# NIMBLE@ITCECNOGRID VS ALCHEMI.NET: A PERFORMANCE COMPARISON

Vijay Dhir[1] ,Dr. Rattan K Datta[2] and Dr. Maitreyee Dutta[3]

[1]Assistant Professor, Department of Information Technology, SBBSIET, Punjab ,India.
vijaykdhir@yahoo.com
[2] Retd. Advisor Govt. of India, MET, New Delhi,India
rkdatta_in@yahoo.com
[3] Associate Professor, Deptt. of CSE,NITTTR,Chandigarh,India
d_maitreyee@yahoo.co.in

***ABSTRACT***

*The present research work includes the development of a novel toolkit namely Nimble@ITCEcnoGrid for calculating the pi() value up to 120 decimal places after decimal. Alchemi.NET is the other popular toolkit based on Windows Platform for doing the same purpose. There is a very interesting comparison between Nimble@ITCEcnoGrid toolkit and Alchemi toolkit as both runs on Windows Operating System. A test bed is build to compare the performance of the two toolkits. The research paper also includes the tables and the graph to illustrate the superiority of Nimble@ITCEcnoGrid.*

***Keywords***

*Computational Grid, Manager, Sub Manager, Executor, Machin's formula and Maclaurin series.*

## 1. INTRODUCTION

Computational grids[1][2][3]that couple geographically distributed resources are becoming the de-facto computing platform for solving large-scale problems such as weather forecasting, Protein Folding etc. Software to enable grid computing has been primarily written for Unix-class operating systems, thus severely limiting the ability to effectively utilize the computing resources of the vast majority of desktop computers i.e. those running variants of the Microsoft Windows operating system. Addressing Windows-based grid computing is particularly important from the software industry's viewpoint where interest in grids is emerging rapidly.
"Grid Computing" [4][5][6] is a special type of parallel computing which relies on complete computers (having own CPU, Storage, NIC, Power supply etc.) connected to the internet by the conventional network interface, such as Ethernet. It can generate the power of the Supercomputer by working collectively. One such Popular Grid toolkit named Alchemi is based on windows operating system. It works on a .NET platform and can be used to solve the complex problems in a real time scenario. A novel homemade toolkit named Nimble@ITCEcnoGrid is compared with the Alchemi Toolkit. The results obtained by the Nimble@ITCEcnoGrid encouraging and much better than Alchemi Toolkit.
Let's discuss the comparison of the Open Source Alchemi toolkit and our own Nimble@ITCEcnoGrid toolkit. Both runs on Windows operating system. To show the efficiency of Nimble@ITCEcnoGrid toolkit, a pi program for finding the 120 digits after the decimal point is run by using the Alchemi and Nimble@ITCEcnoGrid toolkit. Nimble@ITCEcnoGrid computes the value of PI() up to 120 decimal places in milliseconds as

compared to a more than half a minute by its competitor Alchemi.NET( An Open Source toolkit based on windows OS).

## 2. RELATED WORK

Alchemi[7][8] is the .NET framework that provides runtime machinery and programming environment requires constructing desktop Grid. It supports object oriented programming in addition to file based job model cross platform is provided by web server interface. It has no Inter thread communication.

Entropia [9] uses a window desktop grid system by aggregating the raw desktop resources into a single logical resource. There is a one centralized computer, which administrates various desktop clients. But it does not provide web server interface for cross platform.
Condor[10] system is developed by university of Wisconsin at Madison. Unique mechanism enable candor to effectively harness wasted CPU Power from idle desktop workstations. Uses submit their job to condor, condor places them into a queue, chooses when and where to run the jobs based upon a policy monitors is the progress and informs upon completion. It can handle both windows and Unix class resources in its resource pools. This does not have tread-programming model and does not support cross platform web services interface.

| | **Alchemi** | **Condor** | **Entropia** | **Nimble@ITCEcnoGrid** |
|---|---|---|---|---|
| Architecture | Hierarchical | Hierarchical | Centralized | Hierarchical with centralized database |
| Web Services Interface for cross-platform Integration | Yes | No | No | Yes |
| Implementation Technologies | C#, Web services & .NET Framework | C | C++, Win 32 | C#, Web services & .NET Framework 3.5 |
| Thread programming Model | Yes | No | No | Yes |
| Level of Integration of application, Prog. & runtime Environment | Low (General Purpose) | Low (General Purpose) | Low (General Purpose) | High (Single Purpose Single Application Environment) |
| Inter thread Communication | No | No | No | Yes |

Table 1: Comparison of Nimble@ITCEcnoGrid and some related Enterprise Grid Systems[11]

## 3. COMPONENTS OF Alchemi.NET

There are four types of nodes[7] in Alchemi.NET computational grid construction.

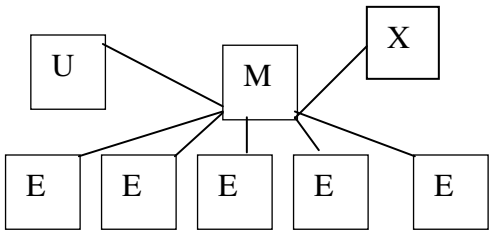



Fig 1: Interaction of Grid Components in Alchemi.NET Toolkit

U- User Node
E- Executor Node
M- Manager Node
X-Cross Platform Manager Node

Alchemi V2.0 toolkit and Nimble@ITCEcnoGrid toolkit both uses C# API and operates in the Windows .NET framework. Users can develop, execute and monitor grid applications using the .NET API and tools which are part of the Alchemi SDK. An optional component is the Cross Platform Manager web service which offers interoperability with custom non-.NET grid middleware.

### 3.1 Alchemi.NET Toolkit Requirements [12]

Alchemi.NET Toolkit H/W & S/W requirements for the test case is

1) 1 GB RAM
2) 1.8 GHz Core-to-Dual Processors
3) Network Interface Card
4) 4 port Network Switch
5) SQL Server 2000 Developer
6) Visual Studio 2003
7) Windows Server 2003 SP2

## 4. COMPONENTS OF Nimble@ITCEcnoGrid

Nimble@ITCEcnoGrid toolkit both uses C# API and operates in the Windows .NET framework.

Nimble@ITCEcnoGrid is a web based Interface, which can be accessed & controlled by the Administrator. It has four components

1) Website Channel
2) Manager
3) Sub Manager
4) Executor

There are Nimble@ITCEcnoGrid Managers at different locations i.e. different managers for different locations. There can be multiple managers at one location. This is because, the failure of one manager at some location will not affect the work. Data after calculation is send to database of Nimble@ITCEcnoGrid Web Interface, which will display the results. There are number of Executors connected to the Sub manager which are in turn connected to Manager. Executors after doing the required task will submit the result to the Sub Manager, which in turn send result to Manager. Manager will further send result to the Database of

Nimble@ITCEcnoGrid Website. More the number of executors, less time the manager will take to produce result. The executors can work on real IP's or LAN or VPN because if it get disconnected in between for some reason, after reconnection the IP address will remain same so that cost of contributing the free cycles can be added economically in the database. The benefit of using the above Model is that, it provides the reliability which was not there in Alchemi.Net as instead of one Manager whose failure can affect the Whole Grid there can be number of Managers at one location and failure of any one or more manager will not stop the work. Figure 2 shows the Model of the Nimble@ITCEcnoGrid.

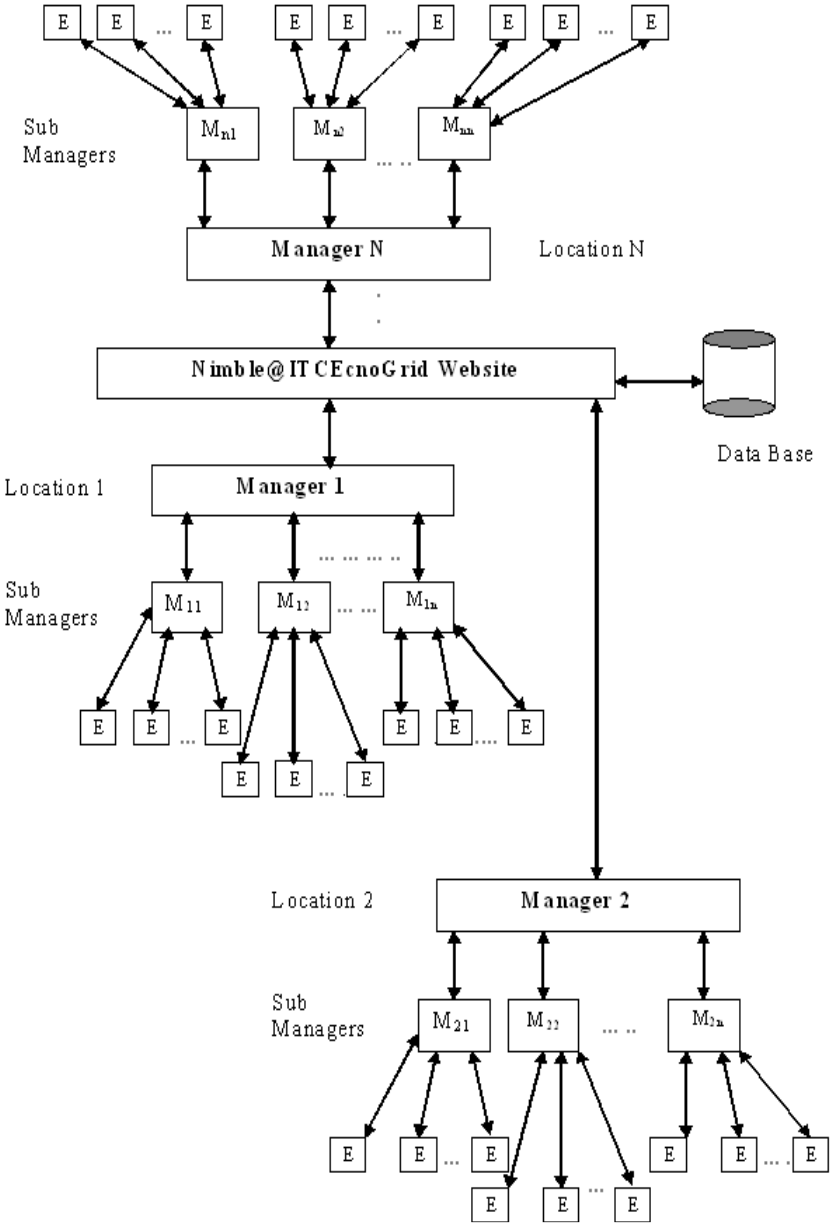



Fig 2: Model of Nimble@ITCEcnoGrid

### 4.1 Hardware & Software Requirements

Nimble@ITCEcnoGrid Toolkit H/W & S/W requirements for the test case is

1) 1 GB RAM
2) 1.8 GHz Core-to-Dual Processors
3) Network Interface Card
4) 4 port Network Switch
5) SQL Server 2008 R2
6) Visual Studio 2008 SP1
7) Windows Server 2008 R2

## 5. DIFFERENCE BETWEEN Alchemi.NET &Nimble@ITCEcnoGrid

Alchemi toolkit has no inter thread communication[13] where as Nimble@ITCEcnoGrid toolkit has inter thread communication with which in case of failure of particular node, work will be distributed to the other executor.

Alchemi has only one Manager whereas Nimble@ITCEcnoGrid toolkit can support number of managers in one or more cities. All the Managers are controlled by the website interface. With this if one manager fails then even we can get the data of that city or other cities (Because there can be multiple sub managers in Nimble@ITCEcnoGrid) whereas in Alchemi if manager stops working, everything stops.
We can deploy more than one manager/Sub manager and associated executors with it. It can give fail-over management in grid but it is not possible in Alchemi. Alchemi has no Economic based policy whereas Nimble@ITCEcnoGrid gives incentives [14] based on the free cycles distribution for grid.

| | **Alchemi.NET Toolkit** | **Nimble@ITCEcnoGrid Toolkit** |
|---|---|---|
| Architecture | Hierarchical | Hierarchical with centralized database |
| Implementation Technologies | C#, Web services & .NET Framework 1.1 | C#, Web services & .NET Framework 3.5 |
| Level of Integration of application, Prog. & runtime Environment | Low (General Purpose) | High (Single Purpose Single Application Environment) |
| Inter thread Communication | No | Yes |
| Economic Based Policy | No | Yes |

Table 2: Comparison between Alchemi.NET & Nimble@ITCEcnoGrid

# 6. IMPLEMENTATION OF Nimble@ITCEcnoGrid TO CALCULATE VALUE OF PI UP TO 120 DECIMAL PLACES AFTER DECIMAL

Pi is one of the most important numbers in mathematics. It is defined as the ratio of a circle's circumference to its diameter, but it crops up in all sorts of places in mathematics. It is an infinitely long non-recurring decimal number.

There are many ways to do this. Some methods converge rapidly but are complicated to implement. Some are simple to implement but converge very slowly. I've chosen a method that is fairly simple and converges reasonably fast. It is based on the following formula:

$$pi / 4 = 4 * \tan^{-1}(1 / 5) - \tan^{-1}(1 / 239)$$

This is Machin's formula[15] and is exact $\tan^{-1}()$ is the Inverse Tangent function, and I use the Maclaurin series [16]to calculate it:

$$\tan^{-1}(z) = z - z^3 / 3 + z^5 / 5 - z^7 / 7 + ...$$

By including sufficiently many terms of this series, we can achieve any desired accuracy. To get 1,000,000 decimal places accuracy for pi, we need about 715,000 terms of the $\tan^{-1}(1/5)$ series and about 210,000 terms of the $\tan^{-1}(1/239)$ series. I have used the program to calculate 120 decimal places accuracy.

# 7. Alchemi.NET TOOLKIT RESULTS

The results of pi program to find 120 digits after decimal point by using Alchemi toolkit ( Show in fig 3, 4 & 5)

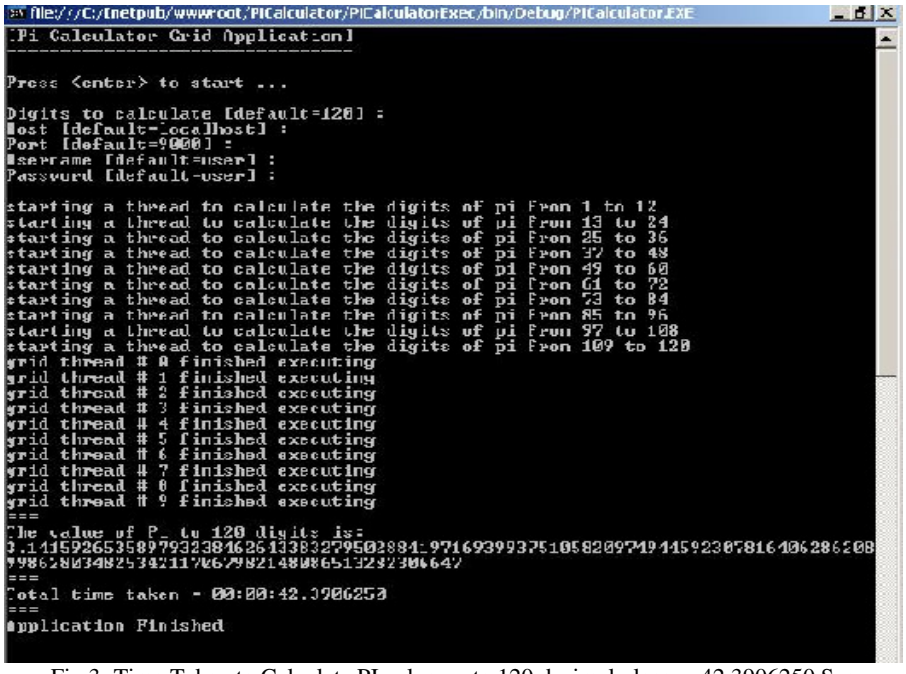



Fig 3: Time Taken to Calculate PI value up to 120 decimal places= 42.3906250 Sec Using 1 Alchemi Executor

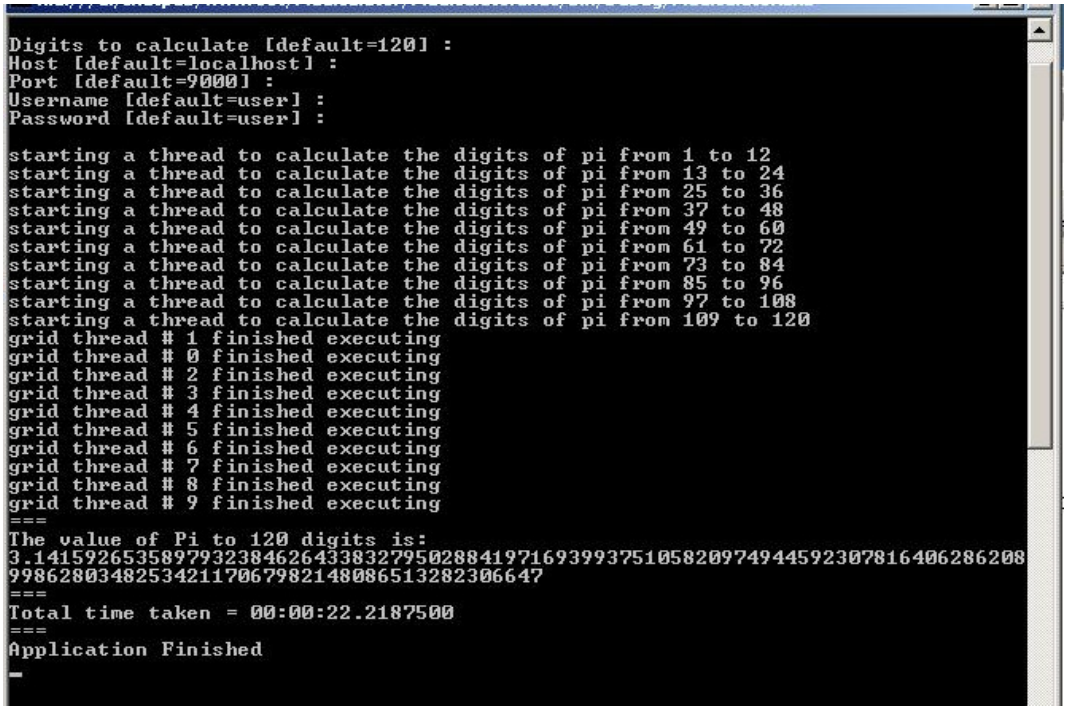



Fig 4: Time Taken to Calculate PI value up to 120 decimal places= 22.2187500 Sec Using 2 Executors of Alchemi

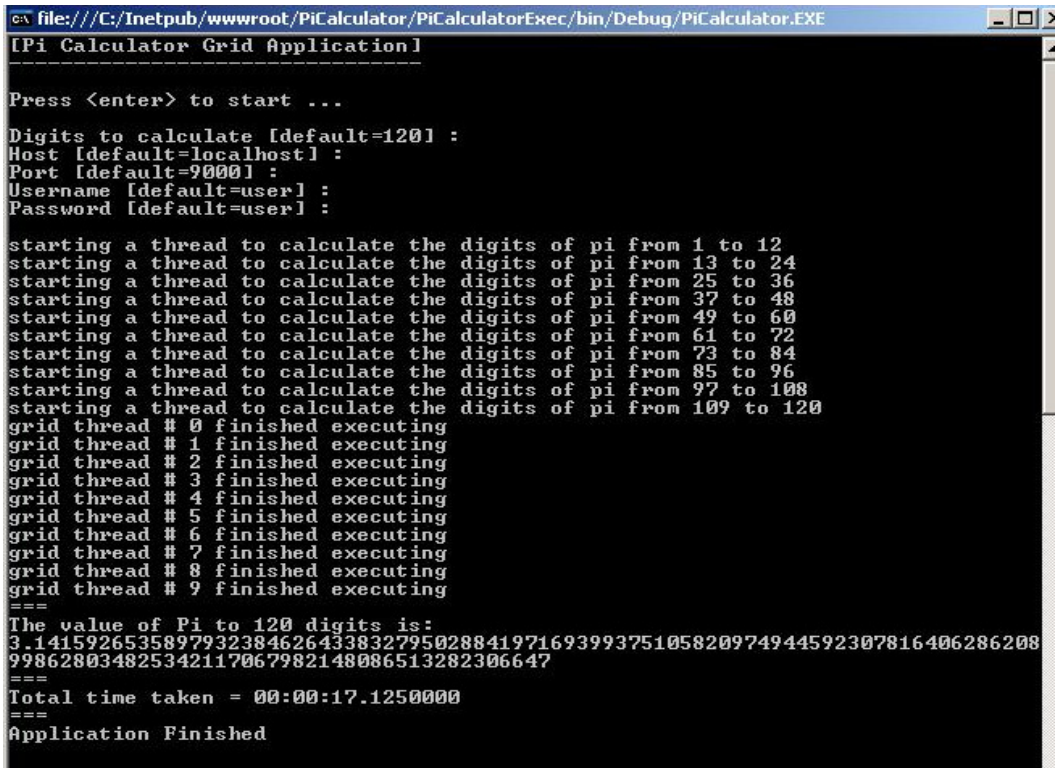



Fig 5: Time Taken to Calculate PI value up to 120 decimal places= 17.1250000 Sec Using Three Executor of Alchemi

# 8. Nimble@ITCEcnoGrid Toolkit Results

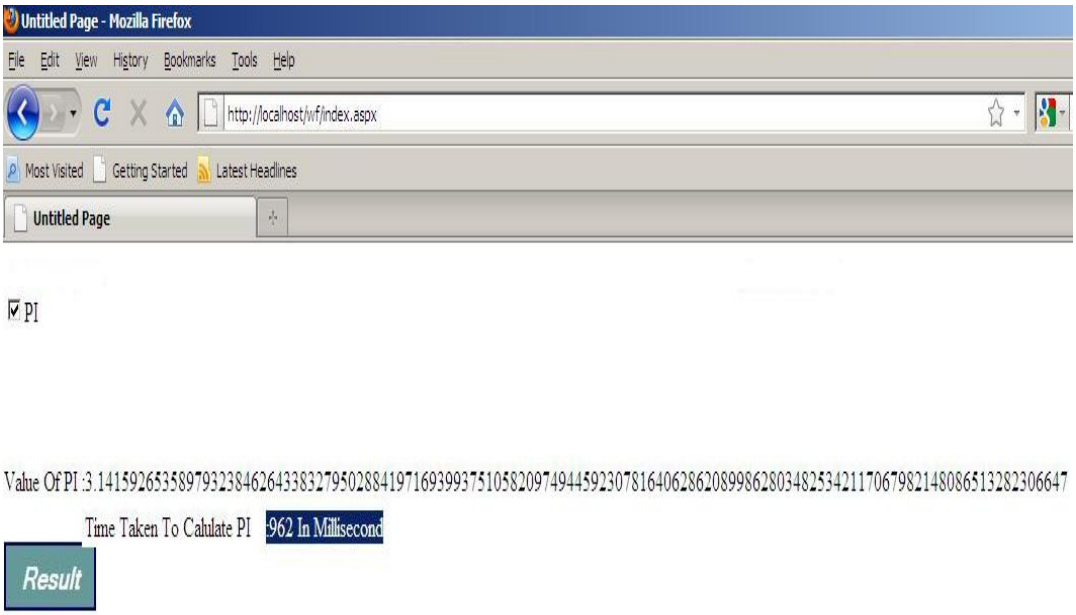



Fig6: Time Taken to Calculate PI value up to 120 decimal places= 962ms
Using 1 Nimble@ITCEcnoGrid Executor

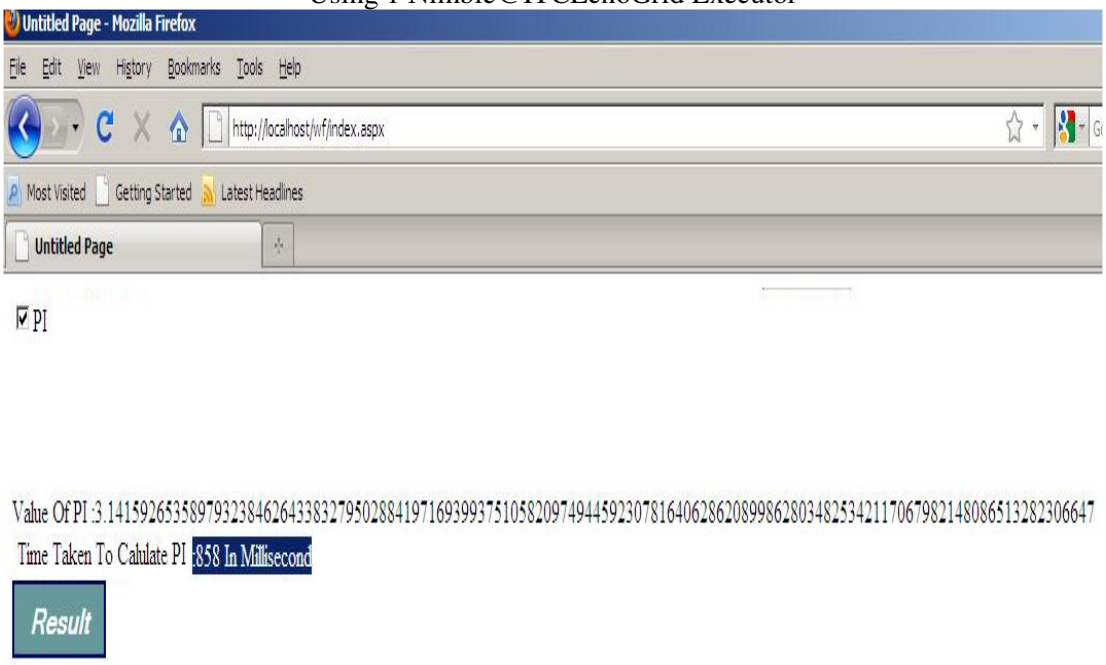



Fig 7: Time Taken to Calculate PI value up to 120 decimal places= 858ms
Using 2 Nimble@ITCEcnoGrid Executors

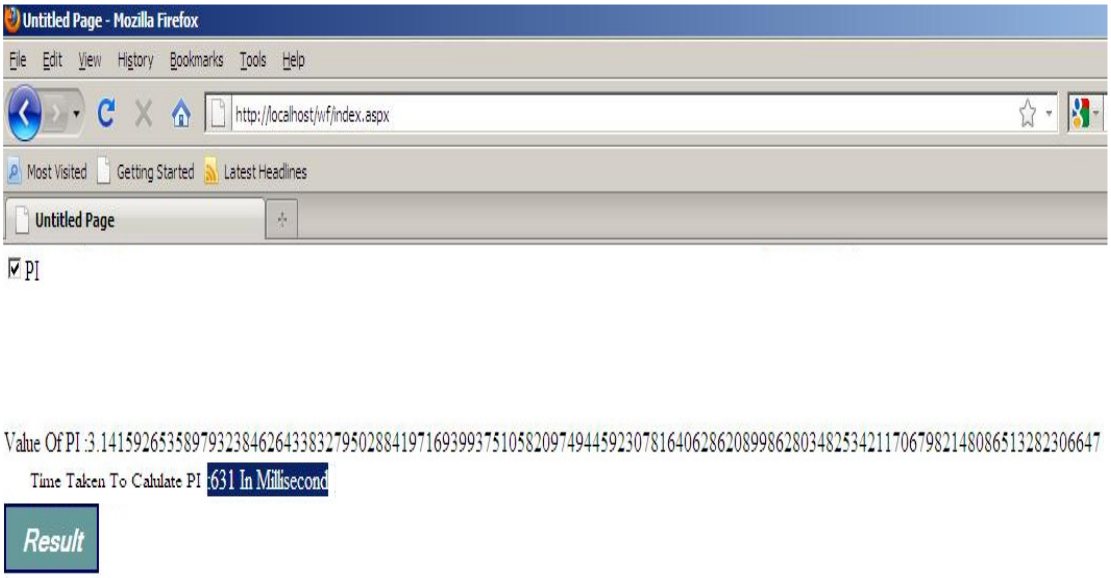


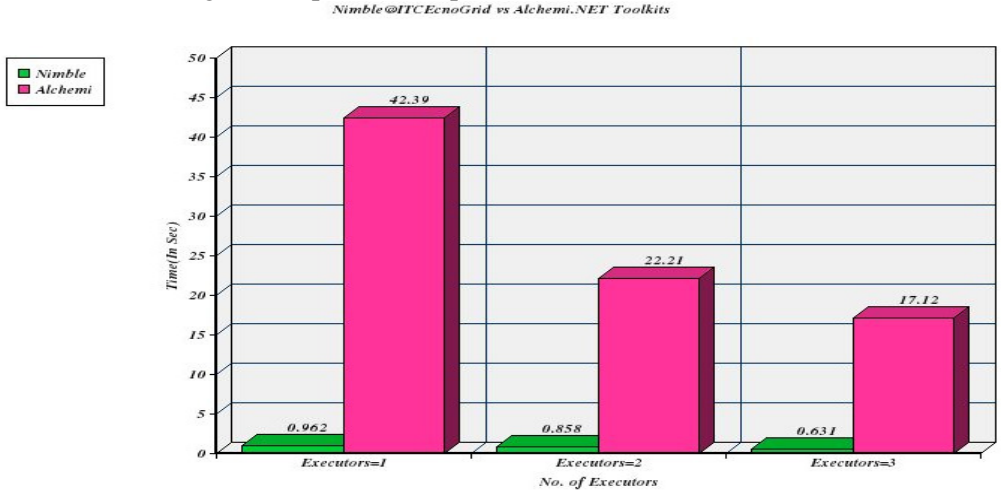

Fig 8: Time Taken to Calculate PI value up to 120 decimal places= 631ms
Using 3 Nimble@ITCEcnoGrid Executors

## 9. Tables & Graph

| No of Executor | Alchemi.NET Toolkit (in sec) | Nimble@ITCEcnoGrid Toolkit (in sec) |
|---|---|---|
| 1 | 42.39 | 0.962 |
| 2 | 22.21 | 0.858 |
| 3 | 17.12 | 0.631 |

Table 3

Fig 9: Comparison Graph of Alchemi.NET and Nimble@ITCEcnoGrid




As shown in fig 9 , we see that the time taken by Nimble@ITCEcnoGrid Toolkit to solve the pi problem is very less( in ms) as compared to Alchemi Toolkit (in Seconds).

# 10. CONCLUSION

The main goal of the research paper is to compare the two grid toolkits namely Alchemi & Nimble@ITCEcnoGrid. It is interesting to find that the results obtained by the Nimble@ITCEcnoGrid toolkit is very encouraging because it takes only a few milliseconds to complete the task as compared to more than three dozen of seconds by Alchemi toolkit. The present research toolkit namely Nimble@ITCEcnoGrid has a feature of Inter thread communication which is missing in Alchemi toolkit. The future scope of this toolkit is that it can be applied to the complex problems like Weather Forecasting, Earth Quake Prediction etc which involves teraflops of pet flops of operations per second.

# REFERENCES

[1] Ian Foster(2002), *What is the Grid? A three point checklist*, Argonne National Laboratory & University of Chicago foster@mcs.anl.gov. July 20, 2002.
www.gridtoday.com/02/0722/100136.html

[2] Rajkumar Buyya (Editor), Kris Bubendorfer (Editor)(2009), *Market-Oriented Grid and Utility Computing* ISBN: 978-0-470-28768-2 Hardcover 643 pages October 2009

[3] White Paper (2009), *Achieving Business Agility with Application Grid* , ,Posted 09 Sept 2009, PDF,11 Pages,Language: English

[4] Ian Foster and Carl Kesselman(1999), *"The Grid: blueprint for a future computing infrastructure*", Mogan Kaufmann Publishers, USA, 1999.

[5] Ian Foster, Carl Kesselman, and S. Tuecke(2001), *"The anatomy of the Grid: enabling scalable virtual organizations*", Journal of Supercomputer Applications, 15(3) pg. 200-222, 2001.

[6] Kalantari **,** Mohammad Kazem Akbari(July 2009), *A parallel solution for scheduling of real time applications on grid environments*, Computer Engineering Department, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran published in Future Generation Computer Systems Volume 25, Issue 7, July 2009, Pages 704-716

[7] Akshay Luther(2005*), "Alchemi: A .NET-Based Enterprise Grid Computing System"*, Proc. 6th Int'l Conf. on Internet Computing (ICOMP'05), Las Vegas, USA, 2005

[8] Akshay Luther, Rajkumar Buyya, Rajiv Ranjan, and Srikumar Venugopal(2005), "*Alchemi: A .NET-Based Enterprise Grid Computing System*", Proceedings of the 6th International Conference on Internet Computing (ICOMP'05), June 27-30, 2005, Las Vegas, USA.

[9] Andrew Chien, Brad calder, Stephen Elbert and Karan Bhatia(2003),"*Entropia: Architecture and Performance of an Enterprise Desktop Grid System*", Journal of parallel and Distributed Coputing, Volume 63, Issue 5, Academics Press, USA, May 2003.

[10] Rajiv Ranjan, Aaron Harwood, Rajkumar Buyya(2006)-"*SLA-Based Coordinated Superscheduling Scheme and Performance for Computational Grids*" -In Proceedings of the 8th IEEE International Conference on Cluster Computing (Cluster 2006), IEEE Computer Society Press, September 27 – 30, 2006, Barcelona, Spain.

[11] James Broberg · Srikumar Venugopal · Rajkumar Buyya(December 2007)**-"***Market-oriented Grids and Utility Computing:The State-of-the-art and Future Directions***"** Received: 18 September 2007 /

Accepted: 12 December 2007 / Published online: 28 December 2007 

[12] http://www.cloudbus.org/~alchemi/documentation.html

[13] Akshay Luther(2005), "*Peer-to-peer grid computing and a .NET-based Alchemi framework*", , Chap 21,403-429, Wiley Press, New Jersey, USA, June 2005.

[14] Saurabh Kumar Garg, Rajkumar Buyya, Howard Jay Siegel(2010), *Time and cost trade-off management for scheduling parallel applications on Utility Grids Future Generation Computer Systems*, Volume 26, Issue 8, October 2010, Pages 1344-1355.

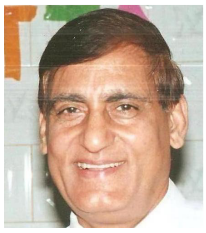
[15] http://mathworld.wolfram.com/MachinsFormula.html

[16] http://mathworld.wolfram.com/MaclaurinSeries.html

**Authors**

**Er. Vijay Dhir**[1] is working as a Assistant Professor in Information Technology at Sant Baba Bhag Singh Institute of Engineering & Technology, Padhiana, Distt: Jalandhar, Punjab, India.. He has a work experience of 11 years. He has published 8 International and 7 National Papers at various Journals/Conferences. He is pursuing his Ph.D in the field of "Grid Computing" since 2007.

**Dr. Rattan K Datta**[2] is working as a C.E.O & Director, MERIT, New Delhi,India. He retired as the Adviser ,Department of Science & Technology, Government of India. He did his Ph.D from IIT (Delhi) on "Monsoon Dynamics & Modelling".He has guided number of students in Ph.D, M.Tech & M.Phil. He has published about 200 research papers in various International & National Journals/Conferences



**Dr. Maitreyee Dutta**[3] is working as a Associate Professor at NITTTR,Chandigarh,India. She has a working experience of 5 years in Research and 11 years in Teaching. She has guided 40 M.Tech students in their thesis. She has published 12 papers in various International journals & conferences.