

# User & Developer Guide



## Mobile Desktop Grid (MDG)

Make mobile device to interact with cluster- and grid-based resources

Visit us at <https://mdg.dev.java.net>

version 1.1.2.2

Prepared by Tan Choo Jun & Ang Wai Heng

Revision on 2010-02-05

# Table of Contents

1.0 Introduction.....	3
2.0 Available MDG modules.....	5
3.0 Step-to-step Guide .....	6
3.1 The 5 Steps to register a new account .....	6
3.2 The steps of adding new cluster .....	9
3.3 The 7 steps to sign in and get MDG solution resources .....	14
3.4 The 3 steps to submit PI job: Using SGE Local Scheduler .....	18
3.5 The 3 steps to submit PI job: Using Fork manager .....	20
3.6 The 5 steps to submit NCBI-Blast job: Using SGE Local Scheduler .....	22
3.7 The 4 steps to submit NCBI-Blast job: Using Fork manager .....	25
3.8 The steps to post and receive notification: At MDG Server .....	28
3.9 The step to receive notification: At MDG Client .....	31
3.10 The steps to add new RSS feed: Sun Blogger RSS .....	32
3.11 The steps to add new RSS feed: Oracle RSS .....	35
4.0 The database logical design.....	38
4.1 Entity Relationship Diagram for MDG Server .....	38
4.2 Entity Relationship Diagram for MDG Client .....	38
5.0 Frequent Answer and Question .....	39
Q: MDG using any open source technologies? .....	39
Q: What is the requirements to join our ROCKS cluster server with MDG solution? .....	40
Q: What is job? .....	40
Q: What is BLAST .....	40
6.0 MDG Projection and Future .....	41
7.0 MDG Appendix.....	42
7.1 Demo clips of work .....	42
7.2 Download .....	42
7.3 MDG Deployment in Mobile Device.....	43
7.3 MDG Achievement.....	44
Open Jive 2009 Malaysia Finals	
Venue: Bukit Jalil Golf & Country Club, Kuala Lumpur Malaysia	
Date: Thursday 2 July 2009 .....	44
OpenJive 2009 Singapore Grand Finals	
Venue: Stamford Ballroom, Raffles City Convention Centre, Singapore	
Date: Tuesday 28 July 2009 .....	47
7.4 Who are the MDG team.....	51

## 1.0 Introduction



### Note:-

The new MDG web site is at URL <https://mdg.dev.java.net>

**Mobile Desktop Grid** (MDG) uses open source technologies and mainly on Java platform that can run on any operating systems. It requires only small hardware requirement as small as any hand held devices that support JAVA (J2SE). Resource contributors around the world need to register their cluster machines joining with MDG Server. The resource users are able to utilize those identified pool resources after registration. Resource contributors are from individual, organization, education institutional and laboratory lab who would like to share out their machine clusters for public use. Whereas resource users are public individuals, university students, organization personnel, lab researchers and officers who require to perform exhaustive computational jobs.

The capabilities of single organization or individual investing in machines assets and computational powers are limited in growing and always constraint by financial budget. The boundary of single organization or individual that owns machines assets and computational powers can be opened up with the MDG solution. Problem of insufficient resources when addressing global issues have the answer from MDG solution. On top of enjoying unlimited computational resource, MDG solution enables resource user to obtain either homogeneous or heterogeneous cluster resources, different CPU processing speed, and various combination of cluster CPU-memory-network environments. Resource users can perform more unit-testing based on these variety environments to obtain more significant testing results in research. As the result, MDG solution is another researcher daily tool option to use in achieving research project mission and vision, such as global research collaboration modeling in against the pandemic threat of Avian flu and other emerging infectious diseases such as virus SARS and A-H1N1.

MDG solution has centralized users management for authentication, external cluster resource recruitment and resource allocation. It also has a universal knowledge sharing platform to enable researchers exploring latest research progress around the world as well as the independent notification platform that enables research execution job running in minimum interruption due to insufficient resources.



## 2.0 Available MDG modules

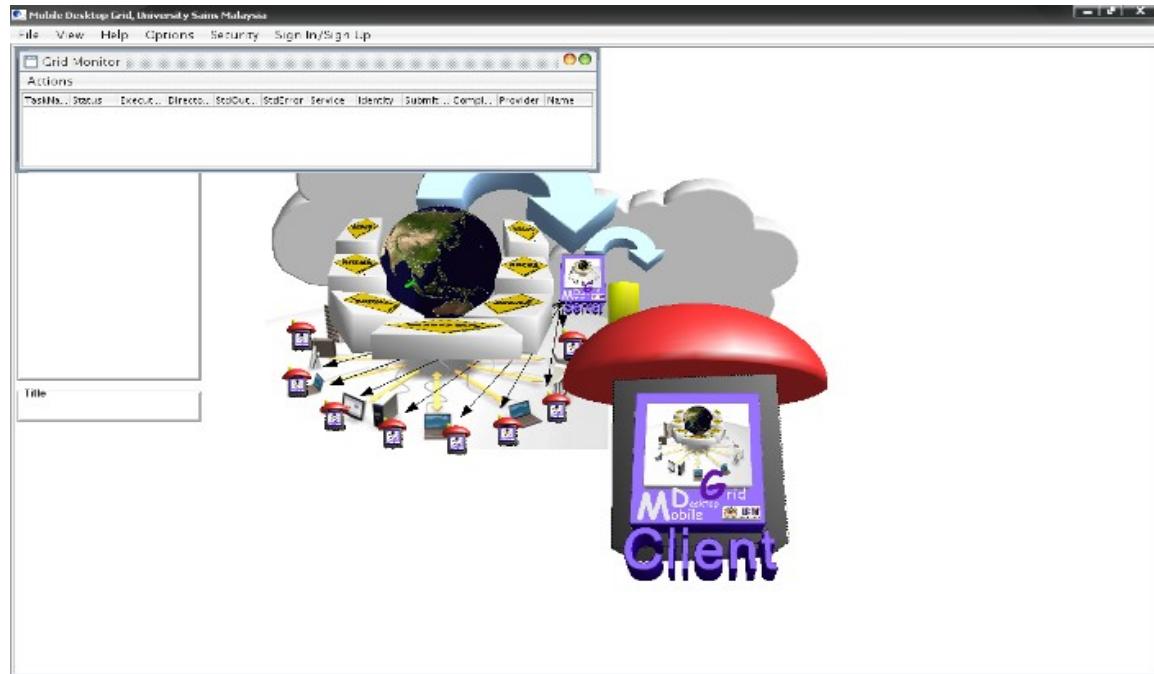


### Note:

As in MDG version 1.1.2.2

# 3.0 Step-to-step Guide

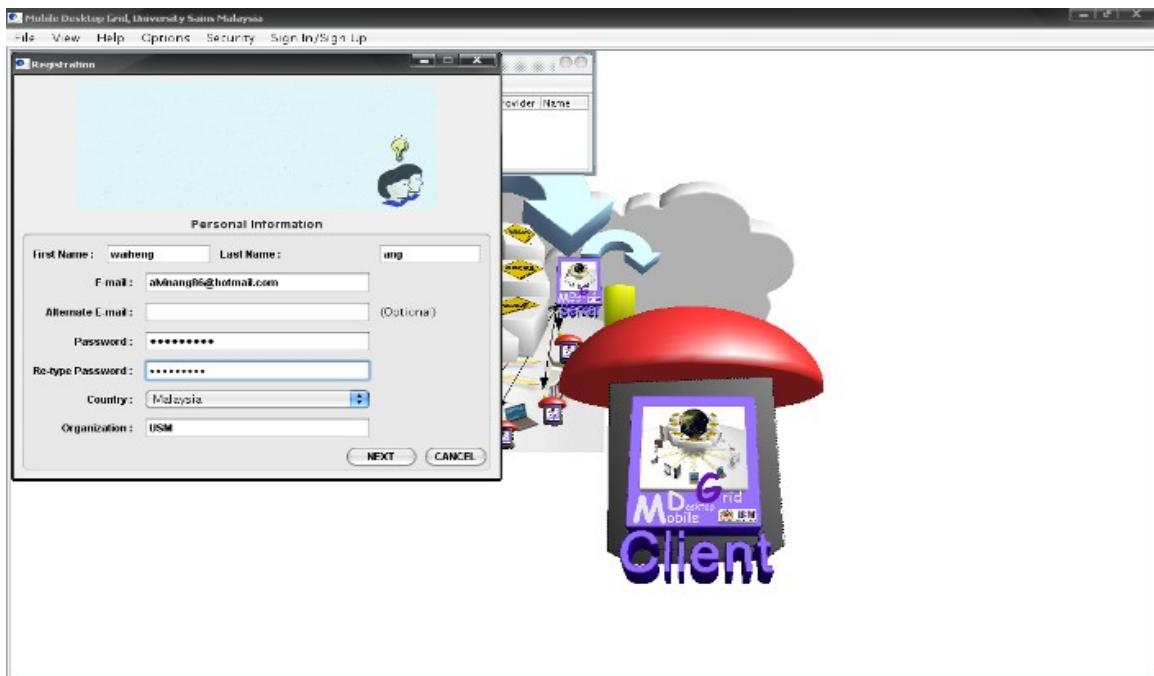
## 3.1 The 5 Steps to register a new account



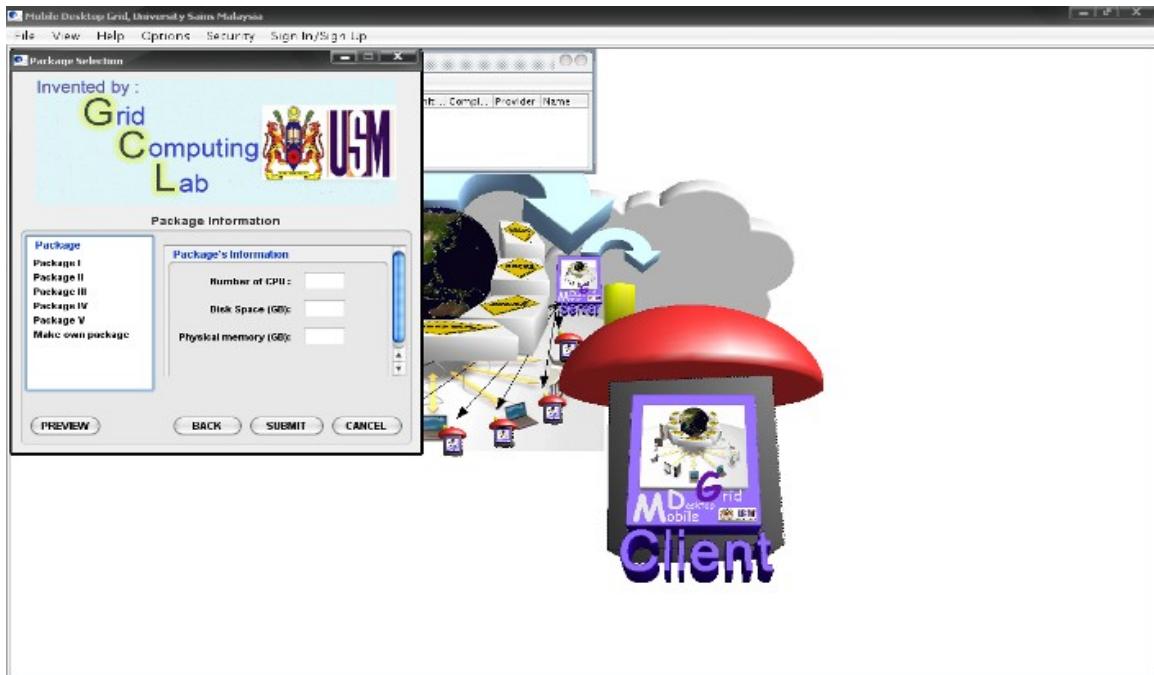
Above shows the MDG Client.



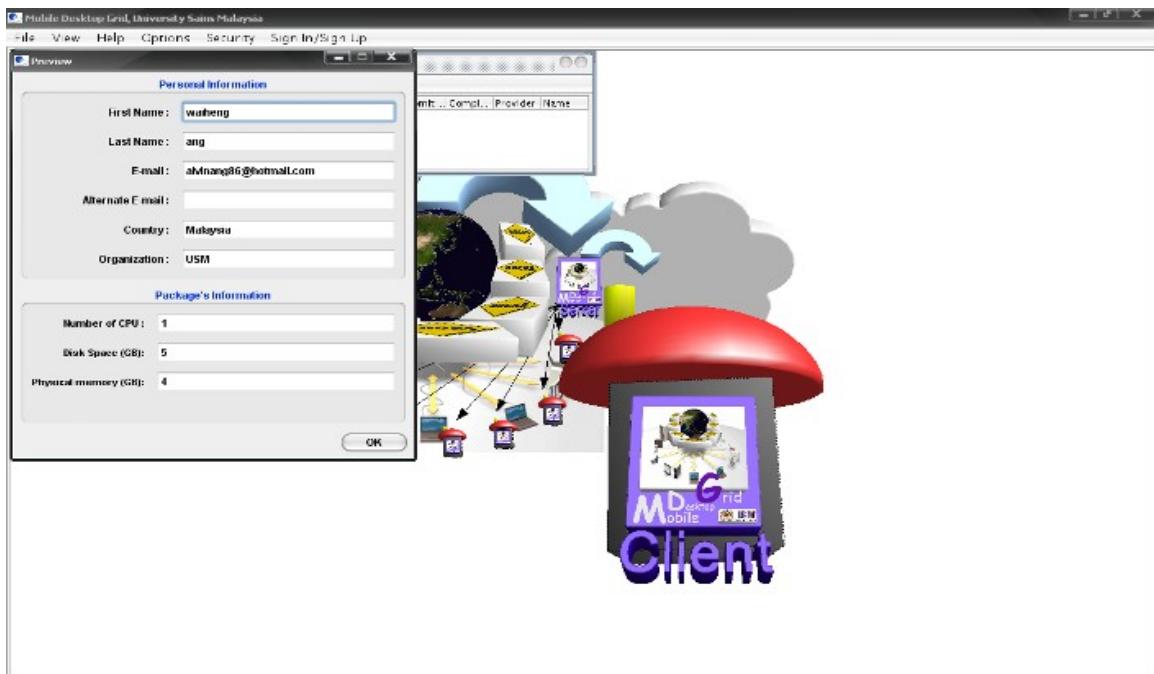
User requires to register himself in order to become the user of MDG solution. User can uses all features in MDG Client upon their registration successfully such as NCBI-Blast and MPICH job submission.



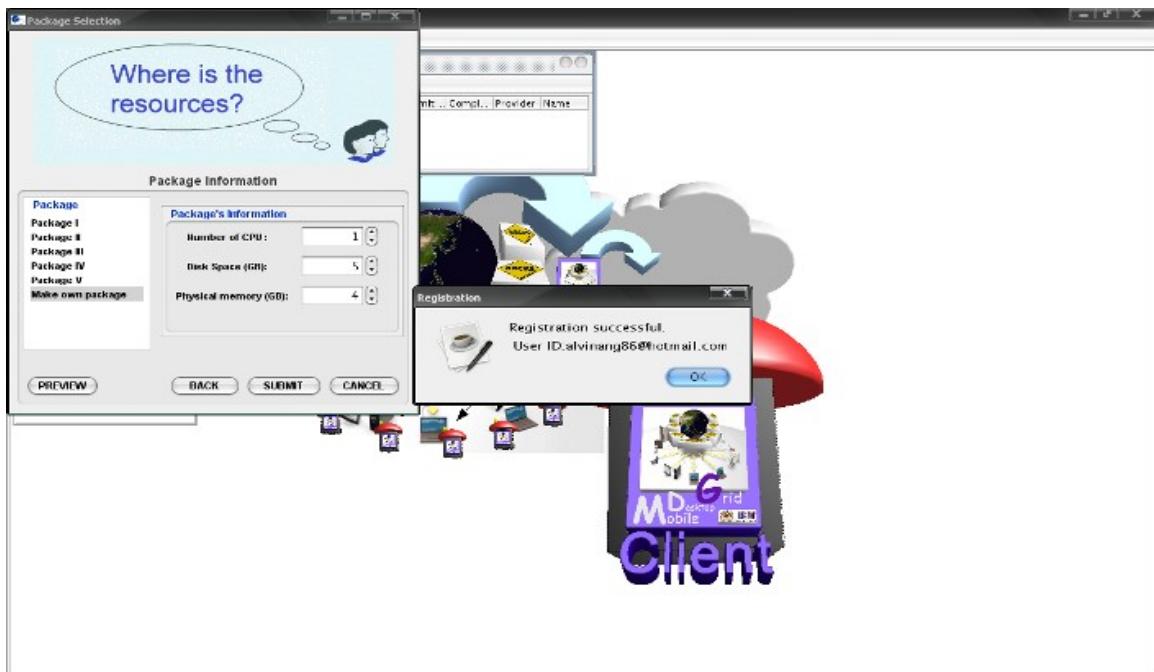
The prompted window require user to enter their personal information, the country of origin and organization. The user email address is user login id in MDG solution using MDG Client.



Above shows the Package Selection interface. User has option to choose the pre-defined resources package or customize the package according to their requirements.

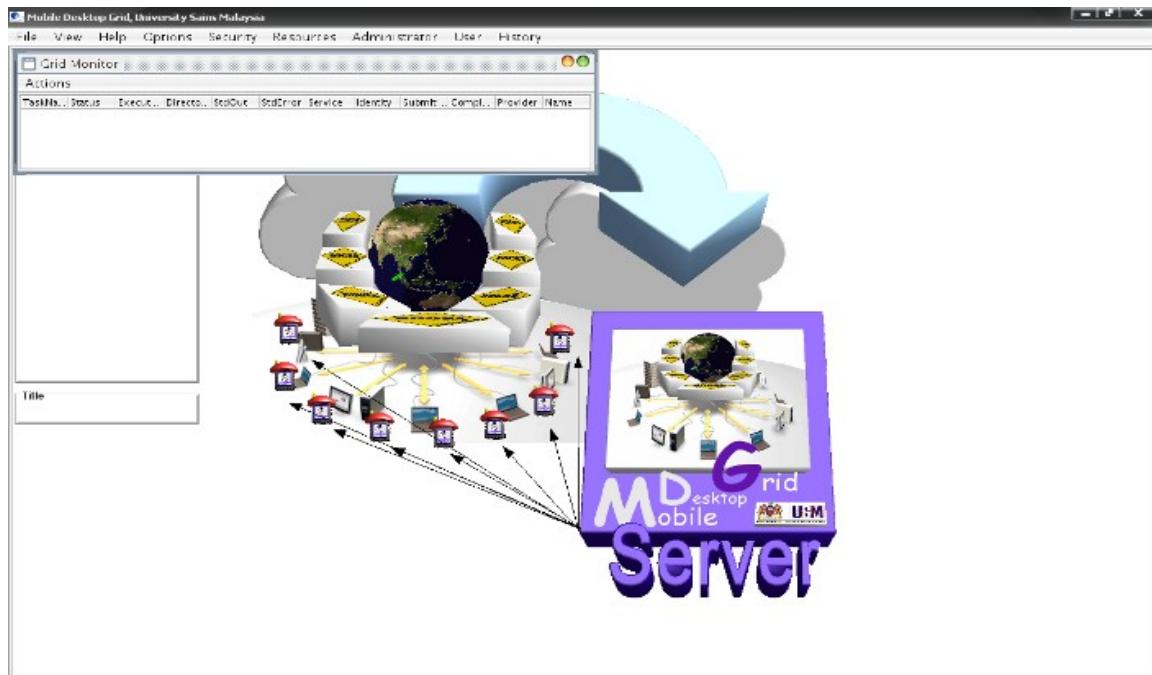


Above shows the detail personal information interface. To preview the user entered personal information, user just need to click on the Preview button at left hand side in Package Selection interface (in previous screen capture).

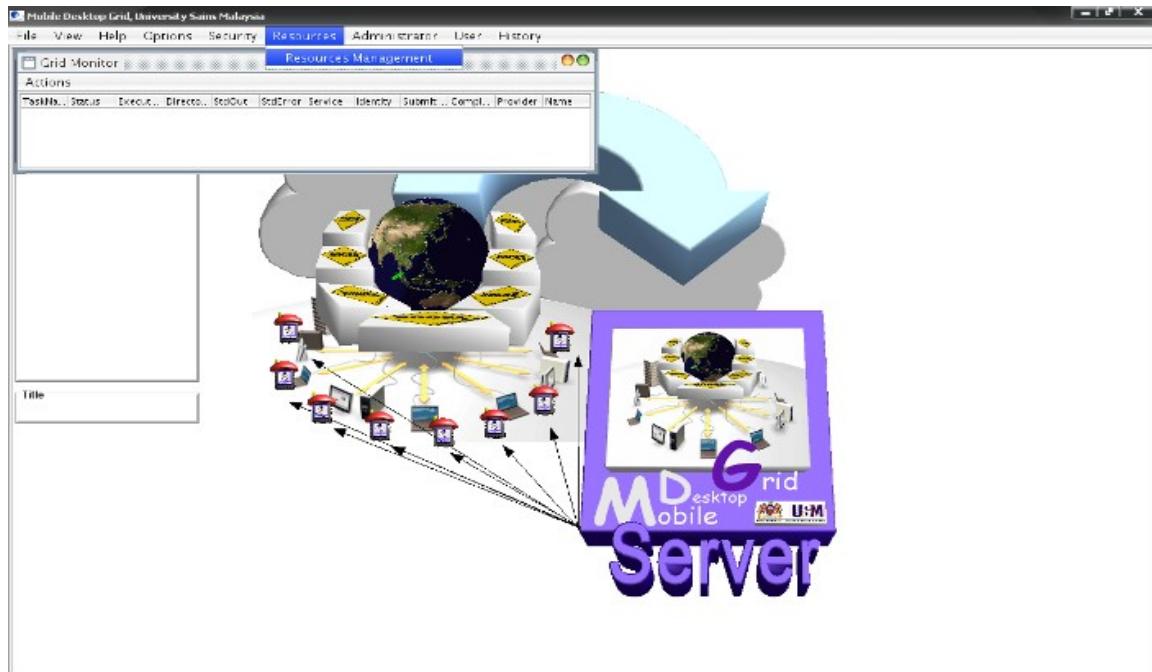


After filling all required information, and submit it by click on the Submit button. Registration is processing in MDG server, and a reply message from server will appear. Above show the successful registration reply. The Back button is for user to go to previous interface (as previous screen capture) to change the entered information to MDG server.

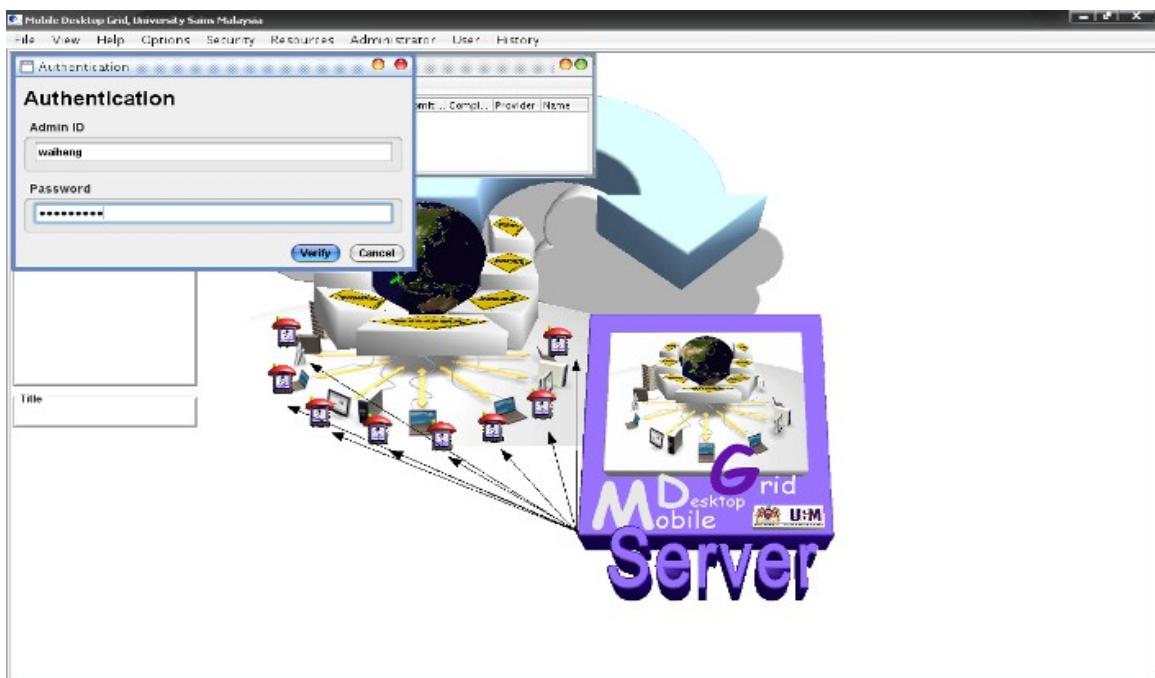
### 3.2 The steps of adding new cluster



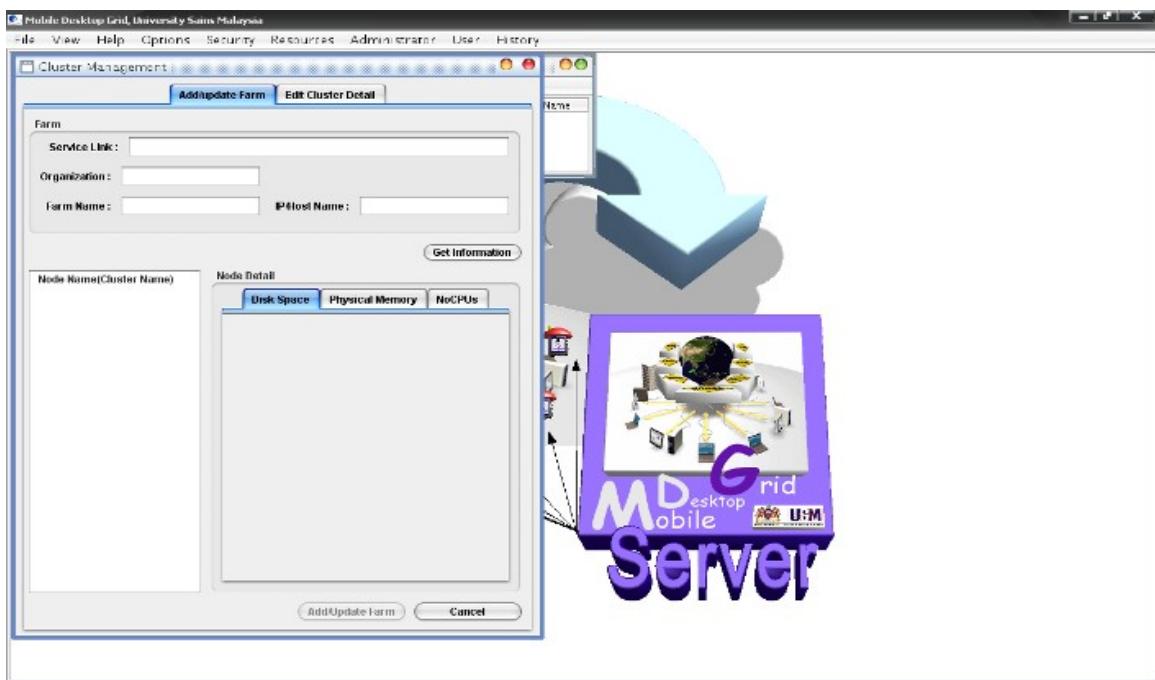
Above is the MDG Server interface.



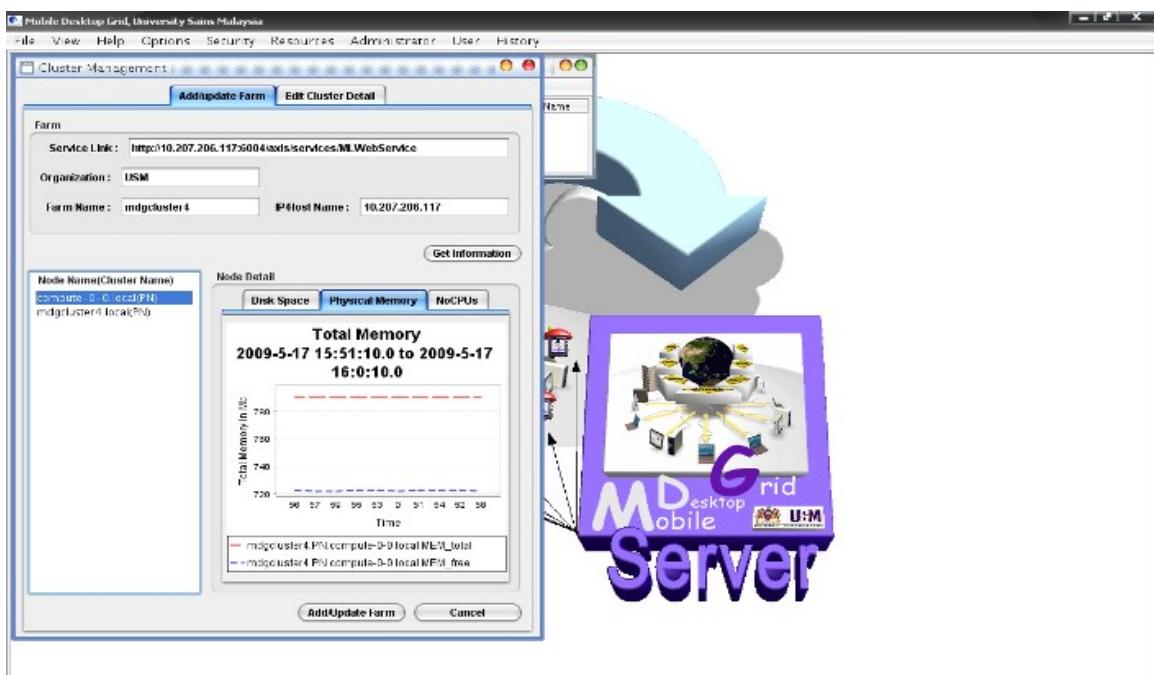
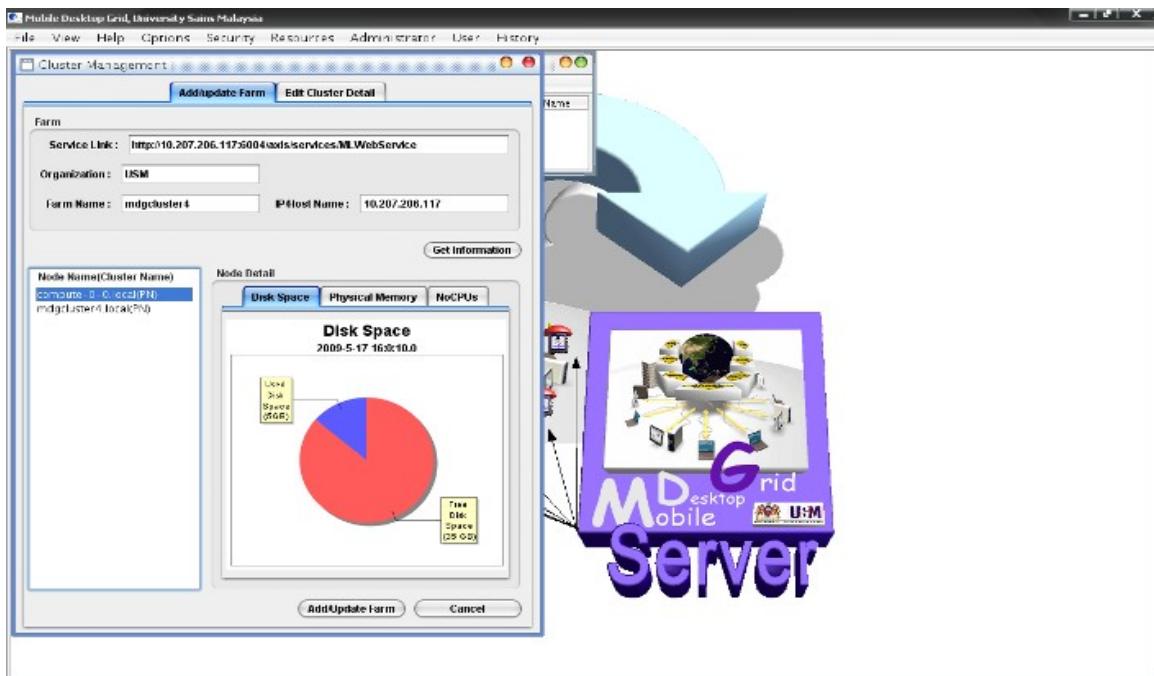
Choose the option of “Resources Management” under category of “Resources” at the menu bar to add new resources into MDG Server resources This task is done on behalf of administrator user.

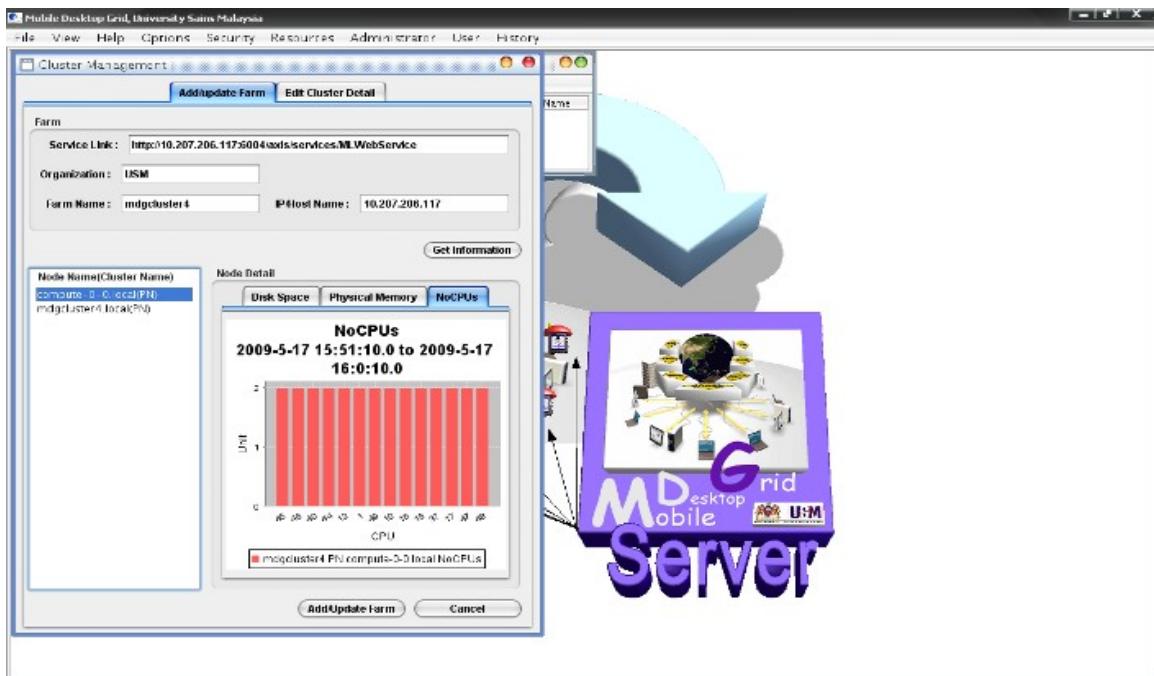


After clicking “Resources Management” option, the “Authentication” window will prompt out to ask for admin id and password.



Administrator user uses this window to manage the MDG clusters. The left tab at the top is for adding new cluster, and updating existing clusters in MDG Server. The right tab is for removing existing clusters from MDG Server resources pool. In the removing cluster activity, the checking is done and show a prompt up warning message to the administrator user. The message consists of the party information who are using the to be removed cluster.

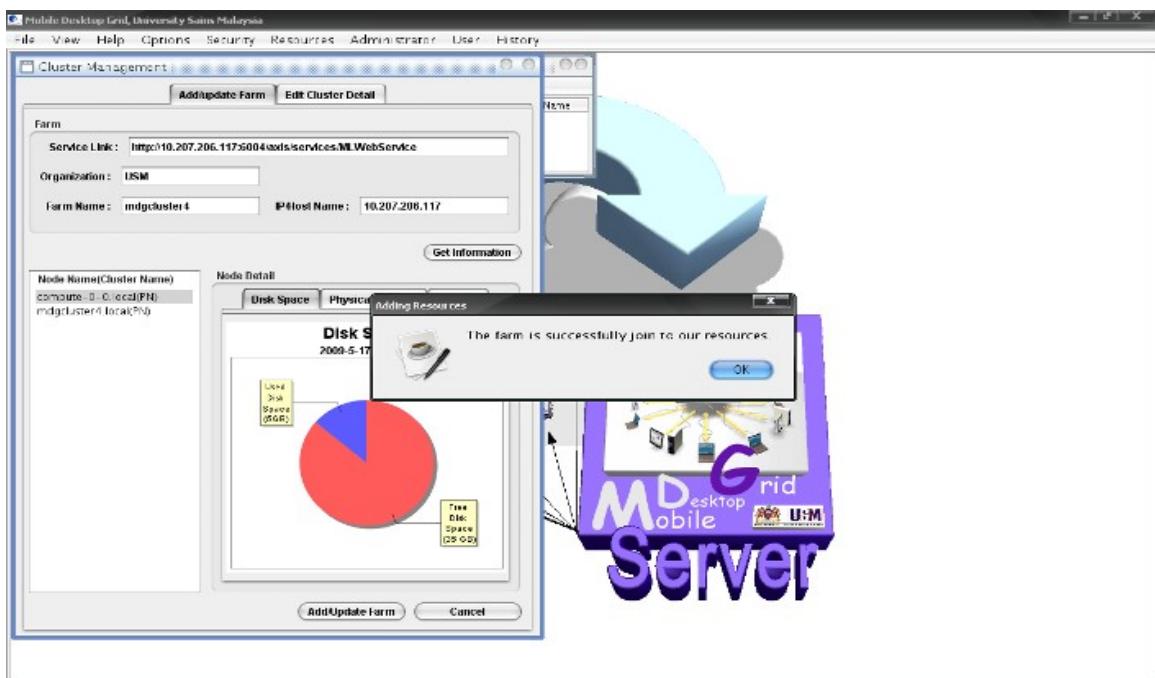




For the adding activity, click in the “Get Information” after the administrator user entered the following information:

1. Service Link
2. Farm Name
3. Cluster IP Address or Host Name
4. Organization Name

With the help of MonALISA service, related cluster information will be presented in JFreeChart format for administrator user previewing the cluster detail information, such as name of available nodes for the cluster, details of nodes in “Disk Space”, “Physical Memory”, and “Number of processor”.

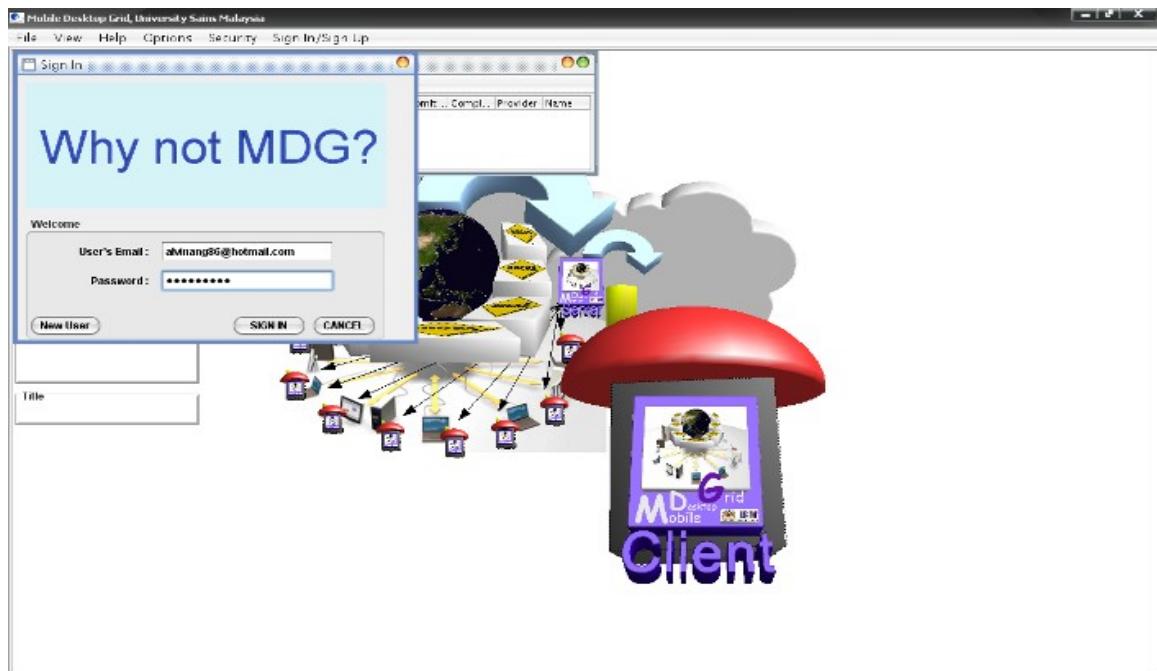


After previewing, administrator user clicks on “Add/Update farm” button to commit the selected cluster into MDG Server database. MDG Server uses embedded database, Apache Derby, to store all remote administrator users committed information.

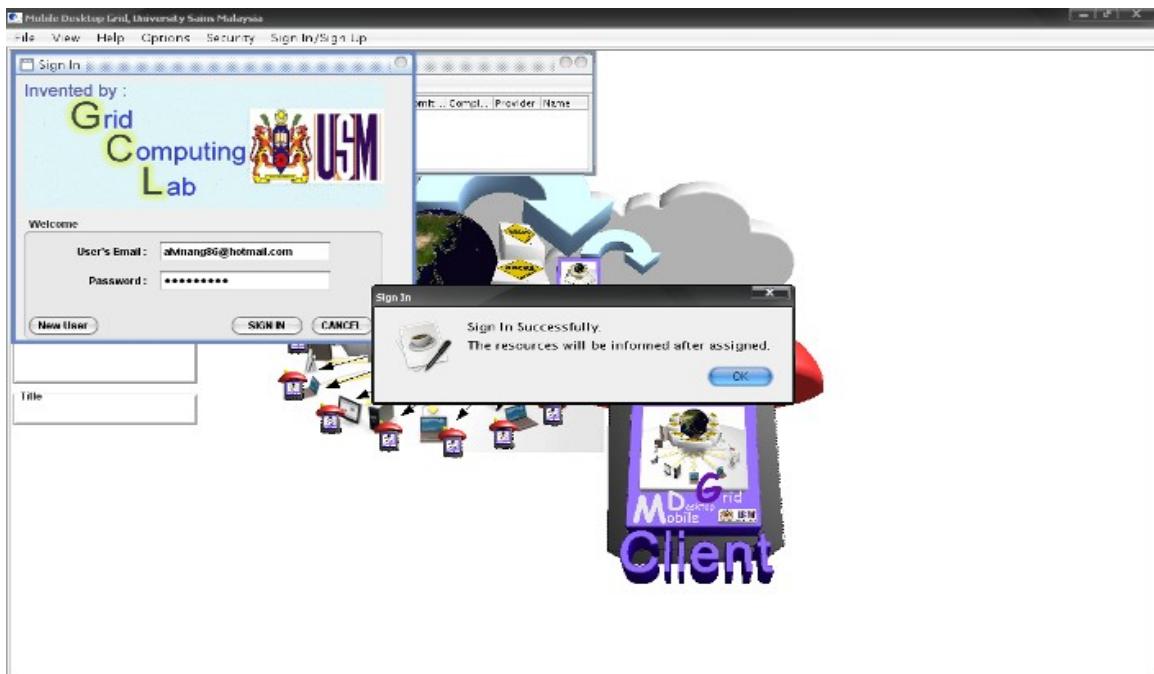
### 3.3 The 7 steps to sign in and get MDG solution resources



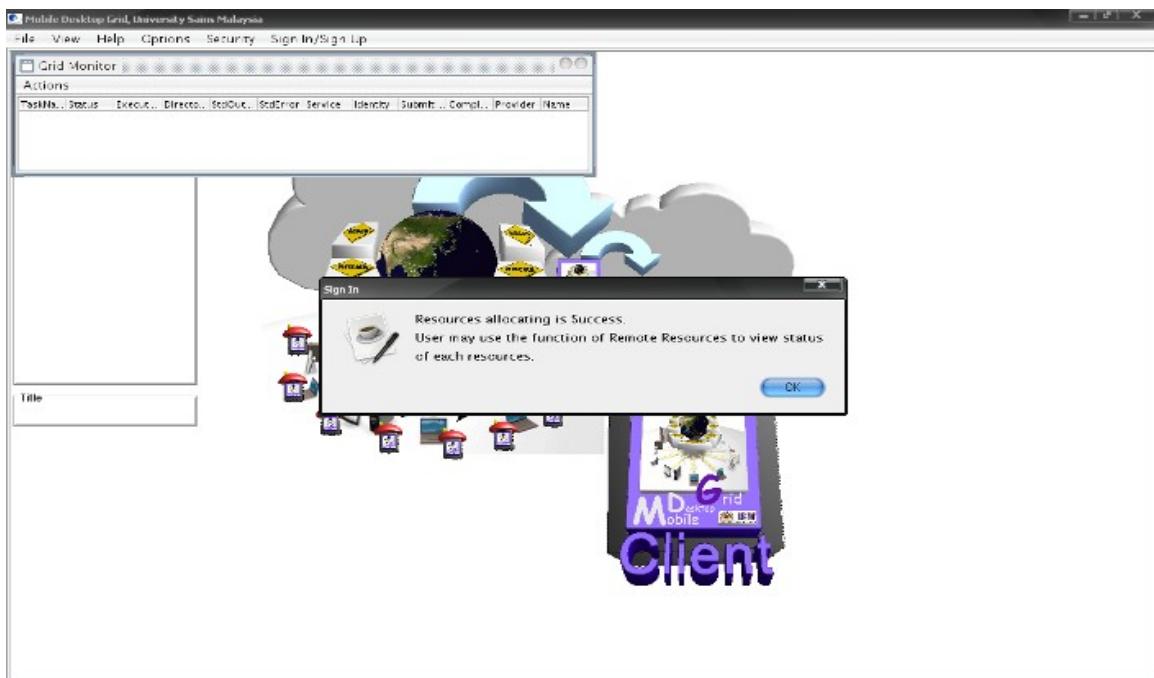
User login into MDG solution using MDG Client by the option “Sign In” under the category of “Sign In/Sign Up” at menu bar.



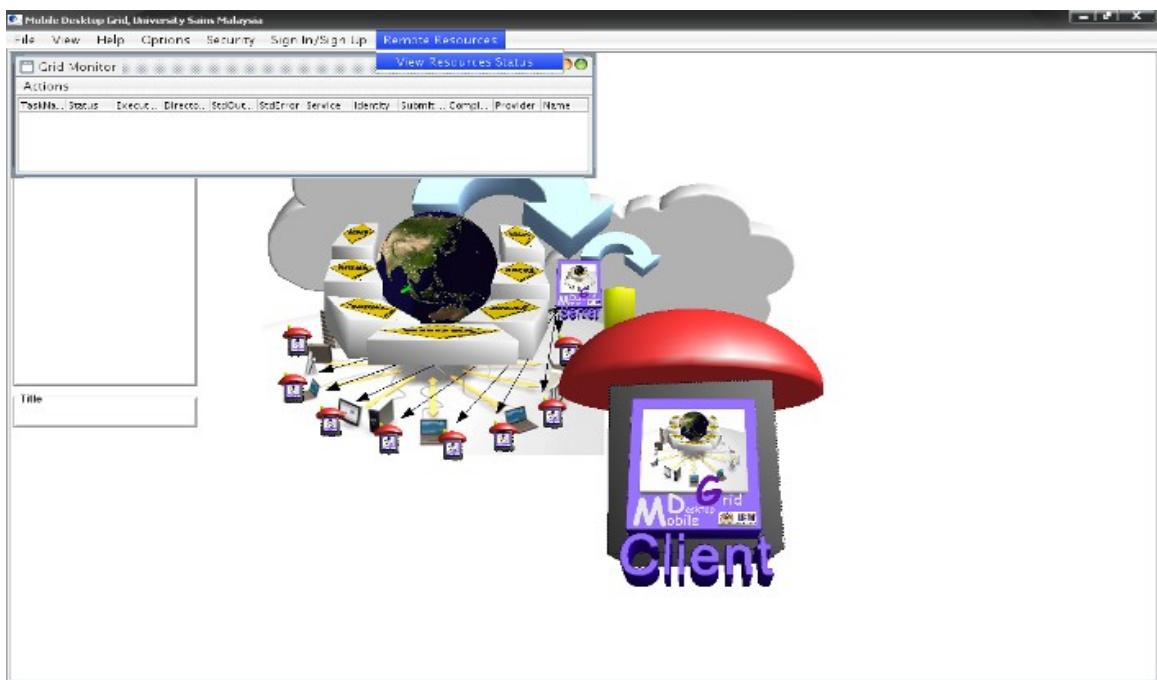
User requires to enter their email address and password to login into MDG solution by click on “Sign In” button. For registering a new account for user, click on “New User” and follow the new user registration wizards appear on window.



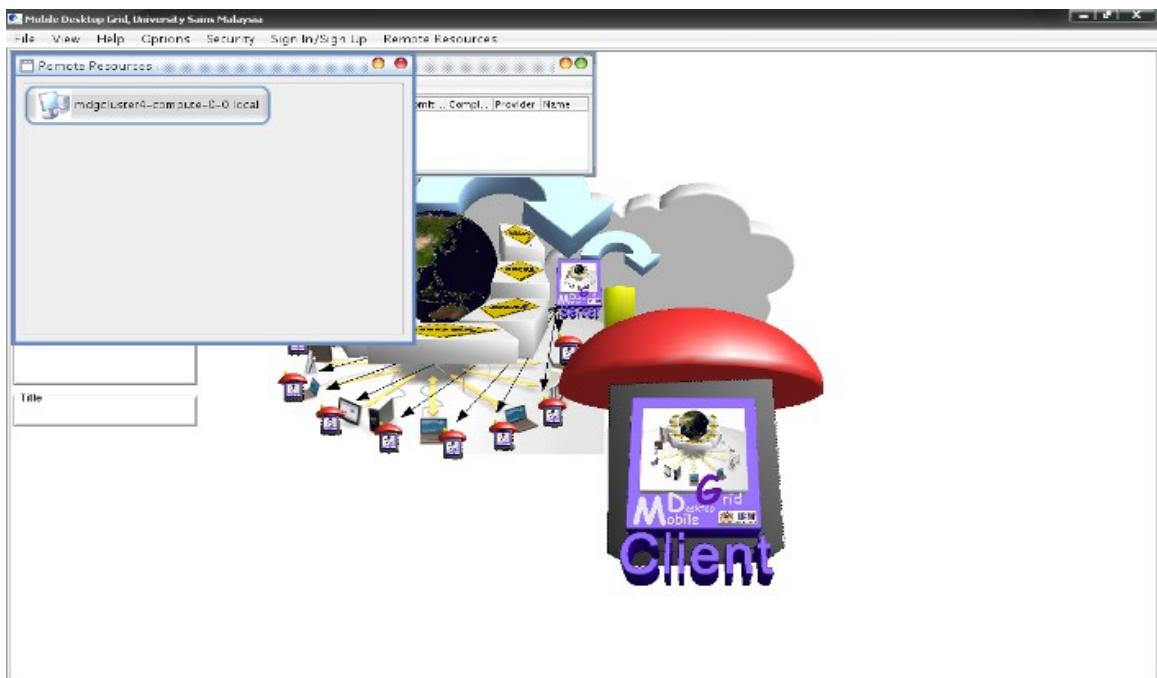
The remote MDG Server will perform the authentication for all MDG Client sign in activities. A prompt up window will show to confirm the successful of user sign in with valid pair of email address and password. For not success case, sign in process will terminate in this interface.



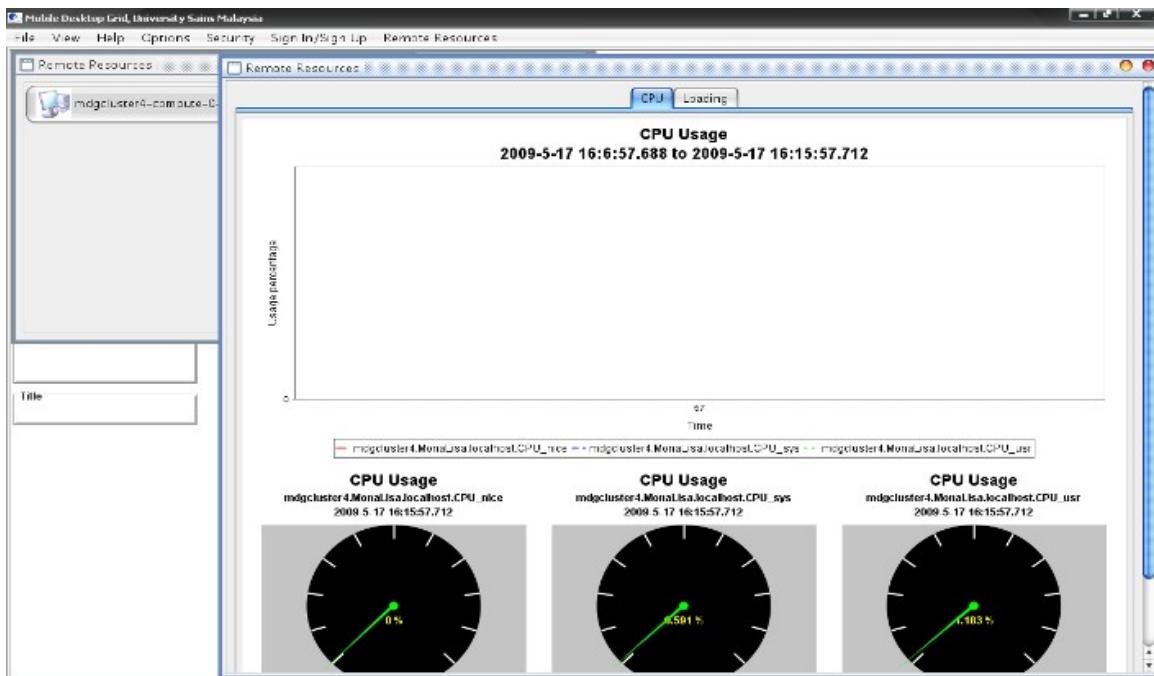
MDG Client will trigger the remote MDG Server, and obtain the configured resources package from resources pool. A prompt up message will show for successful or failure resources assignment of MDG Server remotely. The message consists of detail resources information has been assigned to user.



Upon the successful allocation of resources, a new menu named as “View Remote Resources” is available at menu bar. This option allows user to preview the update-to-date remote resources status.

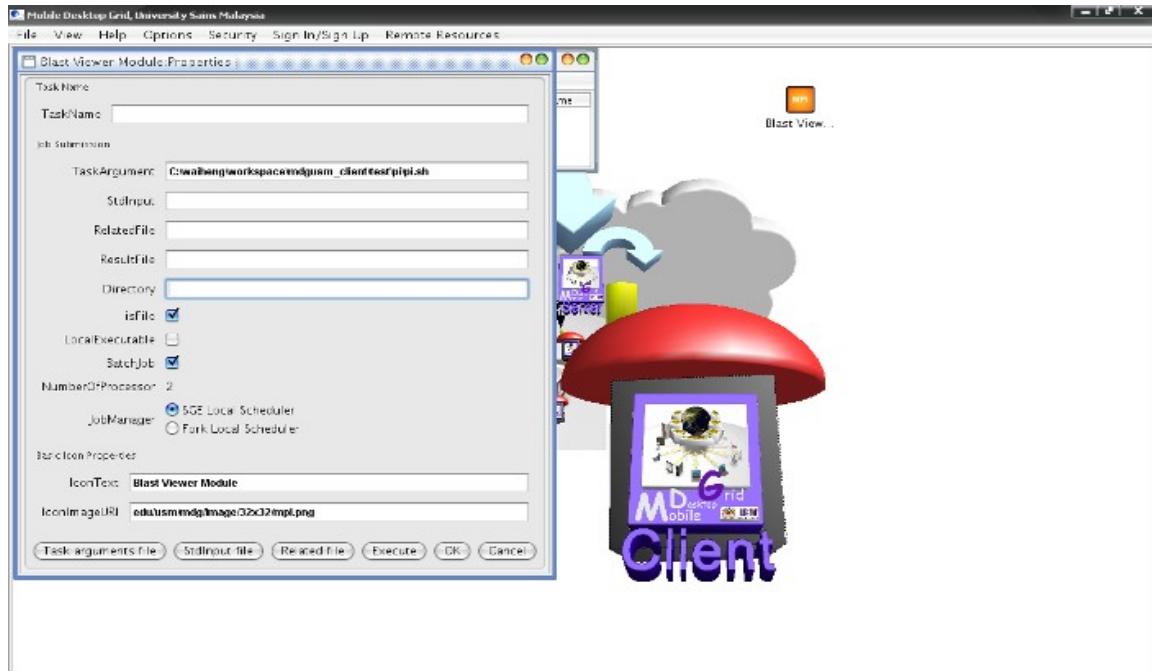


Above screen will be seen after choosing the option “View Remote Resource” at menu bar. The window will show the related resources for user previewing in the format of dynamic button presentation according to number of nodes availability



Above screen shows the detail of particular node in cluster using JfreeChart presentation. It utilizes MonALISA service to pull entire node information.

### 3.4 The 3 steps to submit PI job: Using SGE Local Scheduler

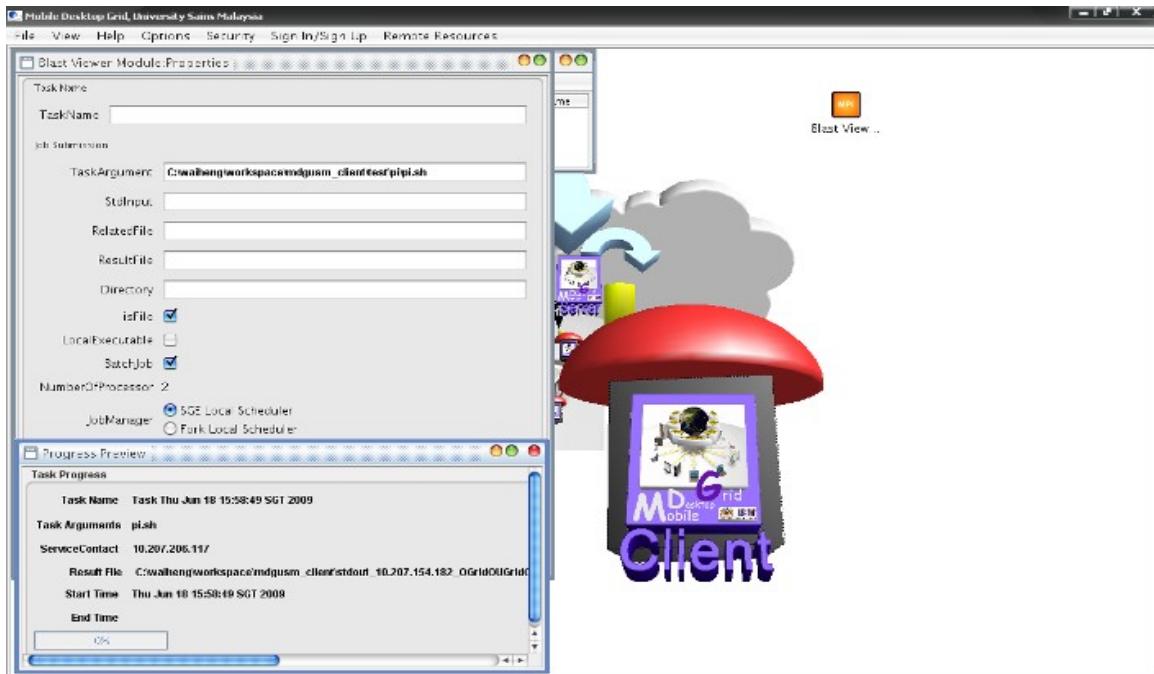


Above shows the submitting MPI job window. Task name is optional, and user can uses it to keep track the executing jobs which will place on MDG Client interface.

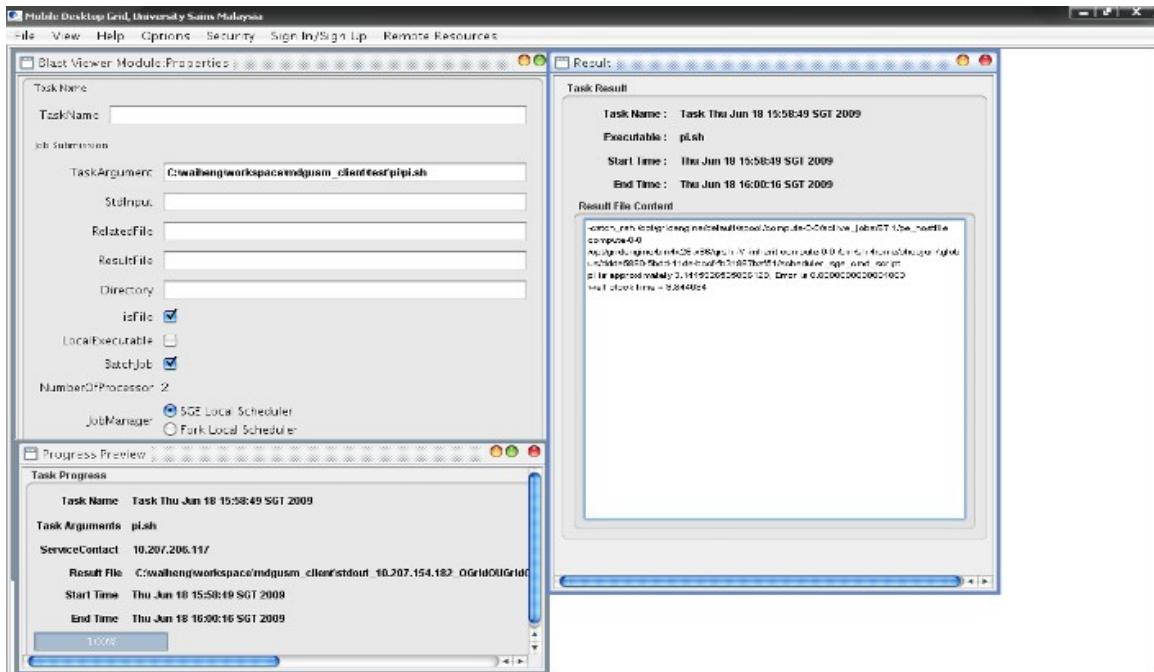
Click on the buttons for Task Argument, StdInput, and Related File for job execution to fill in these mandatory fields.

The Number Of Processor is the maximum availability processor that can be used by user in job execution. This value is declared by remote MDG Server.

There are two options in Job Manager for user to decide the job execution.

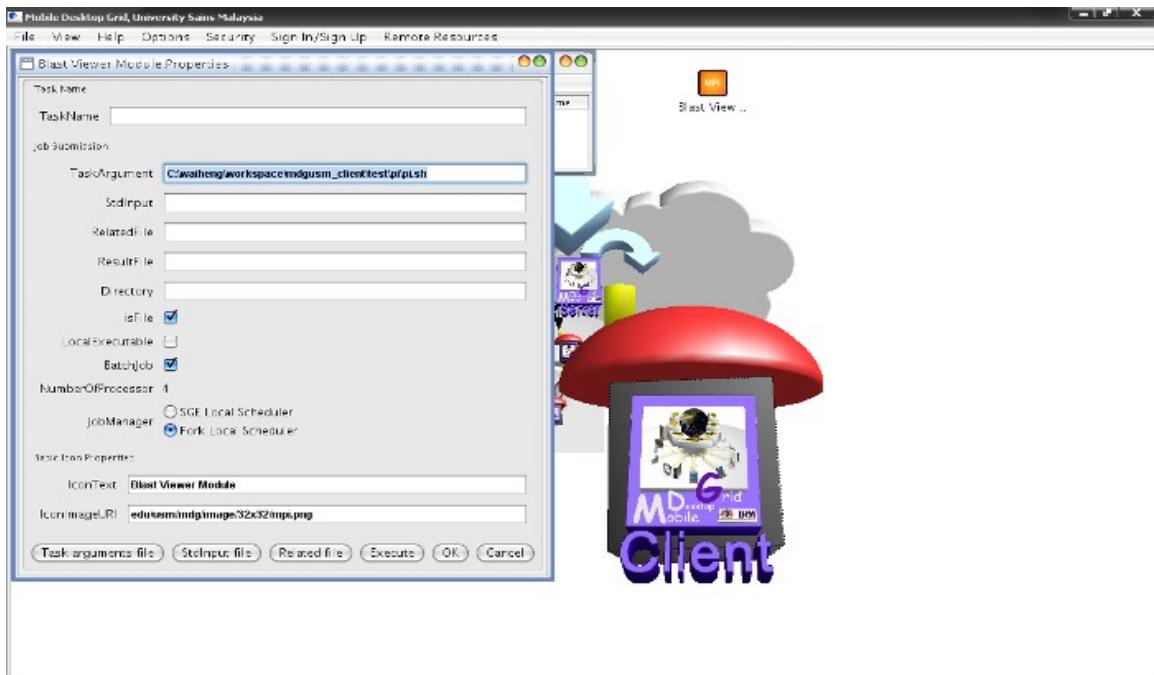


Click on the “Submit” button to submit the job, and a progress bar will show at the bottom of window as remote progress acknowledgment.

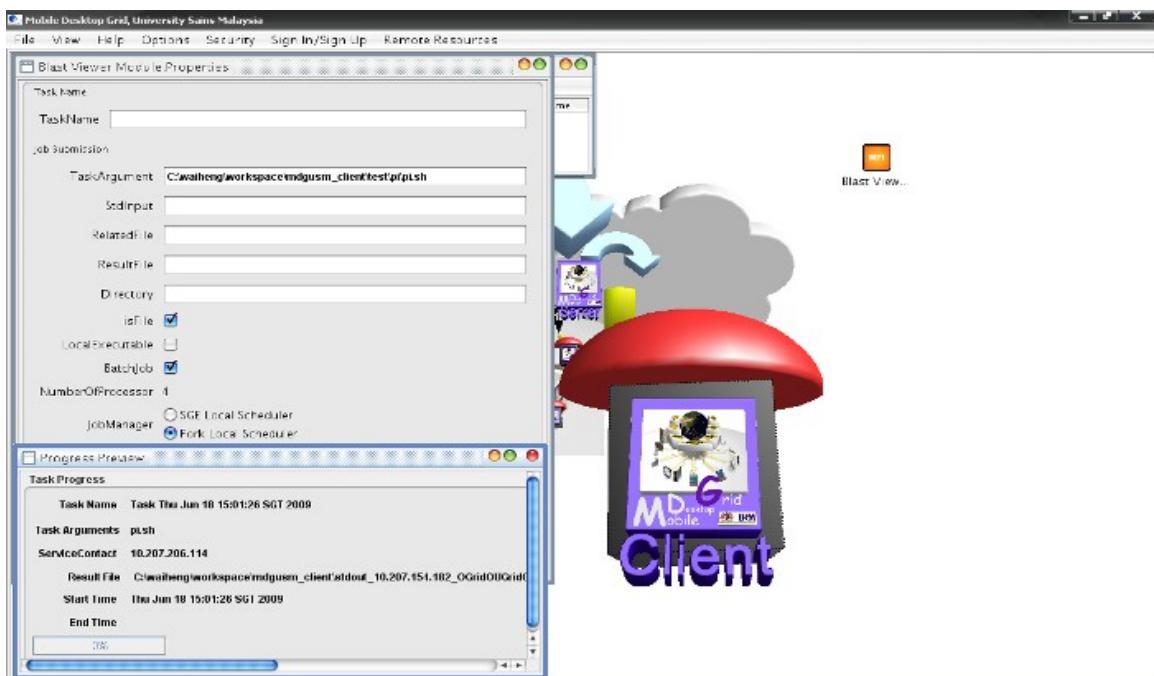


As the remote job execution done, a prompt up shows the result of job StdOut file, and it also records the job start-end time.

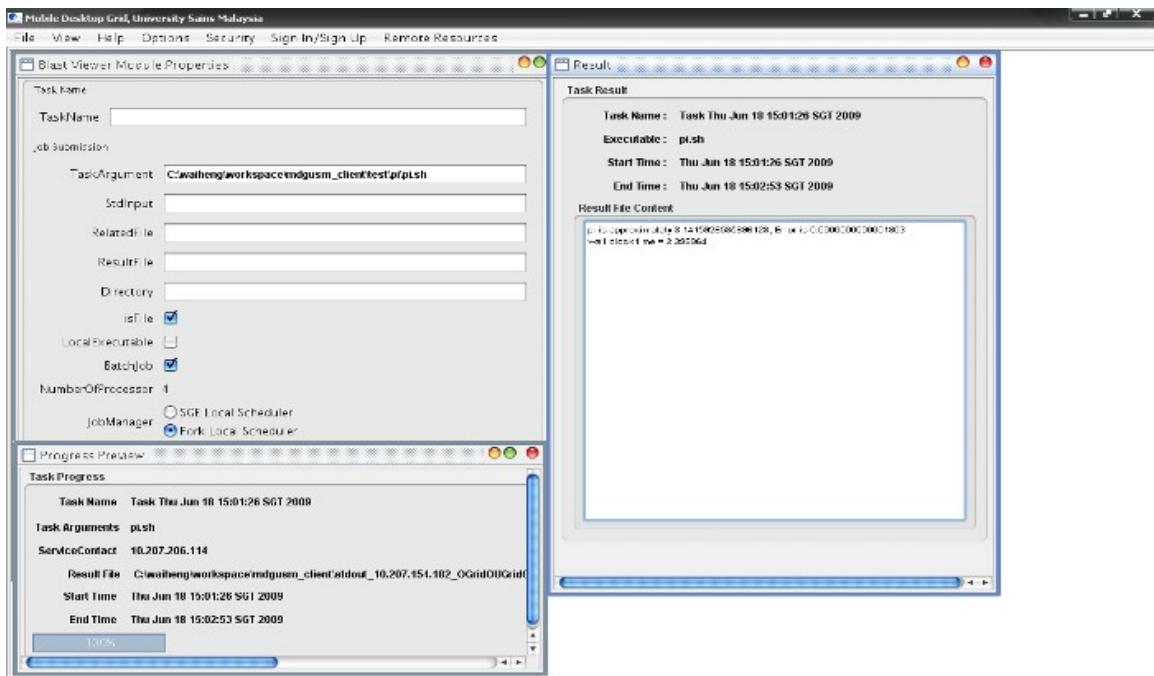
### 3.5 The 3 steps to submit PI job: Using Fork manager



Above screen shows a job of calculating PI in a cluster named Mdgcluster3 (situated in VMware ESXi environment)



After click on the “Submit” button, a progress window show at bottom of window. It captures information of task name, task argument, service contact, result file name, and the submit job time.

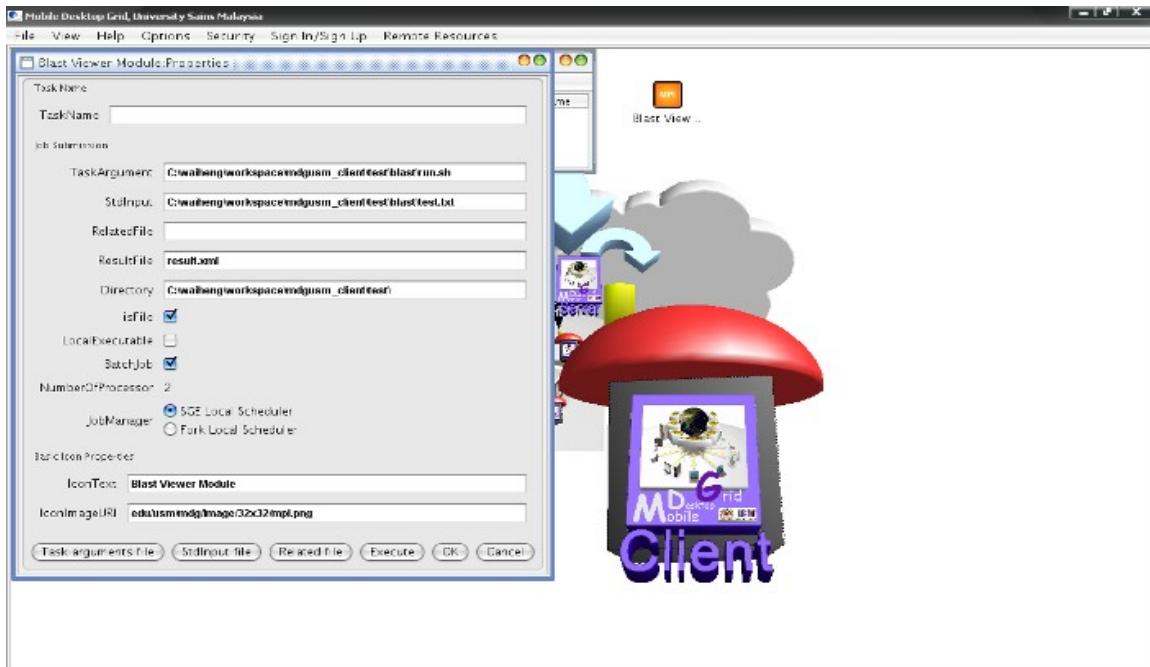


As the job execution done, a prompt window show the job execution result.

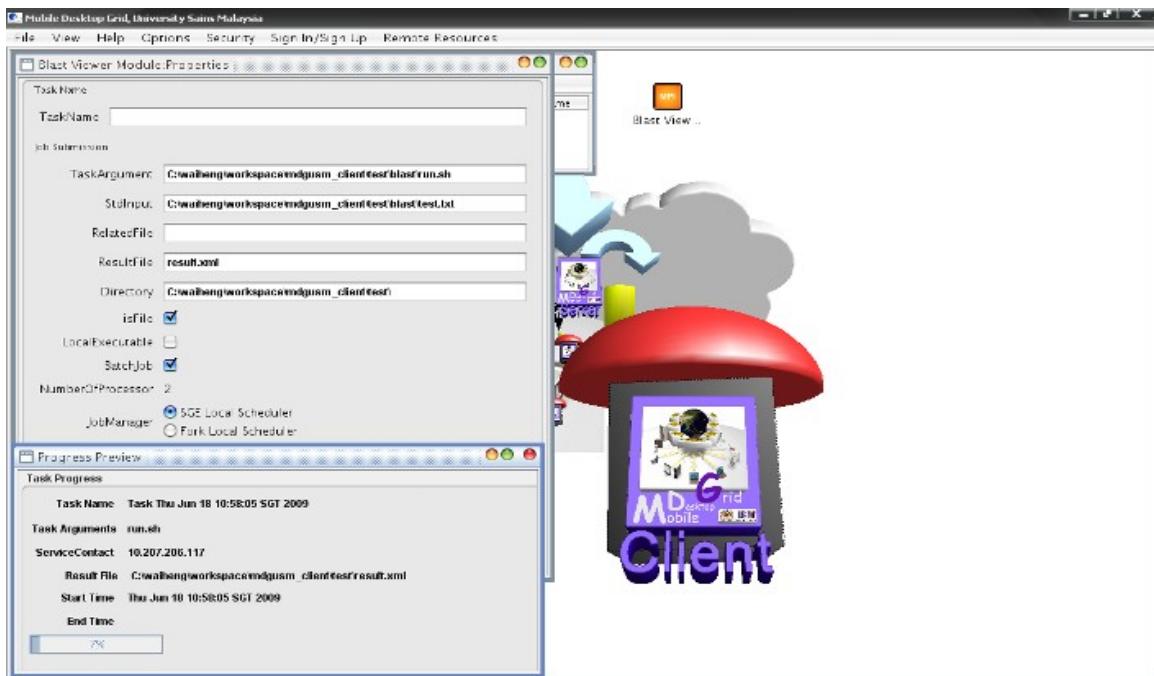
## 3.6 The 5 steps to submit NCBI-Blast job: Using SGE Local Scheduler



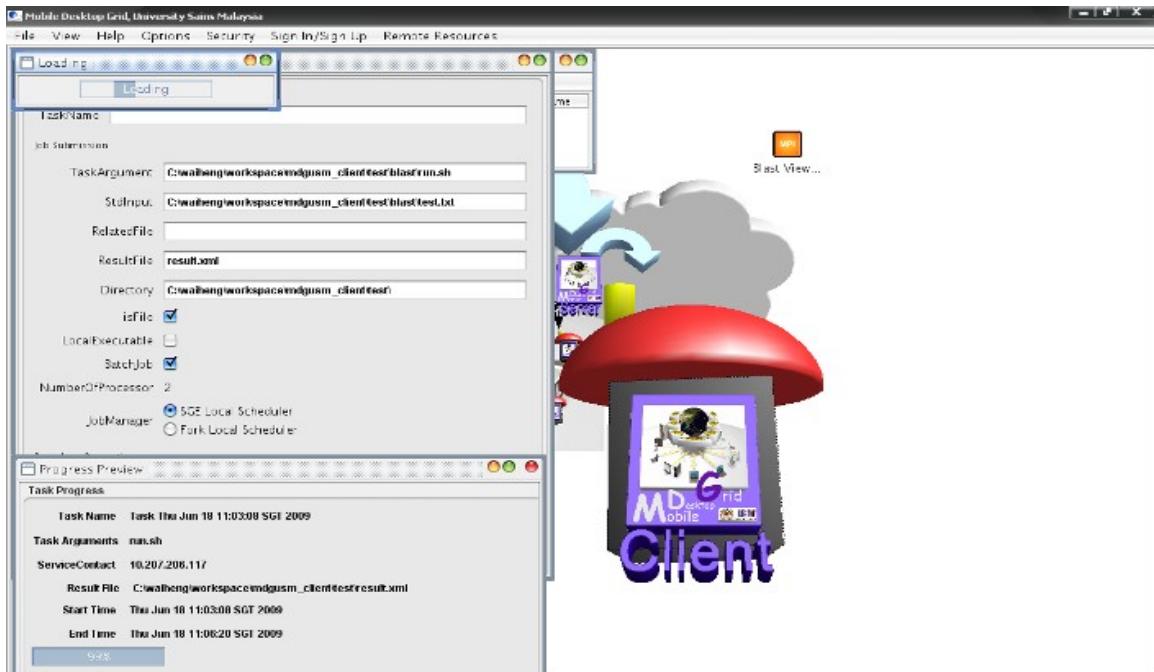
After obtained server assigned resources, MDG Client will unlock user to use right-click options.



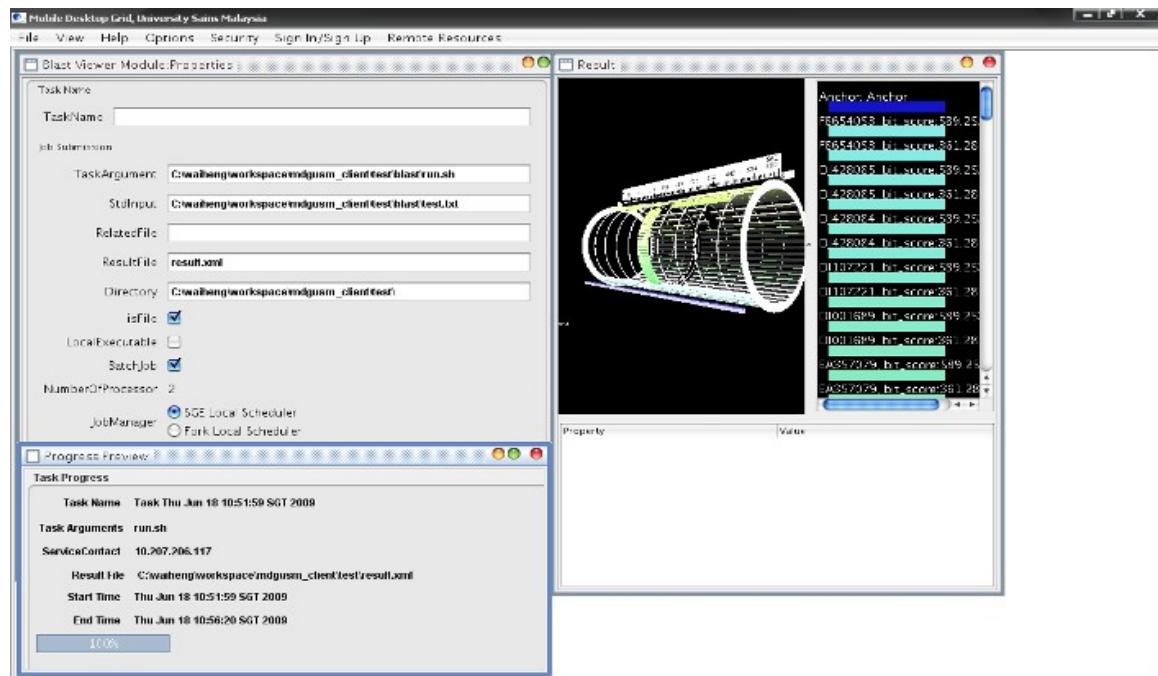
Click the Submit button to execute job remotely after fill in all required information.



The progress bar will show in bottom window after the submitted for remote execution. It has the job start time and related information.

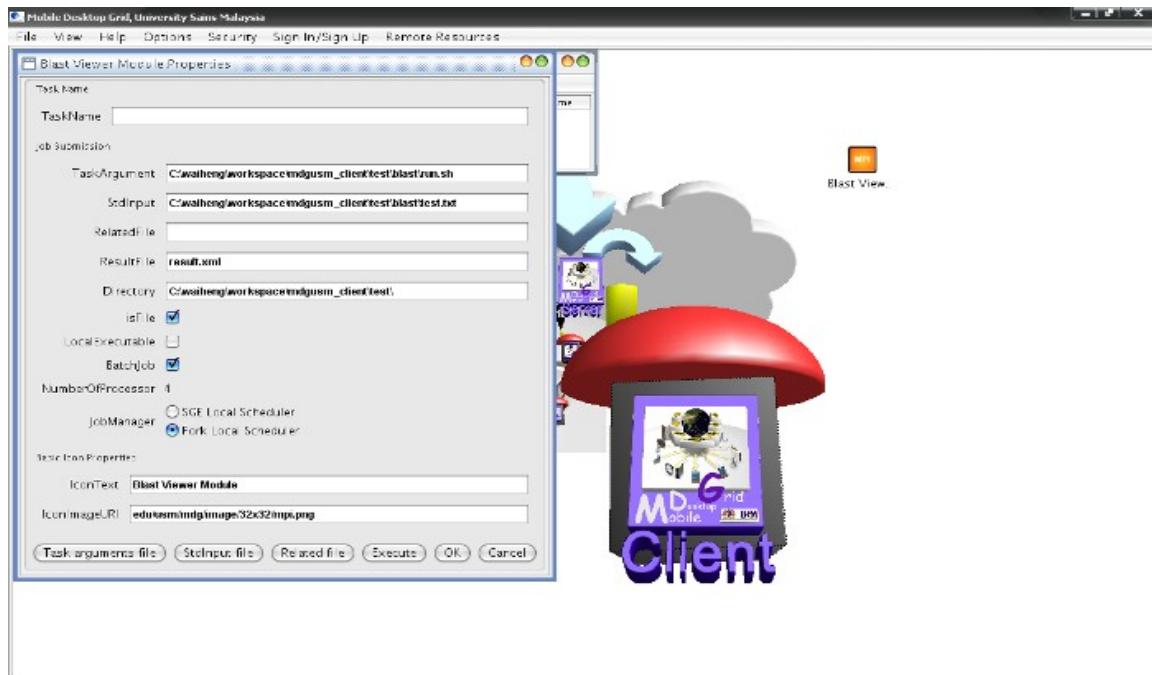


As the Blast job finished executing, the MDG Client will convert the received file from server to XML format.

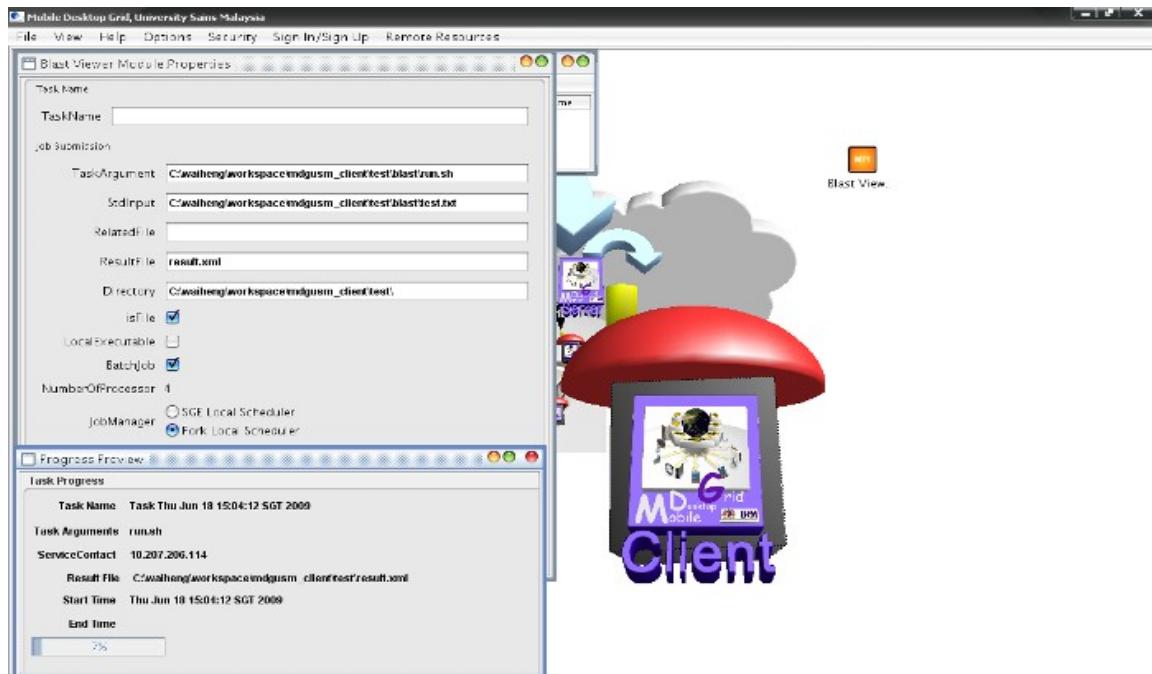


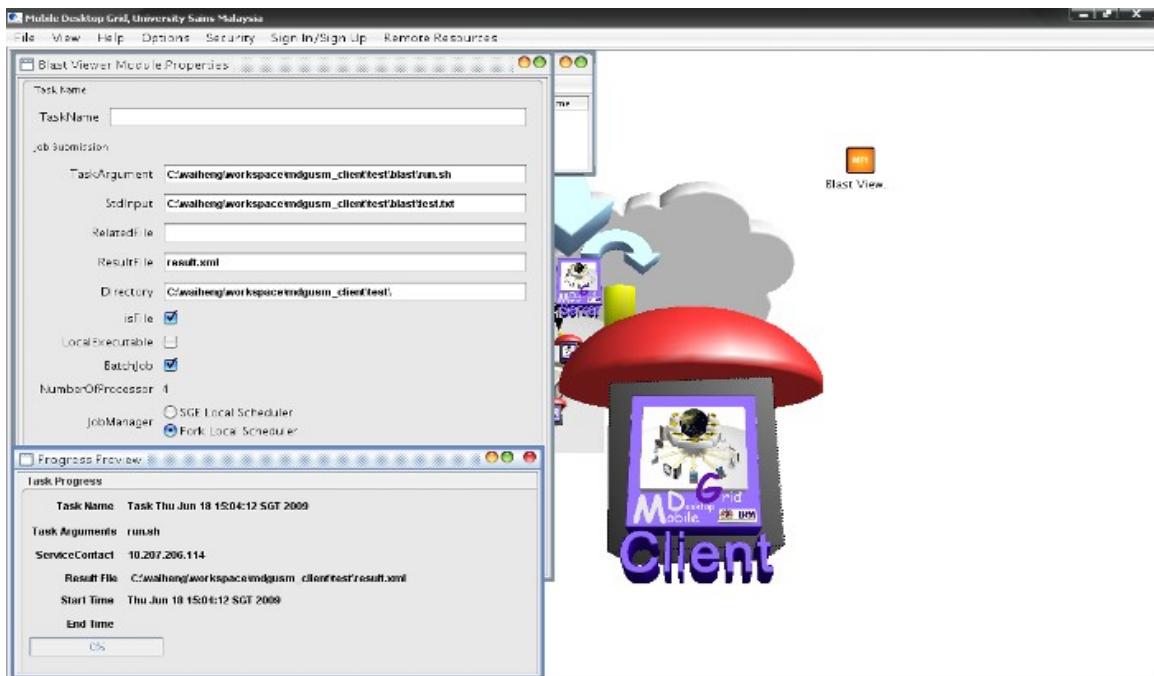
Above screen shows the converted Blast result in 3D rotate-able window.

### 3.7 The 4 steps to submit NCBI-Blast job: Using Fork manager

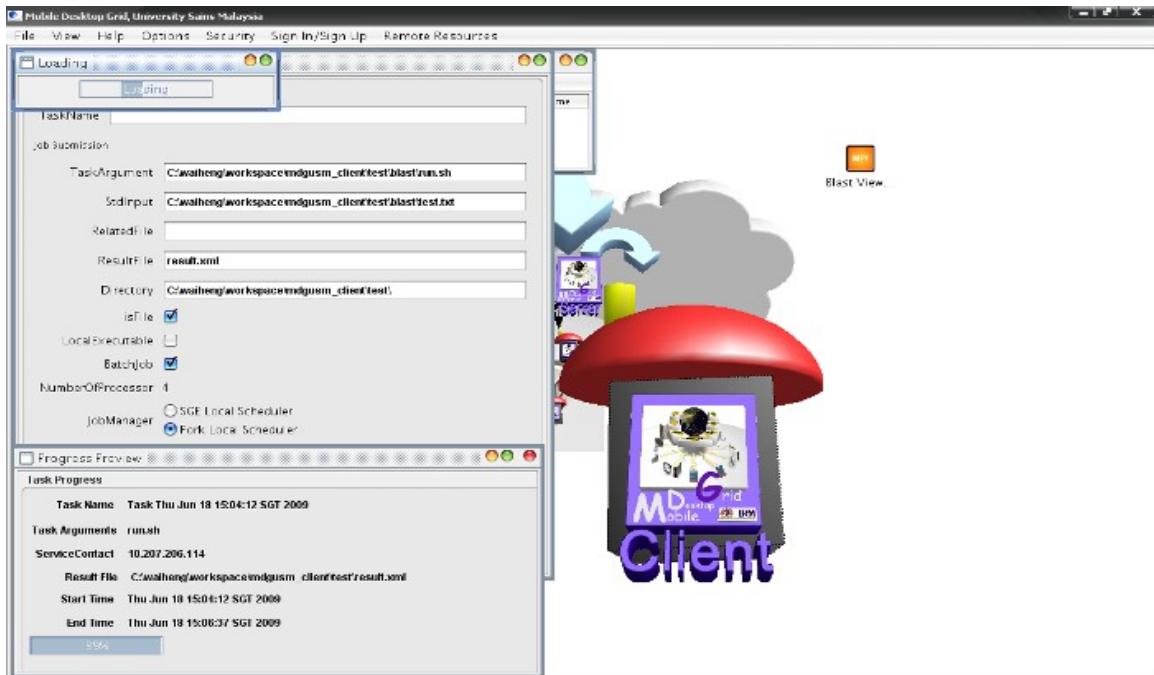


Above screen shows a Blast job in a cluster named Mdgcluster3 (situated in VMware ESXi environment) using Fork Local Scheduler.

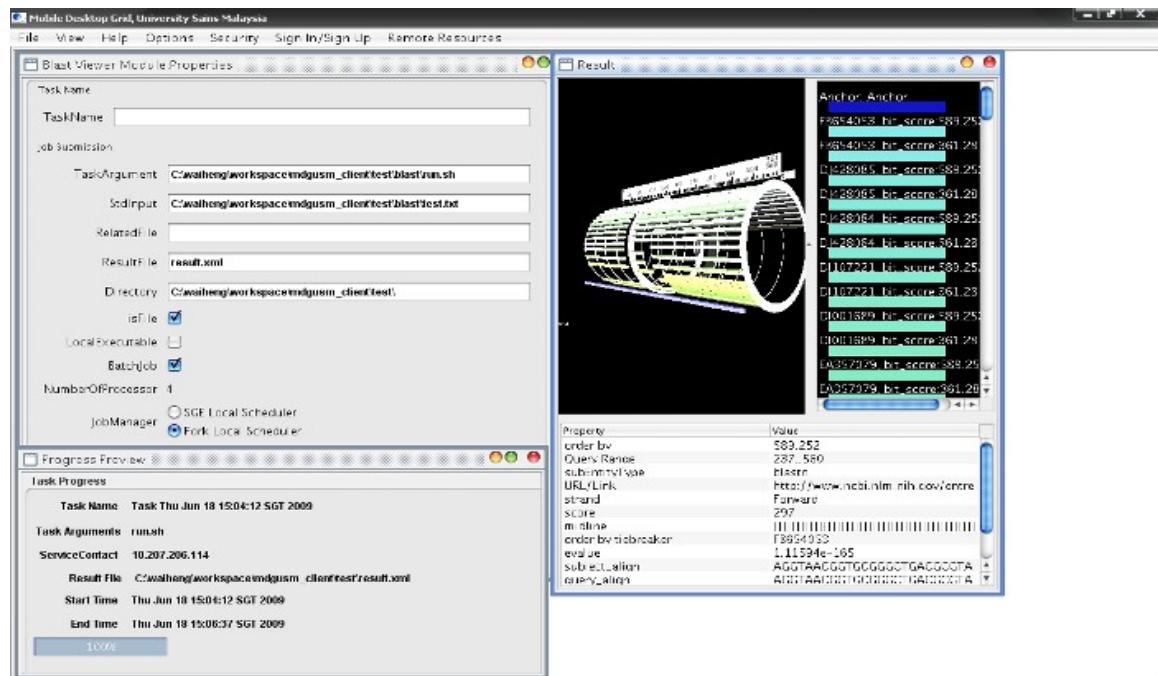




As the job submitted for remote execution, the progress bar show at bottom window with detail information, such as service contact, remote cluster IP address for Mdgcluster3.

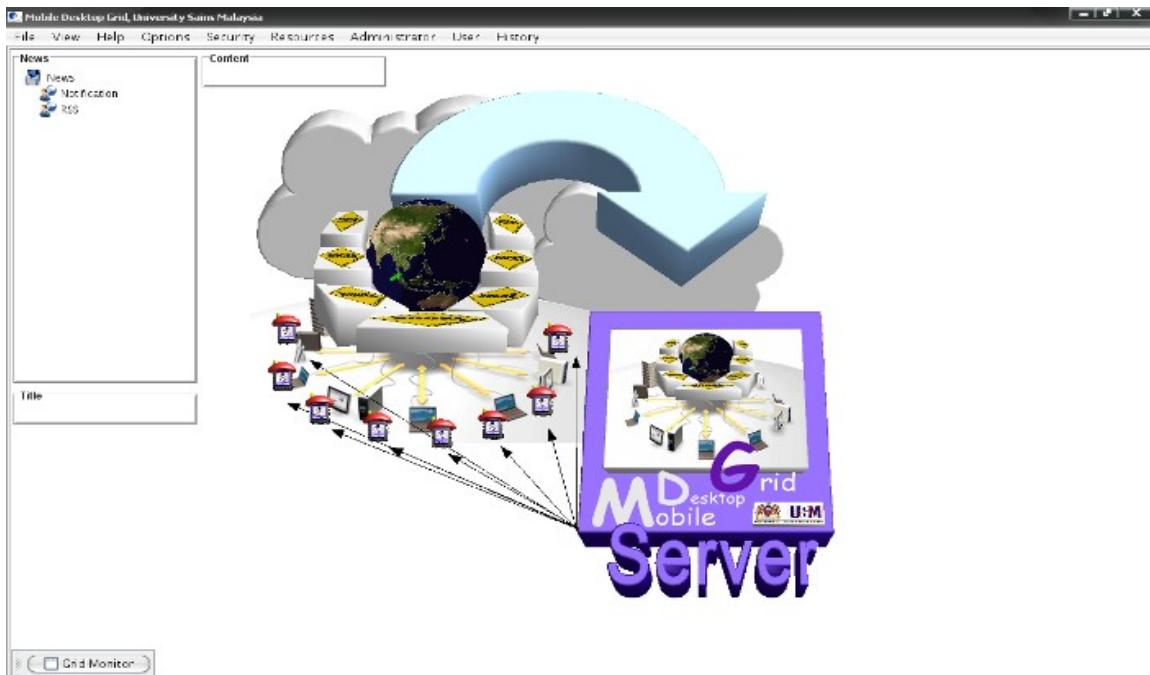


As the Blast job finished executing, the MDG Client will convert the received file from server to XML format.



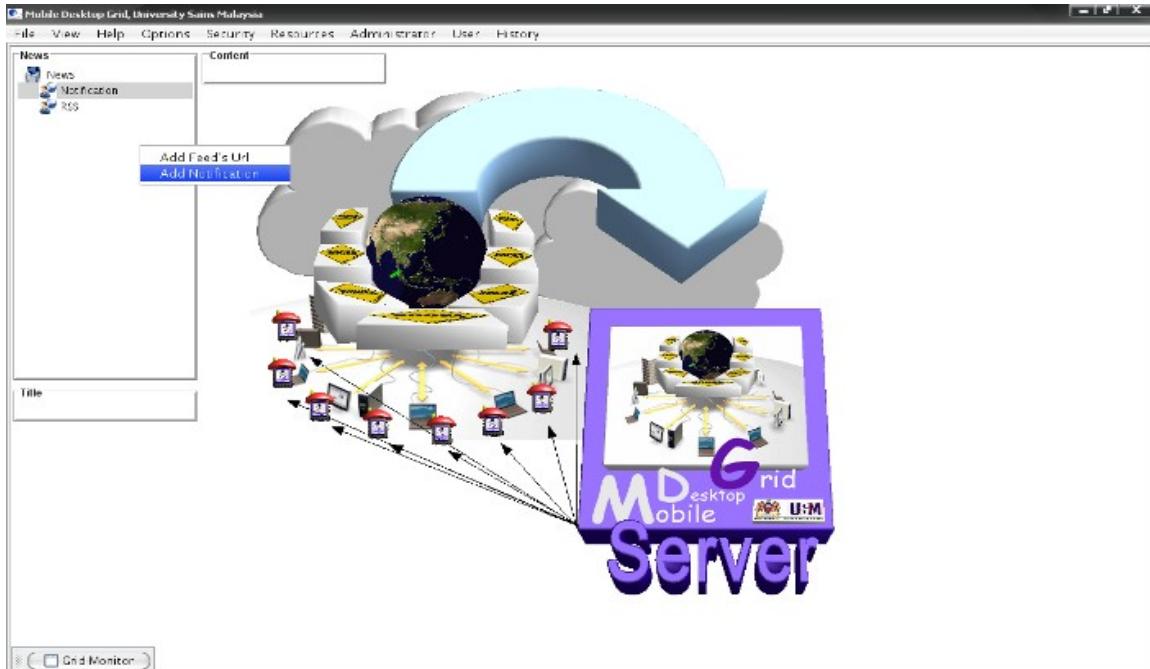
Above screen shows the converted Blast result in 3D rotate-able window.

### 3.8 The steps to post and receive notification: At MDG Server

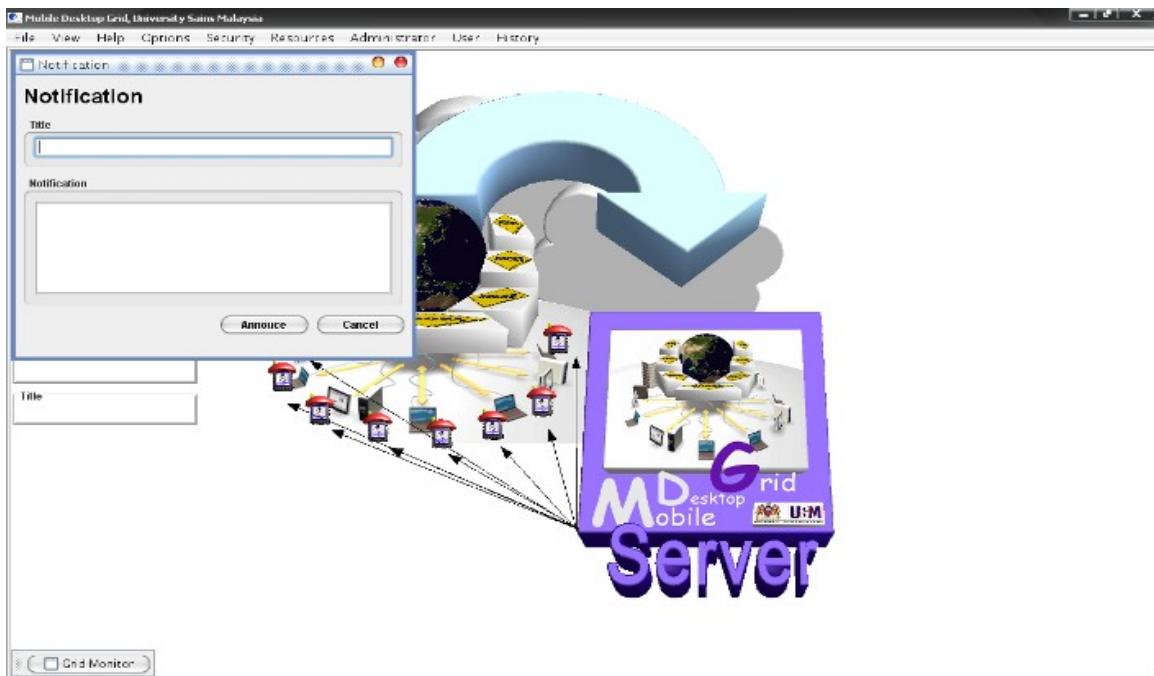


There is a service named “News” situate at the left tree window. This item in MDG Server uses to manage the following services:

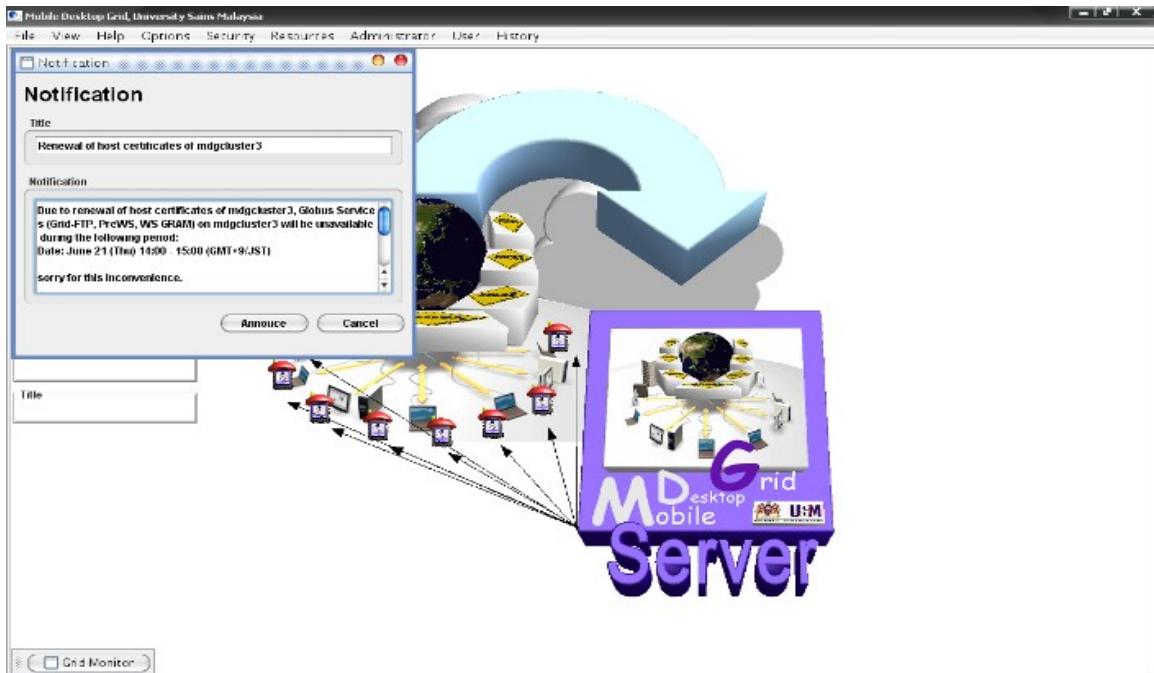
1. to get RSS news from public RSS provider
2. to broadcast-receive notification to-form MDG Clients



Only MDG Server administrator user allows to broadcast notification.



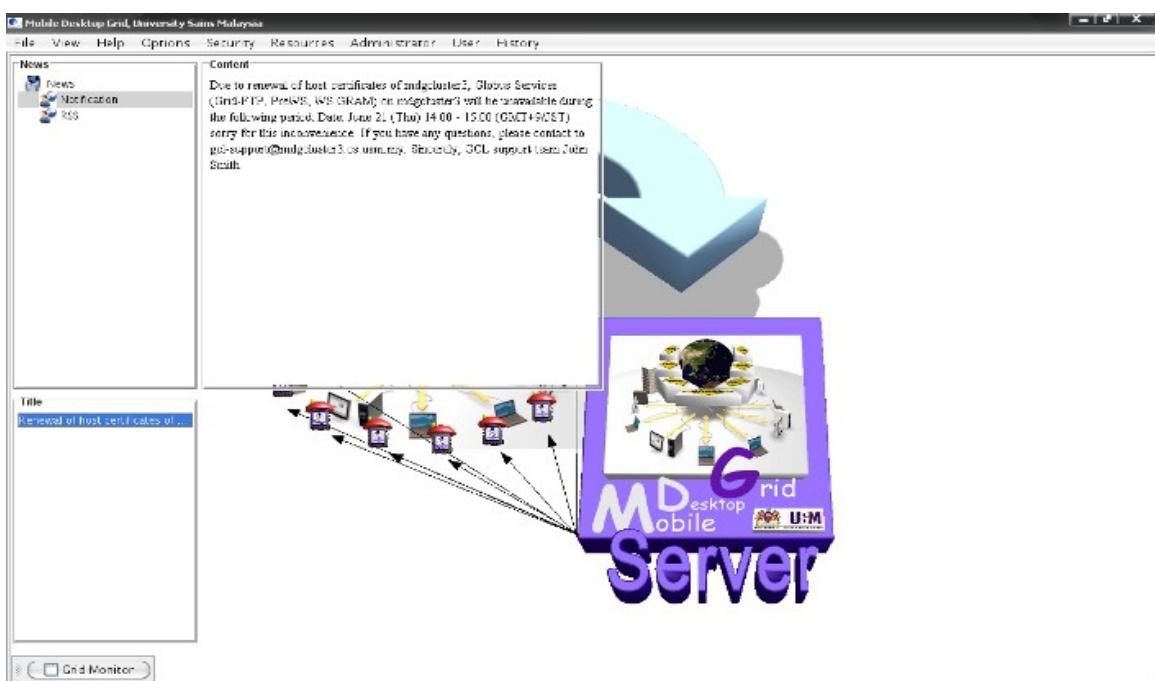
Choose the option “Add Notification”, the MDG Server will prompt up a window for administrator user to enter the notification title and content.



After filled all required information, click on the “Announce” button, and the message will broadcast to all MDG Clients using Sun Open Message Queue.

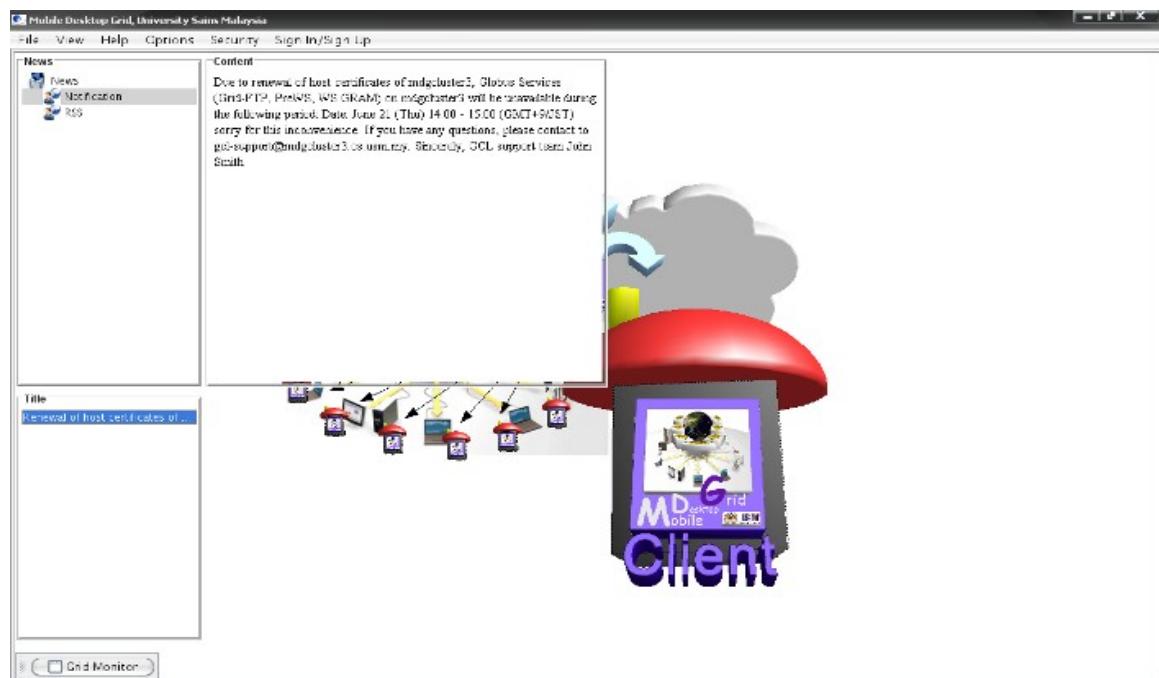


At MDG Server interface, click on the “Notification” of “News” at left window to receive the sent notification.



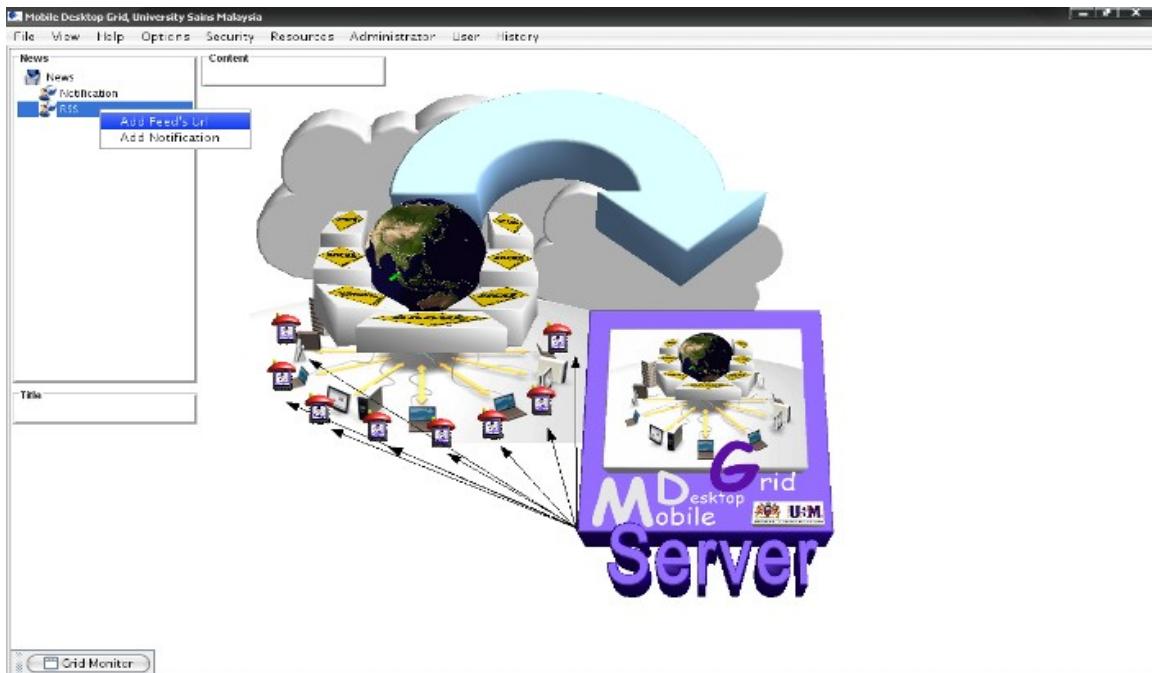
The broadcast notification will show at the right “Content” window. This window will minimize as the administrator user perform another “News” selection at left window.

### 3.9 The step to receive notification: At MDG Client

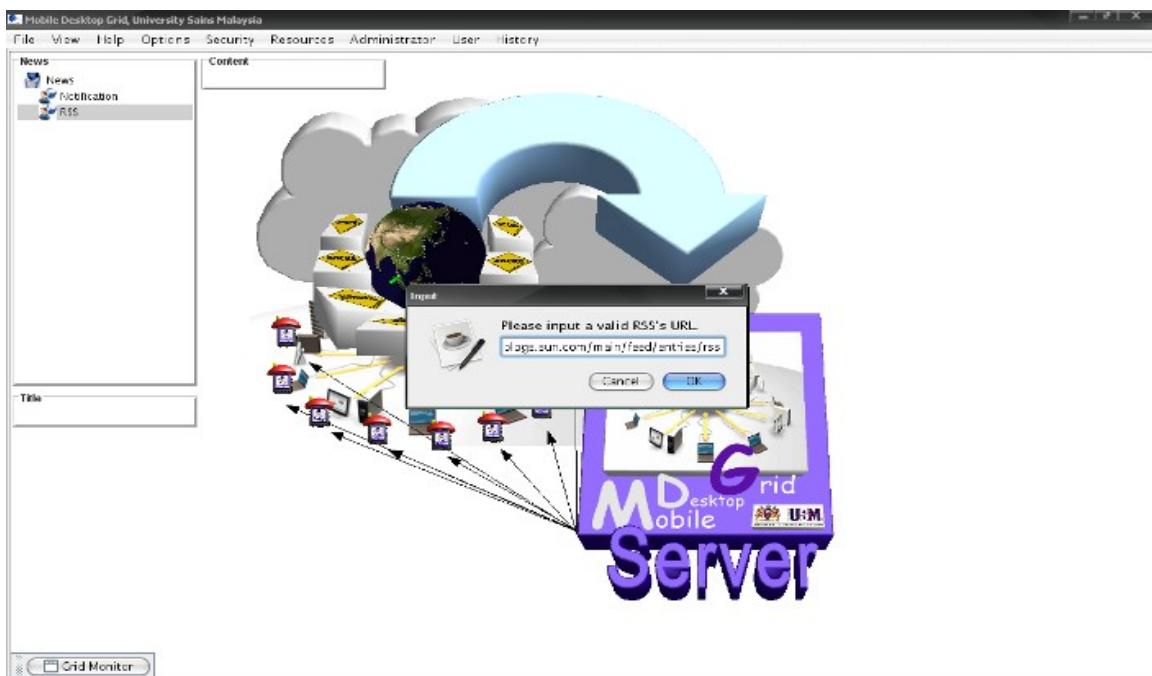


At the MDG Client, click on the “Notification” of “News” at left window to receive the remote MDG Server administrator user broadcast notification.

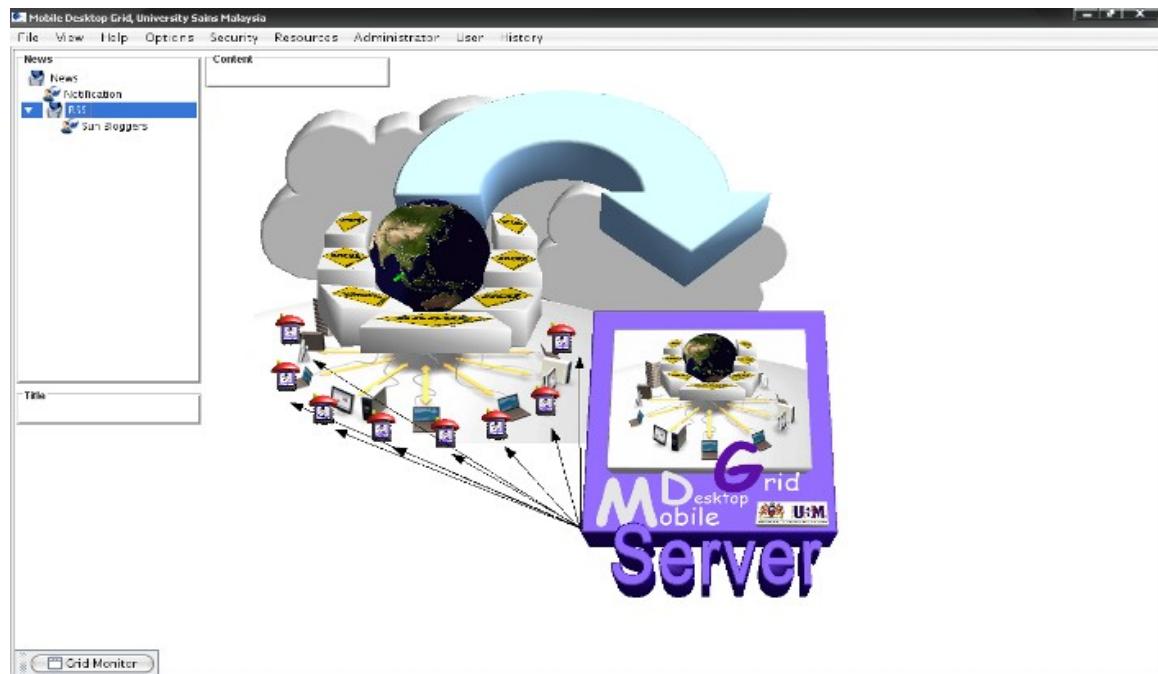
### 3.10 The steps to add new RSS feed: Sun Blogger RSS



At the left window MDG Server (same as MDG Client too!), choose the option “Add Feed’s Url” of “News”



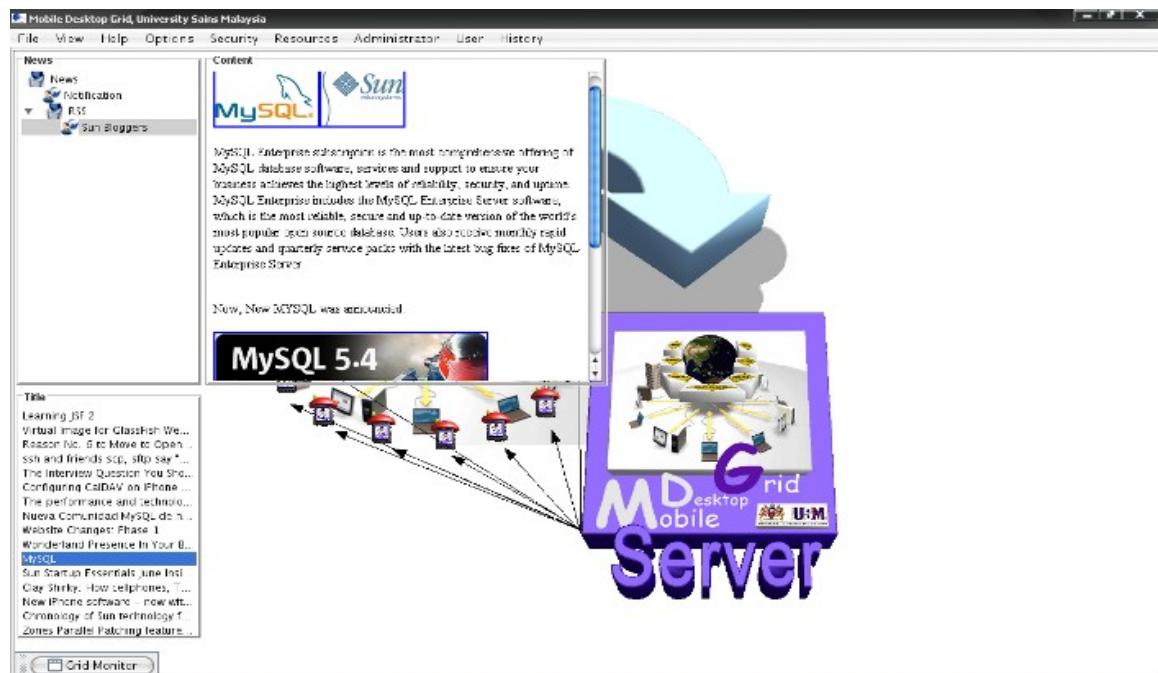
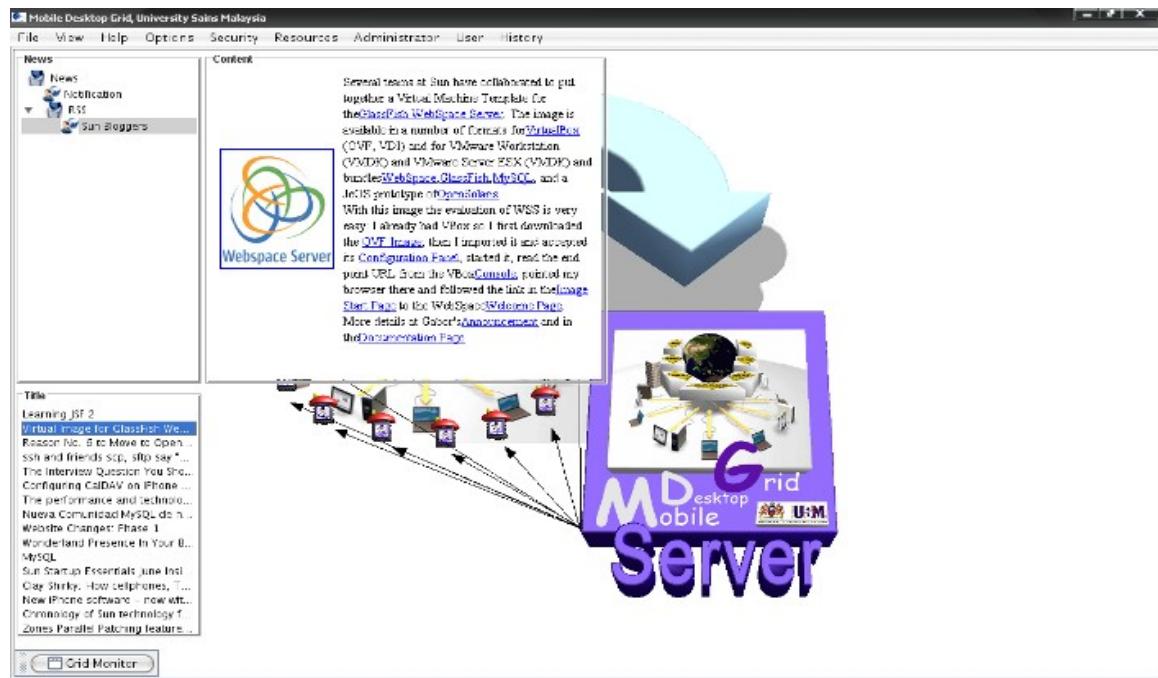
Enter the RSS Url at the prompt up window.



The entered RSS Url will appear at the bottom of "RSS" at the left tree window.

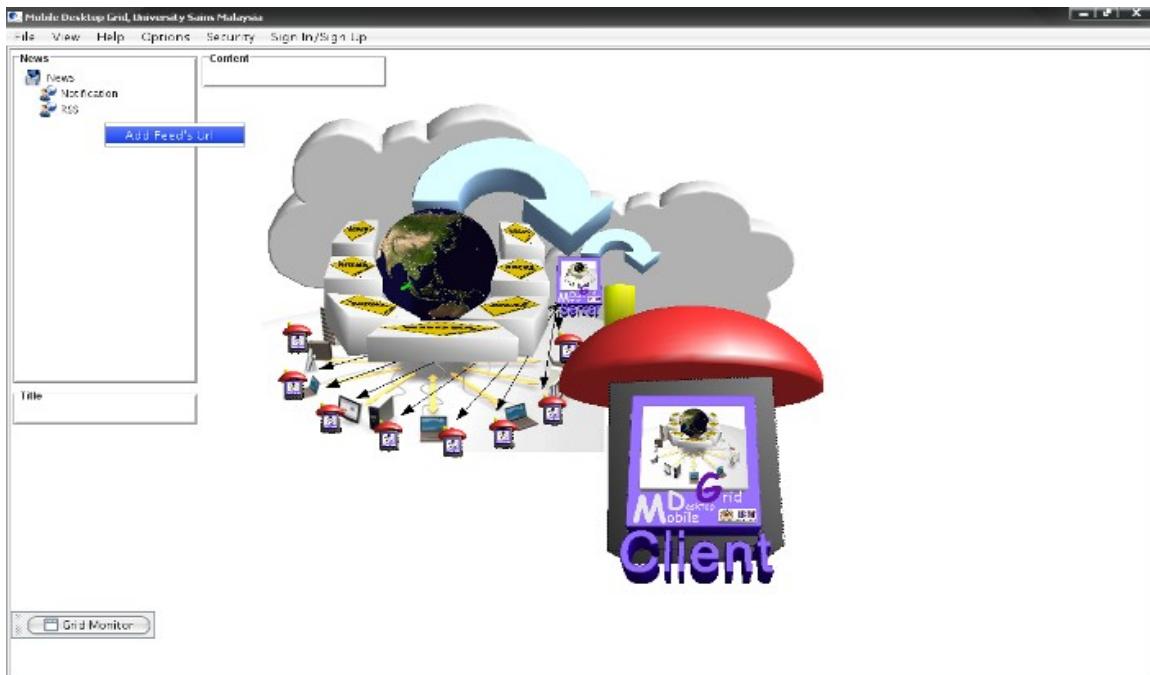


The titles of the entered target RSS will appear at the bottom window.

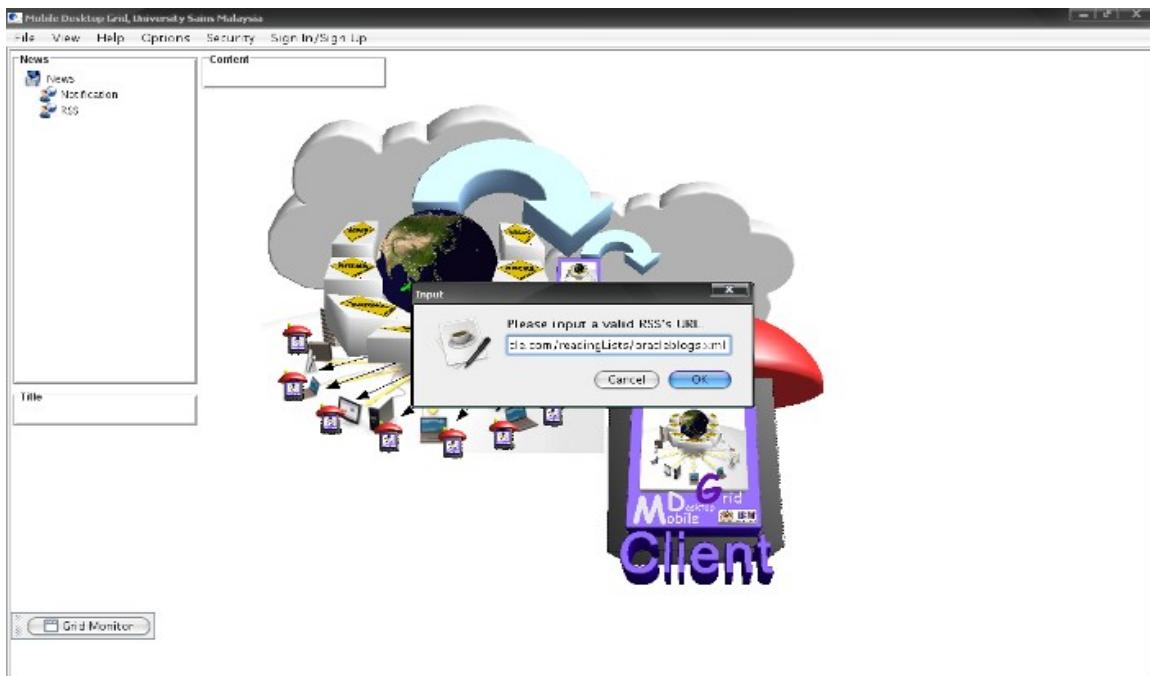


Click on the title of bottom window to retrieve the content. The selected title RSS content will show in "Content" window.

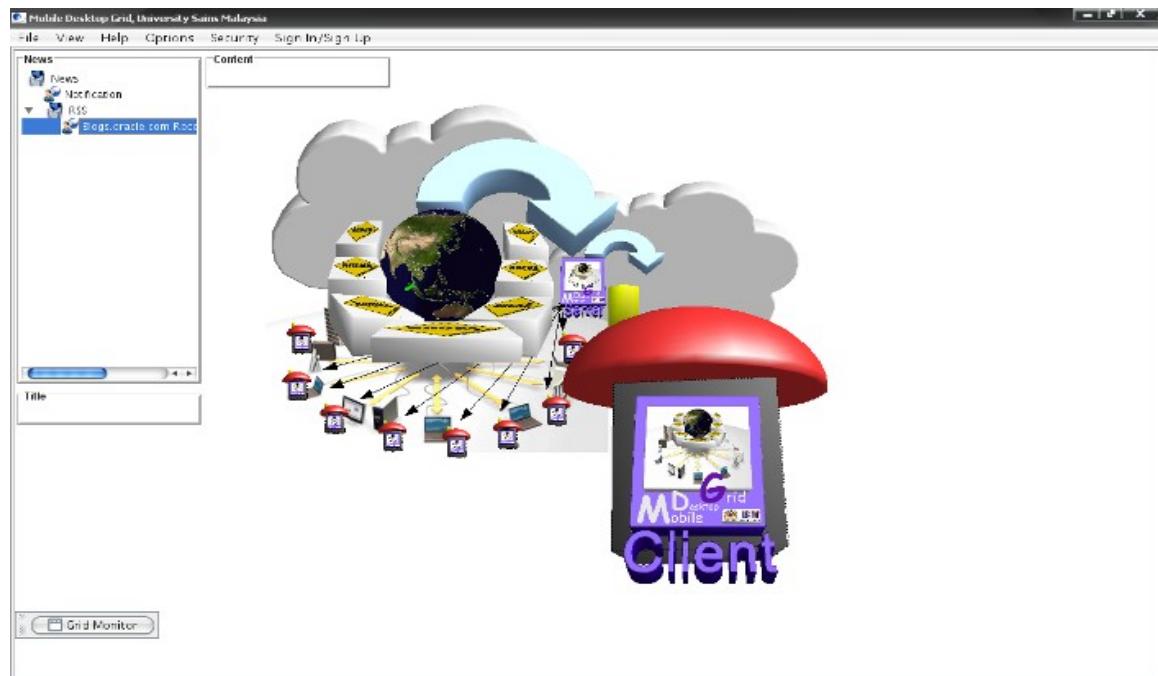
### 3.11 The steps to add new RSS feed: Oracle RSS

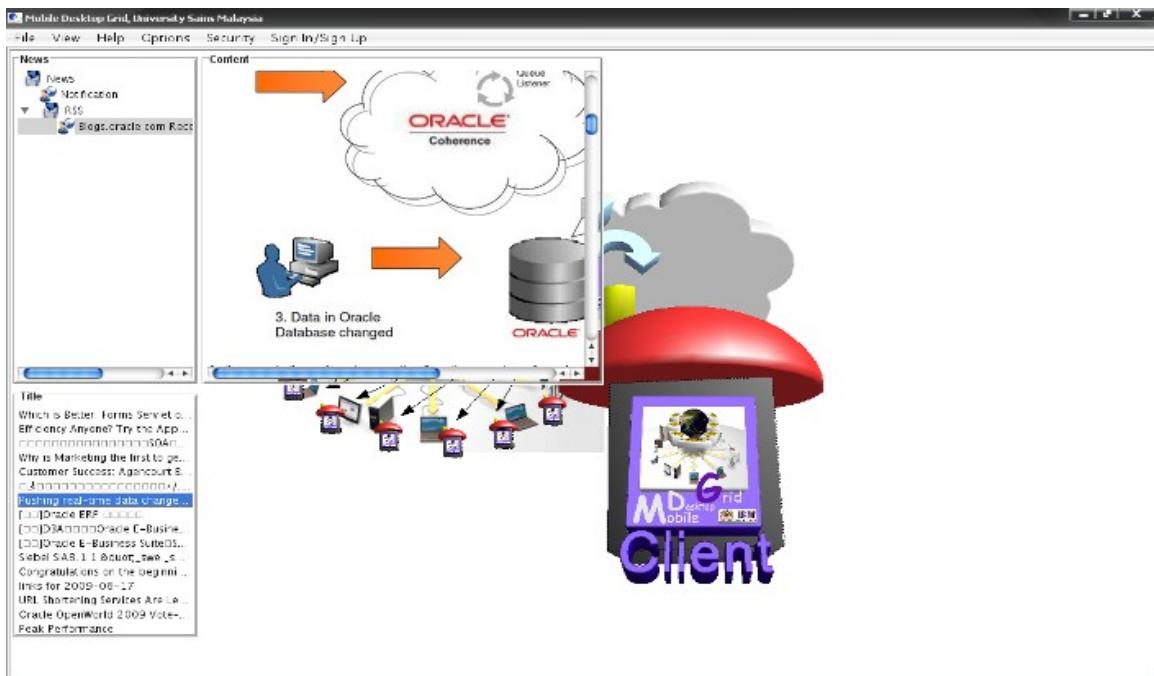


MDG Client user add new RSS feed by right-click the left window



Fill in the destination RSS Url in the prompt up window, and confirm by click on the "OK" button.

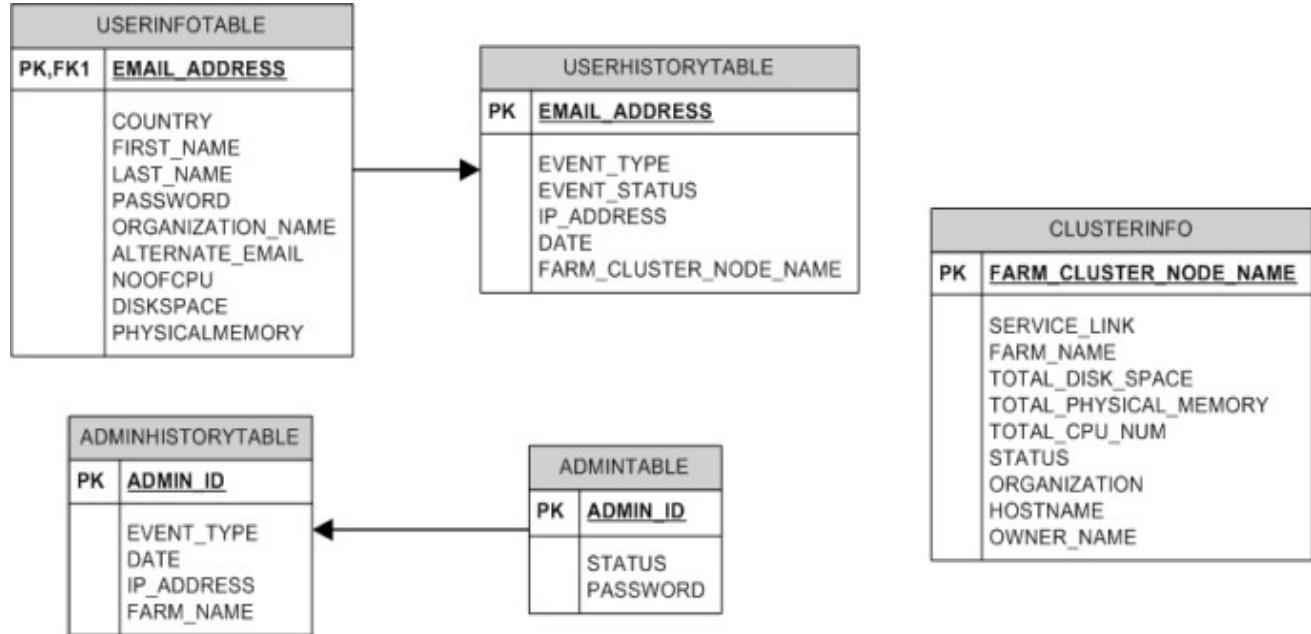




Same as the MDG Server, MDG Client will show the all the destination RSS title at the bottom window. The chosen RSS title content will show in the “Content” window. Both MDG Server and MDG Client support for English and non-English RSS contents display.

# 4.0 The database logical design

## 4.1 Entity Relationship Diagram for MDG Server

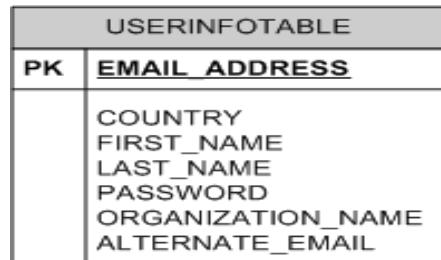


MDG Server consists of five tables in design. Table **Userinfotable** uses to store the personal information of registered user and package information. Table **Userhistorytable** uses to the users activities such as user sign in time, sign up time, and sign out time.

Whereas another two tables, which is **Admintable** and **Adminhistorytable**, are used to keep track administrator activities. Table **Admintable** uses to store administrator sign in id, password, and their status. Administrator status consists of information that is able to differential who are the master administrator in MDG solution. Table **Adminhistorytable** uses to store the administrator activities such as adding and deleting clusters, editing and deleting users.

Table **Clusterinfo** uses to store clusters information in MDG solution. These information are external reference link to MDG solution resources pool, such as MonALISA service links, MonALISA farm names, total disk space as well as all remote cluster hardware information.

## 4.2 Entity Relationship Diagram for MDG Client



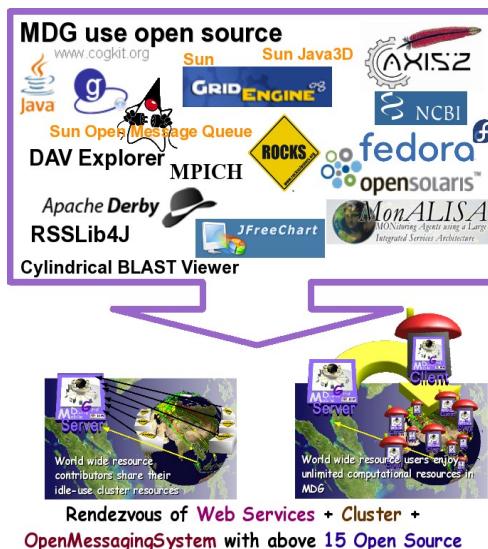
MDG Client uses table **Userinfotable** to store personal information of local users.

# 5.0 Frequent Answer and Question

## Q: MDG using any open source technologies?

A. Yes, the open source components and dependency middlewares for MDG development are

1. Apache Axis 2 Client. <http://ws.apache.org/axis2/>
2. Apache Derby. <http://db.apache.org/derby>
3. Bio Roll & Grid Roll & GridFTP & MPICH from ROCKS 4.2.1.  
<http://www.rocksclusters.org/wordpress> ,  
[http://www.globus.org/grid\\_software/data/gridftp.php](http://www.globus.org/grid_software/data/gridftp.php) ,  
<http://www.mcs.anl.gov/research/projects/mpi/mpich1/>
4. CoGKit. [http://wiki.cogkit.org/wiki/Main\\_Page](http://wiki.cogkit.org/wiki/Main_Page)
5. Cylindrical BLAST Viewer. <http://sourceforge.net/projects/cyl-viewer>
6. DAV Explorer <http://www.davexplorer.org/>
7. JFreeChart. <http://www.jfree.org/jfreechart/>
8. MonALISA web service. <http://monalisa.caltech.edu/monalisa.htm>
9. NCBI Basic Local Alignment Search Tool (BLAST).  
<http://blast.ncbi.nlm.nih.gov/Blast.cgi>
10. RSSLib4J. <http://sourceforge.net/projects/rsslib4j>
11. Sun Grid Engine (SGE) 6.0u8. <http://gridengine.sunsource.net>
12. Sun J2SE 6 and SAXParser. <http://java.sun.com/javase/6>
13. Sun Java3D. <https://java3d.dev.java.net>
14. Sun Open Message Queue (OpenMQ) & <https://mq.dev.java.net/>
15. Sun OpenSolaris. <http://opensolaris.org/os>



## **Q: What is the requirements to join our ROCKS cluster server with MDG solution?**

A: We setup our ROCKS cluster servers with following middleware before servers are able to join into MDG Server

1. Enable WSGRAM (GT4.x) with SGE (6.0u8). Detail setup refer to [http://goc.pragma-grid.net/wiki/index.php/Configure\\_GT4\\_WSGRAM\\_on\\_rocks\\_4.2.1](http://goc.pragma-grid.net/wiki/index.php/Configure_GT4_WSGRAM_on_rocks_4.2.1)
2. MonALISA web service 1.8.x, and enable its web service for remote accessible
3. Pragma user/host certificates and exchange with correct CA files. Detail setup refer to "USM (Malaysia) Cluster Users Guide" and "Preparation Guide for New Site" at URL <http://goc.pragma-grid.net/pragma-doc/default.html>
4. Bio Roll from ROCKS 4.2.1 (for running given NCBI BLAST module only)

## **Q: What is job?**

A: We define user submit work to the MDG solution as job.

## **Q: What is BLAST**

A: Please refer URL <http://blast.ncbi.nlm.nih.gov/Blast.cgi>

# 6.0 MDG Projection and Future

MDG solution has incorporated at least 15 open source middleware in development as shown in above. It has explored cluster resource management using API with web services technologies. Messaging-Oriented platform is MDG solution preferable communication approach. These MDG solution combination technologies open up opportunities for further enhancements in following areas:

1. Educational research exploration in intelligent-agents-based cluster resource allocation and fault-tolerant handling resources.
2. More integration with commercial and open source cluster servers, such as Microsoft HPC Server 2008 and Sun Solaris Cluster.
3. Internet-Service-Provider (ISP) and Grid-Service-Provider (GridSP) generate profitable business strategic by adopting charges via audit resource usage tracing.
4. Parallel and Distributed Computing learning platform for students and developers

MDG Solution Box adopts VMware open source hypervisor ESXi to host MDG solution. The entire machine consists of single Intel Core2 Quad Q6600 processor, 8GB RAM and Gigabit NIC. We show this option to individual and Small-Medium-Enterprise (SME) that with low investment they can use MDG solution as well as enjoy high computational power.



MDG solution is an one-stop solution to obtain worldwide cluster resource for computational use. With MDG solution, world wide physicist, biologist, chemist, laboratory assistant and computer scientist are able to complete their exhaustive computational jobs in a shorter period. MDG solution is a solution to aid researchers addressing global issues. Users can be from non-grid or non-cluster awareness group of people but still they are able to utilize cluster resources with our invented solution. We glad to announce that MDG proudly incorporated with at least 15 freely open source middleware.

We lead to change tomorrow with MDG solution. Public who would like to experience freely parallel and distributed processing implementation model, MDG solution is the answer. Researchers who are addressing global issue in labs have MDG solution as their daily option tools. Let's MDG facilitate you to help the Bottom Billions with your project. MDG team change tomorrow together with you.

# 7.0 MDG Appendix

## 7.1 Demo clips of work

Below are [VLC](#) video clips for version 1.1.x.x

1. [Knowledge sharing platform](#) : Viewing Really Simple Syndication (RSS) news
2. [Real-time notification platform](#) : Viewing MDG Server broadcasting notification messages
3. [Collaborative file sharing platform](#) : Share research finding universally via http and/or https protocol
4. [Resources allocation module](#) : Managing external cluster resources and users authorization
5. Modules for researcher : Modules to materialize parallel and distributed processing in exhaustive computational research tasks
  - o NCBI-BLAST in a module with Globus Toolkit (GT4) using [Fork manager](#)
  - o MPICH module with Globus Toolkit (GT4) using [Fork manager SGE manager](#)
6. Step-by-step guide for MDG Client Mini Version :
  - o Setup in [Fedora WindowsXP](#)
  - o Step-by-step using MPICH module with SGE in [Fedora WindowsXP](#)
  - o Step-by-step using Blast module with SGE in [Fedora](#)

## 7.2 Download

1. Binary Version 1.1.2.2
  - o MDG Client Mini Version [here](#) tested for both Fedora 10 and WindowsXP with service pack 2
  - o MDG Server [here](#), release together with MDG Client Mini Version, tested for both Fedora 10 and WindowsXP with service pack 2
2. The [OpenJive Regional Challenge 2009](#) competition materials
  - o Handout (A4 size) for Singapore Grand Finals [here](#),
  - o General Poster (A2 size) for Singapore Grand Finals [here](#),
  - o Module Poster (A2 size) for Singapore Grand Finals [here](#),
  - o Slide for Singapore Grand Finals on 28 July 2009 [here](#),
  - o Slide for Malaysia Finals on 02 July 2009 [here](#),
  - o Slide for Universiti Sains Malaysia (USM) Celebrating Winners on 07 August 2009 [here](#),
  - o Official USM announcement: for Malaysia Finals [here](#)
  - o Official USM Singapore photos announcement: for Grand Finals [here](#)

## 7.3 MDG Deployment in Mobile Device

MDG Client installed in handtop device. On screen showing the rendered 3D object from BLAST result.



MDG Client installed in handtop device. On screen showing the default screen after authentication.



## 7.3 MDG Achievement

### Open Jive 2009 Malaysia Finals

**Venue:** Bukit Jalil Golf & Country Club, Kuala Lumpur Malaysia

**Date:** Thursday 2 July 2009

MDG won for

#### **1. The Malaysia Champion**

1. RM3300 for MDG team
2. Trophy for each student and mentor
3. Apple iPod Nano 8Gb for each student
4. 2 book for mentor
5. Certificate for each student

#### **2. The Parasoft's Code Quality Challenge award**

1. Western Digital 1TB External Hard Disk for each student
2. Certificate for each student

#### **3. The SUN Technology award**

1. Trophy for each student and mentor
2. Certificate for each student

The stage where we celebrated our joy at Bukit Jalil!



The Malaysia Champion



The Parasoft's Code Quality Challenge award



The Technology award



## **OpenJive 2009 Singapore Grand Finals**

**Venue: Stamford Ballroom, Raffles City Convention Centre, Singapore**

**Date: Tuesday 28 July 2009**

MDG won for

### **1. The SUN Technology award**

1. SGD300 for MDG team
2. Trophy for each student
3. SUN 2GB thumb-drive for each student
4. 1 set of SUN SPOT Java Development Kit (Hardware and Software cost USD750) for MDG team
5. Certificate for each student

#### **Note:**

Sun SPOTS are much more than just an embedded microprocessor that runs Java. SUN have put together a set of technologies that SUN hope will enable the internet of things.

- Embedded Development Platform
  - Extremely flexible hardware and software package
- Easy to program - Java top to bottom
  - Java Even Device Drivers are written in Java
- Connected - Wireless Communication
  - Overlay Network - CTP, IPv6/LowPan
  - Mesh Networking - AODV, LQRP
  - Multi-hop Over the Air Programming
- Mobile
  - Built in Lithium Ion battery charged through USB
- Aware and Active
  - Able to sense and affect surroundings
- Secure
  - Built-in high grade ECC public key cryptography

Source : <http://www.sunspotworld.com/>

The stage where we celebrated our joy at Raffles City Convention Centre!



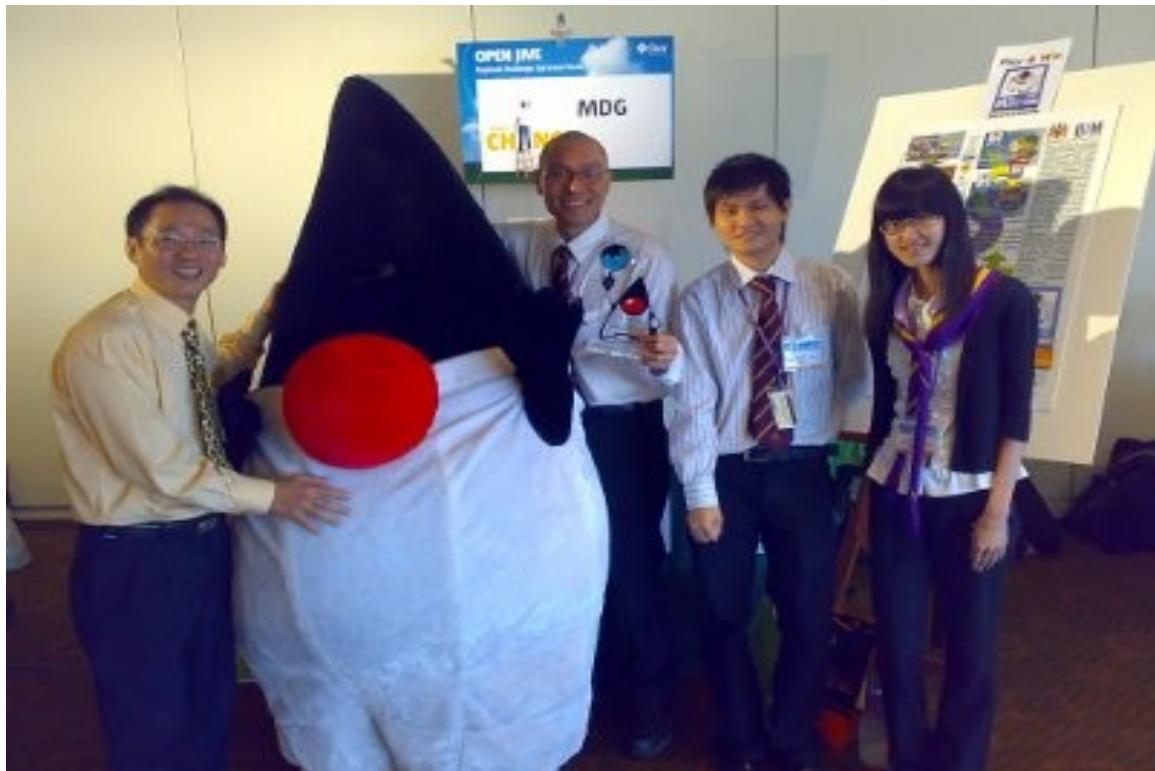
Our demo booth with MDG Solution Box (the PC next to LCD monitor).



We have a big MDG poster, and the organizer prepared a OpenJive logo poster with "MDG" name too!



Java Duke celebrated the joy with us too!



The Technology award from SUN. Change tomorrow, we (MDG team from USM) lead it!



## 7.4 Who are the MDG team

MDG solution is developed by a team of 4 persons in group from Universiti Sains Malaysia (USM @ <http://www.usm.my>). Each of us has different roles and responsibilities to build up MDG solution. Each design, analysis and development consists of 2 persons, We have this practice, even in small team like us, to ensure the final outcomes quality under monitoring. Inside the photo, the machines next to us is our daily development cluster.

