ISSN 1512-1232

GRID COMPUTING WITH ALCHEMI: AN APPRAISAL AND RESEARCH CHALLENGES

[1]Nureni Ayofe Azeez, [2]Isabella M. Venter, [3]Tiko Iyamu

[12]Grid Computing & Distributed Systems (GRIDS) Laboratory, (Security Research Group), Department of Computer Science, University of the Western Cape, Private Bag X1, Bellville, 7535, South Africa 3008814, imventer}@uwc.ac.za

[3]Tshwane University of Technology, Department of Informatics, Faculty of Information and Communication, Soshanguve South, South Africa, iyamut1@tut.ac.za

Abstract

It has become a known fact that the rate at which Grid Computing is rising nowadays in term of applications and demand both in research institutes and industries cannot be over emphasized. The ubiquitous nature of this new innovation and technology couple with open source nature of most of the softwares have led to the research in various Grid Middlewares ; without which grid resources cannot be shared. These middlewares allow submission of requests to execute a computation (called a Job) to the Grid, such that it can be run anywhere on the network. Therefore, grid Middlewares serve as an intermediary layer that allow a reliable and homogeneous access to resources managed locally with different syntax and access techniques. Within the context of availability of various Middlewares for Grid implementation with different features, this paper therefore focuses on various features that are peculiar to Alchemi by taking into consideration its Architecture, the Operating systems, software demand and limitation that are inherent from its usage.

Keywords : *Alchemi, architecture, framework, application, Middlewares*

1.0 Introduction

Alchemi is an open source grid middleware developed at the University of Melbourne, Australia which allows users to effortlessly aggregate and distribute computing power of networked computers into a virtual supercomputer (desktop grid) and to develop applications to run on the grid. It has been designed with the primary goal of being easy to use without sacrificing power and flexibility. Some of the required features that make Alchemi works appropriately include:

- The runtime machinery (Windows executables) to construct computational grids.
- A .NET API and tools to develop .NET grid applications and grid-enable legacy applications.

Since is it known that Alchemi is written for the .NET CLR, therefore all machines running Alchemi middleware must have the .NET framework installed on them without which any application cannot be deployed. One important feature of Alchemi is that it runs with application on windows, however, it is established that it can also run on UNIX operating system. However, Alchemi does offer support for execution of cross-platform applications via web services [5].

Grid Middleware (core services) could be regarded as the backbone of any Computational Grid system without which distinction could not be made between a parallel, distributed or cluster computing. The idea of distributed computing has been around for years. Some of the factors that distinguish Grid computing from typical distributed computing or cluster computing are: that grids tend to be more loosely coupled, heterogeneous, and geographically dispersed. Also, while a computing grid may be dedicated to a specialized application, it is often constructed with the aid of general purpose grid software libraries and middleware. Some of the popular middleware that are available are: Alchemi
• Cosm P2P Toolkit ,• Globus • Gridbus • Grid Datafarm • GridSim: Toolkit for Grid Resource Modeling and Scheduling Simultation • Jxta Peer to Peer Network • Legion: A Worldwide Virtual

Computer • NorduGrid middleware • PUNCH • Simgrid •Storage Resource Broker (SRB) • ProActive •Unicore •Vishwa [1]. All these middlewares have various attributes that make one suitable for a particular application than the other.

Alchemi therefore is considered in this paper based on its cheap and some other desirable features that are lacking among other middlewares for implementing applications most importantly to expose some unique features that are desirable over other grid middlewares.

1.1 Functional Framework of Alchemi

Four distributed components have been identified in the development and implementation of various applications on a computational grid with Alchemy. The components are: Manager, Executor, User and Cross-Platform Manager [2].

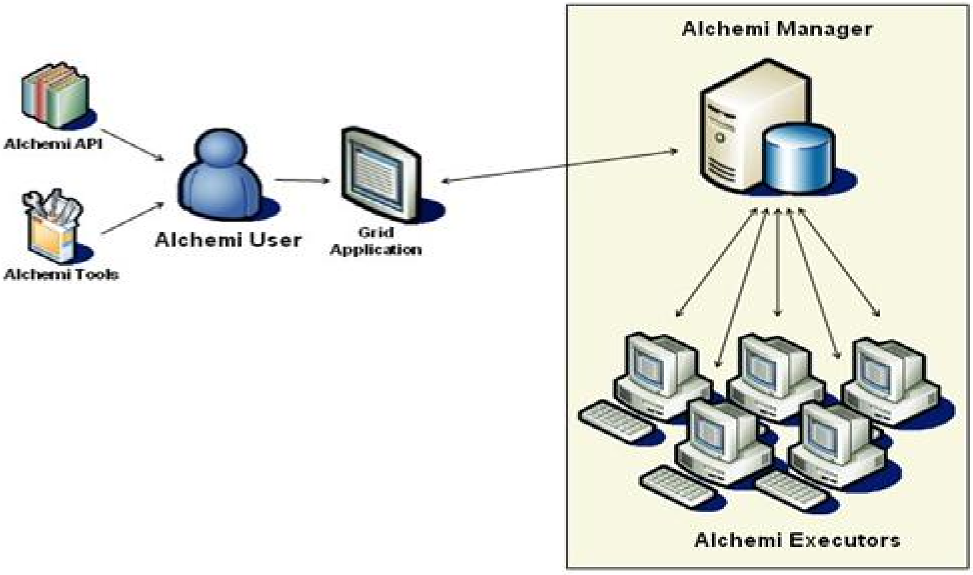

Figure 1. Alchemi Grid Architecture [1]

Supposing 1…n number of machines are to be enabled on a grid, each of these machines must be installed with Executors and consequently link them with a central Manager component. The installation becomes a easier one base on the fact that the window installer setup comes with Alchemi distribution [2].

1.1.1 An Executor can either be dedicated or non-dedicated. A non-dedicated Executor works along NAT servers and firewalls because there is a one-way interaction and communication between the Manager and the Executor. Dedicated Executors are very useful and more applicable in an intranet environment while internet environment is mostly suitable for non-dedicated Executors.

Grid application using Alchemi can be monitored and executed using the .NET API and with other tools which are part of Alchemi SDK. Alchemi provides grid thread programming model which simplifies the construction of grid applications and a grid job model for non- .Net applications.

1.1.2 Cross-Platform Manager

An optional component (not shown in figure 1) is the **Cross Platform Manager** web service which offers interoperability with custom non-.NET grid middleware. This is purely a web based service interface that reveals a section of the functionality of the Manager so as to enable Alchemi manage the job execution on the grid system. Jobs submitted to the Cross-Platform Manager are converted into a form that is understandable and acceptable to the Manager.

1.1.3 User

Any application to be executed on the grid Alchemi is usually done on the user node. The API conceptualizes the grid implementation from the user and performs different services such as submitting application and its constituent threads for job execution. Consequently, it notifies the user of completed job and also informs him of failed threads along with reasons for the said errors.

1.1.4 The Manager

Any grid application(s) and their constituent threads are being managed by the Manager. Executors monitor their status after they have already registered with the manager. As threads are being received from the users they are also placed in a pool and scheduled for execution in various Executors. A priority is placed on the thread when any job is created and submitted. Threads are scheduled on a Priority and First Come First Served (FCFS) basis, in that order. The completed threads are usually returned back to the Manager by the Executors and these jobs will be received by their respective users [4].

2.0 Operating System Requirement of Alchemi

Some of the available grid Middlewares are implemented on the LINUX and UNIX operating system. Alchemi was developed to answer the need and demand within enterprises for a desktop grid solution that utilises the unused computational capacity represented by the vast number of PCs and workstation running Windows within an establishment. Alchemi is implemented on top of the Microsoft .NET Framework and provides the runtime machinery for constructing and managing desktop grids.

3.0 Alchemi application development environment

With the Alchemi Application Programming Interface (API), one can extend the power of the popular Alchemi product line into many other business applications. One can use the Alchemi API to affordably add unstructured data management functionality to business applications, including managing large volumes of unstructured data such as images, computer report files (COLD), MS Office files, Adobe Acrobat PDF files, AutoCAD drawings and other common file formats. In addition, with Alchemy, users of your application can index, archive, query, retrieve, view, print and route this important unstructured data via e-mail. Alchemi gives users fast, easy access to business data for better business decisions and improved customer service. Alchemy can also help users solve IT problems such as compliance and security.

Your developers can easily leverage the software's COM-based objects and ActiveX interfaces to provide powerful information access to your company or clients. Alchemy has a component-based API with dual interfaces that are industry standard: COM and ActiveX. They combine with the existing ODBC interface to provide developers with endless flexibility.

- Integrate Alchemy into applications that need to manage unstructured data
- Control and customize the Alchemy user interface
- Automate common Alchemy tasks

4.0 Suitable Programming Environments for Alchemi

For a grid to be set up using Alchemi as a grid middleware there are things needed to put in place for easy deployment of an application. They are [6]:

- Alchemy version 6 Service Pack 2 or higher installed for both the development environment, and for deployment and distribution of the custom application
- A COM compliant development environment, which should provide an ActiveX container in order to site the Alchemy ActiveX controls. Examples of this type of environment include Microsoft Visual Basic, Microsoft Visual C++, Sybase PowerBuilder, and Inprise C Builder
- Users of the Alchemy API should be proficient in one of these languages and should have a good working knowledge of the Alchemy product

5.0 System Requirements

One can use the Alchemy API by purchasing the Alchemy Application Developer Environment (ADE), a comprehensive product offering that includes all the software, documentation, and education and support services necessary to be a successful Alchemy developer [6]. With the Alchemy API, one can create applications that run on a Microsoft Windows computer.

- Microsoft Windows NT Workstation 4.0
- Microsoft Windows NT Server 4.0
- Microsoft Windows 98
- Microsoft Windows 95

6.0 Key Alchemy Web Benefits

- Instantly provide universal access to all data with the added ability to create separate home pages for separate clients or departments.
- Easily find, view, print and email the business data you need using powerful search tools including full-text, metadata fields and annotations--all in one search operation.
- Clearly view images without powerful image viewing tools including zoom in or out, panning view, multi-page view, and thumbnail view.
- Quickly deploy business documents and data via the web across your global enterprise with our flexible, easy installation.
- Simultaneously browse multiple databases to make your searches even more effective

7.0 Security Issue in Alchemi

Security is undoubtedly the most crucial issue in an environment that is insecure like the internet. Two main aspect of security that Alchemi has addressed are as follow:

(a) it allows users to perform authorized operations whether they are system related or resource related operations and

(b) it allows authorized or non-authorized users to contribute resources.

The problem of allowing users to only perform activities they are authorized to do is addressed using the role-based authorization model. All security-sensitive activities on the Manager are protected in this manner.

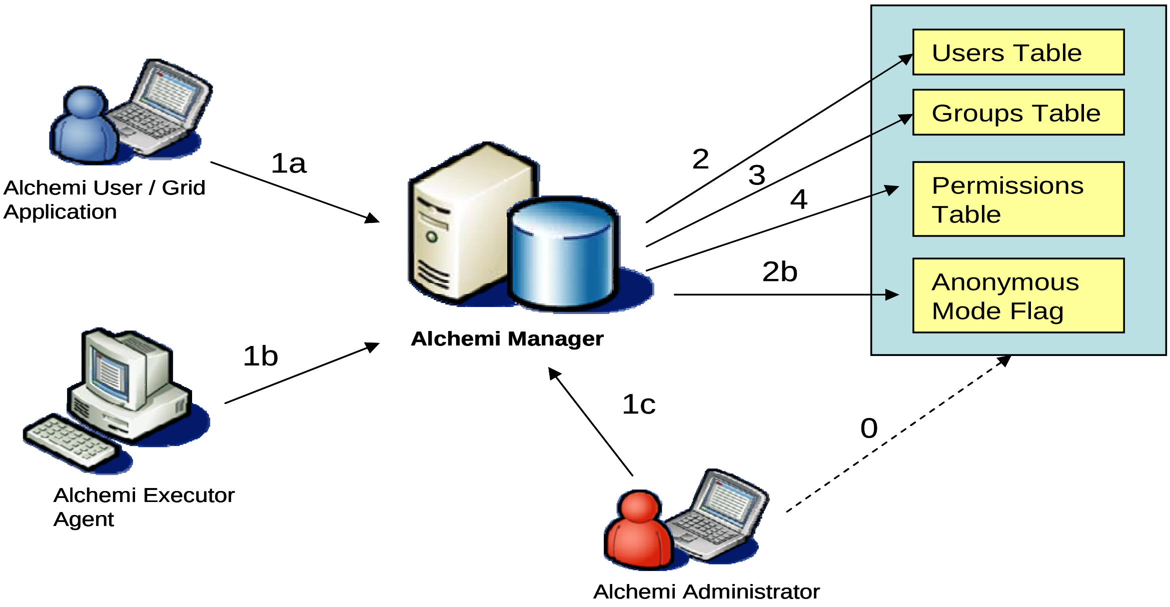

Figure 2: showing Role Based security for Alchemi [7]

- Authentication: Each user is given a unique password for accessing the information on the Alchemi grid enable system. Simple username / password is used to authenticate the validity of a genuine user on the grid [1].
- ***Authorization:*** Role-based security permission is used to confirm whether a user has a specified identity or is a member of a specified role. Principal Permission is the only role-based security permission supplied by the .NET Framework class library [7].
- Auditing: since each user has an account assigned to him in the database, all jobs being executed are recorded against this user to ensure authentication.

8.0 Conclusion

From the foregoing, Alchemy has been identified as an open source grid middleware that offers limitless support for grid applications. Consequently, Alchemi is an easy-to-use .Net Grid-computing framework that aims to lower the barrier of entry into the world of distributed / Grid computing, by making it easier to setup, manage and maintain Grids on a local or wide-area network. However, before an application can be deployed on a grid that requires Alchemi; there is need for the user to modify the open source to suit the purpose of the application. Doing this requires special skill in programming in order to ensure efficient utilization of the grid system.

9.0 References

1. http://www.gridcomputing.com/ by Rajkumar Buyya accessed on: 23rd June, 2010 Alchemi: A .NET-based Enterprise Grid System and Framework by Authors: Akshay Luther, Krishna Nadiminti, Rajkumar Buyya, July 2005. Url: http://www.gridbus.org/~alchemi/files/ 1.0.beta/docs/AlchemiManualv.1.0.htm
2. http://www.flashmagazine.com/news/detail/alchemy_toolkit_-_preview_version_out_on_ adobe_labs/ accessed on: 23rd June, 2010
3. Peer-to-Peer Grid Computing and a .NET-based Alchemi Framework by Akshay Luther, Rajkumar Buyya, Rajiv Ranjan, and Srikumar Venugopal
4. http://sharptoolbox.com/categories/grid-computing.: accessed on 24th June,2010
5. Data Capture Solutions LTD, 345 Edinburgh Avenue, Slough, Berkshire, SL1 4TU, UK. url: http://www.data-management-software.co.uk/alchemy/alchemy-application-development.htm
6. Alchemi: A .NET-based Enterprise Grid Framework: Dept. of Computer Science and Software Engineering, The University of Melbourne, Australia url: www.gridbus.orwww.alchemi.netg /
7. http://msdn.microsoft.com/en-us/library/7sxk9k2h.aspx

Article received: 2011-03-01