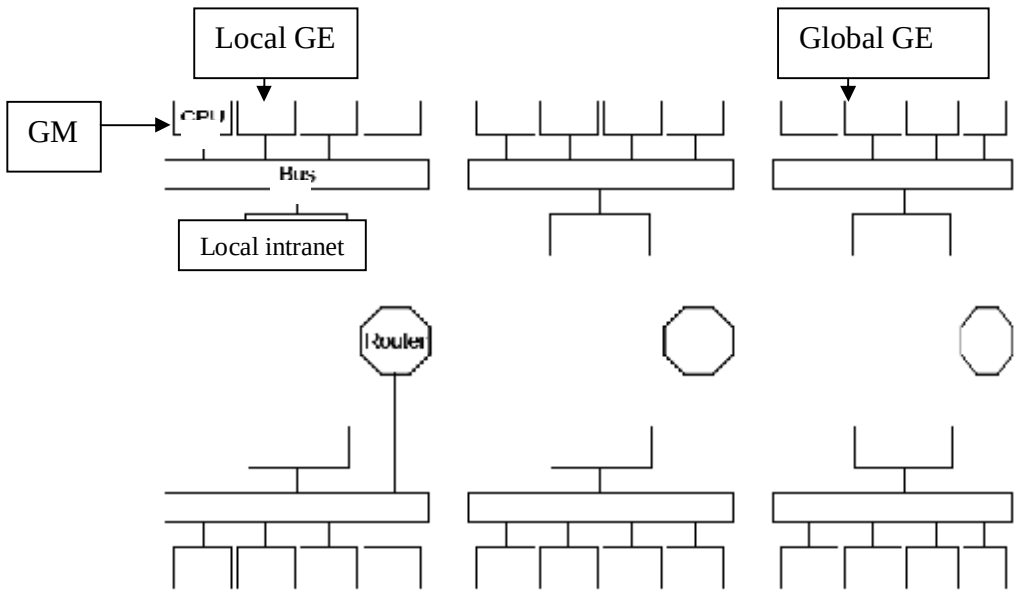
[8] J. King and A. dos Santos, A user-friendly approach to human authentication of messages, Proceedings of the 9th International Conference on Financial Cryptography and Data Security, Roseau, Dominica, 2005, 225–239.

[9] J. Anderson, Multi-writer composite registers, Distributed Computing, 7(4), 1994, 175–196.

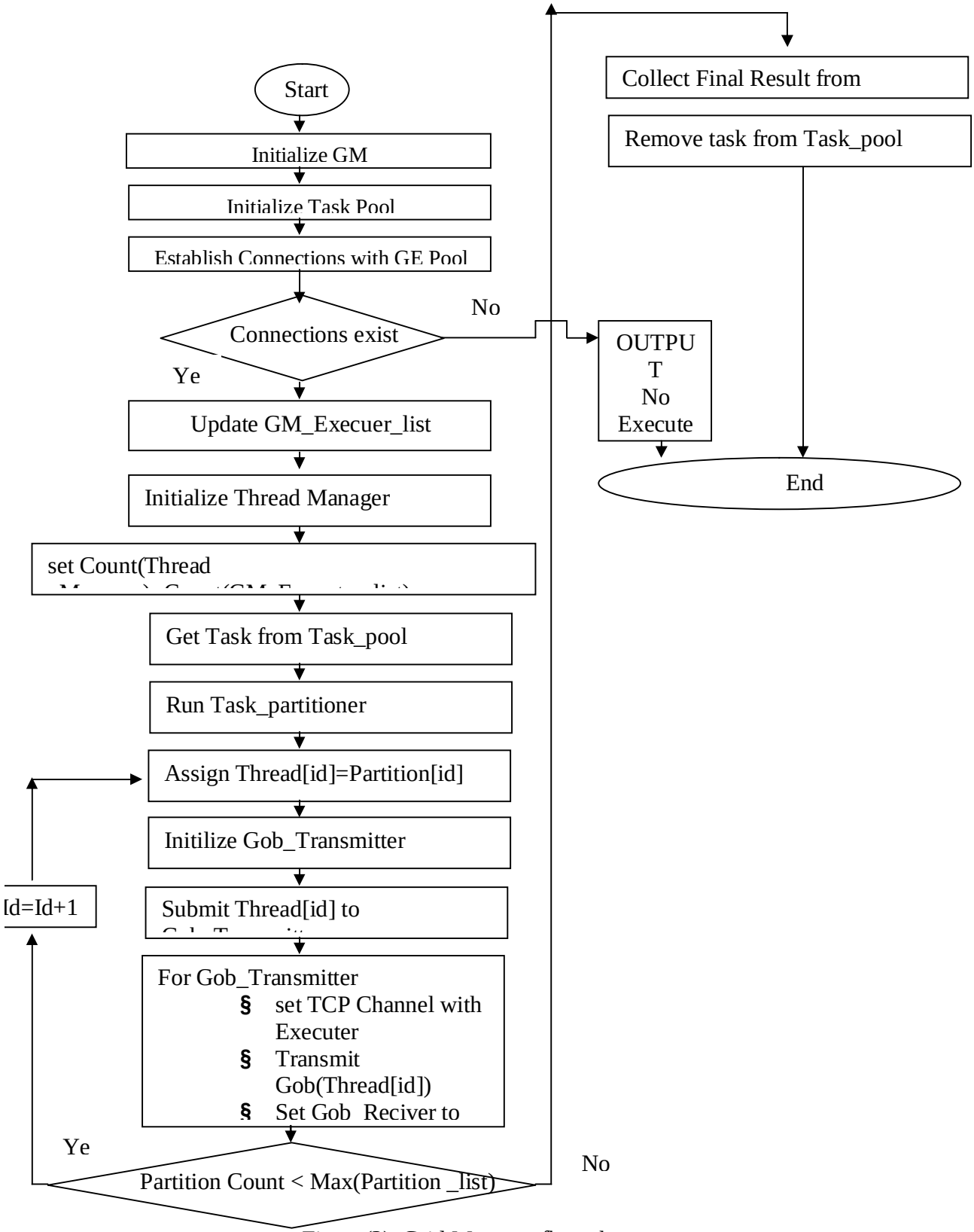
[10] Werner Dubitzky, "Data Mining Techniques in Grid Computing Environments", John Wiley & Sons, Ltd, 2008.



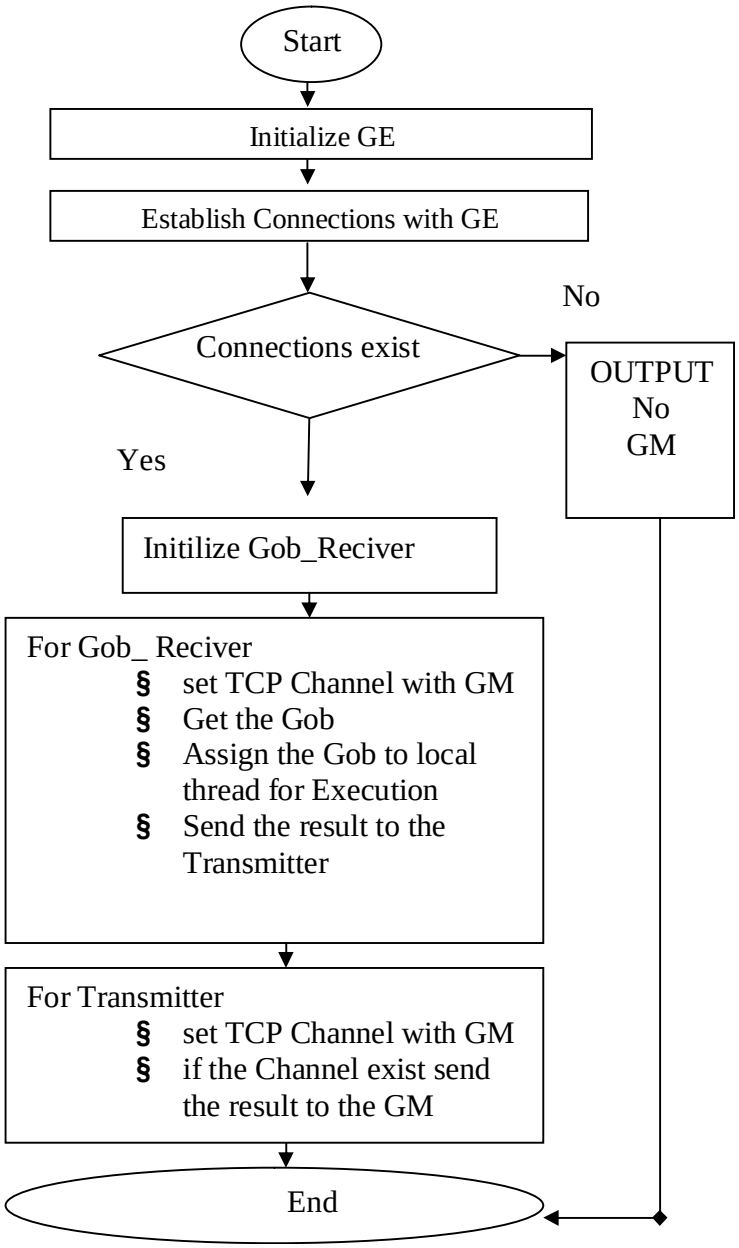
Figure(1). Architecture of Parallel Computer.



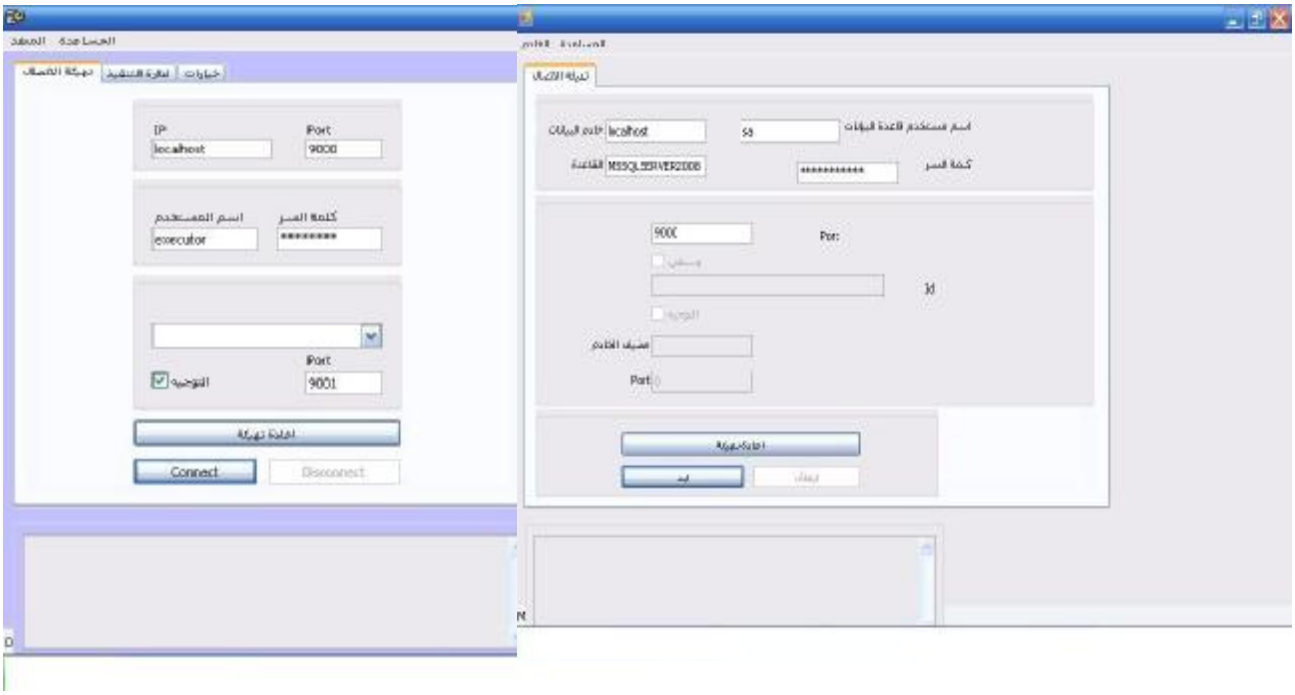
Figure(2). Architecture Virtual Grid-Based Parallel Computer.



Figure(3). Grid Manager flow chart .

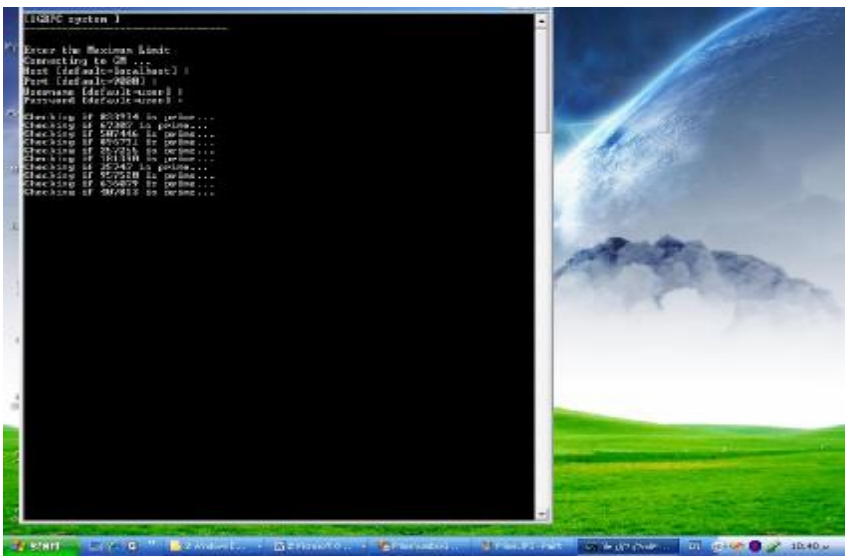


Figure(4). Grid Executer flow chart .



Figure(5).A)Grid Manager GUI B) Grid Executer GUI.

```
class PrimeChecker : VGBPCThread
    {   public readonly int PRM;
        public int F = 0;
        public PrimeNumberChecker(int P)
        { PRM = P;}
        public override void Start()
        {for (int d=1; d<=PRM; d++)
            {if (PRM%d == 0) F++;
```



Figure(6). Prime Checker Class.

J. of university of anbar for pure science : Vol.3:NO.3 : 2009

تصميم وتنفيذ حاسبة متوازية افتراضية معتمدة على الشبكة

د.رباح نوري فرحان
Email:rabahalobaidy@yahoo.com

الخلاصة

تتطلب العديد من التطبيقات في مجالات العلم والصناعة أجراء حسابات كبيره جدا للوصول الى المستوى المطلوب في هذه التطبيقـــات . إن حقل الحساب الموزع يغطي كل مجالات الحساب والوصول الى المعلومات خلال العديد من عناصر الحساب الموصوله بأي شكل من أشكال شبكات الأتصالات. في هذا البحث قمنا بدراسة تصميم وتنفيذ نظام إحتسابي شبكي يعتمد على نظام تشغيل وينـــدوز . إن النظـــام المقترح والذي أسميناه الحاسوب المتوازي الأفتراضي المعتمد على الشبكة قد نفذ بأستخدام (C# and .Net 2008) . النظـــام يـــستخدم حاسبة بعيده حتى تقوم بأجراء مهمة التنفيذ .أن مدير الشبكة تم بناءه ليعمل على الخادم الخاص للشبكة المحلية ويكـــون مـــسؤولا عـــن تجزئة المهمة تحت التنفيذ الى أجزاء ومن ثم نقل هذه الأجزاء عبر الشبكة الى المنفذ على الحاسبة البعيدة .يكون المنفـــذ مـــسؤولا عـــن إستلام هذه الأجزاء من مدير الشبكة وتنفيذها وبعد ذلك إعادة إرسالها الى مدير الشبكة. استخدم النظام المقترح لايجاد وفحـــص الارقـــام الاولية لدراسة خصائصه.