IBM

Grid Computing with the IBM Grid Toolbox

- Introduces IBM Grid Toolbox V3 for Multiplatforms V1.1
- Demonstrates installation and use
- Outlines sample services

Luis Ferreira, Lee B Wilson, Dennis Mosby, Antonio Castro, Michael Brown, Luis G Kiatake, Bradley Smoley, Jon Rossow, and Darwin Dumonceaux

Redbooks

ibm.com/redbooks

International Technical Support Organization

Grid Computing with IBM Grid Toolbox

May 2004

SG24-6332-00

Note: Before using this information and the product it supports, read the information in “Notices” on page ix.

First Edition (May 2004)

This edition applies to the IBM Grid Toolbox V3 for Multiplatforms V1.1.

Contents

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®	iSeries™	Redbooks™
AIX 5L™	LoadLeveler®	Redbooks (logo) ™
Balance®	OS/2®	Tivoli®
Cloudscape™	OS/400®	WebSphere®
@server®	pSeries®	xSeries®
IBM®	POWER™	zSeries®

The following terms are trademarks of other companies:

Intel and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Windowsis a trademark of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Preface

This IBM Redbook is designed to give a comprehensive view of the IBM Grid Toolbox V3 for Multiplatforms V1.1, which is also known as the IBM Grid Toolbox.

Paraphrasing from the product announcement letter:

> "The IBM Grid Toolbox can assist enterprises that deploy, manage, and control grid computing and developers who create products that assist in managing and deploying grids. This grid-enabling toolkit contains standardized development code, much of which was harvested from the open source community, plus an added database and run-time environment."

As the IBM Grid Toolbox is designed in a layered approach, we describe the product by introducing each underlying layer until the whole ecosystem is revealed. The product significantly leverages open standards in the grid computing world, so we show how the IBM Grid Toolbox complements and enhances these standards for the development and deployment of grid services and applications.

The book is organized into the following chapters:

Chapter 1, "Introduction" on page 1

Introduces the IBM Grid Toolbox and presents the architecture components that are part of the product.

Chapter 2, "Planning" on page 11

Discusses some of the planning considerations that should be taken into consideration when installing the IBM Grid Toolbox.

Chapter 3, "Installation and setup" on page 27

Presents the steps necessary to install and configure the IBM Grid Toolbox.

Chapter 4, "Installing related software" on page 59

Presents information about related software and the steps necessary to install and configure the related software.

Chapter 5, "Managing" on page 75

Provides information about managing grid services with the IBM Grid Toolbox and the IBM Grid Services Manager.

Chapter 6, "Samples" on page 105
Presents some of the sample services that are available with the IBM Grid Toolbox.

The team that wrote this redbook

This IBM Redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.

Luis Ferreira, also known as "Luix," is a Senior Software Engineer at IBM Corporation - International Technical Support Organization, Austin Center, working on Linux and grid computing projects. He has 20 years of experience with UNIX®-like operating systems in design, architecture, and implementation, and holds an MSc Degree in Systems Engineering from Universidade Federal do Rio de Janeiro in Brazil. Before joining the ITSO, Luis worked at Tivoli® Systems as a Certified Tivoli Consultant, at IBM Brasil as a Certified IT Specialist, and at Cobra Computadores as a kernel developer and operating systems designer.

Lee B Wilson is an IT Specialist for IBM, currently working as part of the Americas Technical Sales Support organization's Techline team covering the Latin America region. He graduated from Rensselaer Polytechnic Institute in Troy, New York, in May 2002 with a B.S. in Management concentrating on Information Systems and a minor in Computer Science.

Dennis Mosby is a Technical Sales Specialist with Techline, Americas. His responsibilities include technical support for zSeries® hardware, Z/OS, Z/VM, Linux on zSeries, and Grid computing. He joined IBM in 1983 and previously worked in Sales as a Senior Systems Engineer and in IBM Global Services as a Large Systems and Storage IT Specialist and as an IT Availability Manager.

Antonio Castro is a PMI-certified Project Manager. He joined IBM in 1996 in the Strategic Outsourcing organization, then moved to the Sydney Olympics project to manage the testing of the Game Management Systems. He previously worked for another firm in the chemical analysis group developing new analytical methods and software and supported the Doping Control Labs of the Barcelona Olympic Games and Lillehammer Winter Olympics. His main areas of interest include application testing, Linux server and client systems, and project management.

Michael Brown is a Senior Programmer and Sun-certified J2EE architect working in the IBM Linux Integration Center in Austin, Texas, where he is the leader of the team working on projects in the Americas. He has more than 25 years of experience as an application developer and enterprise architect on multiple platforms and operating systems, including UNIX, Linux, AIX®, and

OS/2®. Michael holds Bachelor of Science (Honours) and Master of Science degrees in Computer Science from the University of Western Ontario in London, Ontario, Canada. He worked on the previous GT3 Redpaper and presented GT3 programming sessions at the Colorado Software Summit.

Luis G Kiatake is a Business Developer at Citto Technology and researcher at the Integrated Systems Lab (LSI) of Polytechnic School (EP) of University of São Paulo (USP), Brazil. He got his degree and Masters in electrical engineering from EP-USP, and currently is a PhD candidate. He worked as a technical researcher at LSI from 1990-1998, developing many industrial projects in areas such as high-performance computing, computer graphics, and networks. In 1999 he was project manager for business process mapping and implementation at the National Agency for Electrical Energy (ANEEL), Brasília. From 1999-2002 he joined Getronics Brazil as project manager, then marketing and international account manager. In 2003, he joined Citto Technology, which provides security and network solutions and consulting for the largest banks in Brazil.

Bradley Smoley is a Staff Software Engineer in Rochester, Minnesota. He has more than seven years of experience in the save and restore area on iSeries™. Currently he is one of the support members on the IBM @server® OnDemand Center of Collaboration supporting the IBM Grid Toolbox. He holds a degree in business computer information systems from Saint Cloud State University in Minnesota. His areas of expertise include Backup Recovery and Media Services (BRMS), integrated file system (IFS), save and restore, program temporary fixes (PTFs), installation, and IBM Grid Toolbox. He has written articles about BRMS and general save and restore topics for iSeries Newsletter and iSeries Magazine.

Darwin Dumonceaux is a Staff Software Engineer in Rochester, Minnesota. He has more than five years of experience in the IBM @server Data Access area on iSeries. Currently he is one of the support members on the IBM @server OnDemand Center of Collaboration supporting the IBM Grid Toolbox. He holds a degree in Computer Science from Saint John's University in Minnesota. His areas of expertise include iSeries Access for Windows®, integrated file system (IFS), program temporary fixes (PTFs), IBM Grid Toolbox, and Windows 2000 Pro/Server certified.

Jon Rossow graduated from Winona State University with a BS in Management Information Systems. He works as a Staff Software Engineer at the iSeries Support Center in Rochester, Minnesota. His primary responsibility is level 2 support for IBM @server grid products, including the IBM Grid Toolbox.

Acknowledgements

Thanks to the following people for their contributions to this project:

Joanne Luedtke, Lupe Brown, Arzu Gucer, Betsy Thaggard, Wade Wallace, Chris Blatchley
International Technical Support Organization, Austin Center

Michel Considine, Vladimir Silva, Narsimha Telukuntla, Chris Barry, Joe Caldwell, Mike Williams, Jerry Moody, Ramya Nagarajan, Bob Hansen, Tom Seelbach, James Moore, Vikas Gupta, Joe Miller
Advanced Systems Infrastructure Development - OGSA/I, IBM Poughkeepsie

Anju Kumari, Anagha Gadgil, Kumaravel Ganesan, Deepak K Koshy, Kalyana C Nookala
IBM India Software Labs

Tony White
Worldwide Grid Computing Technical Sales Business Unit Executive, IBM Dallas

Cathy Parker
Systems Group ODOE Market Management, IBM Poughkeepsie

Thomas Visentin
IBM @server Grid PDT Lead, IBM Poughkeepsie

Ernest Segura
Systems Mgmt Infrastructure Software, AIX Development, IBM Austin

Phil Nelson
IBM Grid Toolbox for Linux on iSeries and pSeries®, IBM Rochester

Tony Gargya
Grid solution & Technologies, Development Lab, IBM Boeblingen, Germany

Jakob Carstensen
LTC - Advanced Linux Response Team, IBM Baltimore

Luis Tosta Sa
Mgr, Technical Sales Specialists - iSeries, pSeries, zSeries, and Storage - Techline, Latin America Techline

Scott Klasing
Consulting IT Architect, Grid Computing Practice, Architecture, and Technology COE, IBM Global Services

Special thanks to the following people:

Amy F Hieter, Frank V. Paxhia
Advanced Systems Infrastructure Development, IBM Poughkeepsie

Andy Gangone
IBM Certified Specialist -- Grid Computing Technical Sales, GRID Computing
WW Technical Support Marketing Leader, IBM Systems & Technology Group

Become a published author

Join us for a two- to six-week residency program. Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners, and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks™ to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an e-mail to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM® Corporation, International Technical Support Organization
  Dept. JN9B Building 003 Internal Zip 2834
  11400 Burnet Road
  Austin, Texas 78758-3493

1

Introduction

This chapter introduces the IBM Grid Toolbox and presents the architecture components that are part of the product.

1.1 Introduction to Grid Computing

As industry, research, and academia march forward, the computer application load has followed a very steep curve, surpassing the capabilities of single computers. Supercomputers certainly have a role to play, but they are expensive and tend to be very specialized in the types of problems they are built to solve. The shift from using a single computer for problem solving to using multiple computers is a very dramatic paradigm change.

Staff in computationally heavy areas of interest such as seismic analysis, particle physics, financial modeling, biomedical research, and weather forecasting had to invent new technologies to get their workload processed. Unfortunately they usually ended up with solutions that only solved a specific problem in a very specific hardware or software environment. Clearly something had to be done.

Much of the work that had been done for grid has been targeted at the high-performance distributed computing environment, very specialized and proprietary, which has not helped in building a generic and heterogeneous distributed computing implementations. As grid becomes more important to commercial computing, the need for standardization and open implementation is crucial. Therefore, the Global Grid Forum (GGF) was formed to design and develop a set of grid computing standards that could be used by any industry, in a geographically dispersed, heterogeneous hardware and software environment. Standardization needs were identified in such areas as overall architecture, programming model, network model, data model, security, and management.

The implementation of grid standards has enabled developers and administrators to focus on solving their domain-specific problems, using the grid infrastructure as they would any other portable toolkit. This reduces overall project cost and significantly reduces time-to-deployment.

1.2 Open Standards

To understand the role played by the IBM Grid Toolbox, we have to take a few steps back and discuss some of the fundamental components that the product depends on.

These topics are discussed in depth in other documents and Redbooks, but we summarize them here for those who only need to understand the background at the high levels.

1.2.1 Web services

A Web service can be used to build an application identified by a URL (Uniform Resource Locator, previously called Universal Resource Locator). Web services' interfaces and bindings can be defined, described, and discovered by XML (Extensible Markup Language) artifacts, and support direct interactions with other software applications using XML-based messages via Internet-based protocols. In simpler terms, a Web service is an application that is called using an address similar to the way we address things on the Web, passing parameters in XML format.

Using XML, Web Services Description Language (WSDL) describes network services as a collection of endpoints that operate with messages containing either procedure-oriented or document-oriented information. To define an endpoint, operations and messages are described abstractly and subsequently are bound to an established network protocol. Similarly described endpoints are grouped into abstract endpoints, commonly referred to as services. The key functionality of WSDL is to allow the description of endpoints and their messages regardless of what messaging formats or protocols are used to communicate.

For more information about WSDL, refer to:

http://www.w3.org/TR/wsdl

1.2.2 Open Grid Services Architecture (OGSA)

As mentioned above, the Global Grid Forum was formed to drive standardization in grid computing. From http://www.globus.org/ogsa:

> 'The GGF developed the Open Grid Services Architecture (OGSA) that represents an evolution toward a grid system architecture based on Web services concepts and technologies."

The key item to note here is that OGSA is an architecture based on the existing Web services standards, and also is being used to define many grid standards. Web services standards include XML, SOAP, and WSDL.

1.2.3 Open Grid Services Infrastructure (OGSI)

The Global Grid Forum also promotes the development for standardizing the infrastructure in grid computing. From http://www.globus.org/ogsa:

> "OGSI refers to the base infrastructure on which OGSA is built. At its core is the Grid Service Specification, which defines the standard interfaces and behaviors of a grid service, building on a Web services base."

OGSI provides an infrastructure upon which to define and build OGSA standards. It furnishes technical specifications for the implementation of each factory in OGSA, using grid services to define each interface. The specification is based on a range of Web services standards, with certain extensions to WSDL and XML needed for grid services.

OGSI defines such items as stateful Web services, inheritance of Web service interfaces, asynchronous notification, references to instances of services, collections of service instances, and service state data.

The Web services world has recognized the value in the enhancements made by OGSA/OGSI and work is underway to include some of these enhancements into Web services itself. The line between Web services and grid services is expected to blur as the feature sets converge.

1.2.4 Grid services

Grid services technology is based on SOA (Service Oriented Architecture) that defines an architectural approach where an application is composed of independent and cooperating components called services. These services are building blocks that utilize the component object model for creating open distributed systems and enable companies and individuals to quickly make their digital assets available worldwide.

Additional mechanisms for creating and managing grid services are available when developing a new service for deployment within an OGSA-compliant system. At a glance, these mechanisms are:

Factory A special class responsible for dynamically creating grid service instances - grid service code up and running, waiting for requests.

Registry The interface that enables a set of grid service instances to register their GSHs (Grid Service Handle) into a registry service, to allow for discovery of services in that set.

Discovery The interface that enables clients of the grid services to obtain information about the provided services.

Life cycle Refers to the states between the creation and destruction of grid service instances.

Service data A structured collection of information that is associated with an instance of a grid service.

Notification A mechanism by which a party sends (notification source) a change of state notice to the party who has requested (notification sink) to be notified.

Reliable invocation Techniques that ensure the accuracy of method invocations in case multiple instances have been created and there are redundant grid services in the space.

The important thing to remember is that the only contact between a Grid service and its users (applications running on the grid) is the service interface. These service interfaces are defined by the existing Web Services Description Language (WSDL). Several enhancements to WSDL have been identified for OGSI requirements and are currently being added to the WSDL standard. For more information about WSDL, visit

http://www.w3.org/

1.2.5 The Globus Alliance

As stated on the Globus Web site at http://www.globus.org/, the Globus Alliance is:

> "A research and development project focused on enabling the application of grid concepts to scientific and engineering computing."

The Globus team developed Globus Toolkit 1.0 then Globus Toolkit 2.0 as open source, traditional C-based libraries based on a Globus-defined design and set of protocols.

Globus Toolkit 3.0 represents a significant change in the basis of the Globus project. It is a reference, and open source, implementation of the OGSA architecture, which included an implementation of the OGSI 1.0 specification. Now you can see how OGSA, OGSI, and Globus all tie together. In an attempt to preserve some amount of backward compatibility, Globus Toolkit 3.0 provides updates to most of the functions of Globus Toolkit 2.0. Globus Toolkit 3.0 is a large software framework, including security services, system-level services, base services, user-defined services, options of hosting environments, and a container (OGSI runtime).

1.2.6 Future directions on grid services

Since releasing Globus Toolkit 3.0 in July 2003, the GGF and the Globus Alliance have been working closely to define enhancements to the standards.

In January 2004, they presented the WS-Resource Framework (WSRF), an open framework for modeling and accessing stateful resources using Web services. WSRF defines where Web service standards are evolving to meet grid services elements and requirements (Figure 1-1 on page 6). The specification consists of separate specifications, each one focusing on a specific area.

The document *From Open Grid Services Infrastructure to WS-Resource Framework: Refactoring & Evolution Version 1.1* introduces the following normative WSRF specifications:

WS-ResourceProperties	Specifies stateful Web services
WS-ResourceLifetime	Specifies Web service lifecycle
WS-RenewableReferences	Specifies Web service endpoint reference and addressing
WS-ServiceGroup	Specifies the creation and use of groups of Web services
WS-BaseFault	Specifies fault type used for fault error reporting
WS-Notification	Specifies the notification framework

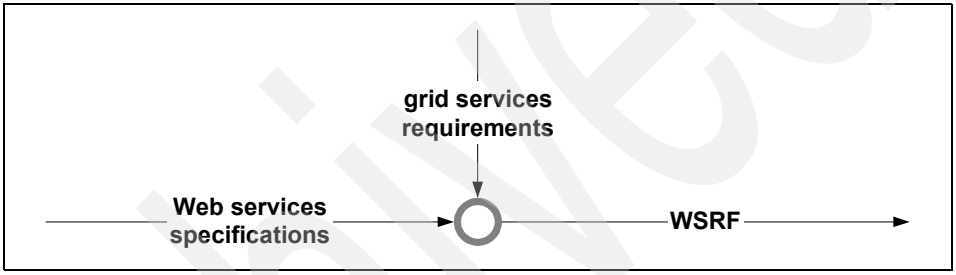


Figure 1-1 Grid and Web service convergence

For more information about OGSA-WG (GGF Workgroup of OGSA) and WSRF, refer to the following Web sites:

http://www.globus.org/
http://www.ggf.org/
http://forge.gridforum.org/
http://www.oasis-open.org/
http://www.globusworld.org/
http://www.ibm.com/developerworks/library/ws-resource/

1.3 Introduction to the IBM Grid Toolbox

IBM has been involved in the Globus project for several years. After identifying some gaps in the feature set of Globus Toolkit 2.4, a package called the IBM Grid Toolbox was developed and provided on an as-is basis on the IBM AlphaWorks Web site:

http://www.alphaworks.ibm.com

As the Globus project evolved, the IBM Grid Toolbox team determined that due to demand from IBM enterprise customers, there was an ongoing need for an IBM product that enhanced Globus.

This product became the IBM Grid Toolbox V3 for Multiplatforms V1.1, the subject of this book. It provides simple installation and integration of middleware, and adds significant value for the two major grid roles:

Grid Developer Tools to develop and test grid services and grid applications
Grid Administrator Tools to host grid services and grid applications

The IBM Grid Toolbox V3 for Multiplatforms V1.1 implements the OGSI standards and provides the tools to build, develop, deploy, and manage grid services. The IBM Grid Toolbox consists of the following:

- A hosting environment capable of running grid services and collaborating with other grid participants in running large tasks.
- A set of tools to manage, monitor, and administer grid services and the grid hosting environment, including a Web-based interface called IBM Grid Services Manager.
- A set of APIs and development tools to create and deploy new grid services and grid applications.
- A set of tools to simplify the installation process and the integration of the embedded middleware, such as: IBM WebSphere® Application Server - Express V5.0.2.

1.3.1 IBM Grid Toolbox goals

The primary goal of the IBM Grid Toolbox is to provide a common infrastructure for Grid Computing, autonomic management, and On Demand Solutions to the IT industry.

Target Audience

The target audience is enterprise customers who typically have much invested in existing applications, are hindered by a business problem, and have identified grid as a possible means for problem resolution or benefit. IBM has worked with these customers for decades and understands their unique and comprehensive requirements.

Development of grid services and applications is done at a very technically demanding level and requires much effort. The IBM Grid Toolbox is positioned for customers who have already decided to implement a grid, have studied and are comfortable with Globus Toolkit 3.0, but are looking for a supported, licensed

product to match the support model of their other hardware and software choices.

Enterprise value

The IBM Grid Toolbox brings value over and above what is available from the open source community:

- Accelerates grid utilization as it provides a more complete development and administration environment than GT3 alone.
- Lowers the risk inherent in developing with GT3 alone, as it is an IBM-supported product. (This support must be purchased.)
- Enables enterprises to leverage the heterogeneous nature of their IT infrastructure.
- Complements IBM grid industry offerings: http://www.ibm.com/grid/solutions/index.shtml
- Installs around the network easily with GUI or automated tools.
- Hosts components and common services within embedded version of the IBM WebSphere Application Server -Express V5.0.2.
- Scales with minimal incremental overhead.
- Coexists with other instances of WebSphere Application Server and other Web services products.
- Interoperates with other standards-compliant implementations.

1.3.2 Platform support and availability

This book is aligned with the IBM Grid Toolbox products that were first announced on December 16, 2003, by Announcement Letters 203-352, 203-348, and 203-349, and followed by additional platforms that were announced on March 23, 2004, by Announcement Letters 204-043 and 204-045.

The IBM Grid Toolbox is supported in the following environments:

- IBM xSeries® servers running Red Hat Enterprise Linux AS 2.1 or SUSE Linux Enterprise Server 8
- IBM pSeries servers running SUSE Linux Enterprise Server 8
- IBM pSeries servers running AIX 5L™ for POWER™ V5.2.0.10
- IBM iSeries servers running SUSE Linux Enterprise Server 8
- IBM zSeries servers running SUSE Linux Enterprise Server 8 (31-bit version only)

At the time of this writing, the product and any fixes can be downloaded from the IBM Grid Toolbox home page at:

http://www.ibm.com/grid/solutions/grid_toolbox.shtml

The IBM Grid Toolbox can also be obtained on CD-ROM by contacting your IBM representative or calling your local IBM Call Center. For customers in the Americas, the telephone number is 800-IBM-CALL.

1.4 An overview of IBM Grid Toolbox components

The IBM Grid Toolbox is a collection of components that includes the Globus Toolkit 3.0. The following overview of components suggests the scope of features that the IBM Grid Toolbox brings to grid developers and administrators:

- Single installation process: The IBM Grid Toolbox ships both wizard-based and silent installation methods, which provide simple one-off or bulk installation around a network.
- Grid services runtime based on the OGSI specification: An embedded version of the IBM WebSphere Application Server - Express V5.0.2 is provided as the grid services container. It replaces the stand-alone container that is provided by GT3 for commercial-grade support.
- Management interface: A browser-based interface called the IBM Grid Services Manager provides easy grid-wide management for administrators.
- An enhanced certificate-based grid security infrastructure.
- Configuration and administration commands: Command-line based scripts for common actions are provided for administrators.
- Development tools: Enhancements and tools are provided that assist with building, packaging, and deploying grid services and applications.
- Additional and enhanced grid services: IBM provides additional functionality including discovery via service group, policy management, and Common Management Models (CMM) Services.

The IBM Grid Toolbox includes a list of base grid services that can be deployed with the installation or deployed or undeployed separately. The following base grid services are included:

- Program Management Services
- Information Services
- Data Management Services
- Common Management Models (CMM) Services
- Policy Services
- Service Group Services

For more information about these services (as presented in Figure 1-2) refer to Section 2.3, “Planning for installation” on page 23.

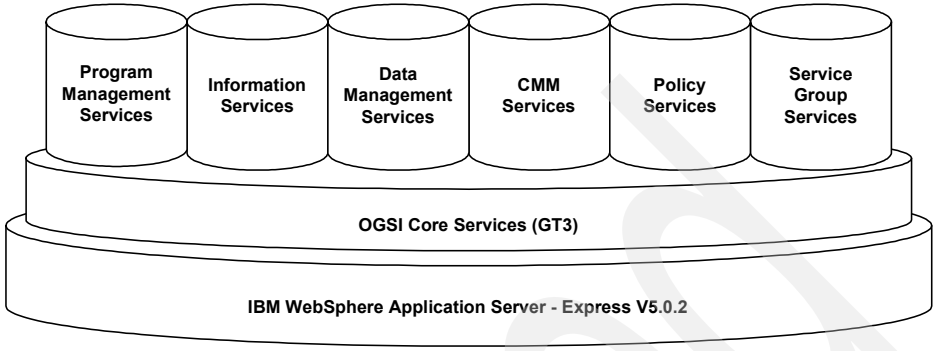


Figure 1-2 Macro components

1.4.1 Hosting environment

A hosting environment is where the application runs. It is the server environment, such as an operating system and the grid application server. The IBM Toolbox is built with an embedded version of the IBM WebSphere Application Server - Express V5.0.2. Although it functions nearly the same as a full-product version of WebSphere Application Server, it is used exclusively for the IBM Grid Toolbox.

A hosting environment can be seen essentially as a grid container running inside of a Java™ engine (Web container) or an EJB application server (WebSphere Application Server). Figure 1-3 gives an example of this hosting environment.

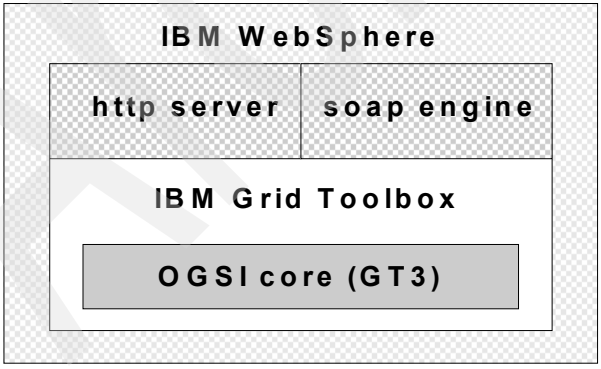


Figure 1-3 Hosting environment

For more information about EJB application servers, visit:

http://www.ibm.com/websphere/

2

Planning

This chapter discusses some of the planning considerations that should be taken into consideration when installing the IBM Grid Toolbox. Specifically, the following topics are discussed:

- IBM Grid Toolbox packaging
- IBM Grid Toolbox requirements
- Planning for installation
- Planning for security
- Planning for related software
- Planning a production environment
- Planning a development environment

2.1 IBM Grid Toolbox packaging

The IBM Grid Toolbox comes with a hosting environment that is capable of running grid services and sharing them with other grid participants. It includes:

- A set of tools to manage and administer grid services and the grid hosting environment, including a Web-based interface, the IBM Grid Services Manager.
- A set of APIs and development tools to create and deploy new grid services and grid applications.

You can access product information at:

http://www.ibm.com/grid/solutions/grid_toolbox.shtml

The IBM Grid Toolbox is available on a variety of IBM @server platforms and operating systems. Table 2-1 summarizes the available distributions of the IBM Grid Toolbox.

Table 2-1 Available IBM Grid Toolbox distributions

	Red Hat AS 2.1	SLES 8.0	AIX	OS/400
iSeries		yes		yes
pSeries		yes	yes	
xSeries	yes	yes		
zSeries		yes		

You can obtain the IBM Grid Toolbox code for each release by following any of these links or by contacting your local IBM representative.

- IBM AIX operating system on IBM @server pSeries

  Order product number 5764-G22 from you local pSeries representative.
- Linux operating system on IBM @server xSeries

  Access the IBM Grid Toolbox code at:

  http://www.ibm.com/grid/solutions/grid_toolbox.shtml
- Linux operating system on IBM @server iSeries, zSeries, and pSeries

  Learn how to obtain the IBM Grid Toolbox code at:

  http://www.ibm.com/grid/solutions/grid_toolbox.shtml

2.2 IBM Grid Toolbox requirements

To successfully install the IBM Grid Toolbox, your system must meet the following requirements, based on hardware and operating systems.

2.2.1 iSeries running Linux

For proper installation of the IBM Grid Toolbox, your system must meet the following system requirements.

Hardware

The minimum hardware requirements for the IBM Grid Toolbox on an iSeries server running Linux are:

- 512 MB of RAM
- 500 MB free space in the /opt directory for the product installation

Software

The minimum software requirements for the IBM Grid Toolbox on an iSeries server running Linux are:

- SUSE Linux Enterprise Server 8.0 - Powered by UnitedLinux V1.0.
- The Linux distribution package must include a version of Perl.

2.2.2 iSeries running OS/400

For proper installation of the IBM Grid Toolbox, your system must meet the following system requirements.

Hardware

For hardware requirements information, visit the eServer Information Center at:

http://publib.boulder.ibm.com/eserver/

Software

The IBM Grid Toolbox for OS/400 requires OS/400 V5R3, with the following software installed:

- IBM WebSphere Application Server – Express 5.0[1]
- OS/400 Host Servers[2]
- TCP/IP Utilities
- IBM Developer Kit for Java Version 1.3

[1] Order and install the latest available PTFs for IBM WebSphere Application Server - Express 5.0.
[2] The host servers must be started with the STRHOSTSVR command before using the installation program to install the IBM Grid Toolbox.

- OS/400 Qshell
- Portable Application Solutions Environment

The following software is required to create a certificate authority and to sign certificate requests:

- OS/400 Digital Certificate Manager
- Crypto Access Provider 128-bit
- IBM HTTP Server[3]

2.2.3 pSeries running Linux

For proper installation of the IBM Grid Toolbox, your system must meet the following system requirements.

Hardware

The minimum hardware requirements for the IBM Grid Toolbox on a pSeries server running Linux are:

- 512 MB of RAM
- 500 MB free space in the /opt directory for the product installation

Software

The minimum software requirements for the IBM Grid Toolbox on an pSeries server running Linux are:

- SUSE Linux Enterprise Server 8.0 - Powered by UnitedLinux V1.0.
- The Linux distribution package must include a version of Perl.

2.2.4 xSeries running Linux

For proper installation of the IBM Grid Toolbox, your system must meet the following system requirements.

Hardware

The minimum hardware requirements for the IBM Grid Toolbox on an xSeries server running Linux are:

- 2.0 GHz processor speed
- 1 GB of RAM
- 500 MB free space in the /opt directory for the product installation
- 250 MB for the installation files

[3] Required to access the user interface for Digital Certificate Manager, which is accessed through a browser and provided via HTTP.

Software

The minimum software requirements for the IBM Grid Toolbox on an xSeries server running Linux are:

- Red Hat Enterprise Linux Advanced Server for Intel® 2.1 or SUSE Linux Enterprise Server 8.0 - Powered by United Linux V1.0
- The Linux distribution package must include a version of Perl
- Linux file system must be set up as follows:

/boot	approximately 100 MB
swap	amount of physical memory installed
/	remaining space on your hard drive

Important: The IBM Grid Toolbox installs into the /opt/IBMGrid directory. If your disk space is partitioned by directory, at least 500 MB of disk space must be associated with /opt for the Toolbox to install.

2.2.5 zSeries running Linux

For proper installation of the IBM Grid Toolbox, your system must meet the following system requirements.

Hardware

The minimum hardware requirements for the IBM Grid Toolbox on a zSeries server running Linux are:

- 512 MB of RAM
- 2 GB of disk space
- 250 MB for the installation files.
- 500 MB in the /opt directory for the product installation.
- 400 MB in the /tmp directory

Software

The minimum software requirements for the IBM Grid Toolbox on a zSeries server running Linux are:

- SUSE Linux Enterprise Server 8.0 (31-bit) Service Pack 2. The Linux distribution package must include a version of Perl.
- Ensure that your Linux file system has the following space allowances:

/boot	approximately 100 MB
swap	amount of physical memory installed
/	remaining space on your hard drive

2.2.6 pSeries running AIX

For proper installation of the IBM Grid Toolbox, your system must meet the following system requirements.

Hardware

The recommended hardware requirements for the IBM Grid Toolbox on a pSeries server running AIX. are:

- 1 GB of RAM
- 18 GB hard disk space
- Network connections
- Graphical console

Note: Your system must also meet the minimum hardware requirements for installing and running AIX 5L Version 5.2.

Software

The minimum software requirements for the IBM Grid Toolbox on a pSeries server running AIX are:

- AIX 5L Version 5.2 with the 5200-01 recommended maintenance package.
- AIX file system must be set up as follows:
  - /opt/IBMGrid (or whatever the directory the IBM Grid Toolbox will be installed in) must have 967 MB associated with it.
  - /tmp must have 440 MB associated with it (200 MB extra if you set an alternate temporary space for the extraction process during the installation).

    Note: Any failures caused by insufficient space during the installation process are logged in the /opt/IBMGrid/log.txt file and the /opt/IBMGrid/installp_isje.log file.

- Mozilla 1.4.1, which can be downloaded from the AIX Web site at http://www.ibm.com/server/aix/browers
- The following rpms:
  - Gtk+1.2.10-3
  - Glib 1.2.10-2, which can be downloaded from http://www.ibm.com/servers/aix/products/aixos/linux/altlic.html

If you intend to install the IBM Grid Toolbox with the install shield and export the display to a server without a graphical console attached, you must have the following X11 software:

- X11.adt.lib
- X11.base.rte
- X11.base.lib
- X11.motif.lib
- X11.motif.mwm

Attention: IBM JDK 1.3.1 is required for the installation of the IBM Grid Toolbox on all platforms.

2.3 Planning for installation

The IBM Grid Toolbox is used to implement the OGSI standards to be able to build a grid, and then develop, deploy, and manage grid services. The toolbox can be used to assist in the following areas:

- Infrastructure optimization
  - Consolidate workload management.
  - Provide capacity for high-demand applications.
  - Reduce cycle times.
- Increase access to data and collaboration
  - Federate data and distribute it globally.
  - Support large multi-disciplinary collaboration.
  - Enable collaboration across organizations and among businesses.
- Resilient, highly available infrastructure
  - Balance workloads.
  - Foster business community.
  - Enable recovery and failure.

Two types of grid services are included in the IBM Grid Toolbox. One type is *core* grid services, which you cannot deploy and undeploy separately from the installation of the IBM Grid Toolbox. The other type is *base* grid services.

The base grid services can be deployed or undeployed separately at any given time. Base grid services that are available with the IBM Grid Toolbox are:

- Information Services
- Data Management Services

- Program Management Services
- Common Management Model (CMM) Services
- Policy Services
- IBM Service Group Services

The following sections describe these six services. Table 2-2 provides an overview of which services you might want to install, depending on whether you are administrating a grid instance or developing grid services.

Table 2-2 Overview of installed services and related software

Package	Development environment	Production environment
Information Services	yes	yes
Data Management Services	yes	yes
Program Management Services	yes	yes
Common Management Model (CMM) Services	yes	yes
Policy Services	yes	yes
IBM Service Group Services	yes	yes
Apache Ant	yes	
GridFTP	yes	yes
Pegasus	yes	yes
SBLIM	yes	yes

2.3.1 Information Services

Grid information services provide information about grid resources for use in resource discovery, selection, and optimization, and they maintain knowledge about resource availability, capacity, and current utilization. This information is critical to the operation of the grid and development of applications. Within any grid, resources fluctuate, depending on their availability to process and share data. As resources become free within the grid, they can update their status within the grid information services. The client, broker, and grid resource manager use this information to make informed decisions about resource assignments.

In the Open Grid Service Architecture (OGSA), everything is represented as a grid service. These grid services can be a static set of persistent services as well as transient services. There may be one or more instances of a particular grid service. Each grid service expresses its state in a standardized way as Service Data Elements (SDEs). Grid information services provides the functionality within which service data can be collected, aggregated, and queried; data feeds can be monitored; and service data can be created dynamically on demand. Grid information services is therefore a broad framework that includes anything that generates, registers, indexes, aggregates, subscribes, monitors, queries, or displays service data in some way.

Within the Information Services is a subset called Index Services. Index Services is used to provide an interface for operations that access, aggregate, generate, and query the service data. The Index Service provides:

- An interface for connecting external Service Data Providers programs to service instances.
- A generic framework for aggregation of service data from other services.
- A Service Group of grid services.
- A dynamic data-generating and indexing node.

2.3.2 Data Management Services

Data Management Services enables data transfer throughout the grid. This service gives grid applications the ability to move data from one node to another; without Data Management Services your data cannot move from node to node. If you need Data Management Services, a Grid File Transfer Protocol (GridFTP) server should be installed. For more information about GridFTP, refer to 4.3, “GridFTP” on page 66.

For more information about Data Management Services, refer to this Globus Web page:

http://www-unix.globus.org/developer/data-management.html

2.3.3 Program Management Services

Program management grid services is the IBM implementation of resource management grid services in the IBM Grid Toolbox V3 for Multiplatforms V1.1. Resource management grid services simplify the use of remote systems by providing a standard interface for requesting and using remote system resources for the submission and control of jobs. This implementation is typically used to support distributed computing applications.

For more information about Program Management Services, refer to this Globus Web page:

http://www-unix.globus.org/developer/resource-management.html

2.3.4 Common Management Model (CMM) Services

Important: The Common Management Model (CMM), was previously referred to as the Common Resource Model (CRM). Many files in the IBM Grid Toolbox still include the old acronym.

The CMM that is included in the IBM Grid Toolbox provides the required infrastructure for representing an instrumented resource as a grid service to enable it to be queried and managed. There are many different methods for gathering information about system resources so that they can be managed over a network. This includes industry standards such as the Common Information Model (CIM). As part of the IBM Grid Toolbox, CMM provides the following:

- An implementation of identified manageable resources as grid services
- A resource adapter compliant with Java Connector Architecture (JCA) specification V1.0 that facilitates communications between the grid service representation of a manageable resource and the CIM instrumentation representing the resource
- A mapping framework that maps the service data and grid service operations to CIM properties and methods
- A sample application that shows one possible use of the core CMM Services in the IBM Grid Toolbox
- The program artifacts used by the sample application to work the with CMM Service

The IBM Grid Toolbox was tested with OpenPegasus as a CIMOM server. OpenPegasus is an open-source implementation of DMTF CIM and WBEM standards. DMTF (Distributed Management Task Force) CIM is a standard for exchanging management information easily and platform-independently through a common protocol. Pegasus provides insight into what processes are being used on a particular system or set of systems, to aid in deciding where to pass the workload. Pegasus itself does not come with the IBM Grid Toolbox; it is only used in a sample.

You can read more about OpenPegasus at:

http://www.openpegasus.org/

For a more in-depth view of DMTF and the standards within, see:

http://www.dmtf.org/about

Note: To use this service, you need additional software that does not come with the IBM Grid Toolbox. Refer to 4.2, “Pegasus and SBLIM” on page 61 for installing additional software.

2.3.5 Policy Services

The policy services in the IBM Grid Toolbox enable administrators to define a set of business goals and to enforce a set of rules that allow their grid to meet those goals. In the IBM Grid Toolbox, a policy identifies the desired outcome for the interactions between different elements in the grid environment. For example, a policy that is used for resource allocation might place limits on how much of a network’s total capacity a certain application may consume. (For example, multicast UDP traffic may not take up more than 10% of the total network capacity.) Other policies may enforce security restrictions by granting or denying access to certain resources to sets of users, processes, or applications. The policy framework in the IBM Grid Toolbox is based on the IETF policy–based network architecture. This includes a core set of services for defining, managing, and applying policies on a grid.

Within the IBM Grid Toolbox Policy Services, there are three major components that are not dependent on any specific discipline:

- Policy Service Manager (PSM)

  The Policy Service Manager provides an interface to management applications that use policies on the grid and:

  - Provides an interface to update, remove, retrieve, and deploy policies.
  - Stores information about the policy system.
  - Includes notification services that enable Policy Services Agents or certain Policy Enforcement Points to receive updates when a policy is changed.
  - Validates policy documents that were received from a management application against a schema stored in the policy repository.

- Policy Service Agent (PSA)

  This agent monitors the policy system for new or changed policies. The PSA notifies the appropriate Policy Enforcement Points of the new or updated policies so that they may be used.

- Policy Repository

  This is where the policies for the administrative domain are stored.

There are also some discipline-specific components of Policy Services. These components are dependent on each unique requirement for the specific discipline:

- Policy Enforcement Point (PEP)

  Services that receive policies from a PSA and apply the necessary configuration settings on the specific resource.

- Policy Transformation Services

  Translates the discipline level policies into a device specific format for a specific PEP.

Within the IBM Grid Toolbox, we also define three different levels of policies:

- Business-level policies

  Defines the high-level user goals for the system.

- Discipline-level policies

  Defines a schema for policy definitions within a specific discipline.

- Device-level policies

  System-specific configuration parameters in a format that each PEP understands.

2.3.6 IBM Service Group Services

A service group is a grid service that maintains information about a group of grid services. Grid applications can query the service group for information about the grid services that are associated with it, including a description of the included services and references to the services themselves. For instance, grid services might be included in a service group to categorize them for application-specific management functions, especially service discovery based on some specified criteria. The service group implementation in the IBM Grid Toolbox includes extensions to the service group services in the core Globus Toolkit. Service group provides a framework that enables grid applications to categorize (group) grid services and later execute queries on the group of services to find specific types of grid services.

Grid application writers might use the service group framework to implement a registry service for their grid. To accomplish this, the service group services provide a set of operations that enable grid services to become members of the service group. The service group service maintains a list of the member services, and some information about them, in its service data. To group the services into related types, the service group defines a set of rules (membershipContentRule) in its service data that defines which grid services can become members.

When a grid service is added to a service group, several operations are performed:

- Checks the grid service to verify that it matches its rules defining the types of services contained in the service group.
- Creates a new grid service to maintain information about the member grid service.
- The service data in the service group service is updated to add an entry for the member grid service.

To be able to support the required operations, the service group service has three port types:

- ServiceGroup
- ServiceGroupEntry
- ServiceGroupRegistration

Note: To see how these services work, refer to 6.5, “Service group sample” on page 121.

For more information about each of these services, refer to the IBM Grid Toolbox V3 for Multiplatforms manual at:

http://publib.boulder.ibm.com/eserver/v1r1/en_US/info/ogsainfo/eicab.pdf

2.4 Planning for security

Security is an integral part of planning and deploying a grid. The IBM Grid Toolbox uses the Grid Security Infrastructure (GSI) to facilitate secure communications over open networks.

Many components are utilized to keep your grid secure. From certificates (x.509) to credential mapping, IBM Grid Toolbox makes security common, robust, and easy to implement. As always, good security is achieved not through technology alone, but through effective use of technology coupled with a well-planned security model. These security components should be considered as part of every grid environment:

- Grid Security Infrastructure (GSI)
- Certificates
- Certificate authority
- Authorization and authentication
- Proxies and delegation
- Grid map file authorization
- Message level security
- OGSI container security

This section provides an overview of security topics to consider when planning a grid. For more information about these topics, refer to the manual included with the IBM Grid Toolbox.

2.4.1 Certificate authority

Security within a grid is established through the use of host and user-level certificates, which are mapped to specific users on the local host. In order to obtain these certificates, host and user certificate requests are made. These requests are sent to a trusted certificate authority (CA), who signs them and returns them to the requestor.

An integral component of these certificates is the distinguished name (DN). The DN is a unique string that identifies a user on the grid. When planning a new grid installation you should consider what naming convention will be used for these DNs. Most of the information in a certificate's DN will be provided by the CA; however, there is a segment of the DN referred to as a Common Name (CN). The CN should be unique for each user on a grid.

The IBM Grid Toolbox does not come with a CA. You must have access to your own CA for the installation. When planning the security of a grid, it is recommended that you consult with the security personnel in your organization as you might have to adhere to some additional practices.

For more information about setting up grid security, see 3.3.5, "Securing the grid" on page 47. For more about the CA, refer to Appendix E, "Certificate authority" on page 225.

2.4.2 Grid map file

After certificates have been obtained, a grid instance must know about them in order to grant access to its resources. This is done with a grid map file, which maps a particular DN to a local user. When mapped, that DN can access resources on a given host as if he were the local user. This enables you to give certain grid users different permission levels based on the users they are mapped to. To deny access to a DN, simply omit it from the grid map file.

For more information about setting up a grid map file, refer to"Adding a grid mapfile entry" on page 53.

2.5 Planning for related software

In addition to the principal IBM Grid Toolbox product, you may need to install additional software components to enhance the functionality of the IBM Grid Toolbox. Related software includes:

- Apache Ant
- SBLIM
- Open Pegasus
- GridFTP

For more information about each component, refer to Chapter 4, “Installing related software” on page 59.

2.6 Planning a production environment

When you build a grid for production follow this list:

1. Make sure that you have the required hardware and software to run the IBM Grid Toolbox.
2. Install the IBM Grid Toolbox and any fixes that are needed at the time that you get the code.
3. Ensure that your grid is secure:
   - Firewalls
   - Virus protection
   - Users
   - Proxies
   - Registration
   - Certificates
   - Certificate authority (CA)
   - Authentication
   - Authorization
   - File permissions
4. Install any additional software that is not included in the IBM Grid Toolbox:
   - If you plan to use the CMM services, you will need to install Pegasus or a compatible CIMOM on those systems that will be managed using CMM. To download Pegasus, go to:

     `http://www.openpegasus.org/`

     Find the documentation at:

     `http://www.openpegasus.org/manual/Introduction.html`

– If you plan to deploy the Data Management Services, you should install GridFTP, which is based on FTP (internet file protocol) and is a high performance, secure, reliable data transfer protocol. For the GridFTP code, download the appropriate GridFTP tar.gz package for your environment from:

  ftp://ftp.globus.org/pub/gt3/3.0/contrib/

5. Deploy your grid services and configure any instances of the IBM Grid Toolbox that you will be using.
6. Back up your grid.

2.7 Planning a development environment

When building a grid for a development environment, ensure that you have everything from the previous list. Besides the additional software needed for a production environment, you should have a tool called Apache Ant installed. You should also have installed the programming tools that you would like to use for that specific platform. Apache Ant is a Java-based build tool that is extended using Java classes and has XML-based configuration files. The IBM Grid Toolbox samples use the Apache Ant build tool for build and deployment. If you intend to develop grid services using the IBM Grid Toolbox, Apache Ant should be installed.

3

Installation and setup

This chapter presents the steps for installing and configuring the IBM Grid Toolbox. We installed the toolbox using both Red Hat and SUSE Linux distributions. For illustration purposes, we present the IBM Grid Toolbox installation using the Red Hat distribution and will note any differences using the SUSE distribution.

The following topics are discussed:

- Lab environment
- Setting up the Linux requirements
- Installing the IBM Grid Toolbox
- Securing the IBM Grid Toolbox
- Verifying the installation

3.1 Lab environment

This section provides an overview of the configuration of the software and hardware used in our lab. It is a simple Grid environment, intended to illustrate the concepts and components of the IBM Grid Toolbox. We used an Ethernet LAN with seven IBM @server xSeries machines on the LAN. We made one of the servers a certificate authority (CA Server). To represent separate virtual organizations, we grouped three servers as the *xingu* organization and grouped the other three servers as the *yanomani* organization. The xingu machines (x1, x2, and x3) were installed with the SUSE SLES 8 Linux distribution and the yanomani machines (y1, y2, and y3) were installed with the Red Hat Advanced Server 2.1 Linux distribution. In addition to being a Grid server, the x3 machine was used as an infrastructure server. Figure 3-1 illustrates this environment with the host names and the functionality of each machine.

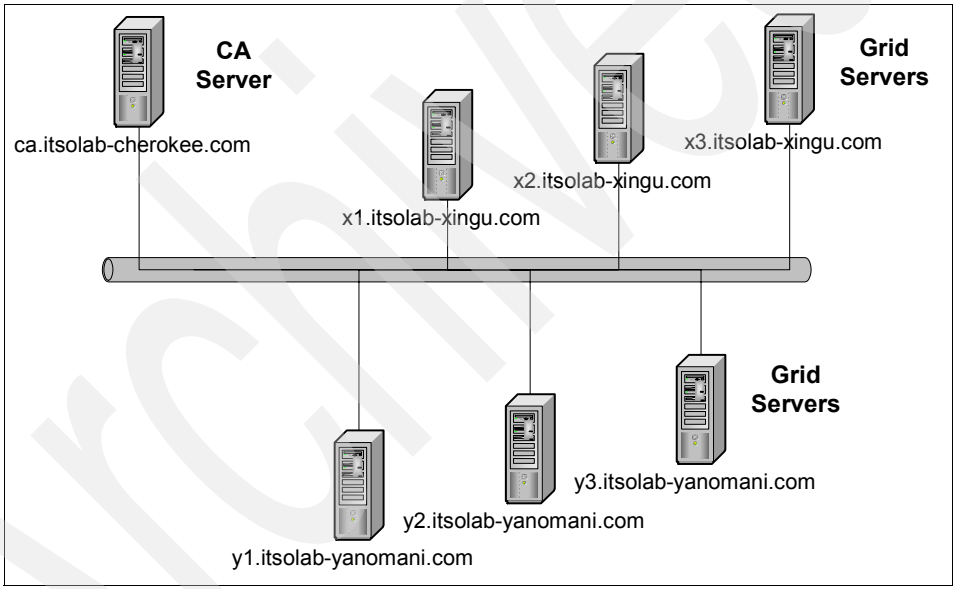


Figure 3-1 Hardware environment

3.1.1 Naming and addressing

Table 3-1 summarizes the names of the machines, their IP addresses, the Linux distribution used, and their primary functions.

Table 3-1 Host names and IP addressing

Host name	IP	Linux distribution	Function
x1.itsolab-xingu.com	192.168.0.11	SUSE SLES 8	Grid Server
x2.itsolab-xingu.com	192.168.0.12	SUSE SLES 8	Grid Server
x3.itsolab-xingu.com	192.168.0.13	SUSE SLES 8	Grid Server
y1.itsolab-yanomani.com	192.168.0.21	Red Hat	Grid Server
y2.itsolab-yanomani.com	192.168.0.22	Red Hat	Grid Server
y3.itsolab-yanomani.com	192.168.0.23	Red Hat	Grid Server
ca.itsolab-cherokee.com	192.168.0.100	Red Hat	CA Server

3.1.2 Certificate authority

The Grid environment requires an existing certificate authority. A simple certificate authority was installed on our CA Server machine. For more information about installing a certificate authority, refer to Appendix E, “Certificate authority” on page 225.

3.1.3 Users and groups

Table 3-2 contains the list of user and group IDs used in our lab.

Attention: The *ibmgrid* user ID is created by the installation of the IBM Grid Toolbox. Do not create the ibmgrid user ID ahead of time.

Table 3-2 User and group IDs

user ID	group ID	password	Function
root	root	<password>	super user
ibmgrid	ibmgrid	<password>	For administrating the grid

3.1.4 Directories

Table 3-3 contains the directories used in our lab environment.

Table 3-3 Directories

Directory	Ownership	Description
/opt/IBMGrid	ibmgrid:ibmgrid	Product base install
/etc/grid-security	root:root	Grid security-related files
/home/ibmgrid/.globus	ibmgrid:ibmgrid	User certificate
/tmp	root:root	Hold certificate request files temporarily

3.2 Setting up the Linux requirements

This section describes the steps that are required to install the Linux environment for using the IBM Grid Toolbox. The major steps to set up this environment are:

1. Install Linux.
2. Configure the network.
3. Configure Network Time Protocol.
4. Mount the infrastructure server directory.

3.2.1 Install Linux

Install Linux on all machines that will be part of the grid. In our lab, we installed Red Hat Advanced Server 2.1 on three machines and SUSE SLES 8 on three machines. We chose the default installation in both distributions. We also chose no firewall protection so that network requests (such as RPC) would not be hindered when we needed to access the infrastructure server. Be sure to adhere to the firewall and network security policy of your installation. We did not use DHCP but used fixed network IP addresses as defined in Table 3-1 on page 29.

Important: Make sure that your installation of Linux meets the requirements listed in 2.2, “IBM Grid Toolbox requirements” on page 13; otherwise, errors may occur when installing the IBM Grid Toolbox. Pay particular attention to space allocation requirements.

3.2.2 Configure network

In our lab we did not use a DNS. Thus, we create the /etc/hosts file on the infrastructure server x3 and then copied the file to the other machines.

Example 3-1 shows the contents of our lab /etc/hosts file.

Example 3-1 /etc/hosts

```
192.168.0.21    y1.itsolab-yanomani.com      y1
192.168.0.22    y2.itsolab-yanomani.com      y2
192.168.0.23    y3.itsolab-yanomani.com      y3
192.168.0.11    x1.itsolab-xingu.com         x1
192.168.0.12    x2.itsolab-xingu.com         x2
192.168.0.13    x3.itsolab-xingu.com         x3
192.168.0.100   ca.itsolab-cherokee.com      ca
```

As root user, issue the command in Example 3-2 to copy /etc/hosts from x3.

Example 3-2 Copy /etc/hosts

```
# scp 192.168.0.13:/etc/hosts /etc/hosts
```

3.2.3 Configure Network Time Protocol (NTP)

For the grid to work properly, the system clocks must be synchronized using NTP. The grid security process creates proxy certificates that are valid for specific times. If the system clocks are not synchronized, the proxy certificates may appear as if they have expired and users may not be able to use the grid.

In our lab environment we used an NTP server on machine y1. In your environment you may choose to set up NTP with a public NTP server. Do your homework in choosing appropriate servers that are geographically near you. Ensure that the server will accept new public connections and whether you must obtain their permission first.

If the package is not already installed:

1. Log in as the root user and enter the following command:

   ```
   $ rpm -ivh <source directory>/ntp-4.1.1-1.i386.rpm
   ```

2. Edit the /etc/ntp.conf file on the machine that is designated to be the time server (in our case, the y1 machine). Leave the four lines shown in Example 3-3 on page 32 as the only uncommented ones, commenting out all of the other lines with a leading # character.

Example 3-3 /etc/ntp.conf file on time server

```
server 127.127.1.0 # local clock
fudge 127.127.1.0 stratum 10
driftfile /etc/ntp/drift
broadcastdelay 0.008
```

3. Also on the NTP server machine, use the **ntsysv** or equivalent command to enable the NTP daemon (ntpd) on the next reboot.
4. Start the ntp daemon with the following command:

   ```
   service ntpd start
   ```

5. On the other machines in the grid (x1, x2, x3, y2, and y3), change the /etc/ntp.conf file to leave only the lines shown in Example 3-4 uncommented.

Example 3-4 /etc/ntp.conf file on other machines

```
server y1.itsolab-yanomani.com
driftfile /etc/ntp/drift
broadcastdelay 0.008
authenticate no
```

Note: It is not necessary to start or restart the ntpd service on a client machine.

6. Execute the following command to have the other machines check for the time from the above time server machine y1:

   ```
   ntpdate -b y1.itsolab-yanomani.com
   ```

   Note: Running **ntpdate** may not respond immediately. You may have to wait a few minutes before **ntpdate** responds.

   This should be executed at least once per boot and could be set up to run periodically using **crond** and **crontab**.

3.2.4 Mount the infrastructure directory

NFS is used to obtain the necessary files provided by the x3 machine.

As `root`, issue the commands shown in Example 3-5 to mount the directory.

Example 3-5 Mount x3 directory

```
mkdir /mnt/x3
mount 192:168.0.13:/export /mnt/x3
```

3.3 Installing the IBM Grid Toolbox

The primary method of installation for the IBM Grid Toolbox is the graphical wizard provided by the toolbox. However, it is possible to install the Toolbox using a built-in command line interface or the silent installation method. This section covers all three installation options.

Before performing either installation method:

- Log in as `root` and create a directory named `/etc/grid-security`.
- Mount the installation CD.

> **Important:** For the installation to work properly, the /etc/grid-security directory must be created ahead of time.

Example 3-6 illustrates how to make the /etc/grid-security directory and mount the installation CD.

Example 3-6 Mounting the source CD

```
[root@y2 /]# mkdir /etc/grid-security
[root@y2 /]# mount /mnt/cdrom /mnt/cdrom
```

Notes:

- For SUSE Linux, the CD-ROM mount command is:

  ```
  mount /dev/cdrom /mnt/cdrom
  ```

- In the case of multiple installations, you could place the contents of the product CD on an NFS server and mount the location of the setuplx.bin file and run the installation remotely.
- The lab environment for this book used primarily xSeries machines running Linux. The name of the installation file is therefore setuplx.bin.
  - For pSeries running AIX, it is setupax.bin.
  - For iSeries and pSeries running Linux it is setupppc.bin.
  - For zSeries running Linux, it is setuplz.bin.

3.3.1 Graphical installation method

1. To begin the graphical installation, switch to the directory containing the binary executable file and issue this command:

   ```
   ./setuplx.bin
   ```

   Example 3-7 on page 34 demonstrates how to begin the installation.

Example 3-7 Beginning the graphical installation

```
[root@y2 /]# ./setuplx.bin -options-record /tmp/response-file-redhat.txt
```

Tip: To speed up the process for subsequent installations, we recommend that a response file be generated with the installation. To record the responses, begin the setup with the following command:

```
./setuplx.bin -options-record /<directory>/<response-file-name>
```

2. When the setup is initialized, a window similar to Figure 3-2 appears.

Figure 3-2 Initial setup window

3. Click **Next** to proceed. A window similar to Figure 3-3 on page 35 appears.

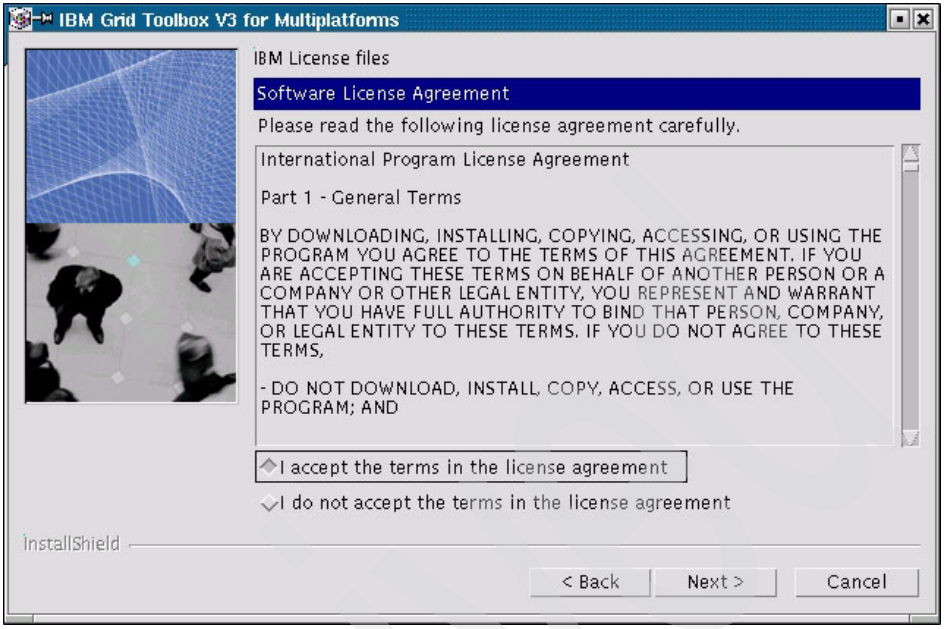


Figure 3-3 License agreement

4. Select **I accept the terms in the license agreement**.
5. Click **Next** to proceed. A window similar to Figure 3-4 on page 36 appears.

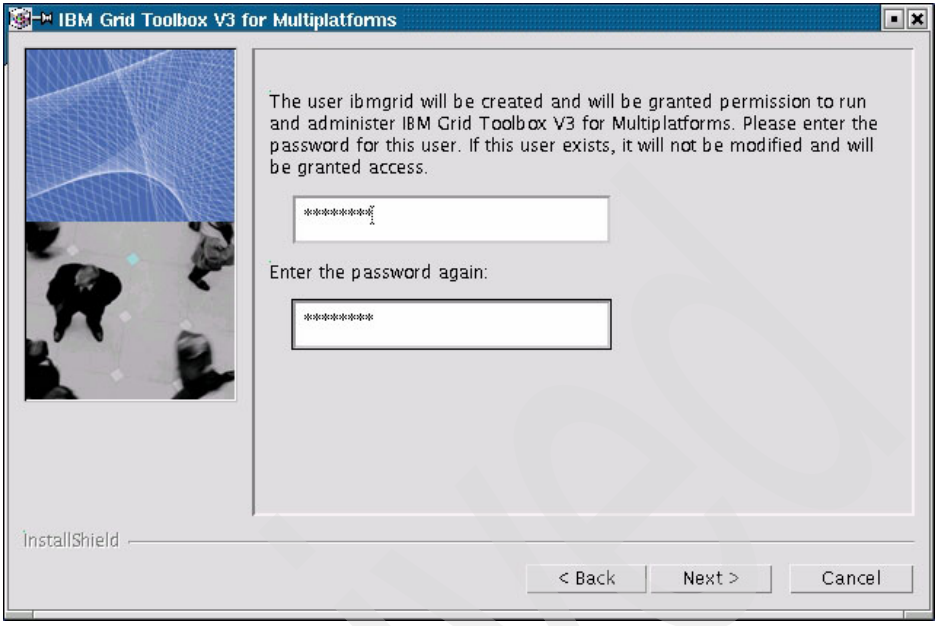


Figure 3-4 ibmgrid password window

6. The setup routine creates an `ibmgrid` user account and group on the installed machine. This user will have access to administer and run the toolbox.
7. Enter a password for ibmgrid.
8. Click **Next** to proceed. A window similar to Figure 3-5 on page 37 appears.

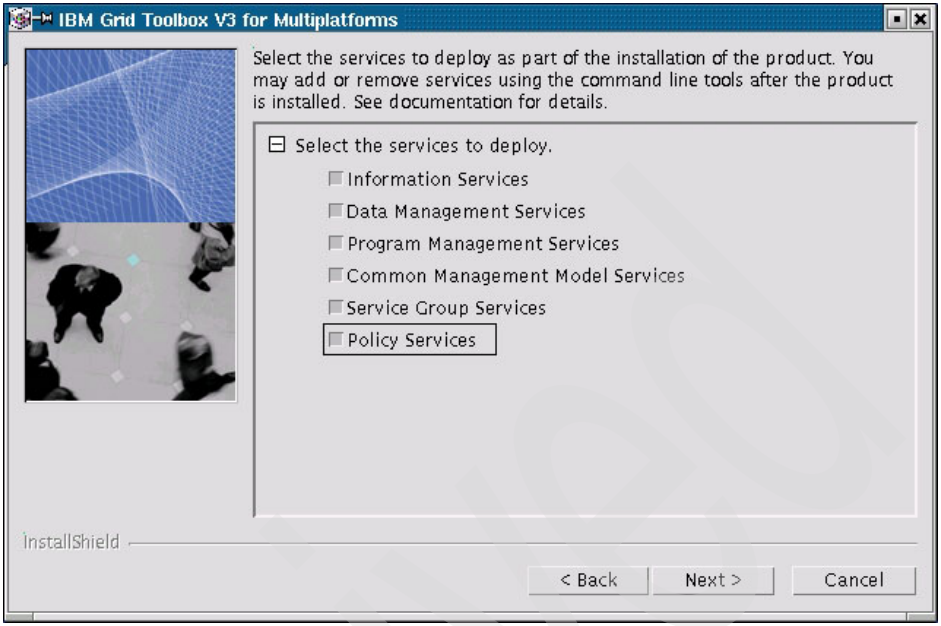


Figure 3-5 Package selection window

9. Select the services to be deployed on this machine. For more information about these services, refer to 2.3, “Planning for installation” on page 17.

> **Important:** If Policy Services are selected, Service Group Services must also be selected.

10. When the desired services are selected, click **Next** to proceed. The window shown in Figure 3-6 on page 38 appears.

> **Note:** If no services are selected for deployment, the user must install it later.

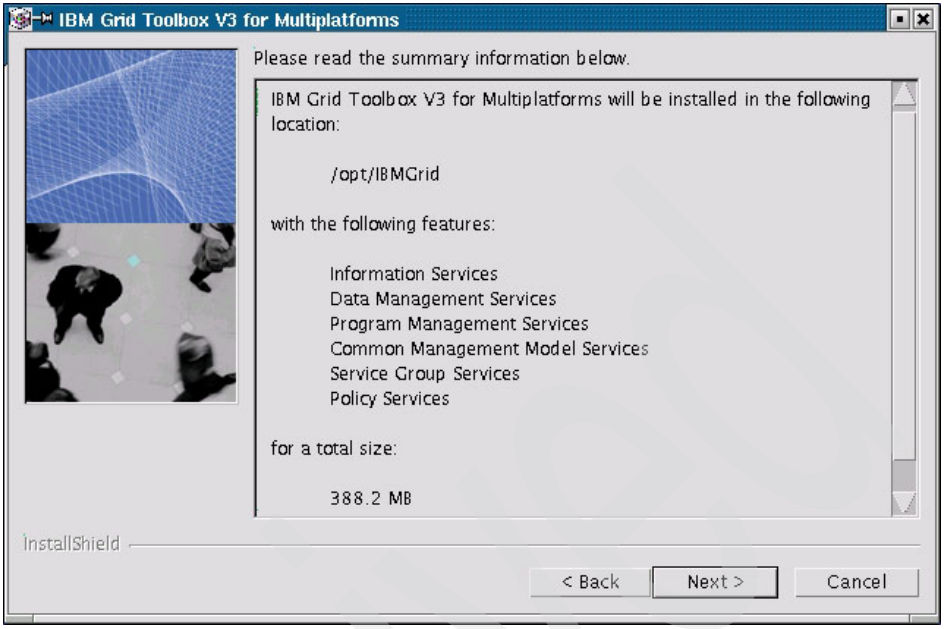


Figure 3-6 Installation summary

11. The installation wizard displays a summary of the packages to install and the directories where the toolbox will be installed. Review the summary to ensure that all is correct, and click **Next** to proceed. The window shown in Figure 3-7 on page 39 appears.

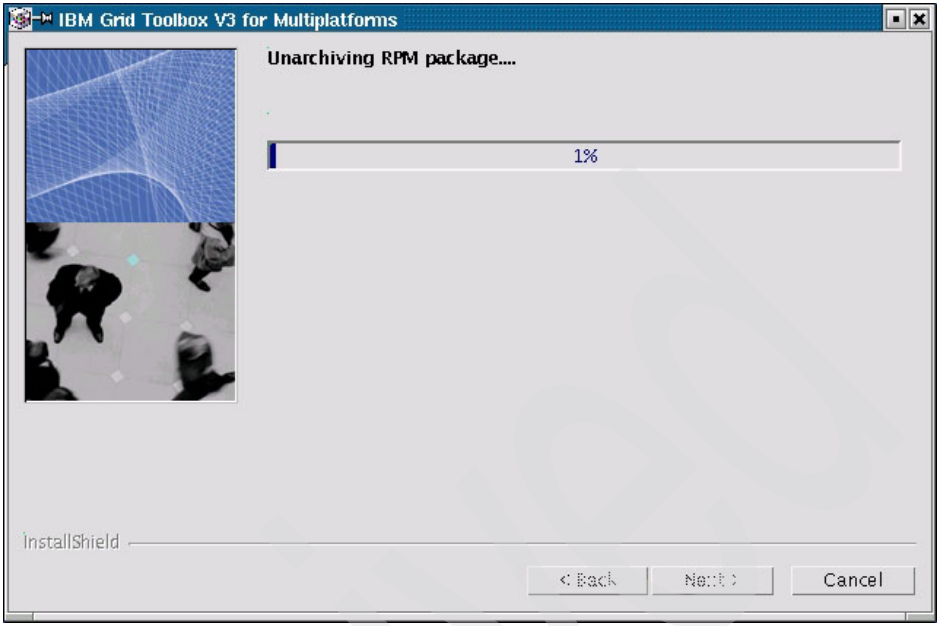


Figure 3-7 Installation progress

12. Installation of the IBM Grid Toolbox V3 for Multiplatforms V1.1 begins.

> **Attention:** Depending on the speed of the machine the installation may take some time. The installation's progress meter may therefore appear stalled at any point. However, the installation is probably still working. To ensure that it is working correctly, run the **top** command to see that Java is running. The output should be similar to Example 3-8.

Example 3-8 Sample top command

```
PID USER     PRI  NI  SIZE  RSS SHARE STAT %CPU %MEM   TIME COMMAND
 5354 root      25   0 31164  30M 10764 R    88.4 12.2   0:10 java
 5033 root      15   0  9064 8524  7736 R     4.5  3.3   0:01 kdeinit
 1139 root      15   0 14048 4004  3548 S     2.5  1.5   1:44 X
    5 root      15   0     0    0     0 SW    1.1  0.0   0:04 kswapd
 5336 root      15   0  1076 1076   832 R     0.5  0.4   0:00 top
    1 root      15   0   512  464   464 S     0.0  0.1   0:03 init
    2 root      15   0     0    0     0 SW    0.0  0.0   0:00 keventd
    3 root      15   0     0    0     0 SW    0.0  0.0   0:00 kapm-idled
    4 root      34  19     0    0     0 SWN   0.0  0.0   0:00 ksoftirqd_CPU0
    6 root      25   0     0    0     0 SW    0.0  0.0   0:00 kreclaimd
    7 root      15   0     0    0     0 SW    0.0  0.0   0:00 bdflush
    8 root      15   0     0    0     0 SW    0.0  0.0   0:00 kupdated
```

```
  9 root      25   0     0    0     0 SW    0.0  0.0    0:00 mdrecoveryd
 13 root      15   0     0    0     0 SW    0.0  0.0    0:01 kjournald
 91 root      16   0     0    0     0 SW    0.0  0.0    0:00 khubd
197 root      15   0     0    0     0 SW    0.0  0.0    0:00 kjournald
682 root      15   0   588  516   516 S     0.0  0.2    0:00 syslogd
```

13. When the installation is complete, you will see the window shown in Figure 3-8. Click **Finish** to end the wizard.

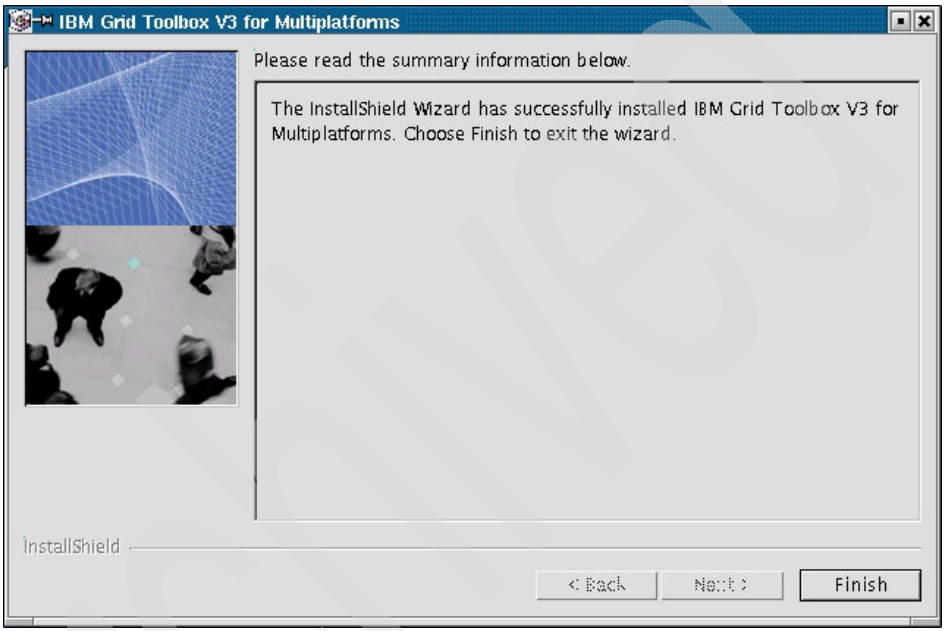


Figure 3-8 Installation is complete

3.3.2 Command line installation method

The IBM Grid Toolbox also offers a text-based installation method for machines that do not have a graphical interface.

1. To begin the command line installation, switch to the directory containing the binary executable file and issue the following command:

   ```
   ./setuplx.bin -console
   ```

   Example 3-9 on page 41 illustrates how to begin the command line installation.

Example 3-9 Beginning the command line installation

```
[root@y2 /]# ./setuplx.bin -options-record /tmp/response-file-redhat.txt
-console
```

Tip: As with the graphical install, it is possible to create a response file to speed up subsequent installations. To record the responses, begin the installation with the following command:

```
./setuplx.bin -options-record /<directory>/response-file-name> -console
```

2. When the setup is initialized, the output shown in Example 3-10 appears.

Example 3-10 Initial setup output

```
[root@y2 cdrom]# ./setuplx.bin -console
InstallShield Wizard

Initializing InstallShield Wizard...

Preparing Java(tm) Virtual Machine...
.................................
.................................
.................................
.................................
.................................
.................................
.................................
.................................
.................................
.................................
.................................
..
-------------------------------------------------------------------------------
Welcome to the InstallShield Wizard for IBM Grid Toolbox V3 for Multiplatforms

The InstallShield Wizard will install IBM Grid Toolbox V3 for Multiplatforms on
your computer.
To continue, choose Next.

IBM Grid Toolbox V3 for Multiplatforms
IBM Corporation
www.ibm.com


Press 1 for Next, 3 to Cancel or 4 to Redisplay [1] 1
```

3. To continue with the installation, type 1 and press Enter. The IBM License agreement shown in Example 3-11 appears.

Example 3-11 IBM license agreement

```
--------------------------------------------------------------------------------
IBM License files

Software Licensing Agreement
Press Enter to display the license agreement on your screen. Please
read the agreement carefully before installing the Program. After
reading the agreement, you will be given the opportunity to accept it
or decline it. If you choose to decline the agreement, installation
will not be completed and you will not be able to use the Program.


     International Program License Agreement

     Part 1 - General Terms

     BY DOWNLOADING, INSTALLING, COPYING, ACCESSING, OR USING
     THE PROGRAM YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF
     YOU ARE ACCEPTING THESE TERMS ON BEHALF OF ANOTHER PERSON
     OR A COMPANY OR OTHER LEGAL ENTITY, YOU REPRESENT AND
     WARRANT THAT YOU HAVE FULL AUTHORITY TO BIND THAT PERSON,
     COMPANY, OR LEGAL ENTITY TO THESE TERMS. IF YOU DO NOT
     AGREE TO THESE TERMS,

     - DO NOT DOWNLOAD, INSTALL, COPY, ACCESS, OR USE THE
     PROGRAM; AND

     - PROMPTLY RETURN THE PROGRAM AND PROOF OF ENTITLEMENT TO
     THE PARTY FROM WHOM YOU ACQUIRED IT TO OBTAIN A REFUND OF
     THE AMOUNT YOU PAID. IF YOU DOWNLOADED THE PROGRAM, CONTACT
     THE PARTY FROM WHOM YOU ACQUIRED IT.

     "IBM" is International Business Machines Corporation or one
     of its subsidiaries.

     "License Information" ("LI") is a document that provides
     information specific to a Program. The Program's LI is
     available at http://www.ibm.com/software/sla/ . The LI may
     also be found in a file in the Program's directory, by the
     use of a system command, or as a booklet which accompanies
     the Program.


Press Enter to continue viewing the license agreement, or, Enter 1 to
accept the agreement, 2 to decline it or 99 to go back to the
```

```
previous screen.

1

Press 1 for Next, 2 for Previous, 3 to Cancel or 4 to Redisplay [1] 1
```

4. To accept the agreement, type 1 and press Enter. Type 1 and press Enter again to continue to the next section. The next section will be similar to Example 3-12.

Example 3-12 ibmgrid password screen

```
--------------------------------------------------------------------------------
The user ibmgrid will be created and will be granted permission to run and
administer IBM Grid Toolbox V3 for Multiplatforms. Please enter the password
for this user. If this user exists, it will not be modified and will be granted
access.

Enter a password:


Enter the password again:

Enter a password:


Press 1 for Next, 2 for Previous, 3 to Cancel or 4 to Redisplay [1] 1
```

The setup routine creates an ibmgrid user account and group on the installed machine. This user will have access to administer and run the toolbox.

5. Enter a password for ibmgrid. Type 1 and press Enter to continue. The screen shown in Example 3-13 appears.

Example 3-13 Package selection screen

```
Select the services to deploy as part of the installation of the product. You
may add or remove services using the command line tools after the product is
installed. See documentation for details.

   IBM Grid Toolbox V3 for Multiplatforms

   To select/deselect a feature or to view its children, type its number:

     1.  [x] Information Services
     2.  [x] Data Management Services
     3.  [x] Program Management Services
     4.  [x] Common Management Model Services
     5.  [x] Service Group Services
     6.  [x] Policy Services
```

```
   Other options:

     0. Continue installing

   Enter command [0] 0

Press 1 for Next, 2 for Previous, 3 to Cancel or 4 to Redisplay [1] 1
```

6. Select the services to be deployed on this machine. Services are selected by typing in their corresponding number and pressing Enter. The output refreshes to reflect the new selections. For more information about these services, refer to 2.3, “Planning for installation” on page 17.

   Important: If Policy Services are selected, Service Group Services must also be selected

7. When the desired services are selected, type 0 (zero) and press Enter to continue. A summary similar to Example 3-14 appears.

Example 3-14 Installation summary screen

```
--------------------------------------------------------------------------------
IBM Grid Toolbox V3 for Multiplatforms will be installed in the following
location:

/opt/IBMGrid

with the following features:

Information Services
Data Management Services
Program Management Services
Common Management Model Services
Service Group Services
Policy Services

for a total size:

 388.2 MB

Press 1 for Next, 2 for Previous, 3 to Cancel or 4 to Redisplay [1] 1
```

8. The setup program provides a summary of the installation. Review the summary and ensure that it is correct. To begin the installation, type 1 and press Enter. A progress meter similar to Example 3-15 on page 45 appears.

Example 3-15 Installation progress meter

```
|-----------|-----------|-----------|------------|
0%          25%         50%         75%         100%
||||||||||||||||||||||||||||||||||||||||||||||||||
```

Attention: Depending on the speed of the machine, the installation may take some time. The installation's progress meter may therefore appear stalled at any point. However, the installation is still working. To make sure it is working correctly, run the **top** command to see that Java is running. The output should be similar to Example 3-16.

Example 3-16 Sample top command

```
PID USER     PRI  NI  SIZE  RSS SHARE STAT %CPU %MEM   TIME COMMAND
 5354 root      25   0 31164  30M 10764 R    88.4 12.2   0:10 java
 5033 root      15   0  9064 8524  7736 R     4.5  3.3   0:01 kdeinit
 1139 root      15   0 14048 4004  3548 S     2.5  1.5   1:44 X
    5 root      15   0     0    0     0 SW    1.1  0.0   0:04 kswapd
 5336 root      15   0  1076 1076   832 R     0.5  0.4   0:00 top
    1 root      15   0   512  464   464 S     0.0  0.1   0:03 init
    2 root      15   0     0    0     0 SW    0.0  0.0   0:00 keventd
    3 root      15   0     0    0     0 SW    0.0  0.0   0:00 kapm-idled
    4 root      34  19     0    0     0 SWN   0.0  0.0   0:00 ksoftirqd_CPU0
    6 root      25   0     0    0     0 SW    0.0  0.0   0:00 kreclaimd
    7 root      15   0     0    0     0 SW    0.0  0.0   0:00 bdflush
    8 root      15   0     0    0     0 SW    0.0  0.0   0:00 kupdated
    9 root      25   0     0    0     0 SW    0.0  0.0   0:00 mdrecoveryd
   13 root      15   0     0    0     0 SW    0.0  0.0   0:01 kjournald
   91 root      16   0     0    0     0 SW    0.0  0.0   0:00 khubd
  197 root      15   0     0    0     0 SW    0.0  0.0   0:00 kjournald
  682 root      15   0   588  516   516 S     0.0  0.2   0:00 syslogd
```

9. When the installation is complete, the output in Example 3-17 is generated.

Example 3-17 Final installation screen

```
Set runAsUser and runAsGroup
Configure custom user registry
Set users.props file
Set groups.props file
Set activeUserRegistry and turn security on
Cur Active = cells/DefaultNode:security.xml#LocalOSUserRegistry
New Active = cells/DefaultNode:security.xml#CustomUserRegistry_1
Saving session

-------------------------------------------------------------------------------
The InstallShield Wizard has successfully installed IBM Grid Toolbox V3 for
```

```
Multiplatforms. Choose Finish to exit the wizard.

Press 3 to Finish or 4 to Redisplay [3] 3
```

10. To finish the installation type `3` and press Enter.

3.3.3 Silent installation method

You also can install the IBM Grid Toolbox using a silent method. This method requires a response file to provide the installation options. A response file can be generated either by using the -options-record flag in the graphical and command line methods earlier in this chapter or by executing this command to create a template response file. This file can then be edited to set the installation options.

```
./setuplx.bin -options-template /<directory>/<response-file-name>
```

For a sample response file that was created using the -options-record flag on a graphical installation, refer to Appendix D, “Response file” on page 215.

To begin the installation using the response file, run the following command:

```
./setuplx.bin -options /<directory>/<response-file-name> -silent
```

The setup procedure begins and a prompt is returned when setup is complete.

Example 3-18 shows how to begin the silent installation with a response file that was created using the -options-record flag on a graphical install.

Example 3-18 Begin the silent installation

```
[root@y2 /]# ./setuplx.bin -options /tmp/response-file-redhat.txt -silent
```

Attention: During this procedure, no output will be generated.

Tip: To run the setup in the background, append an ampersand (&) to the end of the setup command.

Tip: If you are installing the IBM Grid Toolbox on a large number of machines, you can automate some of the installation steps. For more about automating parts of the installation, see Appendix C, “Script the installation” on page 201.

3.3.4 Post installation

After the toolbox is installed on a machine, log files for the installation can be found in the /opt/IBMGrid/logs directory.

In order for the IBM Grid Toolbox to function properly, its environment variables must be set. This can be done by executing the /opt/IBMGrid/igt-setenv.sh script with the following command:

```
/opt/IBMGrid/igt-setenv.sh
```

Tip: We recommend that the sourcing of the /opt/IBMGrid/igt-setenv.sh script be added to the system or user profile so that the environment can be set up automatically upon logon. In our lab, we copied igt-setenv.sh to the /etc/profile.d directory.

3.3.5 Securing the grid

In order to use resources in a grid, you must first request and install security certificates from a reputable certificate authority (CA).

Note: For a description of how to establish and use a certificate authority, refer to Appendix E, "Certificate authority" on page 225.

Obtain a CA certificate

To begin securing a machine on a grid, you must first obtain a CA certificate from your CA. The IBM Grid Toolbox can import CA certificates in three formats:

- pem
- der
- gt3

After the certificate has been obtained and copied to the /tmp directory, it must be imported into the /etc/grid-security directory. As `root`, issue this command:

```
/opt/IBMGrid/bin/igt-import-ca -type PEM -cert /tmp/cacert.pem -default
```

Note: To import DER-type CA certificates, set the -type flag to `DER` instead of `PEM`.

The output from the **`igt-import-ca`** command should be similar to that in Example 3-19.

Example 3-19 Sample igt-import-ca command

```
[root@y2 /]# /opt/IBMGrid/bin/igt-import-ca -type PEM -cert /tmp/cacert.pem
-default
******************************************************
grid security folder: /etc/grid-security/
has been backed up to:
/etc/grid-security//backup-gsi-Fri-Feb-27-17-14-56-CST-2004.tar
```

```
*******************************************************
Configuring GSI security...
Cert Hash: e82f5117

Creating /etc/grid-security//certificates/grid-security.conf.e82f5117
Creating /etc/grid-security//certificates/globus-host-ssl.conf.e82f5117
Creating /etc/grid-security//certificates/globus-user-ssl.conf.e82f5117

Saving policy file in: /etc/grid-security//certificates/e82f5117.signing_policy
Certificate DN: /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca

Copying /tmp/cacert.pem to /etc/grid-security//certificates/e82f5117.0

Setting CA e82f5117, as default...
linking /etc/grid-security//certificates/grid-security.conf.e82f5117 to
        /etc/grid-security//grid-security.conf

linking /etc/grid-security//certificates/globus-host-ssl.conf.e82f5117 to
        /etc/grid-security//globus-host-ssl.conf

linking /etc/grid-security//certificates/globus-user-ssl.conf.e82f5117 to
        /etc/grid-security//globus-user-ssl.conf

Done.
[root@y2 /]#
```

To import a GT3-type certificate, execute the following command:

```
/opt/IBMGrid/bin/igt-import-ca -type gt3 -tar
globus_simple_ca_[CA-HASH]_setup.tar.gz -default
```

Important: This command should appear as a single line.

To verify the default CA, run this command:

```
/opt/IBMGrid/bin/igt-grid-default-ca
```

The output from **igt-grid-default-ca** will look similar to Example 3-20.

Example 3-20 sample igt-grid-default-ca command

```
[root@y2 etc]# /opt/IBMGrid/bin/igt-grid-default-ca
The available CA configurations installed on this host are:

1) e82f5117 -  /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca

The default CA is:  e82f5117
```

```
Enter the index number of the CA to set as the default:
1

setting the default CA to: /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca

linking /etc/grid-security//certificates//grid-security.conf.e82f5117 to
        /etc/grid-security//grid-security.conf

linking /etc/grid-security//certificates//globus-host-ssl.conf.e82f5117 to
        /etc/grid-security//globus-host-ssl.conf

linking /etc/grid-security//certificates//globus-user-ssl.conf.e82f5117 to
        /etc/grid-security//globus-user-ssl.conf


...done.

[root@y2 etc]#
```

Request a host certificate

The next step in securing the grid is to request a host certificate for a particular machine. First begin by creating a certificate request for the host, running the following command as root:

```
/opt/IBMGrid/bin/igt-grid-cert-request -host <hostname> -dir /tmp
```

The resulting output will be similar to Example 3-21.

Example 3-21 Sample igt-grid-cert-request command

```
[root@y2 /]# /opt/IBMGrid/bin/igt-grid-cert-request -host
y2.itsolab-yanomani.com -dir /tmp

A private host key and a certificate request has been generated
with the subject:

/C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=host/y2.itsolab-yanomani.com
---------------------------------------------------------
The private key is stored in /tmp/hostkey.pem
The request is stored in /tmp/hostcert_request.pem

This certificate request (/tmp/hostcert_request.pem) should be processed
with your trusted Certificate Authority, consistent
with your site security policy.
[root@y2 /]#
```

The following files were created in the /tmp directory as a result:

- hostcert_request.pem
- hostkey.pem
- hostcert.pem (This is only a placeholder file of 0 KB)

After a request has been generated, it can be submitted to your CA for signing. The filename for the host certificate request will be `hostcert_request.pem`. Your CA then sends back a signed certificate, which should be renamed to `hostcert.pem`.

Next, you must install the new host certificate. As `root`, issue this command:

```
/opt/IBMGrid/bin/igt-install-certs -host -cert /tmp/hostcert.pem
```

This generates output similar to Example 3-22.

Example 3-22 Sample igt-install-certs command

```
[root@y2 /]# /opt/IBMGrid/bin/igt-install-certs -host -cert /tmp/hostcert.pem
*******************************************************
Installing host cert: : /tmp/hostcert.pem
Host: y2.itsolab-yanomani.com
Destination: /etc/grid-security/
*******************************************************
INFO: Copying /tmp/hostkey.pem to /etc/grid-security//hostkey.pem...

Done.
[root@y2 /]#
```

To verify the installation of the host certificate, issue the following command:

```
/opt/IBMGrid/bin/grid-cert-info -file /etc/grid-security/hostcert.pem
-subject
```

This should produce output similar to Example 3-23.

Example 3-23 Sample grid-cert-info command

```
[root@y2 /]# /opt/IBMGrid/bin/grid-cert-info -file \
/etc/grid-security/hostcert.pem -subject
/C=US/ST=Texas/O=ITSOLAB/CN=host/y2.itsolab-yanomani.com
```

Request a user certificate

In addition to certificates on a host level, users also require certificates to participate in a grid. To obtain a user certificate, log in as `ibmgrid` and run the request certificate command:

```
/opt/IBMGrid/bin/igt-grid-cert-request -dir /tmp
```

Tip: By default, the IBM Grid Toolbox sets the Common Name (CN) as `IBM Grid Toolbox V3 for Multiplatforms`. This may cause problems with your CA if you plan to request multiple user certificates. To specify a unique Common Name, use the -cn flag in the **igt-grid-cert-request** command:

```
/opt/IBMGrid/bin/igt-grid-cert-request -dir /tmp -cn “usery”
```

The output for the **igt-grid-cert-request** command should be similar to Example 3-24.

Example 3-24 Sample igt-grid-cert-request command

```
[ibmgrid@y1 /]$ /opt/IBMGrid/bin/igt-grid-cert-request -dir /tmp -cn "usery"
Using user id: ibmgrid
Common Name: usery

A certificate request and private key is being created.
You will be asked to enter a PEM pass phrase.
This pass phrase is akin to your account password,
and is used to protect your key file.
If you forget your pass phrase, you will need to
obtain a new certificate.

Using configuration from /etc/grid-security/globus-user-ssl.conf
Generating a 1024 bit RSA private key
.............................................++++++
.......++++++
writing new private key to '/tmp/userkey.pem'
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:

A private key and a certificate request has been generated with the subject:

/C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=usery

If the CN=usery is not appropriate, rerun this
script with the -force -cn "Common Name" options.

Your private key is stored in /tmp/userkey.pem
Your request is stored in /tmp/usercert_request.pem

This certificate request (/tmp/usercert_request.pem) should be processed
with your trusted Certificate Authority, consistent
with your site security policy.

[ibmgrid@y1 /]$
```

The following files were created in the /tmp directory as a result:

- usercert_request.pem
- userkey.pem
- usercert.pem (This is only a placeholder file of 0 KB)

After a request has been generated, it can be submitted to your CA for signing. The filename for the user certificate request will be `usercert_request.pem`. Your CA then sends back a signed certificate, which should be renamed to `usercert.pem`.

Next, install the new user certificate. Begin by logging on as `ibmgrid` and running the following command:

```
/opt/IBMGrid/bin/igt-install-certs -cert /tmp/usercert.pem
```

The output from this command should be similar to Example 3-25.

Example 3-25 Sample igt-install-certs command

```
[ibmgrid@y2 /]$ /opt/IBMGrid/bin/igt-install-certs -cert /tmp/usercert.pem
**********************************************************
Installing user cert: /tmp/usercert.pem
User: ibmgrid
Destination: /home/ibmgrid/.globus/usercert.pem
Cert DN: /C=US/ST=Texas/O=ITSOLAB/CN=usery
**********************************************************
Creating /home/ibmgrid/.globus
Copying /tmp/usercert.pem to /home/ibmgrid/.globus/usercert.pem...
Setting permissions...

WARNING: ibmgrid has not been added to the grid-map-file. root/sysadmin access
required.

INFO: Copying user key: /tmp/userkey.pem to
/home/ibmgrid/.globus/userkey.pem...

Done.
[ibmgrid@y2 /]$
```

After it is installed, you can verify the installation by running:

```
/opt/IBMGrid/bin/grid-cert-info -subject
```

The output of this command should be similar to Example 3-26.

Example 3-26 Sample grid-cert-info command

```
[ibmgrid@y2 /]$ /opt/IBMGrid/bin/grid-cert-info -subject
/C=US/ST=Texas/O=ITSOLAB/CN=usery
```

```
[ibmgrid@y2 /]$
```

Adding a grid mapfile entry

Now that the certificates have been signed and installed, users must be added to the grid mapfile so that they can access resources on a grid host. The mapping consists of associating a grid user's DN with a local user on the host.

Begin by logging on as `ibmgrid` and finding the DN for the ibmgrid user with the following command:

```
/opt/IBMGrid/bin/grid-cert-info -file /home/ibmgrid/.globus/usercert.pem
-subject
```

Important: This command should appear on a single line.

Tip: You can also use this command to find the DN of the user certificate for the user you are currently logged in as:

```
/opt/IBMGrid/bin/grid-cert-info -subject
```

Take note of the complete distinguished name that is associated with the user. It should be similar to Example 3-27.

Example 3-27 Sample grid-cert-info

```
[ibmgrid@y2 /]$ /opt/IBMGrid/bin/grid-cert-info -file
/home/ibmgrid/.globus/usercert.pem -subject

/C=US/ST=Texas/O=ITSOLAB/CN=usery
[ibmgrid@y2 /]$
```

Tip: To find the distinguished name of other users, replace `/home/ibmgrid/` in the above command with the desired user's home directory.

Log in as `root` and issue the following command to add the user to the mapfile:

```
/opt/IBMGrid/sbin/grid-mapfile-add-entry -dn "<DN>" -ln "<localuser>"
```

Important: Make sure to replace <DN> with the results of the **`grid-cert-info`** command above.

The output should be similar to Example 3-28.

Example 3-28 Sample grid-mapfile-add-entry command

```
[root@y2 /]# /opt/IBMGrid/sbin/grid-mapfile-add-entry  -dn
"/C=US/ST=Texas/O=ITSOLAB/CN=usery" -ln "ibmgrid"
```

```
/etc/grid-security/grid-mapfile does not exist... Attempting to create
/etc/grid-security/grid-mapfile
(1) entry added
[root@y2 /]#
```

You can verify that the entry was added by running the following two commands:

```
cat /etc/grid-security/grid-mapfile
/opt/IBMGrid/sbin/grid-mapfile-check-consistency
```

Note: grid-mapfile-check-consistency generates no output when successful.

The output of these two commands should be similar to Example 3-29.

Example 3-29 Verifying the grid mapfile

```
[root@y1 /]# cat /etc/grid-security/grid-mapfile
"/C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=usery" ibmgrid
[root@y2 /]# /opt/IBMGrid/sbin/grid-mapfile-check-consistency
[root@y2 /]#
```

3.3.6 Verifying the installation

To verify that the IBM Grid Toolbox has been installed successfully and that grid security was implemented correctly, complete the following procedure:

1. Log in to your system using the newly created `ibmgrid` user ID.

   Note: You must log off of your system completely. Switching to ibmgrid will not work.

2. Run the following command to start the IBM Grid Toolbox container:

   ```
   /opt/IBMGrid/bin/igt-start-container
   ```

   When the process is complete, a message indicates that the server is `open for e-business` along with the process ID as shown in Example 3-30.

Example 3-30 Starting the IGT container

```
[ibmgrid@y2 /]$ /opt/IBMGrid/bin/igt-start-container
/opt/IBMGrid/AppServer/bin/startServer.sh: ulimit: cannot modify limit:
Operation not permitted
ADMU0116I: Tool information is being logged in file
           /opt/IBMGrid/AppServer/logs/server1/startServer.log
ADMU3100I: Reading configuration for server: server1
```

```
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is
10051
```

Note: igt-start-container may take some time to complete.

3. Start your browser and open the IBM Grid Services Manager at:

   `http://hostname:port/gsm/`

   – `hostname` is the host you are working with and `port` is the port used by the Web application that is associated with the OGSI run-time environment. The default port is 12080.

   Note: Your browser must have cookies enabled to access the IBM Grid Services Manager.

4. On the IBM Grid Services Manager logon page, enter the `ibmgrid` user ID and password that were created during the install process. Click **Logon**.

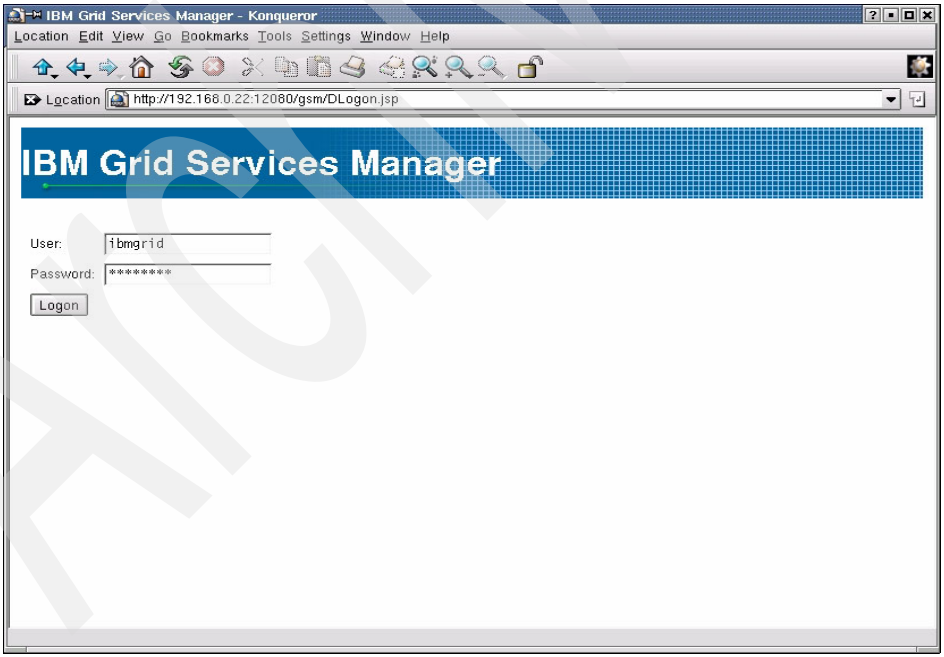


Figure 3-9 IBM Grid Services Manager logon page

5. The GSI Proxy Generation page should appear next. Enter the Pass Phrase created during the Setting Up Grid security process. Click **Create Proxy**.

   Figure 3-10 shows the GSI Proxy Generation page in our lab environment.

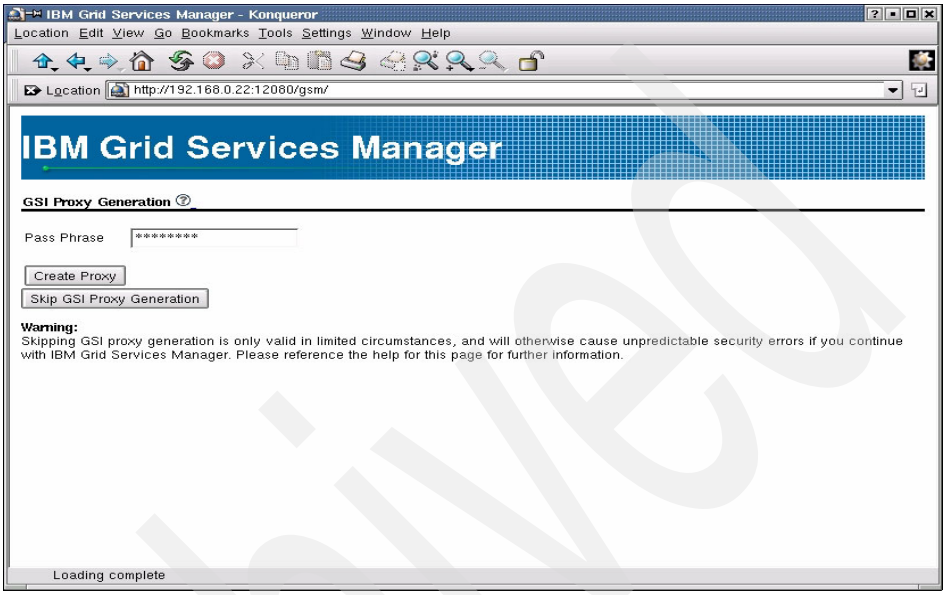


Figure 3-10 GSI Proxy Generation page

6. The IBM Grid Services Manager - Manage Available Instances page should appear. Figure 3-11 on page 57 shows this page as seen in our lab environment.

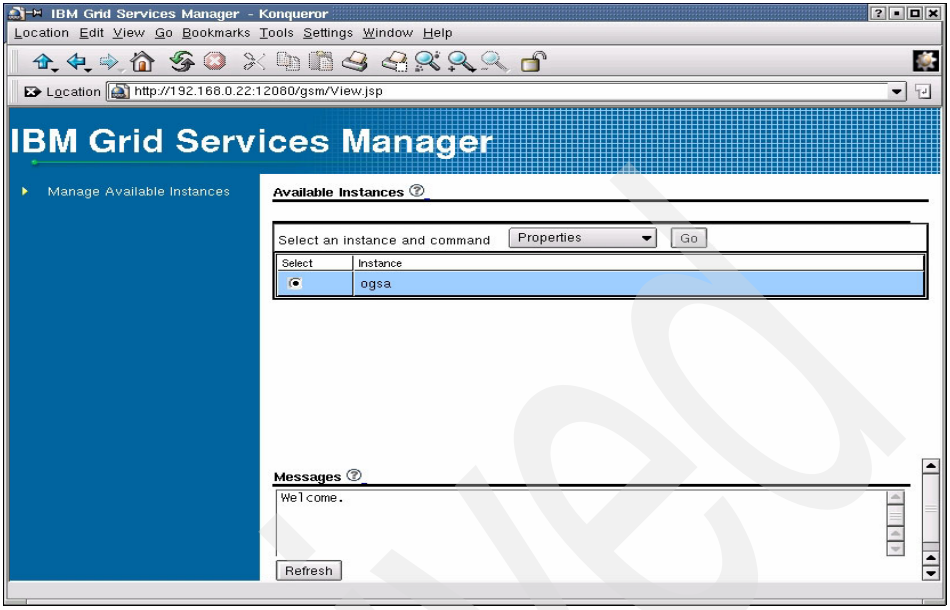


Figure 3-11 Manage Available Instances page

7. To complete the verification, expand the **Manage Available Instances** link in the left-side navigation pane. In the new branch, click **ogsa**, then click **Manage Grid Services**.

   If the Grid Services page appears, then the installation completed successfully. Figure 3-12 on page 58 shows this page as seen in our lab environment.

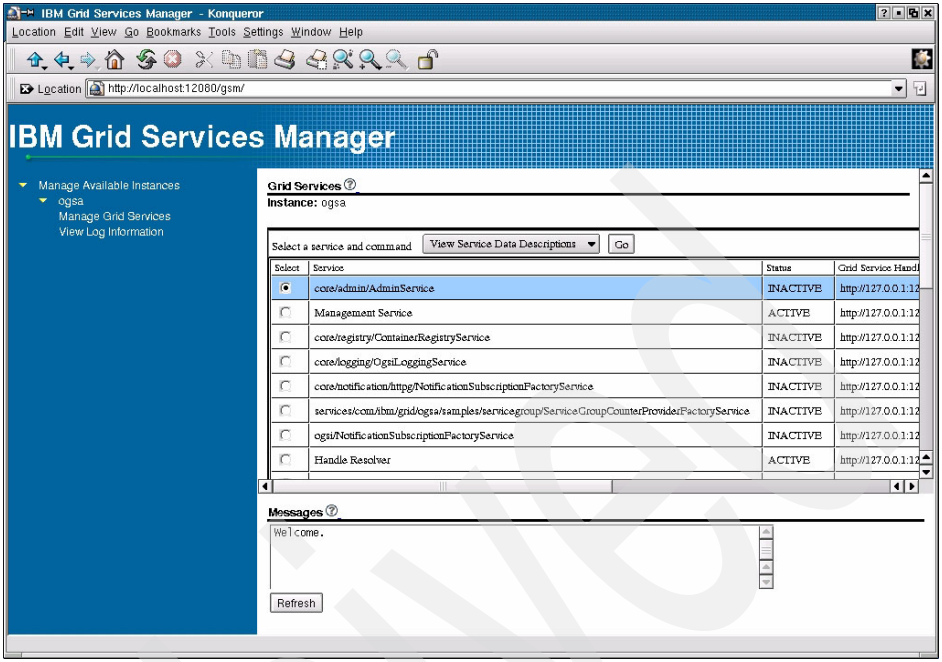


Figure 3-12 Grid Services page

Tip: An additional method of verifying the installation is to run the samples found later in this book. For more information about these samples, refer to Chapter 6, “Samples” on page 105.

4

Installing related software

This chapter presents information about related software and the steps that are necessary to install and configure it.

The following software is presented:

- Apache Ant
- Pegasus and SBLIM
- Grid File Transfer Protocol (GridFTP)

4.1 Apache Ant

Apache Ant is a Java-based build tool for developing software across multiple platforms. Unlike other build tools (Make, for example) that are extended using shell-based commands, Apache Ant is extended using Java classes. Instead of writing shell commands, the configuration files are XML-based.

The IBM Grid Toolbox samples use the Apache Ant build tool for build and deployment. If you intend to develop grid services using the IBM Grid Toolbox, you should install Apache Ant.

Ant's build files are written in XML. Each build file contains one project and at least one (default) target. Targets contain task elements. For writing simple build files and running Ant, refer to the Ant documentation.

For documentation about Apache Ant, visit the Apache Ant Web site:

http://ant.apache.org

4.1.1 Acquire Apache Ant

We downloaded Apache Ant 1.6.1 binary, apache-ant-1.6.1-bin.tar.gz, into the /tmp directory on our y1 machine from the following URL:

http://ant.apache.org/bindownload.cgi

4.1.2 Set up the environment variable, path and directory

1. Log on as the root user.
2. Make a directory for Ant. Example 4-1 shows the command we used in our lab environment.

Example 4-1 Ant directory

```
mkdir /opt/apache-ant
```

3. Set the ANT_HOME environment variable and add the bin directory to your path. In our lab, we chose to update the igt-setenv.sh file in our /etc/profile.d directory. Example 4-2 shows the statements we added to our igt-setenv.sh file.

Example 4-2 Ant variable and path statement

```
export ANT_HOME=/opt/apache-ant/apache-ant-1.6.1
export PATH=$ANT_HOME/bin:$PATH
```

4. Start the shell again to invoke the changes above.

4.1.3 Install Ant

Install Apache Ant using the commands in Example 4-3.

Example 4-3 Commands to install Apache Ant

```
cd /opt/apache-ant
tar -zxvf /tmp/apache-ant-1.6.1-bin.tar.gz
```

To verify that everything is working properly, enter the following command:

```
ant -help
```

If the Ant help menu is displayed, then you have installed Ant successfully.

4.1.4 Uninstall Apache Ant

To uninstall Apache Ant, refer to the Ant documentation appropriate to your operating environment.

4.2 Pegasus and SBLIM

Pegasus is an open-source implementation of the Distributed Management Task Force (DMTF), Common Information Model (CIM), and Web Based Enterprise Management (WBEM) standards.

For Common Management Model (CMM) Services to be fully operational, Pegasus should be installed. For more information about CMM, refer to Chapter 2, “Planning” on page 11.

The Standards Based Linux Instrumentation for Manageability (SBLIM) is an Open Source project to enhance the manageability of GNU/Linux systems. SBLIM is one of the instrumentation providers that can be registered to Pegasus.

4.2.1 Acquire Pegasus

The IBM Grid Toolbox has been tested with Pegasus 2.2.1. We downloaded the Pegasus release snapshot, pegasus-2.2.1.tar.gz, into the /tmp directory on our y1 machine from:

http://www.openpegasus.org

Click **Release Snapshots** on the left side of the page, then click the **GZ** link for Release 2.2.1 to download.

4.2.2 Install Pegasus

The following steps were used to install Pegasus in our lab:

1. Log on as the root user.
2. Make a directory for Pegasus. The IBM Grid Toolbox uses the /opt directory, so we decided to keep Pegasus there. Example 4-4 shows the command we used in our lab environment.

Example 4-4 Pegasus directories

```
mkdir /opt/pegasus
mkdir /opt/pegasus/pegasus_home
```

3. Next, we changed to the /opt/pegasus directory and unzipped the pegasus-2.2.1.tar.gz file. These commands are listed in Example 4-5.

Example 4-5 Pegasus unzip commands

```
cd /opt/pegasus
tar -zxvf /tmp/pegasus-2.2.1.tar.gz
```

4. Review the “Pegasus Dependencies” section in the following file and update your software if needed:

   ```
   /opt/pegasus/pegasus-2.2.1/readme.txt
   ```

   Note: Our Red Hat Advanced Server 2.1 and SUSE SLES 8 Linux distributions met the Pegasus requirements. We did not upgrade any software and the installation completed successfully.

5. Next, set up your environment. In our lab environment, we set our environment variables and path as listed in Example 4-6.

Example 4-6 Pegasus environment variables and path

```
export PEGASUS_ROOT=/opt/pegasus/pegasus-2.2.1
export PEGASUS_HOME=/opt/pegasus/pegasus_home
export PEGASUS_PLATFORM=LINUX_IX86_GNU
export PATH=$PATH:$PEGASUS_HOME/bin
```

   To make it easier, we put these commands in a script file called pegasus.sh and placed the script file in /etc/profile.d.

   As instructed, we also added a line to the /etc/ld.so.conf file. Example 4-7 on page 63 shows the line we added in our lab.

Example 4-7 /etc/ld.so.conf addition

```
/opt/pegasus/pegasus_home/lib
```

6. Now make the Pegasus code by changing to the pegasus-2.2.1 directory and entering the commands in Example 4-8 in the order shown.

Example 4-8 Make commands

```
cd /opt/pegasus/pegasus-2.2.1
make
make repository
make tests
```

The first make command may run for a while, depending on your server's hardware capabilities.

4.2.3 Acquire SBLIM

1. We made a directory for the SBLIM packages with the following command:

   ```
   mkdir /tmp/sblim
   ```

2. The SBLIM CMPI packages were downloaded into the directory from:

   http://www.ibm.com/sblim/sblimpack.html

3. Click the **Overview** link at the bottom of the page. On the next page, click each individual package name.
4. The IBM Grid Toolbox was tested with these SBLIM providers. We downloaded the following packages into our /tmp/sblim directory:
   - sblim-cmpi-adapter-0.8.1.tar.gz
   - sblim-cmpi-base-1.2.2.tar.gz
   - sblim-cmpi-fsvol-1.2.3.tar.gz
   - sblim-cmpi-network-1.2.3.tar.gz

4.2.4 Install SBLIM

The following steps were used to install the SBLIM packages in our lab:

1. Log on as the `root` user.
2. Open the /tmp/sblim directory and unzip the downloaded files (Example 4-9).

Example 4-9 SBLIM unzip commands

```
cd /tmp/sblim
tar -zxvf sblim-cmpi-adapter-0.8.1.tar.gz
tar -zxvf sblim-cmpi-base-1.2.2.tar.gz
tar -zxvf sblim-cmpi-fsvol-1.2.3.tar.gz
```

```
tar -zxvf sblim-cmpi-network-1.2.3.tar.gz
```

3. Ensure that the environment variables for Pegasus that were set in Pegasus installation step 5 on page 62 are in effect. Use the **export -p** command or **echo $PEGASUS_HOME** command. Set up the variables again, if needed.
4. Change to the /tmp/sblim/sblim-cmpi-adapter-0.8.1/pegasus directory and make its modules. These commands are shown in Example 4-10.

Example 4-10 sblim-cmpi-adapter commands

```
cd /tmp/sblim/sblim-cmpi-adapter-0.8.1/pegasus
make
make install
```

5. Change to the /tmp/sblim/sblim-cmpi-base-1.2.2 directory and make its modules. These commands are shown in Example 4-11.

Example 4-11 sblim-cmpi-base commands

```
cd /tmp/sblim/sblim-cmpi-base-1.2.2
make
make install
```

6. Change to the /tmp/sblim/sblim-cmpi-fsvol-1.2.3 directory and make its modules. These commands are shown in Example 4-12.

Example 4-12 sblim-cmpi-fsvol commands

```
cd /tmp/sblim/sblim-cmpi-fsvol-1.2.3
make
make install
```

7. Change to the /tmp/sblim/sblim-cmpi-network-1.2.3 directory and make its modules. These commands are shown in Example 4-13.

Example 4-13 sblim-cmpi-network commands

```
cd /tmp/sblim/sblim-cmpi-network-1.2.3
make
make install
```

4.2.5 Start Pegasus and add a user

Assuming that all of the Pegasus and SBLIM packages installed correctly, you are now ready to start Pegasus, verify that the providers are correct, and add a CIM user:

1. Log on as the root user.

2. Start the Pegasus server with the following command:

```
cimserver
```

Example 4-14 shows the **cimserver** startup command from our lab.

Example 4-14 cimserver command

```
[root@y1]# cimserver
Logs Directory = /opt/pegasus/pegasus_home/./logs
CIM Server 2.2
Built Mar 13 2004 09:51:27
Starting... No SLP
Listening on HTTP port 5988
Started.
[root@y1]#
```

3. Verify that the providers are correct with the following command:

```
cimprovider -l
```

Example 4-15 shows the cimprovider command from our lab.

Example 4-15 cimprovider command

```
[root@y1]# cimprovider -l
OperatingSystemModule
ComputerSystemModule
ProcessModule
IBM_CIMOMStatDataProviderModule
ProcessorProviderModule
IPProviderMod
OSBase_ComputerSystemProviderModule
OSBase_OperatingSystemProviderModule
OSBase_UnixProcessProviderModule
OSBase_ProcessorProviderModule
OSBase_RunningOSProviderModule
OSBase_OSProcessProviderModule
OSBase_CSProcessorProviderModule
OSBase_LocalFileSystemProviderModule
OSBase_NFSProviderModule
OSBase_HostedFileSystemProviderModule
OSBase_BootOSFromFSProviderModule
OSBase_IPProtocolEndpointProviderModule
OSBase_LocalLoopbackPortProviderModule
OSBase_EthernetPortProviderModule
OSBase_TokenRingPortProviderModule
OSBase_CSNetworkPortProviderModule
OSBase_NetworkPortImplementsIPEndpointProviderModule
[root@y1]#
```

4. Add a CIM user with the following command:

```
cimuser -a -u <userid> -w <password>
```

Example 4-16 shows the cimuser command from our lab.

Example 4-16 cimuser command

```
[root@y1]# cimuser -a -u ibmgrid -w passw0rd
User added successfully.
[root@y1]#
```

Note: Keep a record of the user ID and password that you add. They will be needed when adding a CMM Connection factory. For information about adding a CMM Connection factory, refer to 5.5.1, “Adding a connection” on page 100.

5. To stop the cimserver, enter the following command:

```
cimserver -s
```

4.2.6 Uninstall Pegasus and SBLIM

To uninstall Pegasus or SBLIM, refer to the appropriate Pegasus or SBLIM documentation for your operating environment.

4.3 GridFTP

Grid File Transfer Protocol (GridFTP) is an extension of the existing FTP standard for use in Grid environments. GridFTP provides advantages over traditional FTP in the following areas:

- Security
  - Transfers secured through integration with the Grid Security Infrastructure (GSI).
  - Certificates to identify grid users to those mapped in a hosts grid mapfile.
  - Use of x509 certificate authentication.
- Performance:
  - Enables tuning of buffer sizes for optimal performance.
  - Provides improved transfer speeds by using parallel transfers.

- Flexibility: Transfers can occur though HTTP, HTTPS, FTP, GSIFTP, and more.
- Reliability:
  - Implements standard FTP error detection and recovery technologies.
  - Enables partial file transfers supporting retransmission from the last checkpoint.

For additional information about GridFTP, refer to:

http://globus.org/datagrid/gridftp.html

4.3.1 Acquire GridFTP

GridFTP can be downloaded from the following FTP location:

ftp://ftp.globus.org/pub/gt3/3.0/contrib

Important: The FTP site contains many different versions of GridFTP. Make sure to select the appropriate one for your operating system.

In the lab environment for this book, the following GridFTP file was used:

```
gridftp-1.10-xl.tar.gz
```

4.3.2 Install GridFTP

Begin the installation of GridFTP by making a directory where the server files can be stored.

1. As root, run the following command:

   ```
   mkdir /opt/gridftp
   ```

2. Copy the file downloaded that you downloaded in the previous section, “Acquire GridFTP,” to the gridftp directory:

   ```
   cp /tmp/gridftp-1.10-xl.tar.gz /opt/gridftp/gridftp-1.10-xl.tar.gz
   ```

3. Use the **gunzip** command to unzip the GridFTP package.

   ```
   gunzip gridftp-1.10-xl.tar.gz
   ```

   Note: Some operating systems use a utility other than **gunzip**. Refer to the documentation for your operating system for details on how to unzip files.

4. Untar the GridFTP file with the following command:

   ```
   tar -xvf gridftp-1.10-xl.tar
   ```

Note: Some operating systems use a utility other than **tar**. Refer to the documentation for your operating system for details on how to untar files.

Tip: An alternative method to installing GridFTP is to use the -zxvf flags while running the **tar** command on the gridftp-1.10-xl.tar.gz file.

```
tar -zxvf gridftp-1.10-xl.tar.gz
```

Example 4-17 shows the installation process used in the lab environment.

Example 4-17 GridFTP installation

```
[root@y2 /]# mkdir /opt/gridftp
[root@y2 /]# cp /tmp/gridftp-1.10-xl.tar.gz
/opt/gridftp/gridftp-1.10-xl.tar.gz

[root@y2 /]#cd /opt/gridftp
[root@y2 gridftp]# gunzip gridftp-1.10-xl.tar.gz
[root@y2 gridftp]# tar -xvf gridftp-1.10-xl.tar
sbin/ftpcount
sbin/ftpshut
bin/ftprestart
bin/ftpwho
sbin/in.ftpd
etc/ftpaccess
etc/ftpconversions
man/man1/ftpcount.1
man/man1/ftpwho.1
man/man5/ftpaccess.5
man/man5/ftpconversions.5
man/man5/ftphosts.5
man/man5/ftpservers.5
man/man8/ftpd.8
man/man8/ftprestart.8
man/man8/ftpshut.8
[root@y2 gridftp]#
```

5. After the GridFTP files have been installed, copy the ftpaccess file used by GridFTP to the /opt/IBMGrid/etc/ directory.

Important: The ftpaccess file must be copied to /opt/IBMGrid/etc/ before starting a GridFTP server in order for all functions to work properly.

Example 4-18 on page 69 shows the command that is used to copy the ftpaccess file.

Example 4-18 Copying the ftpaccess file

```
[root@y1]# cp /opt/gridftp/etc/ftpaccess /opt/IBMGrid/etc/
[root@y1]#
```

4.3.3 Test GridFTP

After installation, GridFTP can be tested locally using a single host or multiple hosts if you have the resources available. This section discusses both methods.

Important: It is crucial for the dates and times on all machines involved in GridFTP to be synchronized. Failure to do so will cause GridFTP to fail.

Single host test

The single host test for GridFTP transfers a file from the local GridFTP server to another local directory using GSI. The purpose of this test is to simulate a transfer from a secure remote host using GSI.

1. Log on to the host as `root`.

   Important: Make sure to source the /opt/IBMGrid/igt-setenv.sh script to properly set up the grid environment. For more information about sourcing your environment, refer to 3.3.4, “Post installation” on page 46.

2. Run the following command to start the GridFTP server, replacing <port> with the desired port number:

   ```
   /opt/gridftp/sbin/in.ftpd -a -S -p <port>
   ```

   Example 4-19 shows the start of the lab GridFTP server using port 12345.

Example 4-19 Starting the GridFTP server

```
[root@y2 /] /opt/gridftp/sbin/in.ftpd -a -S -p 12345
[root@y2 /]
```

Tip: There may be more than one in.ftpd service on your machine. To start the GridFTP daemon and not another FTP daemon, be sure to use the fully qualified path, /opt/gridftp/sbin/in.ftpd, as shown in Example 4-19.

3. After the server has been initialized, log on as the ibmgrid user.
4. Ensure that you have a valid grid proxy. This can be checked by running the following command:

   ```
   /opt/IBMGrid/bin/grid-proxy-info
   ```

The results should be similar to those in Example 4-20.

Example 4-20 grid-proxy-info command

```
[ibmgrid@y2 /]# grid-proxy-info
subject  : /C=US/ST=Texas/O=ITSOLAB/CN=usery2/CN=991620727
issuer   : /C=US/ST=Texas/O=ITSOLAB/CN=usery2
identity : /C=US/ST=Texas/O=ITSOLAB/CN=usery2
type     : Proxy draft compliant impersonation proxy
strength : 512 bits
path     : /tmp/x509up_u500
timeleft : 11:59:59
[ibmgrid@y2 /]#
```

5. Make sure that the timeleft field has a value larger than 00:00:00. If it does not, create a new proxy with the following command:

   ```
   /opt/IBMGrid/bin/grid-proxy-init
   ```

   The results of this command should be similar to those in Example 4-21.

Example 4-21 grid-proxy-init command

```
[ibmgrid@y2 /]# grid-proxy-init
Your identity: /C=US/ST=Texas/O=ITSOLAB/CN=IBM Grid Toolbox V3 for
Multiplatforms
Enter GRID pass phrase for this identity:
Creating proxy ............................................. Done
Your proxy is valid until: Mon Mar  1 13:55:18 2004
```

6. For testing purposes, create a /tmp/file1 file. This is the file you are going to transfer.
7. Run the **globus-url-copy** command (the actual command line program that is used to transfer files in GridFTP):

   ```
   /opt/IBMGrid/bin/globus-url-copy gsiftp://localhost:<port>/tmp/file1
   file:///tmp/file2
   ```

   Example 4-22 shows the results of the single host test in our lab environment.

Example 4-22 Single host test results

```
[ibmgrid@y2 /]# /opt/IBMGrid/bin/globus-url-copy
gsiftp://localhost:12345/tmp/file1 file:///tmp/file2
[ibmgrid@y2 /]# ls -al /tmp
...(author omits text)...
-rw-r--r--    1 ibmgrid  ibmgrid        19 Mar  2 04:01 file1
-rw-r--r--    1 ibmgrid  ibmgrid        19 Mar  2 04:05 file2
...(author omits text)...
[ibmgrid@y2 /]
```

Tip: If the transfer fails, the most common cause is a discrepancy in the system clocks of the server or the host, or the valid date range of the certificates. If you receive a message similar to Example 4-23, check the time and date settings for all machines that are involved in the transfer.

Example 4-23 Sample globus-url-copy error

```
[ibmgrid@y2 /]# /opt/IBMGrid/bin/globus-url-copy
gsiftp://localhost:12345/tmp/file1 file:///tmp/file2
error: globus_l_ftp_control_send_cmd_cb: gss_init_sec_context failed

 GSS failure:
GSS Major Status: Authentication Failed
GSS Minor Status Error Chain:

init_sec_context.c:169: gss_init_sec_context: SSLv3 handshake problems
globus_i_gsi_gss_utils.c:879: globus_i_gsi_gss_handshake: Unable to verify
remote side's credentials
globus_i_gsi_gss_utils.c:851: globus_i_gsi_gss_handshake: SSLv3 handshake
problems: Couldn't do ssl handshake
OpenSSL Error: s3_clnt.c:840: in library: SSL routines, function
SSL3_GET_SERVER_CERTIFICATE: certificate verify failed
globus_gsi_callback.c:349: globus_i_gsi_callback_handshake_callback: Could not
verify credential
globus_gsi_callback.c:416: globus_i_gsi_callback_cred_verify: The certificate
is not yet valid: Cert with subject:
/C=US/ST=Texas/O=ITSOLAB/CN=host/x3.itsolab-xingu.com is not yet valid- check
clock skew between hosts.
```

Multiple host test

The multiple host test is similar to the single host test, with one exception: The files being transferred are going from one host to another instead of staying within the same host.

1. To begin the test, follow the procedure in 4.3.2, “Install GridFTP” on page 67.
2. Next, run the single host test in “Single host test” on page 69 to ensure that you can transfer files with GridFTP on the localhost.

Important: The single host test must work before running the multiple host test.

3. When the single host test is successful, two changes must be made on each GridFTP host in order to allow bi-directional secure file transfers:

   a. Make sure that CA certificates are installed on each host for each CA they may have to deal with. For example, in a two-host scenario in which each host used a different CA, each host must have two CA certificates installed (one for each CA used). For instructions for installing CA certificates, refer to 3.3.5, “Securing the grid” on page 47.

      Note: When installing additional CA certificates it is not necessary to include the -default flag.

   b. Make sure to update your grid-mapfile to include the distinguished name of any external user who may access your GridFTP server. To add entries to your grid-mapfile, refer to “Adding a grid mapfile entry” on page 53.

4. For testing purposes, create a /tmp/file1 file. This is the file that you are going to transfer.

   To transfer a file from your local file system to a remote host, use the following command:

   ```
   /opt/IBMGrid/bin/globus-url-copy file:///tmp/file1 \
   gsiftp://x3.itsolab-xingu.com/tmp/remote-file1
   ```

   To transfer a file from a remote host to your local file system, use the following command:

   ```
   /opt/IBMGrid/bin/globus-url-copy
   gsiftp://x3.itsolab-xingu.com/tmp/remote-file1 file:///tmp/file2
   ```

   Note: The command above should appear as a single line.

   Important: In order to work properly, the host names that are used must match the host name supplied by your host certificates.

   Example 4-24 demonstrates the use of both **globus-url-copy** commands.

Example 4-24 globus-url-copy command

```
[ibmgrid@y2 /]#/opt/IBMGrid/bin/globus-url-copy file:///tmp/file1
gsiftp://x3.itsolab-xingu.com/tmp/remote-file1

[ibmgrid@y2 /]#/opt/IBMGrid/bin/globus-url-copy
gsiftp://x3.itsolab-xingu.com/tmp/remote-file1 file:///tmp/file2

[ibmgrid@y2 /]#
```

Note: When GridFTP is functioning, you can use the Reliable File Transfer (RFT) service that is part of the IBM Grid Toolbox. For more information about RFT, refer to 6.7, “Reliable File Transfer” on page 147.

4.3.4 Configure GridFTP

No configuration is required for GridFTP to function properly. However, you can modify **inetd** or **xinetd** to start a GridFTP server automatically. This section covers how to modify both.

As root, edit the /etc/services and add the following line. The default port is 2811.

```
gsiftp <port>/tcp
```

Attention: In some cases, the `gsiftp <port>/tcp` line has already been added to /etc/services by the grid installation, so no edit is necessary.

inetd configuration

1. If you are running AIX, edit the /etc/inetd.conf file as root and add the following line:

   ```
   gsiftp stream tcp nowait root /usr/bin/env env
   LIBPATH=/opt/IBMGrid/lib/opt/gridftp/in.ftpd -l -a -G /opt/gridftp
   ```

   If you are running Linux, edit the /etc/inetd.conf file as root and add the following line:

   ```
   gsiftp stream tcp nowait root /usr/bin/env env
   LD_LIBRARY_PATH=/opt/IBMGrid/lib/opt/gridftp/in.ftpd -l -a -G
   /opt/gridftp
   ```

   Important: The lines above should appear as a single line.

   Note: If you do not provide a -G flag, the GridFTP server will default to the /etc/ftpaccess file.

2. Refresh the inet daemon:
   - In AIX, use the following command:

     ```
     refresh -s inetd
     ```

   - In Linux, use the following command:

     ```
     killall -HUP inetd
     ```

Tip: On some systems these commands may have no effect. If your system does not respond use the following command:

```
ps aux | grep inetd | awk '{print $2}' | xargs kill HUP
```

xinetd configuration

1. As root, add a file called grid-ftp to the /etc/xinet.d/ directory with the following content:

```
service gsiftp
{
instances = 1000
socket_type = stream
wait = no
user = root
env = LD_LIBRARY_PATH=/opt/IBMGrid/lib
server = /opt/gridftp/sbin/in.ftpd
server_args = -l -a -G /opt/gridftp
log_on_success += DURATION USERID
log_on_failure += USERID
nice = 10
disable = no
}
```

2. Restart xinetd with the following command:

```
/etc/rc.d/init.d/xinetd restart
```

4.3.5 Uninstall GridFTP

GridFTP is not uninstalled as part of the regular IBM Grid Toolbox uninstall procedure. To uninstall GridFTP, you must manually remove the files that were installed with GridFTP:

1. As root, run the following command from the /opt directory:

```
rm -rf gridftp
```

2. In addition, you must manually reverse any changes to configuration files that you modified for GridFTP. Some of the files that may have been modified are:
   - /etc/services
   - /etc/inetd.conf
   - /etc/xinetd.d/grid-ftp
   - /etc/ftpaccess

5

Managing

This chapter provides information about the management, administration, and deployment of grid services with the IBM Grid Toolbox and the IBM Grid Services Manager. It does not discuss managing a grid in general but is specific to the IBM Grid Toolbox. The following topics are discussed:

- IBM Grid Services Manager
- Deploying and undeploying services
- Managing Information Services
- Managing a policy
- Managing connections for Common Management Model (CMM) Services
- Backing up a grid

5.1 IBM Grid Services Manager

The IBM Grid Services Manager is the user's interface to managing grid services. It is a Web application and is accessed with a Web browser. Each install of the IBM Grid Toolbox is set up with its own IBM Grid Services Manager. However, you can add multiple instances of the IBM Grid Toolbox to a single IBM Grid Services Manager, giving you the capability to manage several instances from one place.

Tip: When using the IBM Grid Services Manager, you can get page-specific help information by clicking on the help icon (the ? character in a circle).

5.1.1 Starting the IBM Grid Services Manager

Complete the following steps from a Web browser on a system that has network access to the IBM Grid Toolbox instance to be managed. The instance must have the container started with the **igt-start-container** command.

1. Start your Web browser and open the IBM Grid Services Manager at the following URL:

   `http://hostname:port/gsm/`

   - *hostname* is the host of the instance you are working with.
   - *port* is the port you are using.The default port is 12080.

   Note: You must have cookies enabled on your Web browser to access the IBM Grid Services Manager.

2. Log on to the IBM Grid Services Manager using a user ID and password for this instance. The default user ID is `ibmgrid`. Figure 5-1 shows the IBM Grid Services Manager logon page from our lab.

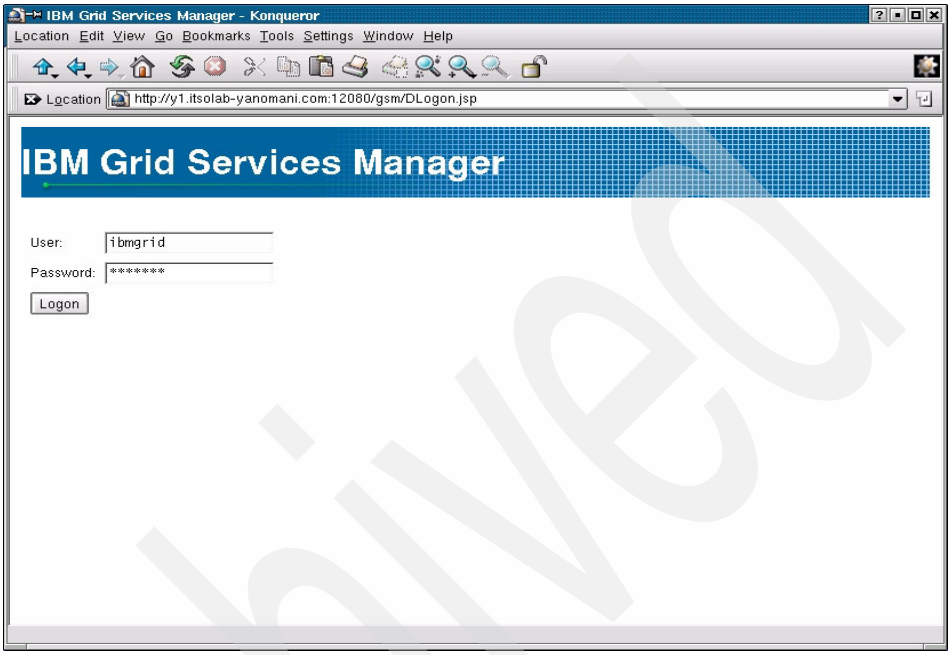


Figure 5-1 IBM Grid Services Manager logon page

3. If the GSI proxy has expired or is missing, the GSI Proxy Generation page appears.

   If this page appears, it is time to regenerate a long-term proxy according to your local policy.

   To continue with a temporary short-lived proxy, enter the pass phrase that was used to create your user certificate request and click **Create Proxy**. Figure 5-2 shows the GSI Proxy Generation page from our lab.

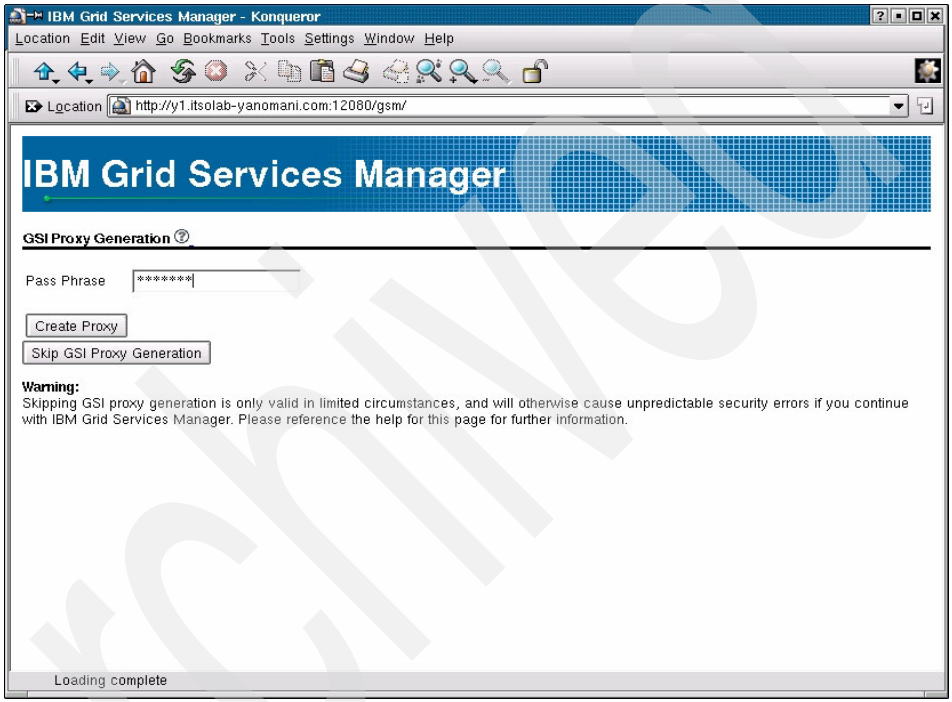


Figure 5-2 GSI Proxy Generation page

4. The Manage Available Instances page appears if everything is entered correctly. An example from our lab is shown in Figure 5-3.

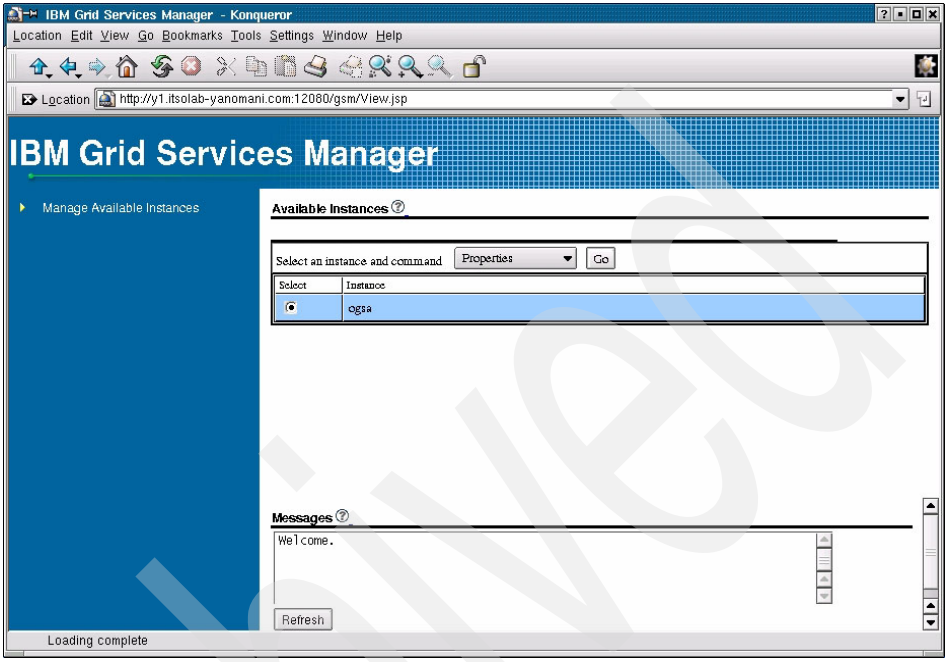


Figure 5-3 Manage Available Instances page

Adding a user

The default user ID created during the installation of the IBM Grid Toolbox is ibmgrid. You can add other user IDs that can access the IBM Grid Services Manager using the **igt-add-user** and **igt-set-admin-user** commands.

Note: In our lab, while testing the **igt-add-user**, **igt-set-admin-user**, and **igt-delete-user** commands, we encountered the following error:

```
Error code: JAVA java0x5
ADMA0094E: Application Management operation setApplicationInfo is not
available in your WebSphere installation.
```

Correcting this problem requires a service pack to the IBM Grid Toolbox. Refer to Appendix J, “Software support for the IBM Grid Toolbox” on page 259 for information about obtaining service packs for the IBM Grid Toolbox for your server platform.

Example 5-1 shows the results of the commands that we executed in our lab. Notice that there is no output from the **igt-add-user** command.

Example 5-1 igt-add-user and igt-set-admin-user commands

```
[ibmgrid@z1 IBMGrid]$ igt-add-user ibmtest -p passw0rd -n "Ibmtest Fullname"
[ibmgrid@z1 IBMGrid]$ igt-set-admin-user ibmtest
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7357I: By request, this scripting client is not connected to any server
process. Certain configuration and application operations will be
available in local mode.

Starting processing
Processing user: ibmtest
ibmtest
Finished processing users
ibmtest
{Administrator No No ibmtest {}}
Added users, ibmtest , to administrator role

Saving configuration
[ibmgrid@z1 IBMGrid]$
```

Tip: In our lab, we added the new admin user while the IBM Grid Toolbox was running. To be able to use the new user, we had to stop and then start the container.

To display current users, issue the **igt-list-user** command.

Removing a user

To remove an user ID that can access the IBM Grid Services Manager, issue the following command:

```
igt-delete-user
```

Example 5-2 shows the results of the command that we executed in our lab. Notice that there is no output from this command.

Example 5-2 igt-delete-user command

```
[ibmgrid@z1 IBMGrid]$ igt-delete-user ibmtest
[ibmgrid@z1 IBMGrid]$
```

Note: In our lab, while testing the **igt-add-user**, **igt-set-admin-user**, and **igt-delete-user** commands, we encountered the following error:

```
Error code: JAVA java0x5
ADMA0094E: Application Management operation setApplicationInfo is not
available in your Websphere installation.
```

A service pack for the IBM Grid Toolbox is required to correct this problem. Refer to Appendix J, “Software support for the IBM Grid Toolbox” on page 259 for information about obtaining service packs for the IBM Grid Toolbox for your server platform.

5.1.2 Adding instances

To add an instance to the list of available instances in the IBM Grid Services Manager, complete the following steps:

1. Select **Add instance** from the list of commands in the pull-down menu.
2. Click **Go**.

3. On the Add Instance page, enter the Instance name, Host name, Port, and Application context and click **OK** to add the instance. Figure 5-4 shows how we completed the Add Instance page in our lab.

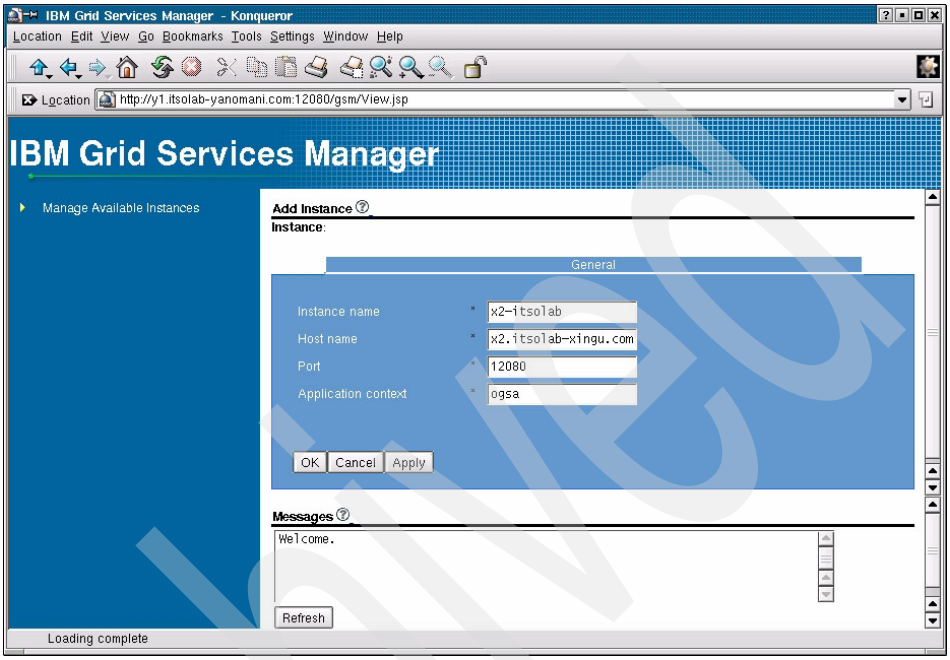


Figure 5-4 Add Instance page

Figure 5-5 shows the results of Add Instance.

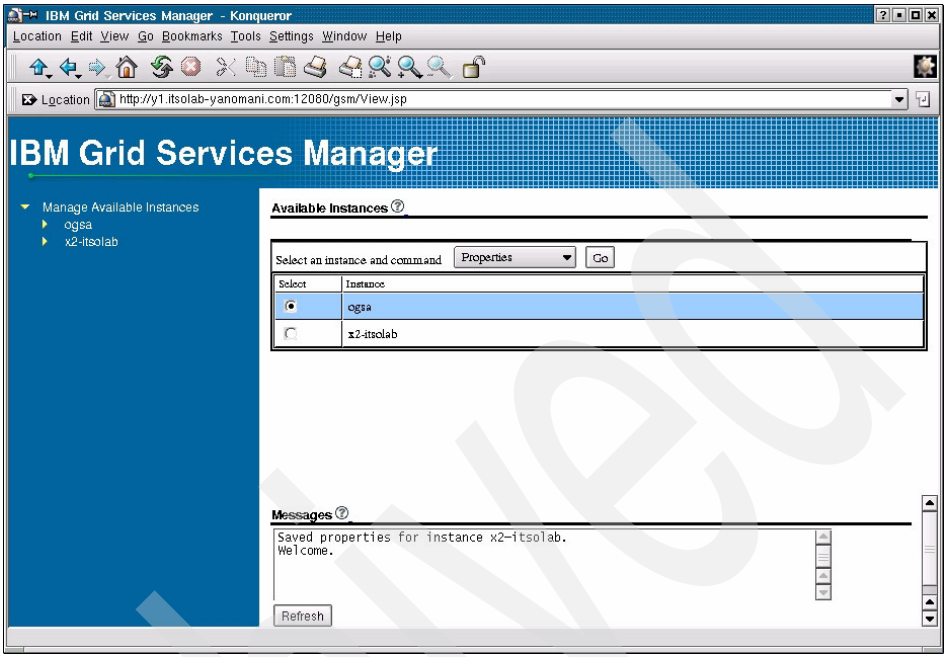


Figure 5-5 Results of adding an instance

Tip: If you want to change the name of any instance in the list of Manage Available Instances, including the default `ogsa` instance, you can add the instance with the name you desire using the steps outlined in this section, and then remove the ogsa instance you are trying to rename by using the steps in 5.1.3, “Removing instances” on page 83. In summary, you must remove whatever instance you intend to rename.

5.1.3 Removing instances

To remove an instance from the list of available instances in the IBM Grid Services Manager, complete the following steps:

1. Select the instance to be removed.
2. Select **Remove instance** from the list of commands in the pull-down menu.
3. Click **Go**.
4. Figure 5-6 on page 84 shows the confirmation page for removing this instance. Click **OK**.

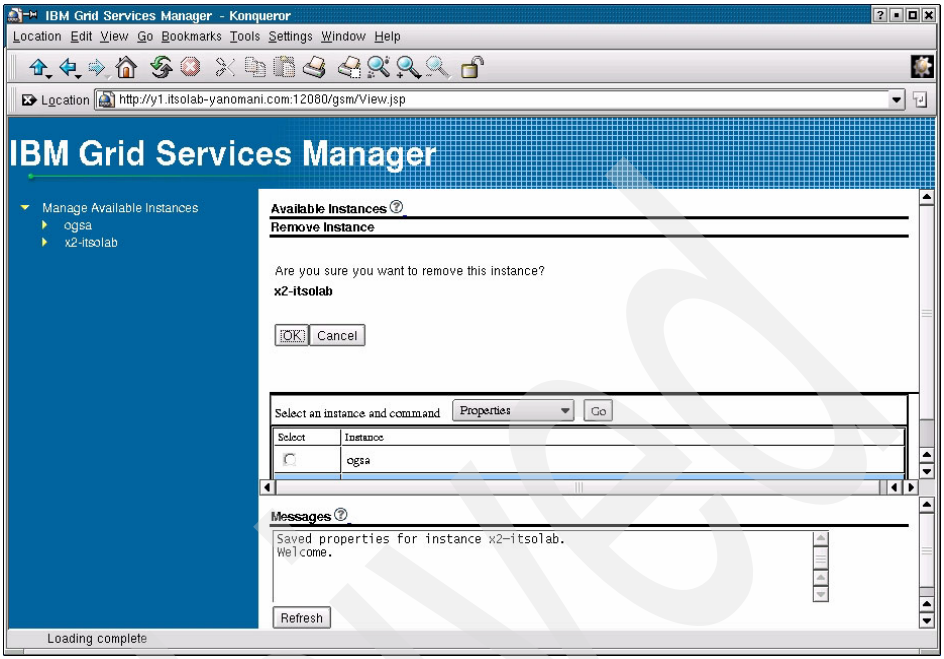


Figure 5-6 “Are you sure?” prompt

5.1.4 Viewing and editing properties, statistics, and logging

Properties

To view and edit the properties of an instance, complete the following steps from the Available Instances page:

1. Select the instance whose properties you want to view.
2. Select **Properties** from the list of commands.
3. Click **Go**.
4. In the Instance Properties page that appears, you can change the following items:

Host name	The host name or IP address of the system hosting the OGSI instance you have selected
Port	The TCP/IP port number on which the selected OGSI instance is listening
Application context	The Enterprise Application Context for the Web application associated with the OGSI instance you have selected

5. If desired, enter a change. Click **OK** to make changes.

   Figure 5-7 is an Instance Properties page from our lab.

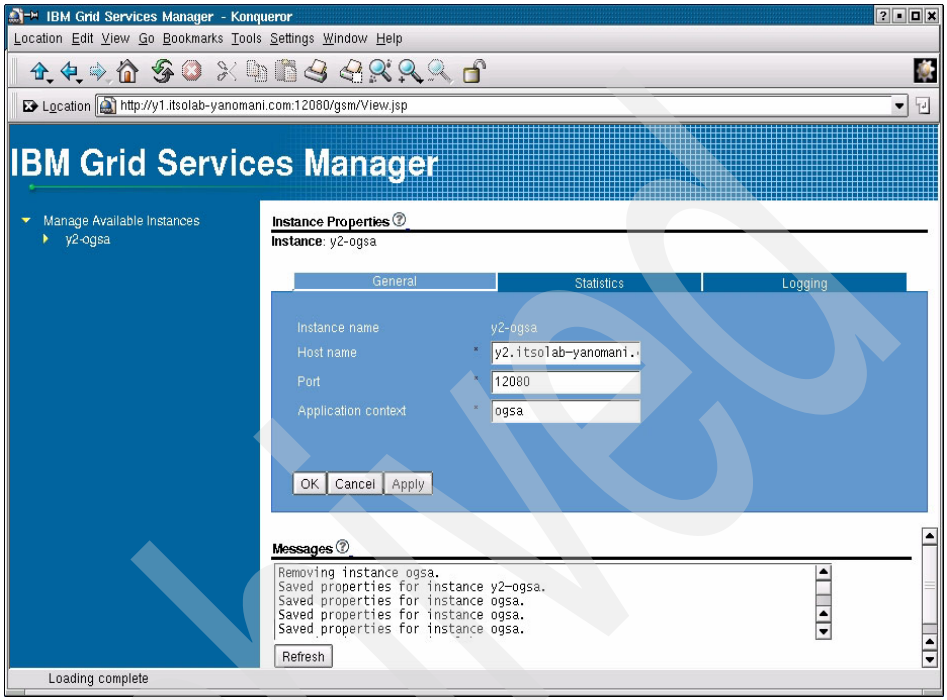


Figure 5-7 Instance properties page

Statistics

To display information about the grid services that were deployed to this OGSI instance, click the **Statistics** tab.

The following information is displayed:

Available services The total number of active and inactive services. This total might include services that are created during run-time, (for example, by a factory service).

Deployed services The total number of services that were deployed on the OGSI instance at start-up.

Active services The number of services that were started by a consumer application or by the IBM Grid Services Manager and are currently running.

Inactive services The number of services that were deployed using the available tooling, but not started by a consumer application or the IBM Grid Services Manager.

Figure 5-8 shows a Statistics page from our lab.

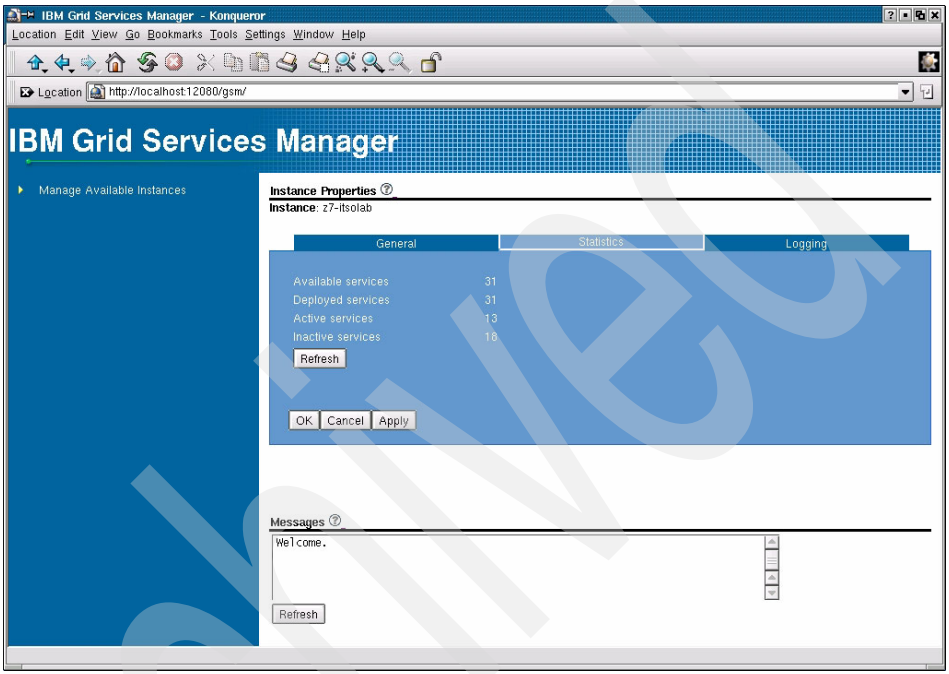


Figure 5-8 Statistics

Logging

To configure logging, click the **Logging** tab. The Logging properties page enables you to configure the logging level of the existing loggers for an OGSI instance and to organize these loggers into groups for easier management.

Figure 5-9 shows a portion of the logging pane from our lab.

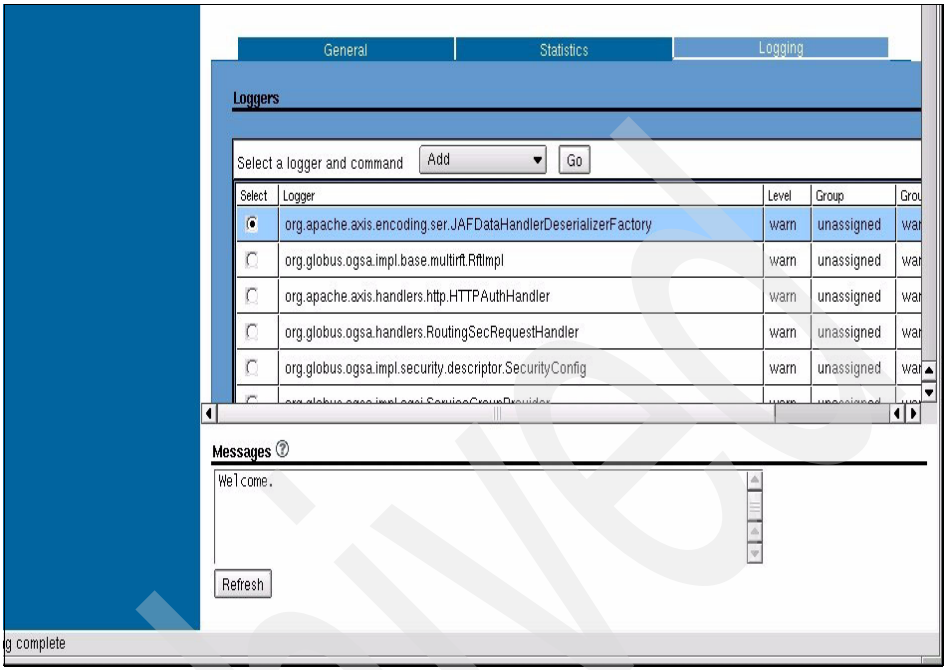


Figure 5-9 Logging pane

To change a logger, complete the following steps on the Loggers page:

1. Select the logger whose properties you want to change.
2. Select a command from the pull-down menu. Available commands are **Add**, **Remove**, **Edit**, **Edit Group Level**, and **Remove Group**.
3. Click **Go**.

You can set each Logger's level to Debug (All), Trace, Info, Warning, Error, Fatal, or None (Off). Loggers will start receiving the selected type of information as soon as they are added, with the exception of Trace data. You can group and manage Loggers as a set by specifying a group name. Specifying a group can make it easy to change the logging level for an entire set of loggers, rather than changing each one individually. A logger can be added or removed from a group by changing its group property.

Figure 5-10 shows the result of an **Edit** command in our lab.

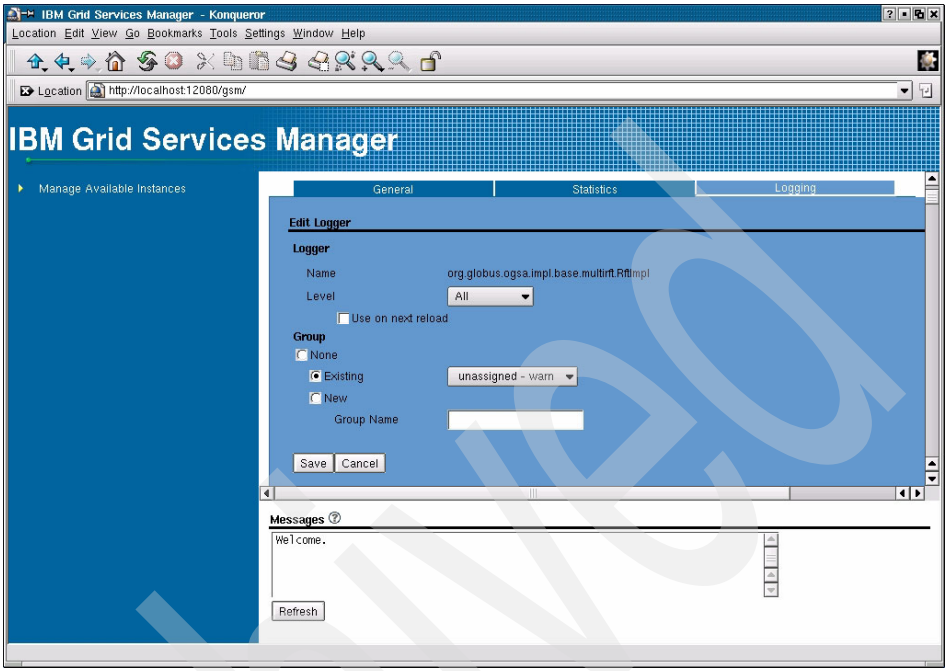


Figure 5-10 Edit Logger command

For more information about logging, use the help icon (the ? character in a circle) from the Instance Properties page. For information about log files, refer to Appendix G, “Logging & Error Messages” on page 245.

5.1.5 Managing a grid service

The IBM Grid Services Manager enables you to manage the status of deployed grid services and to inspect the service data, port types, and WSDL associated with a grid service.

To work with grid services, select the instance that is hosting the service you want to work with on the navigation tree in the left pane. Then click **Manage Grid Services**.

This displays the list of all grid services that are available on the selected OGSI instance, and shows the name, state, and grid service handle for each of those services.

Service The service name specified in the service definition. This is generated using the tooling.

Status A grid service can be either active or inactive, depending on whether it was started. A service can be started by either the IBM Grid Services Manager or a consumer application. Inactive services have a defined address but they are not currently using system resources such as memory or CPU. Active grid services are running and use system resources.

Grid service handle The grid service handle is a unique identifier that is assigned to the grid service. Other applications and grid components that work with the grid service identify it through this handle.

Manage the status of a service

To change the state of a service (from INACTIVE to ACTIVE, for instance), complete the following steps:

1. Select the service you want to change.
2. Select **Activate Service**, **Deactivate Service**, **End Service**, or **End Service Force** from the list of commands in the pull-down menu.
3. Click **Go**.

> **Important:** Verify that you have selected the correct service to be activated or ended. The action is taken immediately after you click **Go**.

Table 5-1 provides information about each command.

Table 5-1 Description of manage status commands

Command	Description
Activate Service	Starts an inactive grid service.
Deactivate Service	Stops the grid service so that it is no longer consuming system resources.
End Service	Terminates the grid service so that it is no longer available for use. Ended services are not listed in the IBM Grid Services Manager until the instance is restarted. This command *attempts* to use the destroy operation to remove the grid service.
End Service Force	Terminates the grid service and does not depend on the service's destroy operation.

Viewing service information

To view the service data information, complete the following steps:

1. Select the service you want to view.
2. Select **View Service Data Descriptions**, **View Service Port Types**, or **View Service WSDL** from the list of commands in the pull-down menu.
3. Click **Go**.

Each command will be discussed in more detail. Table 5-2 provides brief information about each command.

Table 5-2 Description of view command

Command	Description
View Service Data Descriptions	Displays a list of service data descriptions that are contained in the selected grid service. The command enables you to inspect the service data element's current value in XML.
View Service Port Types	Displays a list of port types that are used by the selected grid service. The command enables you to view the operations that are supported on each port type and the attributes and input/output parameters for each operation.
View Service WSDL	The command displays the complete WSDL file that is associated with the selected grid service.

View Service Data Descriptions

The Service Data Descriptions pane lists all of the service data that is provided by the selected grid service. The following information is listed:

Service Data Name The name assigned to a service data element during the service's implementation.

Port Type The port type that contains this service data element. This was defined during the service's implementation.

Min occurs Indicates the minimum number of SDE values that can appear in the service instance's SDE values or the portType staticServiceDataValues. If minOccurs equals zero, then this SDE is optional. Default value = 1.

Max occurs Indicates the maximum number of SDE values that can appear in the service instance's SDE values or the portType staticServiceData Values. Default value = 1.

Mutability Defines whether the service data element can be changed during run time.

Type	Defines the XML schema type of the service data value.
Modifiable	A mechanism to specify a read-only or write-only service data element. If writable, you can use setServiceData to change its SDE value based on mutability, min, and max constraints. Default value = false (all SDEs are by default read-only).
Nillable	Indicates whether an SD value can have a nil value. You can declare this SDE value as true. Default value = false.
Namespace URI	The namespace URI for this service data, as defined in the grid service implementation.

Figure 5-11 is a portion of the Service Data Description pane from our lab, after selecting **View Service Data Descriptions**.

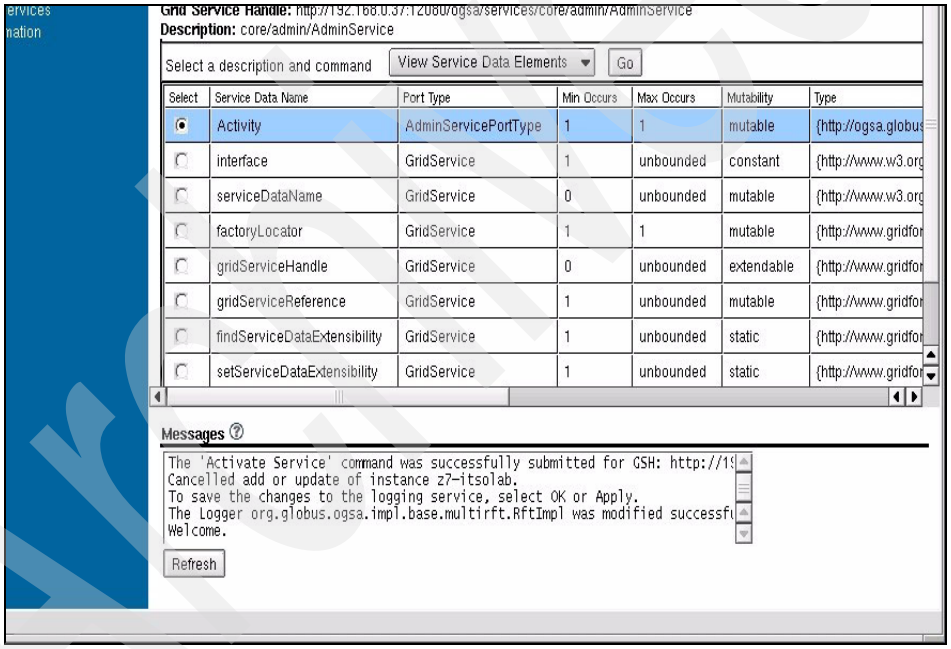


Figure 5-11 Service Data Description pane

You can view the contents of the service data by doing the following:

1. Select the service data description you want to inspect.
2. Select **View Service Data Elements** from the list of commands.
3. Click **Go**.

View Service Port Types

Every grid service implements one or more port types as defined by the grid service specifications. The Service Port Types pane displays each of the port types implemented by the selected grid service.

Figure 5-12 is a portion of the Service Port Types pane from our lab, after selecting **View Service Port Types**.

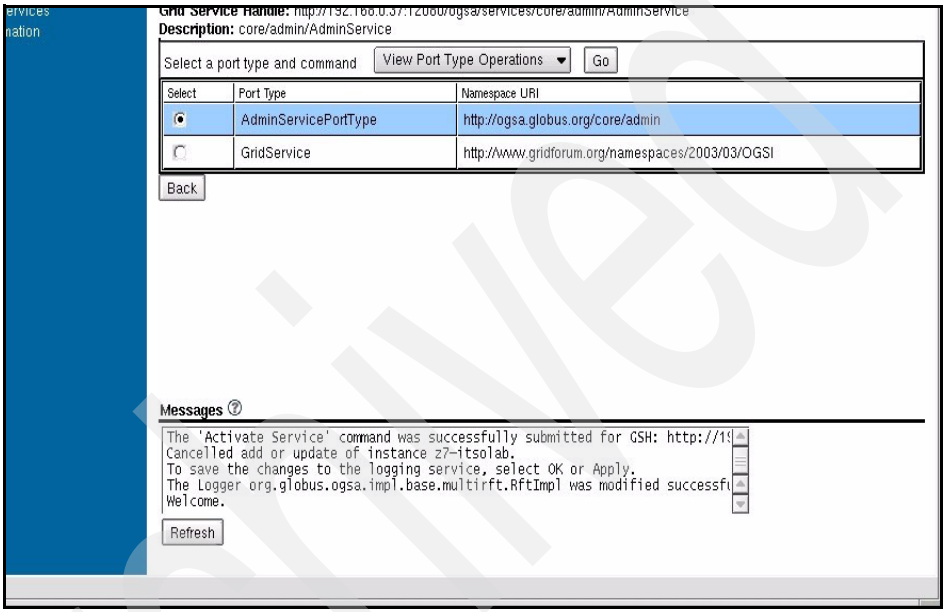


Figure 5-12 Service Port Types

You can also display the operations that are supported on each of the port types by doing the following:

1. Select the port type you want to inspect.
2. Select **View Port Type Operations** from the list of commands.
3. Click **Go**.

If you have selected a factory port type, you can create a new service from that factory by doing the following:

1. Select the factory port type.
2. Select **Create New Service From Factory** from the list of commands.
3. Click **Go**.

Figure 5-13 is a portion of the Service Port Type Operations pane from our lab, after selecting **View Port Type Operations**.

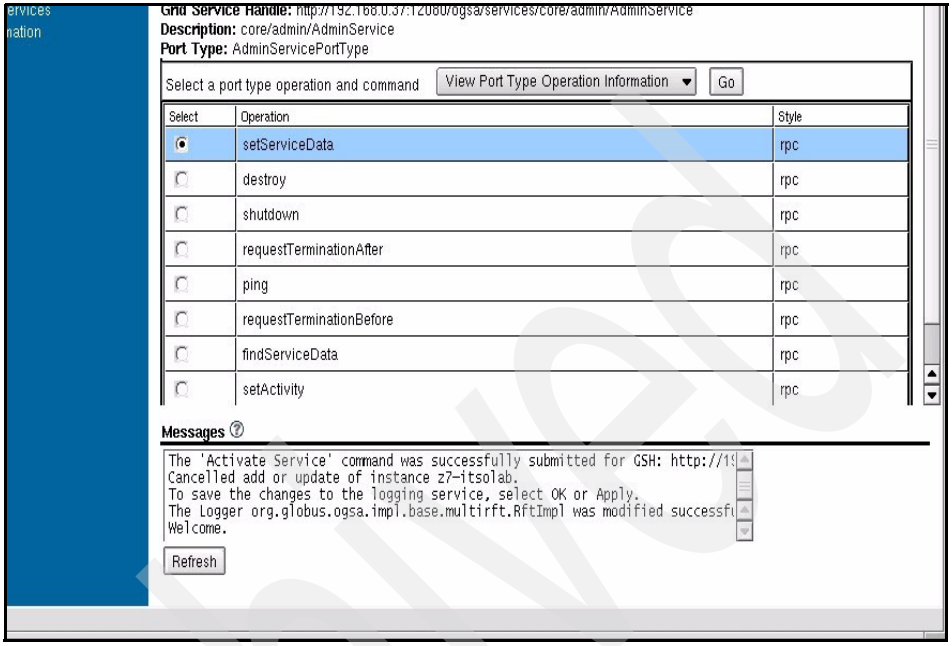


Figure 5-13 Service Port Type Operations pane

The Service Port Type Operations pane displays the operations that are available for a selected port type, including the following:

Name Name of the port type operation as defined during the service's implementation.

Style Port type operation can be implemented as either a document or remote procedure call (`rpc`). The style is determined during the service's implementation.

You can also view the attributes and input/output parameters for a selected port type operation by doing the following:

1. Select the Port Type Operation you want to inspect.
2. Select **View Port Type Operation Information** from the list of commands.
3. Click **Go**.

View Service WSDL

The Service WSDL Contents pane displays the complete WSDL file associated with the selected grid service.

Figure 5-14 is a portion of the Service WSDL Contents pane from our lab, after selecting **View Port Type Operations**.

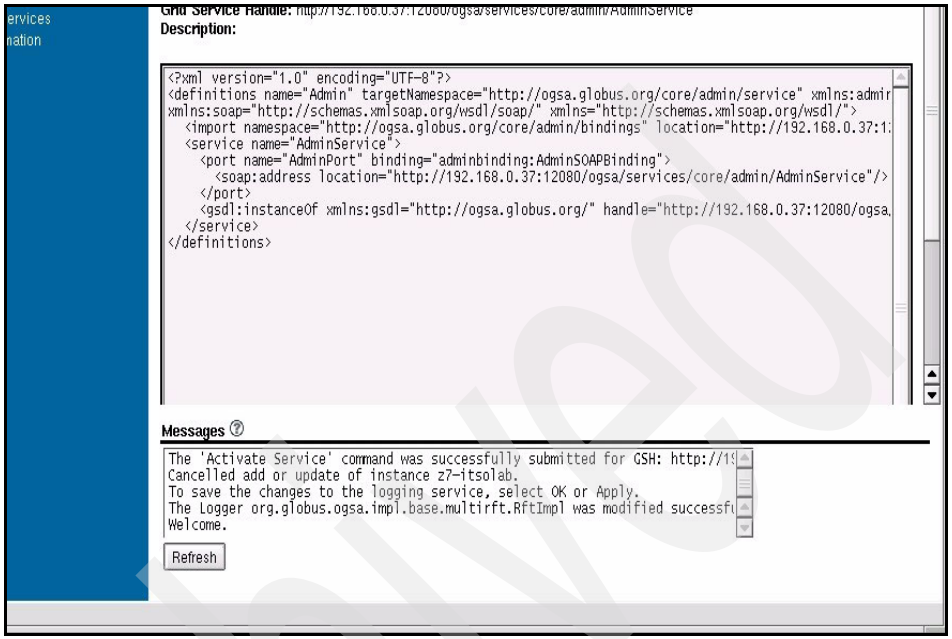


Figure 5-14 Service WSDL Contents pane

5.1.6 Stopping the IBM Grid Toolbox instance

To stop the instance of the IBM Grid Toolbox, you can stop the IBM Grid container (Web server) with the following command:

```
igt-stop-container
```

5.2 Deploying and undeploying grid services

The IBM Grid Toolbox enables the deploying and undeploying of services using a simple command line interface. This section illustrates how to deploy and undeploy a service using the basic counter service as an example.

5.2.1 Deploying

To deploy a new service:

1. Execute the **igt-deploy-gar** command as `ibmgrid` with the following syntax:

   ```
   igt-deploy-gar /<directory>/<gar name>
   ```

   Tip: Before deploying a service, make sure to source the igt-setenv.sh script.

   The basic counter service is in /opt/IBMGrid/gars/IGTCounterSamples.gar

   Example 5-3 shows how we deployed the basic counter service in the lab environment.

Example 5-3 igt-deploy-gar command

```
[ibmgrid@y2 /]$ igt-deploy-gar /opt/IBMGrid/gars/IGTCounterSamples.gar
2004-03-08 08:50:37: Starting
Number of gars to deploy: 1
   /opt/IBMGrid/gars/IGTCounterSamples.gar
...(author omits output)...
Generating the undeployment information ...
2004-03-08 08:51:08: Done processing
[ibmgrid@y2 /]$
```

2. Restart the container. First, stop the container with the **igt-stop-container** command and then start it with the **igt-start container** command.
3. To verify that the services have been deployed use the Managed Grid Services feature of the Grid Service Manager as described in 5.1.5, “Managing a grid service” on page 88.

Testing the basic counter service

To test the basic counter service and verify that is was properly deployed:

1. Open a Web browser and go to the following address:

```
http://<hostname>:<port>/ogsa/samples/counters/basic
```

The resulting Web application should look similar to Figure 5-15.

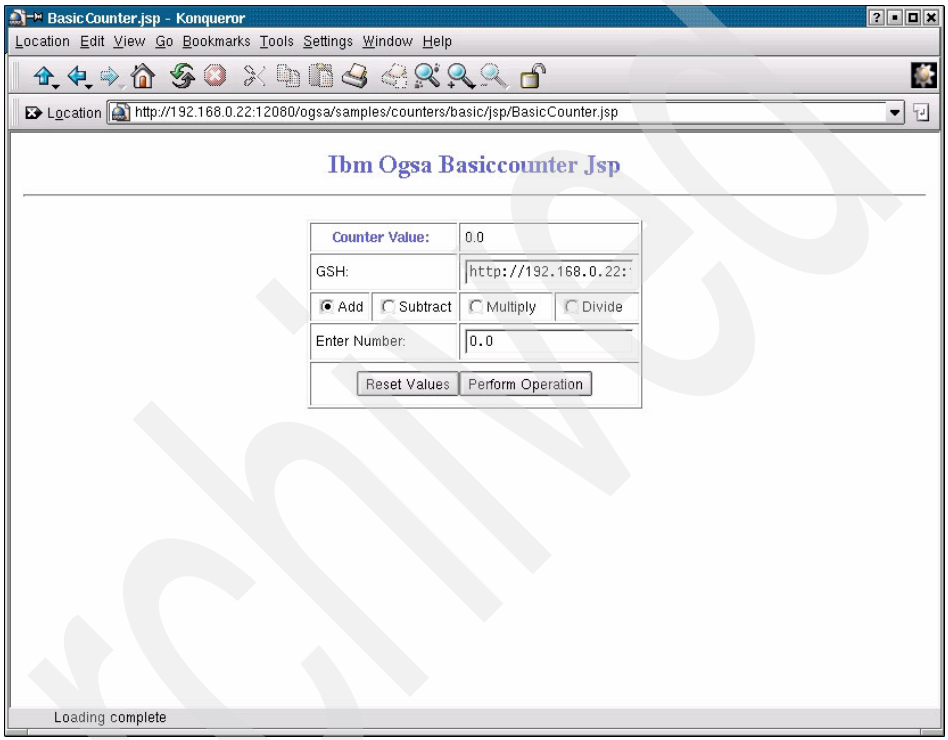


Figure 5-15 Basic counter service application

2. Restart the container. First stop the container with the **`igt-stop-container`** command, then start it with the **`igt-start container`** command.

5.2.2 Undeploying

To undeploy a service, execute the **`igt-undeploy-gar`** command as `ibmgrid`. The syntax is:

```
igt-undeploy-gar <gar ID>
```

The **`igt-deploy-gar`** command takes the name of the gar they deployed minus the extension (.gar) as its argument.

Example 5-4 shows the command and output for undeploying the basic counter service.

Example 5-4 igt-undeploy-gar

```
[ibmgrid@y2 /]$ igt-undeploy-gar IGTCounterSamples
2004-03-08 08:49:56: Starting
2004-03-08 08:49:56: Running
/opt/IBMGrid/undeploy/IGTCounterSamples-undeploy.sh
Undeploying IGTCounterSamples
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
2004-03-08 08:50:19: Done processing
[ibmgrid@y2 /]$
```

5.2.3 Updating a deployed service

To update an already deployed service, use the following steps as ibmgrid:

1. Update the service artifacts, and create a new gar file.
2. Undeploy the old service as described in 5.2.2, “Undeploying” on page 96.
3. Deploy the new service as described in 5.2.1, “Deploying” on page 95.
4. If the services container is running, first stop the container using **igt-stop-container** and then restart it using **igt-start-container**.

5.2.4 Adding security

When deploying services other than those provided by the IBM Grid Toolbox, it is possible to interact with those services through the IBM Grid Services Manager (GSM). Add the service’s security settings to the following configuration file:

```
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/OGSAConfig.war
/schema/InstanceSecurityConfig_ogsa.xml
```

Example 5-5 is an example of the security settings for a service in the InstanceSecurityConfig_ogsa.xml file.

Example 5-5 Service security settings example

```
<service name="base/multirft/MultiFileRFTFactoryService">
   <security>
      <authentication>
         <secureConversation>
```

```
            <protection>Privacy</protection>
            <delegation>LIMITED</delegation>
         </secureConversation>
      </authentication>
      <authorization>
         <clientAuthorization>NONE</clientAuthorization>
      </authorization>
   </security>
</service>
```

Note: If the new service resides on another instance of the IBM Grid Toolbox and it does not already exist, you must add that instance in the IBM Grid Services Manager first. For adding an instance to GSM, see 5.1.2, "Adding instances" on page 81. This creates a new security file for that instance called InstanceSecurityConfig_<instance name>.xml. Edit this file to interact with the new service using the IBM Grid Services Manager.

5.3 Managing Information Services

The IBM Grid Toolbox follows the OGSI specification for Information Services. This section describes some Information Services attributes that are specific to the IBM Grid Toolbox. For information about Information Services, see Chapter 2, "Planning" on page 11.

5.3.1 Information Services startup status

When the IBM Grid Toolbox is installed and the instance is started with the **igt-start-container** command, the following services are INACTIVE the first time the Toolbox is used:

- core/registry/ContainerRegistryService
- ogsi/NotificationSubscriptionFactoryService
- Index Service

Depending on what Information Services function you intend to use, you must ACTIVATE one or more of these services using the process defined in 5.1.5, "Managing a grid service" on page 88.

5.3.2 File location

The configuration file for grid service indexing is called index-service-config.xml. The path of the configuration file is specified by the parameter serviceConfig in the Index Service deployment descriptor in the server-config.wsdd file.

For the IBM Grid Toolbox instance, the server-config.wsdd file is located in this directory:
/opt/IBMGrid/Appserver/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF

Example 5-6 lists part of the IndexService section from the server-config.wsdd file for the IBM Grid Toolbox. Notice that the value of the serviceConfig parameter is `etc/index-service-config.xml`.

Example 5-6 IndexService section of server-config.wsdd file

```
</service>
 <service name="base/index/IndexService" provider="Handler" style="wrapped"
... (author omits text) ...
<parameter name="serviceConfig" value="etc/index-service-config.xml"/>
... (author omits text) ...
<parameter name="name" value="Index Service"/>
  <parameter name="xindiceEnabled" value="false"/>
 </service>
```

Consequently, the index-service-config.xml file is located in this directory:
/opt/IBMGrid/Appserver/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/etc

For a detailed description of the index-service-config.xml file and how to use it, refer to the Globus Web pages at:

http://www.globus.org/ogsa/releases/final/docs/infosvcs/indexsvc_ug.html

5.4 Managing a policy

With Policy Services, administrators can define a set of business goals and establish a set of rules that enable their grid to reach those goals. Policy Service Manager (PSM) is the administrator's interface to add, update, remove, retrieve, and deploy policies.

The IBM Grid Toolbox includes a sample application that demonstrates the Policy framework and Policy management. Deploying and following the sample application is a good way to learn about managing policies. For more information, read 6.6, "Policy application sample" on page 126.

5.5 Managing connections for CMM Services

The IBM Grid Toolbox Common Management Model (CMM) Services provide the infrastructure that is required to represent an instrumented resource as a grid service.

CMM Services communicates with an underlying Common Information Model Object Manager (CIMOM), like Pegasus. The CIMOM returns the requested information about the instrumented resource to the CMM Services. CMM Services uses the mapping framework to convert the CIM data and operations into information that is useful in a grid context.

For this communication to take place, a connection factory must be defined.

5.5.1 Adding a connection

After CMM Services have been deployed, complete the following steps to create a connection factory:

1. Log in as the ibmgrid user.
2. Start the container with the **igt-start-container** command.
3. Enter the following command to create a connection factory:

```
igt-add-cmmconnectionfactory \
   -cfName <name of connection factory> \
   -jndiName <jndi name of connection factory> \
   -serverName <shortname or IP address of CIMOM server> \
   -portNumber <port on which CIMOM listens> \
   -userName <CIMOM username> \
   -password <CIMOM password> \
   -nameSpace <CIMOM namespace>
```

All values are required. The values for -cfName and -jndiName are names of your choosing. Values for the remaining parameters must match the target CIMOM environment.

Example 5-7 illustrates how the command was entered in our lab environment and the resulting output.

Example 5-7 igt-add-cmmconnectionfactory command

```
[ibmgrid@z1 /]$ igt-add-cmmconnectionfactory -cfName cf2 -jndiName cimom1
-serverName 192.168.0.31 -portNumber 5988 -userName guest -password guest
-nameSpace root/cimv2
option=-cfName
option=cf2
option=-jndiName
option=cimom1
```

```
option=-serverName
option=192.168.0.31
option=-portNumber
option=5988
option=-userName
option=guest
option=-password
option=guest
option=-nameSpace
option=root/cimv2
PROPS=-javaoption -Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer
option=-c
option=ra-connection-factory
option=-rarName
option=CimRA
option=-cfName
option=cf2
option=-jndiName
option=cimom1
option=-serverName
option=192.168.0.31
option=-portNumber
option=5988
option=-userName
option=guest
option=-passw
option=guest
option=-nameSpace
option=root/cimv2
option=-adapterType
option=cim
/opt/IBMGrid/AppServer/bin/wsadmin.sh -connytpe NONE -javaoption
-Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer -f
/opt/IBMGrid/scripts/AppServer/ra-connection-factory.jacl -rarName CimRA
-cfName cf2 -jndiName cimom1 -serverName 192.168.0.31 -portNumber 5988
-userName guest -passw guest -nameSpace root/cimv2 -adapterType cim
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7357I: By request, this scripting client is not connected to any server
process. Certain configuration and application operations will be
available in local mode.
createCF: Retrieving the Cim Resource Adapter Details
ConnectionFactory cf2 has been created
[ibmgrid@z1 /]$
```

Note: Record these settings, as they will be needed when updating the CRMResourceBinding.xml file. Refer to 6.4, “Common Management Model (CMM) service” on page 115.

5.5.2 Deleting a connection

To delete a connection factory, enter the following command:

```
igt-delete-cmmconnectionfactory -cfName <name of connection factory>
```

Example 5-8 shows how we deleted the connection factory that was created in our lab environment.

Example 5-8 igt-delete-cmmconnectionfactory command

```
[ibmgrid@z1 /]$ igt-delete-cmmconnectionfactory -cfName cf2
option=-cfName
option=cf2
PROPS=-javaoption -Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer
option=-c
option=remove-ra-connection-factory
option=-raName
option=CimRA
option=-cfName
option=cf2
/opt/IBMGrid/AppServer/bin/wsadmin.sh -connytpe NONE -javaoption
-Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer -f
/opt/IBMGrid/scripts/AppServer/remove-ra-connection-factory.jacl -raName CimRA
-cfName cf2
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7357I: By request, this scripting client is not connected to any server
process. Certain configuration and application operations will be
available in local mode.
ConnectionFactory cf2 has been deleted
[ibmgrid@z1 /]$
```

5.6 Backing up a grid

It is recommended that you back up each instance of the IBM Grid Toolbox periodically and that these backups be integrated with your organization's backup strategy.

5.6.1 Backing up files

To back up an instance of the IBM Grid Toolbox, back up the following directories, including subdirectories:

- /opt/IBMGrid/AppServer/
- /opt/IBMGrid/Database/system/
- /opt/IBMGrid/undeploy/

- /opt/IBMGrid/deploy/
- /opt/IBMGrid/gars/
- /home/ibmgrid/.globus/
- $HOME/.globus (for other users with certificates)
- /etc/grid-security/

The backup should also include any grid-related files that you have modified or created.

5.6.2 Restoring files

To restore your instance of the IBM Grid Toolbox, follow these steps:

1. Install the IBM Grid Toolbox. For more information, refer to Chapter 3, “Installation and setup” on page 27.
2. Install any necessary related software. For more information, refer to Chapter 4, “Installing related software” on page 59.
3. Apply the backup files for your instance.

6

Samples

This chapter presents some of the sample services that are available with the IBM Grid Toolbox. The lab environment that is used in these samples is the same as that used during the installation process. We discuss these specific samples:

- Service data counter service
- Notification counter service
- Secure counter service
- Common Management Model service
- Service group sample
- Policy application sample
- Reliable File Transfer
- Managed-job-globusrun sample

6.1 Service data counter service

The service data counter sample illustrates how data is maintained in a grid service. This sample also demonstrates how service data is queried by a client application using a Web application similar to the basic counter service in 5.2, “Deploying and undeploying grid services” on page 94.

6.1.1 Setting up the service data sample

To run the service data counter sample, deploy the IGTCounterSamples.gar file. As ibmgrid, issue this command to deploy the IBM Grid Toolbox counter samples.

```
igt-deploy-gar /opt/IBMGrid/gars/IGTCounterSamples.gar
```

Note: For additional information about deploying services, refer to 5.2, “Deploying and undeploying grid services” on page 94.

6.1.2 Running the service data sample

After the service has been deployed, you can interact with the service data counter sample by using a Web application at the following address:

```
http://<hostname>:<port>/ogsa/samples/counters/servicedata
```

The default port number is 12080. The Web application should look similar to Figure 6-1 on page 107.

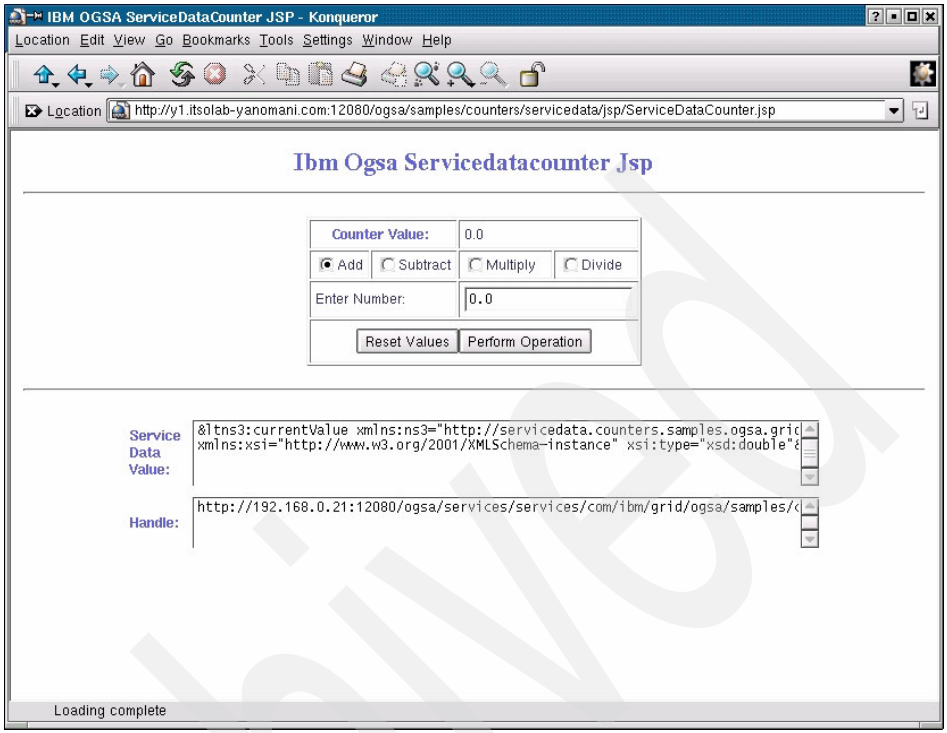


Figure 6-1 Service data counter sample

The service data counter sample implements two port types, which are required for all grid services. The first is the findServiceData, which enables a service to search for service data, and setServiceData, which enables a grid service to work with the service data.

To update the service data in our sample:

1. Select an operation to perform (**Add**, **Subtract**, **Multiply**, **Divide**)
2. Type in a number in the Enter number field.
3. Click **Perform operation**.

Figure 6-2 helps to illustrate the relationship between the Web application and underlying service data counter grid service.

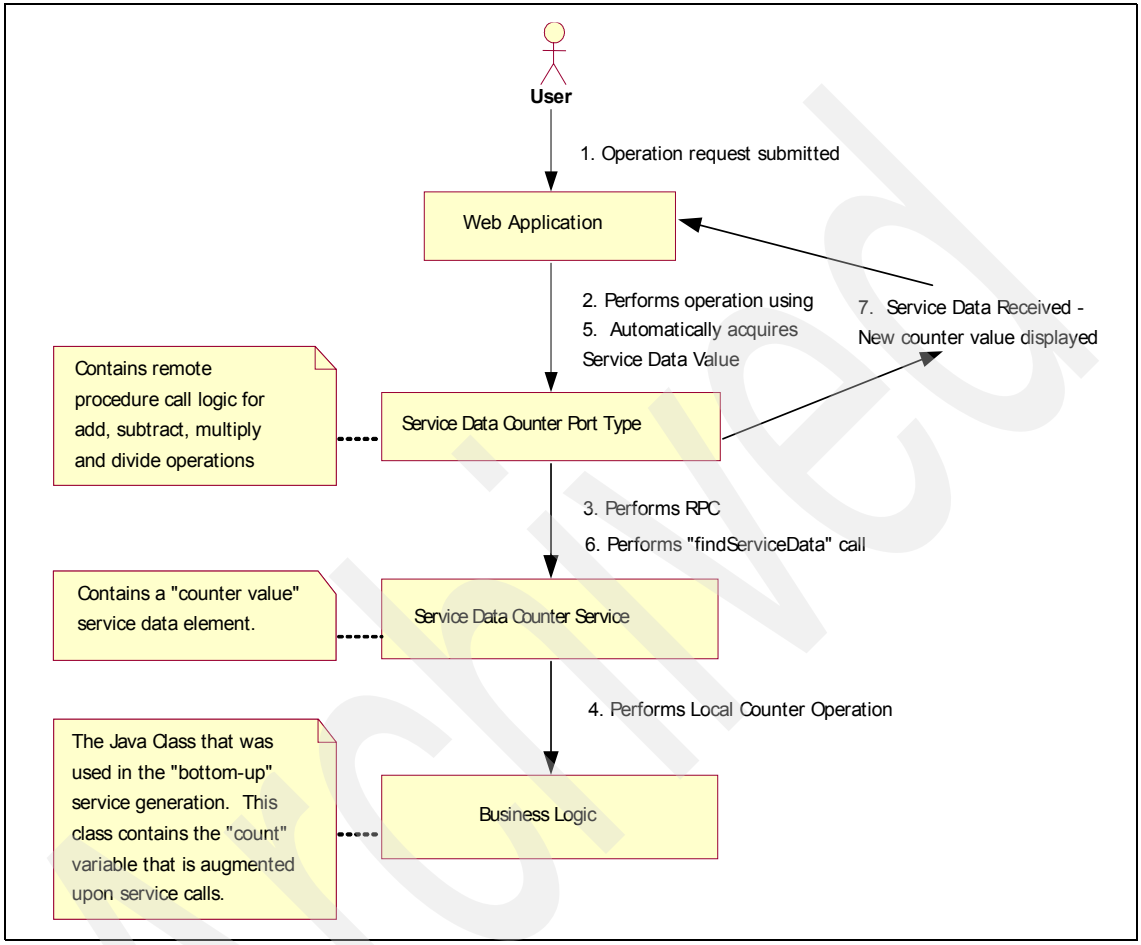


Figure 6-2 Service data counter relationships

When the operation is initiated, the Web application issues the desired command (add, subtract, multiply, or divide) on the ServiceDataCounter port type that interacts with the Grid Service. The operation is performed on the business logic to change the *count* variable, which is bound to the service data value. This means that every time that the class variable is changed, the service data is updated automatically. After the update is performed, the Web application displays the new currentValue in the Counter Value box of the Web application and outputs the currentValue service data element in the Service Data Value box. Example 6-1 on page 109 shows the original service data value at the beginning of the sample.

Example 6-1 Original service data value

```
<ns3:currentValue
xmlns:ns3="http://servicedata.counters.samples.ogsa.grid.ibm.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xsd:double"&gt0.0</ns3:currentValue>
```

Example 6-2 shows the updated service data value after performing an add 10 operation. Notice the new value toward the end of the service data value.

Example 6-2 Updated service data value

```
<ns3:currentValue
xmlns:ns3="http://servicedata.counters.samples.ogsa.grid.ibm.com"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="xsd:double"&gt0.0</ns3:currentValue>
```

For additional information about the background procedures used by this sample, refer to the manual that is included with the IBM Grid Toolbox.

6.2 Notification counter service

This notification counter service sample illustrates how making changes to service data can trigger a notification to be sent to other services. The sample uses a Web application similar to that in 6.1, “Service data counter service” on page 106.

6.2.1 Setting up the notification counter sample

To run the notification counter sample. deploy the IGTCounterSamples.gar file. As ibmgrid, issue the following command to deploy the IBM Grid Toolbox counter samples:

```
igt-deploy-gar /opt/IBMGrid/gars/IGTCounterSamples.gar
```

Note: For additional information about deploying services, refer to 5.2, “Deploying and undeploying grid services” on page 94.

Also, in order to use notification services with the IBM Grid Toolbox, it is necessary to start the Open JMS environment. Use the following command:

```
/opt/IBMGrid/OpenJMS/bin/startup.sh
```

Tip: For Open JMS to start properly, the IBM Grid Toolbox container *must* be running.

With the Open JMS environment running it is now necessary to restart the IBM Grid Toolbox container with the **igt-stop-container** and **igt-start-container** commands. More information about Open JMS can be found at:

http://openjms.sourceforge.net/

Tip: Open JMS will continue to run as long as it is monitoring for messages. To have it run in the background, add an ampersand (&) to the end of the /opt/IBMGrid/OpenJMS/bin/startup.sh command.

6.2.2 Running the notification counter sample

After the service has been deployed and Open JMS has been started, you can interact with the notification counter sample using a Web application at the following address:

```
http://<hostname>:<port>/ogsa/samples/counters/notification
```

The default port number is 12080. The Web application should look similar to Figure 6-3.

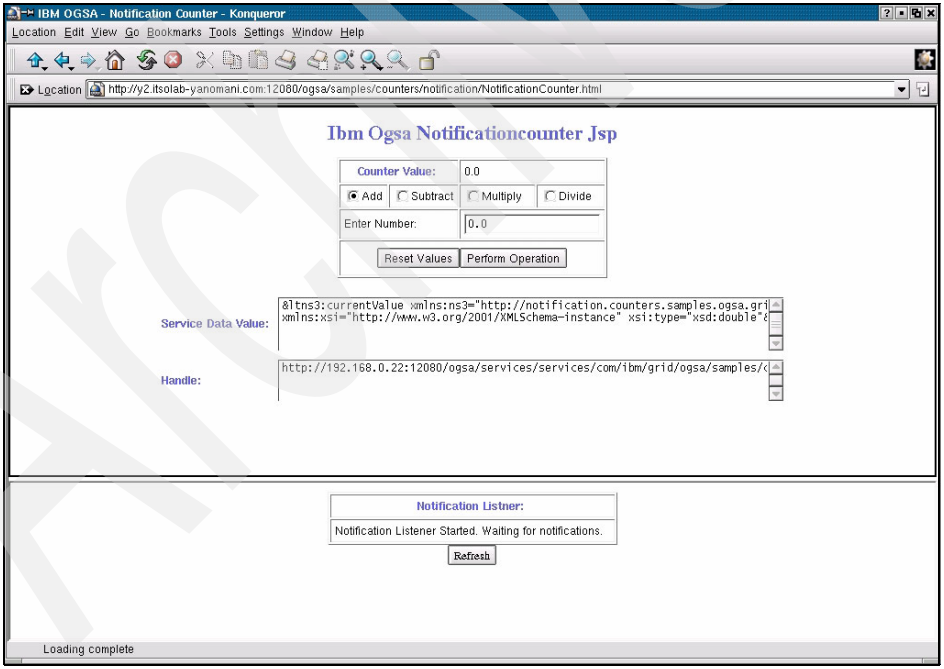


Figure 6-3 Initial notification counter sample page

The notification counter sample is similar to the service data counter sample. However, the notification counter sample calls an additional notification service to provide information about changes to the service data value in the lower frame of the Web application.

Figure 6-4 illustrates the relationship between the Web application and the notification service.

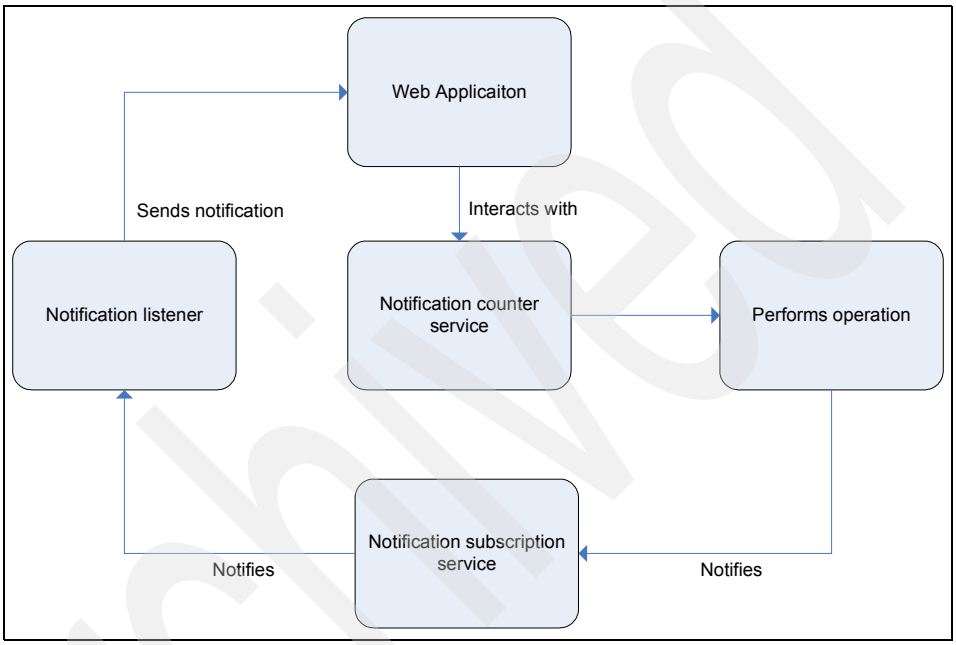


Figure 6-4 Notification counter service relationship

Tip: Because there are two grid services being initialized, you may experience time-out problems with this sample. Refer to the manual that is included with the IBM Grid Toolbox for more details.

To update the service data in our sample and receive a notification of the change:

1. Select an operation to perform (Add, Subtract, Multiply, Divide).
2. Type in a number in the Enter number field.
3. Click **Perform operation**.

After the operation is performed, the Service Data Value field will be updated to reflect the new value. Shortly after the service data value has been updated, you will see the notification in the lower frame of the Web application.

Figure 6-5 shows that a new notification is received when adding a value of 10 to an existing number.

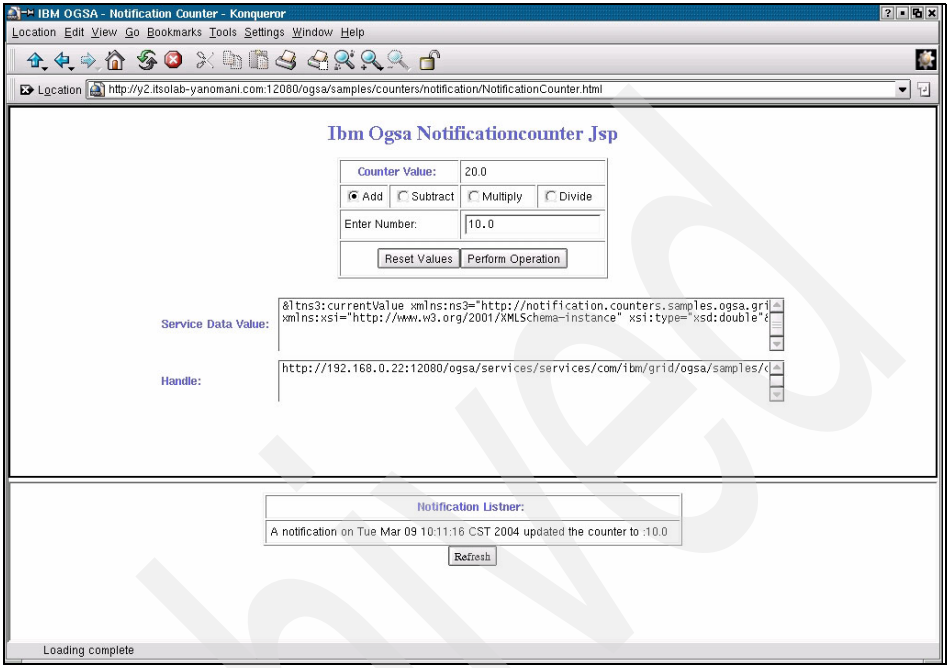


Figure 6-5 New notification

For additional information about the background procedures used by this sample, refer to the manual included with the IBM Grid Toolbox.

6.3 Secure counter service

The secure counter service sample demonstrates some of the basic concepts of grid security. For this sample, you must set up a grid proxy before accessing the Web application that comes with the sample. The application is similar to that in 6.1, “Service data counter service” on page 106.

6.3.1 Setting up the secure counter sample

To run the secure counter sample, deploy the IGTCounterSamples.gar file. As ibmgrid, issue the following command to deploy the IBM Grid Toolbox counter samples:

```
igt-deploy-gar /opt/IBMGrid/gars/IGTCounterSamples.gar
```

Note: For additional information about deploying services, refer to 5.2, “Deploying and undeploying grid services” on page 94.

6.3.2 Running the secure counter sample

To begin the sample:

1. Issue the following command as *ibmgrid* to create a grid proxy:

   ```
   grid-proxy-init
   ```

   Tip: If you already have a proxy set up, you can use the **`grid-proxy-info`** command to learn the details of the proxy.

2. Open the Web application at the following address. The default port number is 12080.

   ```
   http://<hostname>:<port>/ogsa/samples/counters/secure
   ```

The secure counter service application should look similar to Figure 6-6.

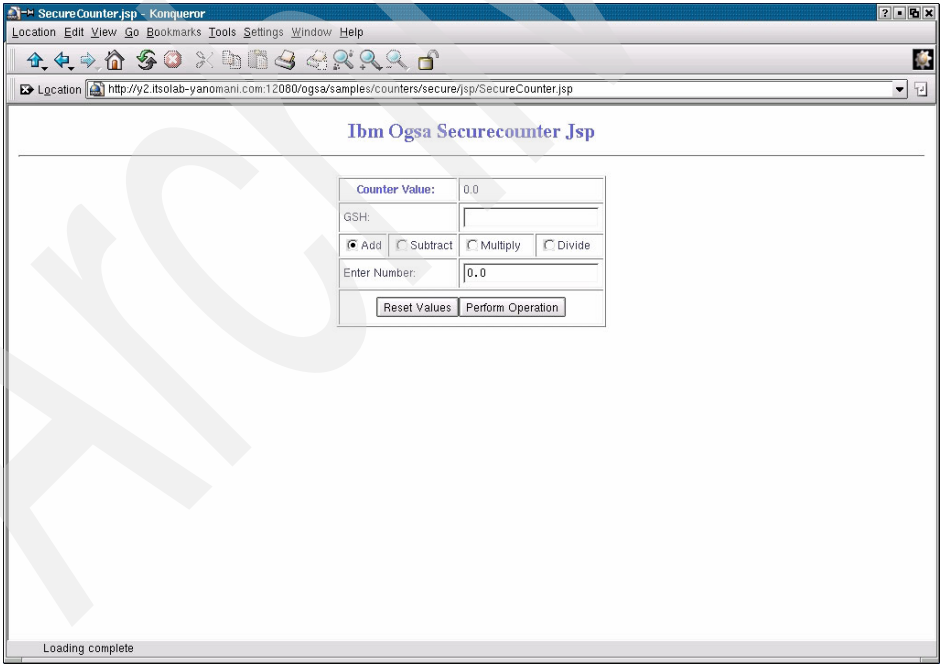


Figure 6-6 Secure counter service application

The secure counter service sample is similar to the original counter service in 5.2, “Deploying and undeploying grid services” on page 94. However, it includes some additional security parameters in the service deployment descriptor.

To update the value in our sample:

1. Select an operation to perform (Add, Subtract, Multiply, Divide).
2. Type in a number in the Enter number field.
3. Click **Perform operation**.

Figure 6-7 provides a basic overview of how the secure counter service works.

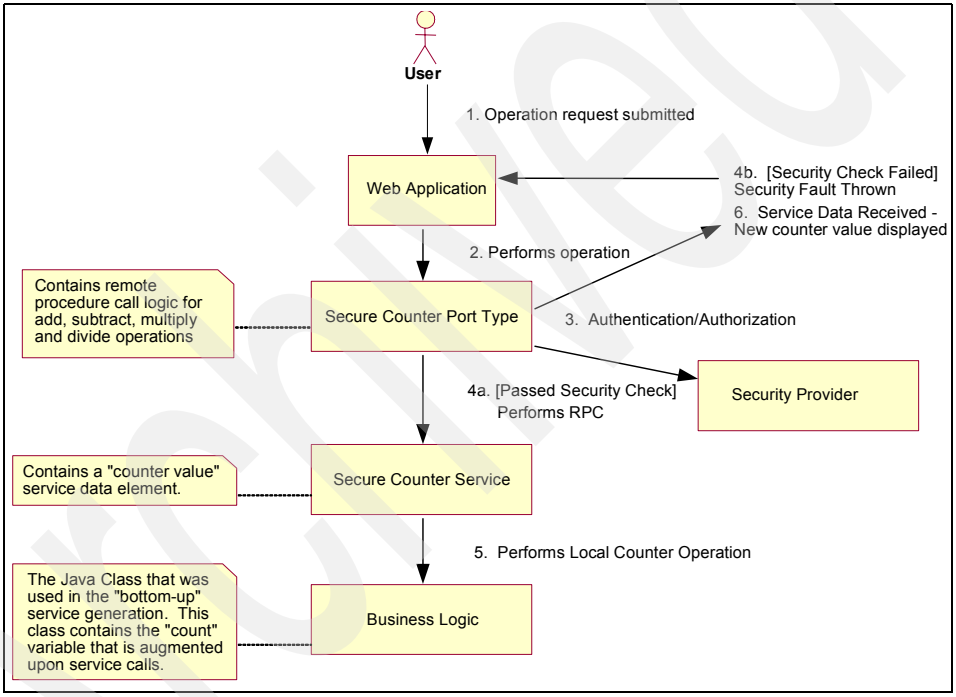


Figure 6-7 Secure counter service relationship

The service performs the security checks before the operation is performed. The service flow is as follows:

1. Web application interacts with grid service.
2. User Authentication/Authorization is performed.
3. Operation is performed on business logic.

For additional information about the background procedures used by this sample, refer to the manual included with the IBM Grid Toolbox.

6.4 Common Management Model (CMM) service

The CMM Process Tracker sample illustrates one possible use of the CMM core services in the IBM Grid Toolbox. This sample communicates with a CIM Object Manager, or CIMOM, to get information about processes that are executing on an operating system.

The sample includes the following services:

- The Manageable Resource Registry service (MR Registry)
- The Manageable Resource handleResolver service (MR Handle Resolver)

The process tracker application queries the MR Registry for a list of all processes that are executing on the selected operating system. The MR Registry uses the mapping framework to convert this query to a format that can be consumed by the CIMOM. This query is sent via the CIM JCA adapter to the CIMOM, which returns a list of processes that are running on the specified operating system. The MR Handle Resolver takes the GSH (which uses the CIM naming scheme) and uses this to get a reference to the CIM object associated with the selected process. The MR Handle Resolver then creates a new grid service instance for the selected process. The MR Handle Resolver then passes the GSR for the process service back to the Process Tracker Web application. Figure 6-8 on page 116 depicts an overview of the concepts.

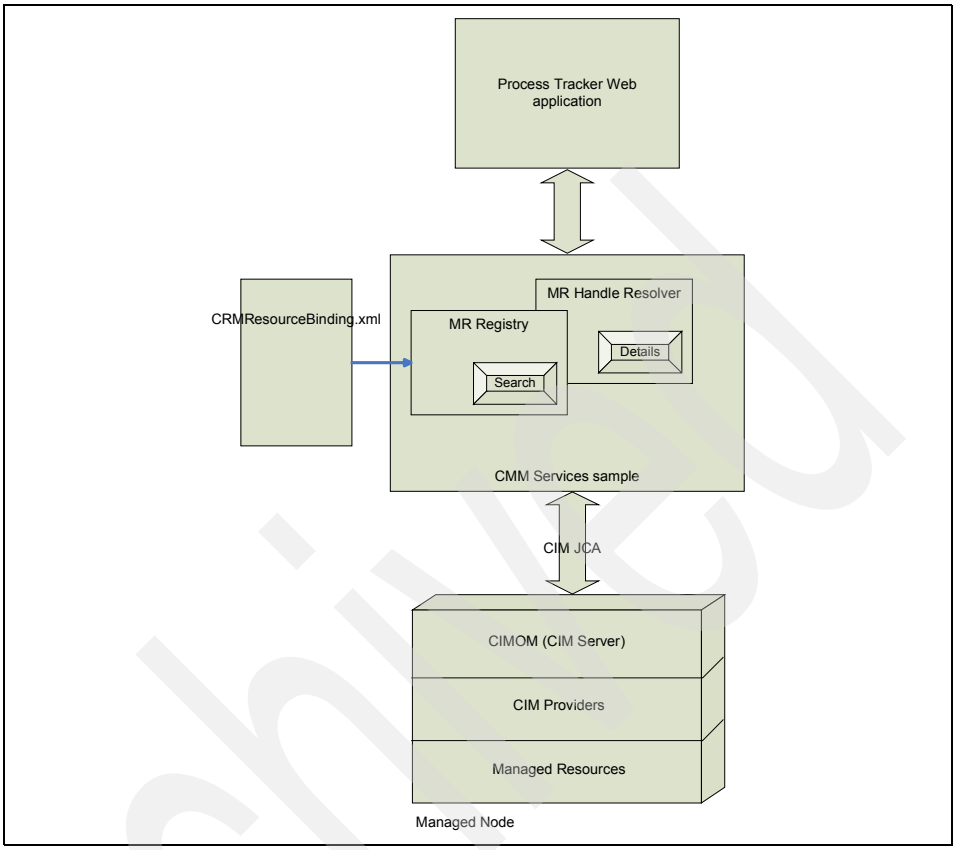


Figure 6-8 ProcessTracker Web application concepts

6.4.1 Setting up the CMM sample

Before running the CMM sample, follow these steps to set up the sample:

1. As the ibmgrid user, deploy the CMM sample. Run the following command from the /opt/IBMGrid/ directory:

```
igt-deploy-gar  /opt/IBMGrid/gars/crm-sample.gar
```

2. Pegasus and SBLIM packages must be installed. Refer to Chapter 4, “Installing related software” on page 59. As the root user, start Pegasus and add a `cimuser`. Example 6-3 shows the commands that we used in our lab.

Example 6-3 Start Pegasus and add cimuser commands

```
cimserver
cimuser -a -u ibmgrid -w passw0rd
```

Important: The cimserver must be initialized as root. If not, the CIM server may not work properly and you could damage your CIM server installation.

3. Create a connection factory so that CMM services can communicate with the CIMOM. For details of this command and how to manage connections, see 5.5, “Managing connections for CMM Services” on page 100. Example 6-4 shows the command we used in our lab.

Example 6-4 igt-add-cmmconnectionfactory command

```
[ibmgrid@z1 /]$ igt-add-cmmconnectionfactory -cfName conn1 -jndiName cimom1
-serverName 192.168.0.31 -portNumber 5988 -userName ibmgrid -password passwOrd
-nameSpace root/cimv2
```

4. Configure the sample application with the connection factory. To enable the sample to use these connection factories, edit the /opt/IBMGrid/AppServer/installedApps/DefaultNode/crm-sample_war.ear/crm-sample.war/WEB-INF/CRMResourceBinding.xml file.

   Example 6-5 shows the settings that we used in our lab.

Example 6-5 CRMResourceBinding.xml file

```
<resourceBindings>
      <connection name="cimom1">
        <connectionProperties name="AdapterType" value="CIM"/>
        <connectionProperties name="NameSpace"   value="root/cimv2"/>
        <connectionProperties name="ServerName"  value="192.168.0.31"/>
        <connectionProperties name="PortNumber"  value="5988"/>
      </connection>
  </resourceBindings>
```

Important: The `connection name` value must match the JNDI Name that was used when creating a new CMM Connection factory. For more information about creating a CMM Connection factory, refer to 5.5, “Managing connections for CMM Services” on page 100.

5. Disable security for the CMM sample.
   a. Use a text editor to open the following configuration file: /opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/schema/base/crm/security/cmm-security-config.xml
   b. Search in the configuration file for the line shown in Example 6-6 on page 118.

Example 6-6 CMM sample original security

```
<securityConfig xmlns="http://www.globus.org">
<auth-method>
<pkey/>
<gsi/>
</auth-method>
</securityConfig>
```

c. Replace that line, as shown in Example 6-7.

Example 6-7 CMM sample new security

```
<securityConfig xmlns="http://www.globus.org">
<auth-method>
<none/>
</auth-method>
</securityConfig>
```

d. Save your changes to the configuration file.

6. Stop and restart the OGSI container:

```
igt-stop-container
igt-start-container
```

6.4.2 Running the CMM sample

A Web application that is capable of interacting with the CMM services was deployed with the IBM Grid Toolbox installation.

Start the Process Tracker Web application

To access this user interface, open the following URL in your browser:

```
http://host name:port/crm/ProcessTrackerServlet
```

– *host name* is the name of the host system on which the sample was deployed.
– *port* is the port used by the OGSI container, such as 12080.

The results of accessing this URL are shown Figure 6-9.

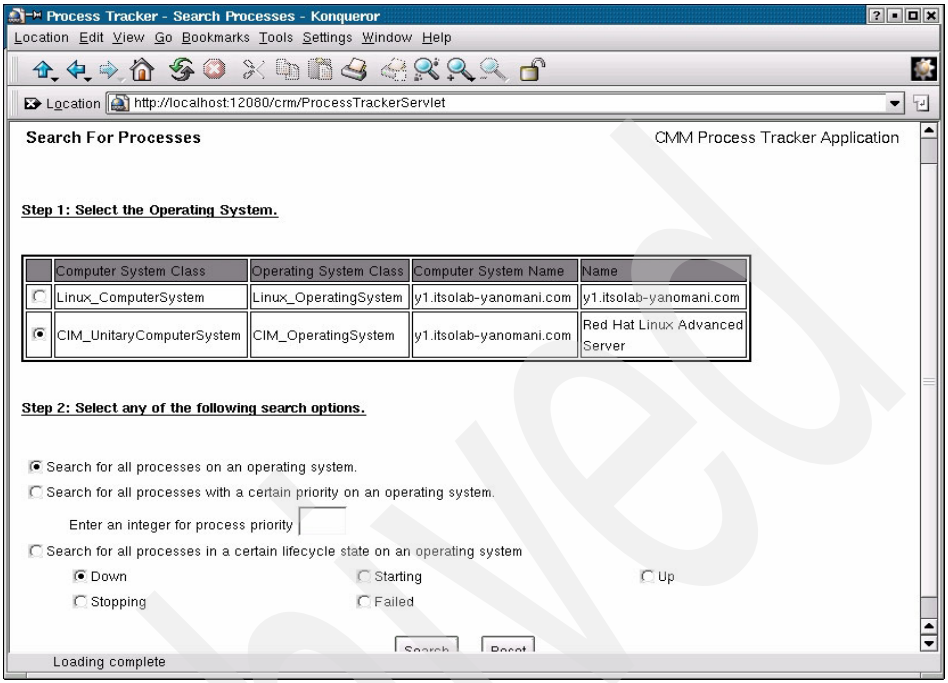


Figure 6-9 ProcessTracker

Use the CMM process tracker application

The Process Tracker Web application communicates with the CMM services to request dynamic information about managed resources that have been exposed as grid services. To do this, the Web application sends a request to the MR Registry to get information about a certain type of resource:

1. Select a system from the list of operating systems.
2. Select the **Search for all processes on an operating system** search option.
3. Click **Search**.

   An example is shown in Figure 6-10.

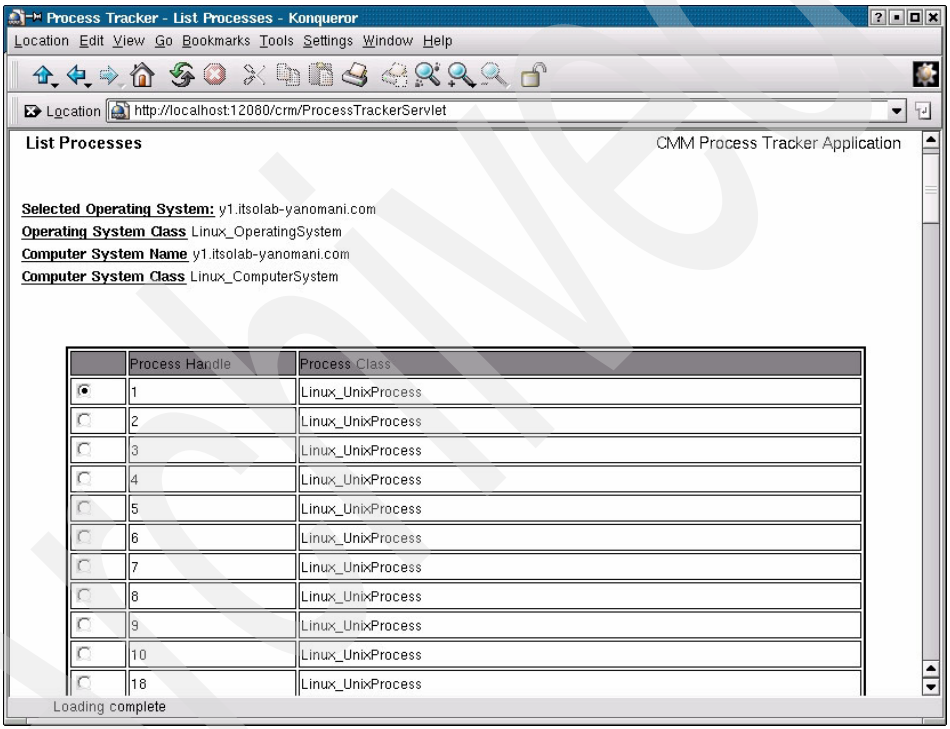


Figure 6-10 ProcessTracker List Processes

4. Select a process from the list.
5. Click **Get details**.

   The results are shown in Figure 6-11.

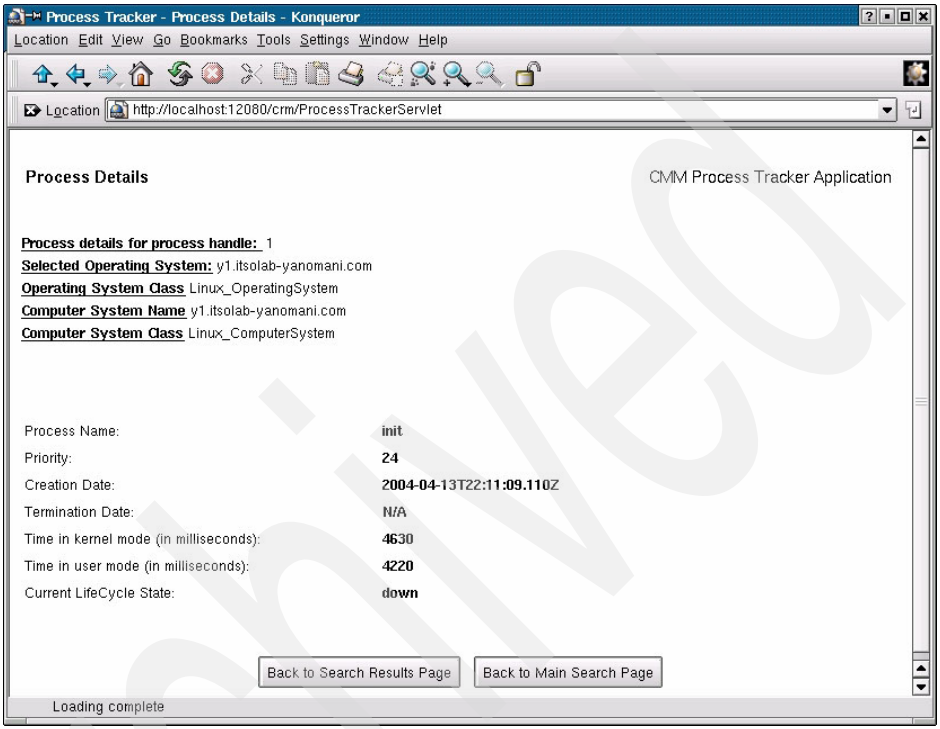


Figure 6-11 ProcessTracker Process Details

6.5 Service group sample

The service group sample illustrates how grid services can be collected into meaningful groups. These groups can then be used by applications to access all necessary services from a single location instead of searching the grid for all of the services. The service group sample provides a Web application interface through which counter services can be aggregated to or removed from the group.

6.5.1 Setting up the service group sample

Before using the Service group sample, it must be deployed on your grid instance. As ibmgrid, issue the following command:

```
igt-deploy-gar /opt/IBMGrid/gars/ServiceGroupCounterService.gar
```

Because the Service group sample was created to interact with unsecure services, you must disable security for this service. Follow these steps:

1. Open the file in Example 6-8 in your preferred editor.

Example 6-8 Service group sample security file

```
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/schema
/security/servicegroup/ServiceGroupRegistrationDefaultSecurity-Server.xml
```

2. Search the configuration file for the text shown in Example 6-9.

Example 6-9 Original security setting

```
<securityConfig xmlns="http://www.globus.org">
   <auth-method>
      <pkey/>
      <gsi/>
   </auth-method>
</securityConfig>
```

3. Replace the text in Step 2 with the text in Example 6-10.

Example 6-10 New security setting

```
<securityConfig xmlns="http://www.globus.org">
   <auth-method>
      <none/>
   </auth-method>
</securityConfig>
```

4. Save the changes to the configuration file.

Important: This service group sample only runs with security disabled.

After deployment and when security has been disabled, restart the container using the **igt-stop-container** and **igt-start-container** commands.

6.5.2 Running the service group sample

Open the Web application at the following address:

```
http://<hostname>:<port>/sgsample/ServiceGroupSampleCounter.jsp
```

The default port number is 12080.

Tip: If the Web application fails to load, it may be the result of a time-out error. Refer to the manual included with the IBM Grid Toolbox for more information.

Note: If security has not been configured, you may receive error messages when starting the Web application.

After it has loaded, the initial application page should look similar to Figure 6-12.

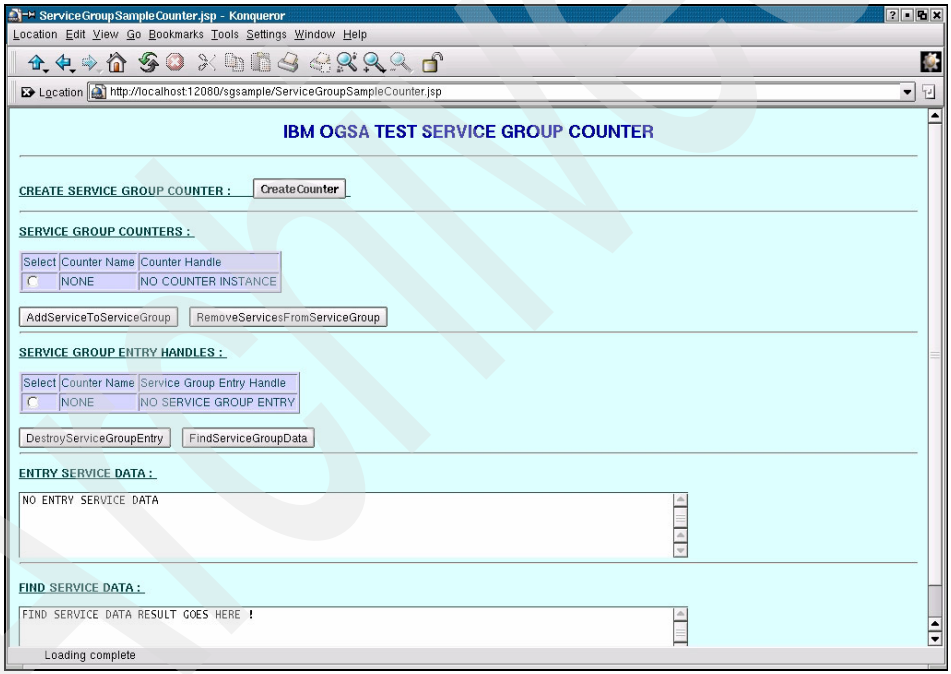


Figure 6-12 Initial service group application page

Adding a service to the group

To create a few sample counter services and add them to the counter service group, use the following steps:

1. Click **Create Counter**. This creates a sample counter service and adds it to the Service Group Counters table, also providing the grid service handle for the sample counter service.
2. Create two more counter services by repeating Step 1.
3. Select **CNTR 1** and click **AddServiceToServiceGroup**. This adds the selected service to the counter service group, passing the service's name and unique handle to the service group service. This information is then stored in a service data element called Entry. The service group creates one Entry service data element for each service added to the group.
4. Repeat Step 3 for CNTR 2 and CNTR 3.

Figure 6-13 shows the output from the service group application after all three counter services have been added.

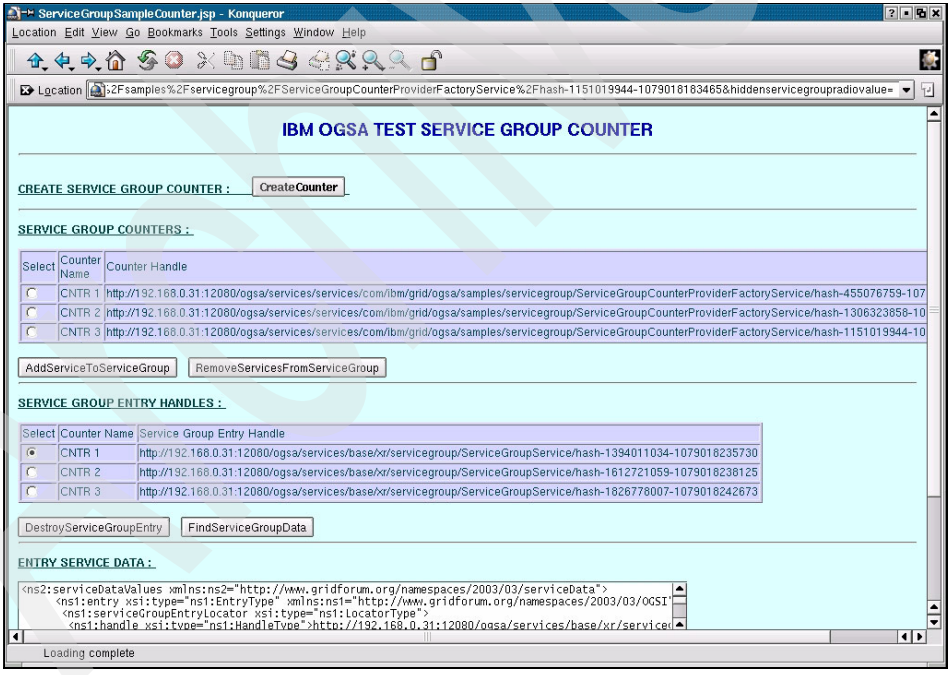


Figure 6-13 Service group application output

Find service data on a member service

Now that the counter services have been added to the service group, you can find and examine the service data associated with those services. Follow these steps:

1. Select **CNTR 1** from the SERVICE GROUP ENTRY HANDLES table.
2. Click **FindServiceGroupData**. The Web application runs a query for the Entry service data element that is associated with CNTR 1 and displays it in the FIND SERVICE DATA field. Example 6-11 shows the value of the service data element that is associated with CNTR 1.

Example 6-11 Sample service group service data element

```
MEMBER LOCATOR SERVICE DATA = <ns1:memberServiceLocator
xmlns:ns1="http://www.gridforum.org/namespaces/2003/03/OGSI"
xmlns:ns3="http://xml.apache.org/xml-soap"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="ns3:Element">
<ns1:LocatorType xmlns:ns1="http://www.gridforum.org/namespaces/2003/03/OGSI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns1:LocatorType">
<ns1:handle
xsi:type="ns1:HandleType">http://192.168.0.21:12080/ogsa/services/services/com/
ibm/grid/ogsa/samples/servicegroup/ServiceGroupCounterProviderFactoryService/ha
sh-455076759-1079018177295</ns1:handle>
</ns1:LocatorType>
   </ns1:memberServiceLocator> CONTENT SERVICE DATA = <ns1:content
xmlns:ns1="http://www.gridforum.org/namespaces/2003/03/OGSI"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="ns1:EntryContentType"><ns1:content
xmlns:ns1="http://www.gridforum.org/namespaces/2003/03/OGSI">
<definitions name="TestContents" xmlns="http://schemas.xmlsoap.org/wsdl/"/>
</ns1:content>
</ns1:content>
```

Destroying a service group entry

The service group service also enables you to destroy an entry service data element that is related to an instance of a service in the group. Follow these steps:

1. Select **CNTR1** in the SERVICE GROUP ENTRY HANDLES table.
2. Click **DestroyServiceGroupEntry**. The Web application looks for a matching entry service data element in the service group and destroys it. The application then updates the ENTRY SERVICE DATA field to reflect the change.

Removing a member service

Finally, the service group service enables you to remove all instances of a service from the group at once.

First, add a second instance of CNTR 2 to the service group. Then you can remove all instances of CNTR 2 using the following steps:

1. Select **CNTR 2** from the SERVICE GROUP COUNTERS table.
2. Click **RemoveServiceFromServiceGroup**. The application finds and removes all instances of CNTR 2 from the group and refreshes the application.

6.6 Policy application sample

The policy application sample illustrates how the core policy framework works in the IBM Grid Toolbox. The sample shows how to work with policy settings for a sample discipline and a sample role and how to apply these policies to sample resources. This sample uses a Web application to interact and query the policies.

6.6.1 Setting up the policy application sample

Before using the policy application sample, it must be deployed on your grid instance. As ibmgrid, issue the following command:

```
igt-deploy-gar /opt/IBMGrid/gars/policy-sample.gar
```

Because the policy application sample was created to interact with unsecure services, you must disable security for this service. Follow these steps:

1. Open each of the files listed in Example 6-12 in your preferred editor.

Example 6-12 Policy application security files

```
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/schema/ba
se/policy/PSASecurityServerConfig.xml

/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/schema/ba
se/policy/PSMSecurityServerConfig.xml
```

2. Search each file for the lines shown in Example 6-13.

Example 6-13 Original security settings

```
<securityConfig xmlns="http://www.globus.org">
   <auth-method>
      <pkey/>
```

```
      <gsi/>
   </auth-method>
</securityConfig>
```

3. Replace the lines in Step 2 with those in Example 6-14 for each security file.

Example 6-14 New security settings

```
<securityConfig xmlns="http://www.globus.org">
   <auth-method>
      <none/>
   </auth-method>
</securityConfig>
```

4. Next, to use the policy application sample with the IBM Grid Toolbox it is necessary to start the Open JMS environment. Use the following command:

   ```
   /opt/IBMGrid/OpenJMS/bin/startup.sh
   ```

   Tip: For Open JMS to start properly, the IBM Grid Toolbox container *must* be running.

5. With the Open JMS environment running and security for the policy application disabled, it is necessary to restart the IBM Grid Toolbox container with the **`igt-stop-container`** and **`igt-start-container`** commands. More information about Open JMS can be found at:

   `http://openjms.sourceforge.net/`

   Tip: Open JMS will continue to run as long as it is monitoring for messages. To have it run in the background, add an ampersand (&) character to the end of the command in step 4.

6. After the service has been deployed, security has been configured, and Open JMS has been started, you can interact with the policy application sample using a Web application at the following address:

   ```
   http://<hostname>:<port>/policy/policy/AdminLogin
   ```

   The Web application should look similar to Figure 6-14 on page 128.

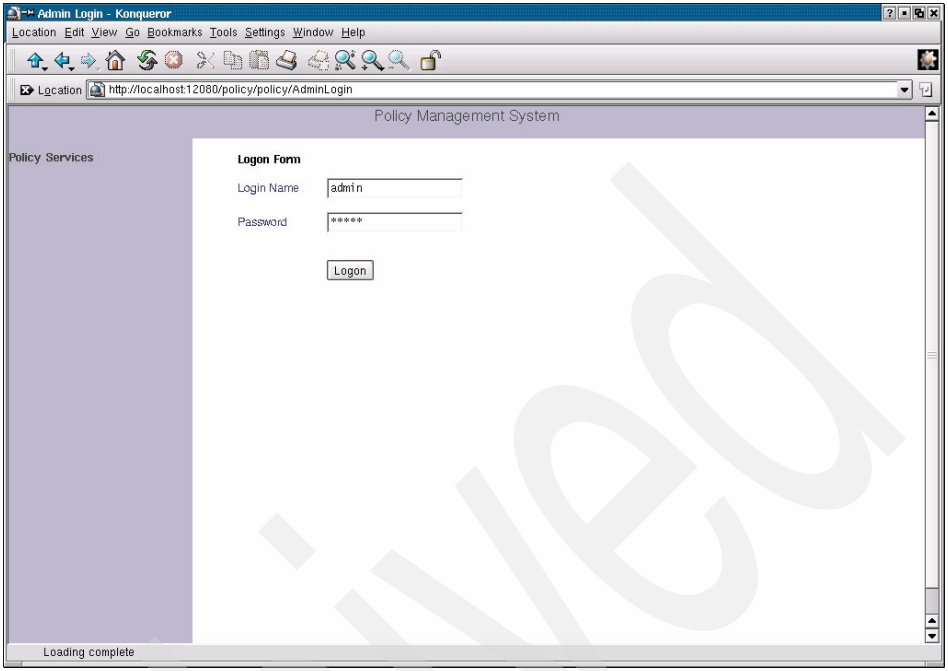


Figure 6-14 Initial policy application page

7. The policy application fills in the user name and password fields for you. Click **Logon** to enter the application.

6.6.2 Creating the policy services

Before you can begin to apply and manage policies, you must set up the infrastructure for managing policies by doing the following:

1. Create a new topology.
2. Create a Policy Service Manager.
3. Create a Policy Service Agent.
4. Create a Policy Enforcement Point.

Create a new topology

A policy topology defines the administrative domain under which you can configure and enforce policies. To create a new topology:

1. Type in a name for the new topology in the Topology name field.
2. Click **Create**.

Tip: The topology name must not contain any spaces or special characters.

3. After a topology has been created, the Manage Topology page opens. Here you can create Policy Service Managers, Policy Service Agents, and Policy Enforcement Points. To properly use policies, you should create at least one of each of these services. The Manage Topology page should appear similar to Figure 6-15.

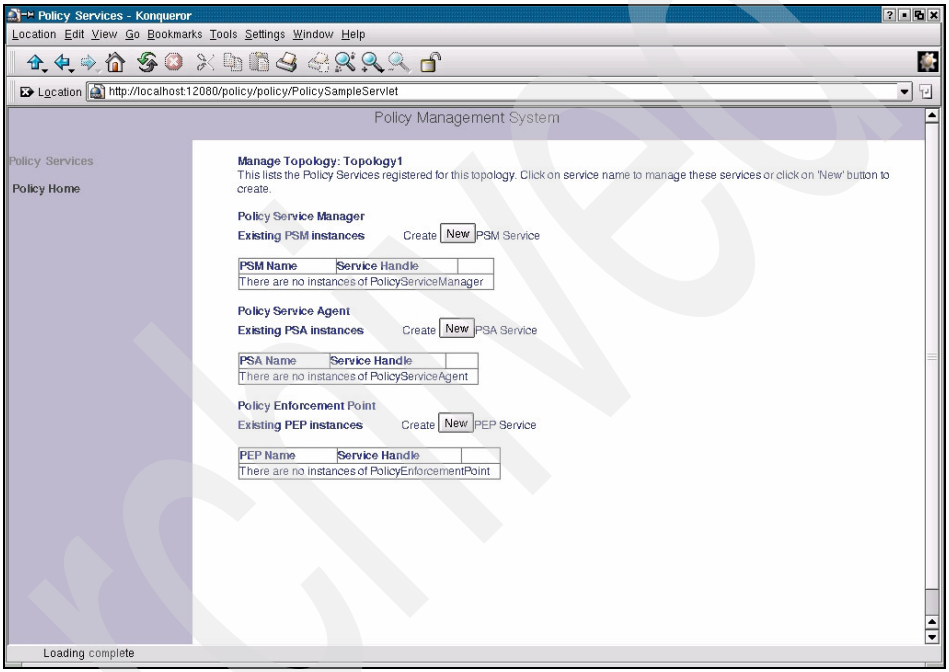


Figure 6-15 Manage Topology page

Create a Policy Service Manager

Policy Service Manager (PSM) is a grid service that enables management applications to interact directly with the core policy service in the IBM Grid Toolbox. In addition, it provides administrators with a variety of policy management functions.

To create a new Policy Service Manager:

1. Click **Create New PSM Service** under the Policy Service Manager heading. This opens the Create PSM Instance page similar to Figure 6-16 on page 130.

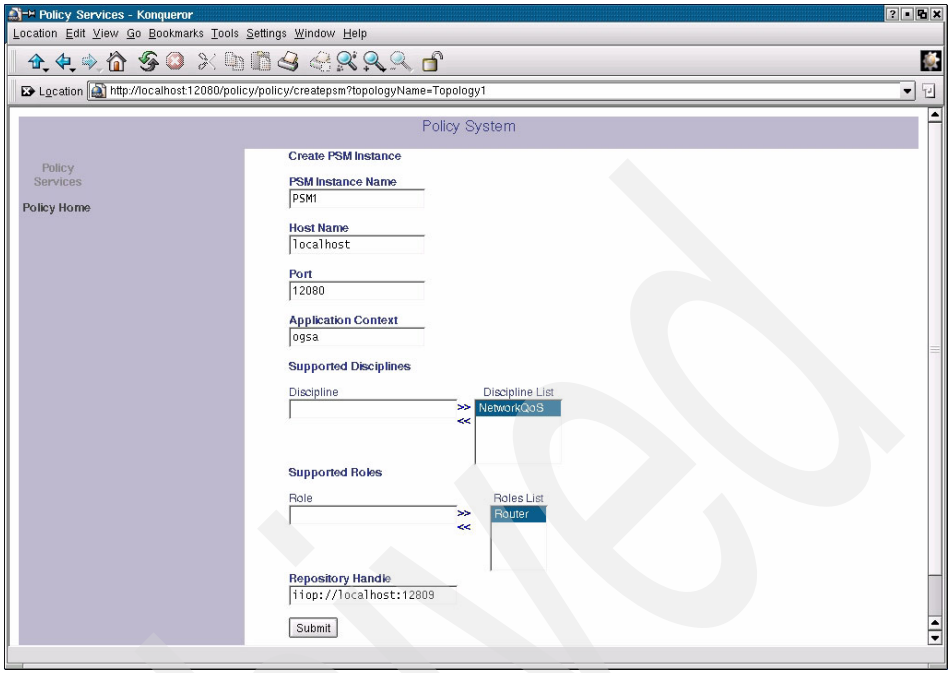


Figure 6-16 Create PSM Instance page

2. Fill in the required PSM configuration information. Table 6-1 lists the fields, their descriptions, and the values used in the sample.

Table 6-1 PSM creation fields

Field name	Description	Value
PSM Instance Name	Unique identifier for PSM service	PSM1
Host Name	Host name of system hosting the PSM service	localhost
Port	Port PSM will listen on	12080
Application Context	Application context for use by PSM	ogsa
Supported Disciplines	List of disciplines supported by the PSM	NetworkQoS
Supported Roles	List of roles supported by the PSM	Roles
Repository Handle	Handle for the policy repository service	Provided by application

3. Click **Submit** to create a new Policy Service Manager.

After this creation, the policy application returns to the Manage Topology page and shows the newly created PSM.

Create a Policy Service Agent

The Policy Service Agent (PSA) monitors the policy system for new or changed policies and makes these known to the Policy Enforcement Point.

To create a new PSA from the Manage Topology page:

1. Click **Create New PSA Service** to open the Create PSA Instance page, which should look similar to Figure 6-17.

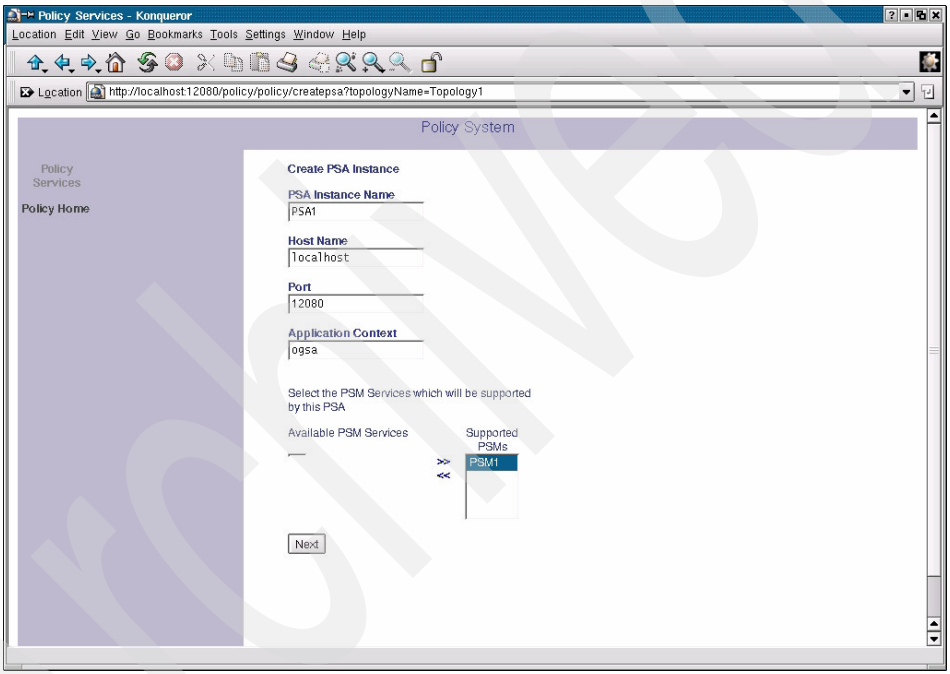


Figure 6-17 Create PSA Instance page

2. Fill in the required configuration information. See Table 6-2 for a listing of the fields, their descriptions and the values used in the sample.

Table 6-2 PSA creation fields

Field name	Description	Value
PSA Instance Name	Unique identifier for PSA service	PSA1
Supported PSMs	List of available PSM services	PSM1

3. Click **Next** to open the Assigning Disciplines page. Here you can associate PSM disciplines with the new PSA.
4. Select **NetworkQoS** then click **Next**. This opens the Assigning Roles page. Here you can associate PSM roles with the new PSA.
5. Select **Router** then click **Create**. The policy application creates the new PSA and returns you to the Manage Topology page. You now see the new PSA under the Policy Service Agent heading.

Create a Policy Enforcement Point

A Policy Enforcement Point (PEP) is a grid service that represents a device or network resource in a grid for purposes of enforcing the policies that are applied to that resource.

To create a new PEP from the Manage Topology page:

1. Click **Create New PEP Service**. This opens the Create PEP Instance page, which should look similar to Figure 6-18.

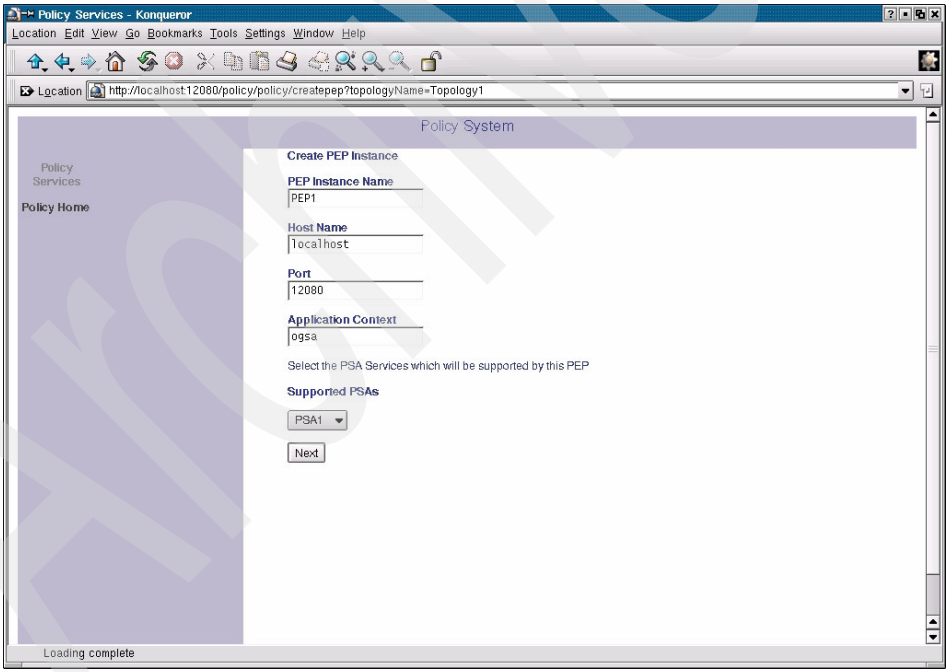


Figure 6-18 Create PEP Instance page

2. Fill in the required PSA configuration information. Table 6-3 on page 133 shows the fields, their descriptions, and the values used in the sample.

Table 6-3 PEP creation fields

Field name	Description	Value
PEP Instance Name	Unique identifier for PEP service	PEP1
Supported PSAs	List of available PSA services	PSA1

3. Click **Next**. This opens the Assigning Disciplines page. Here you can associate disciplines supported by the selected PSA with the new PEP.
4. Select **NetworkQoS** then click **Next**. This opens the Assigning Roles page for associating roles supported by the selected PSA with the new PEP.
5. Select **Router** then click **Create**. The policy application creates the new PEP and returns to the Manage Topology page. The new PEP now appears under the Policy Enforcement Point heading.

After you have created a PSM, PSA, and PEP, your Manage Topology page should look similar to Figure 6-19.

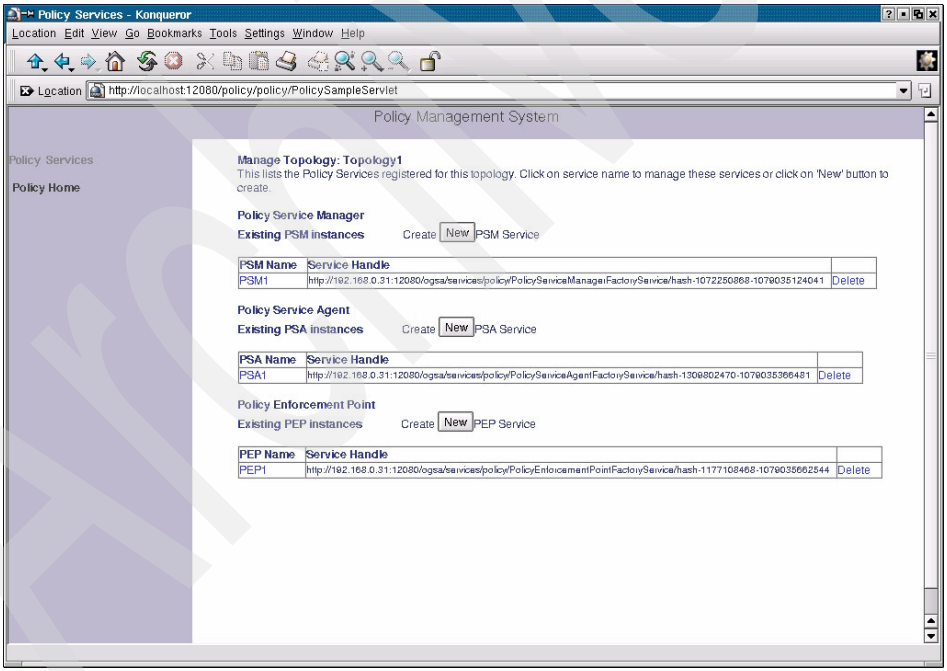


Figure 6-19 Manage Topology page

6.6.3 Managing policies

The policy application sample provides grid administrators with a variety of tools to create, update, and deploy policies.

Using the Policy Service Manager

The Policy Service Manager provides a variety of tools that aid grid administrators in defining and managing policies on a grid, and distributes these new or updated policies to the entire policy system.

To begin the PSM tool, the name of the Policy Service Manager you want to manage. This opens the page shown in Figure 6-20.

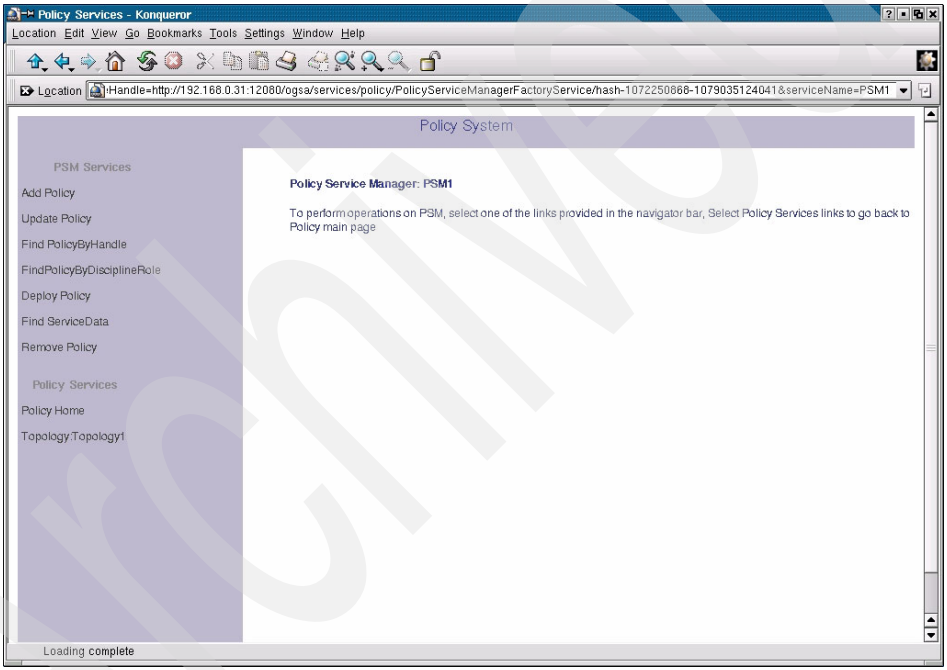


Figure 6-20 Initial PSM managing page

Add a policy

To add a policy in the policy application sample:

1. Click **Add Policy** on the left navigation frame. This opens the Add Policy page shown in Figure 6-21 on page 135.

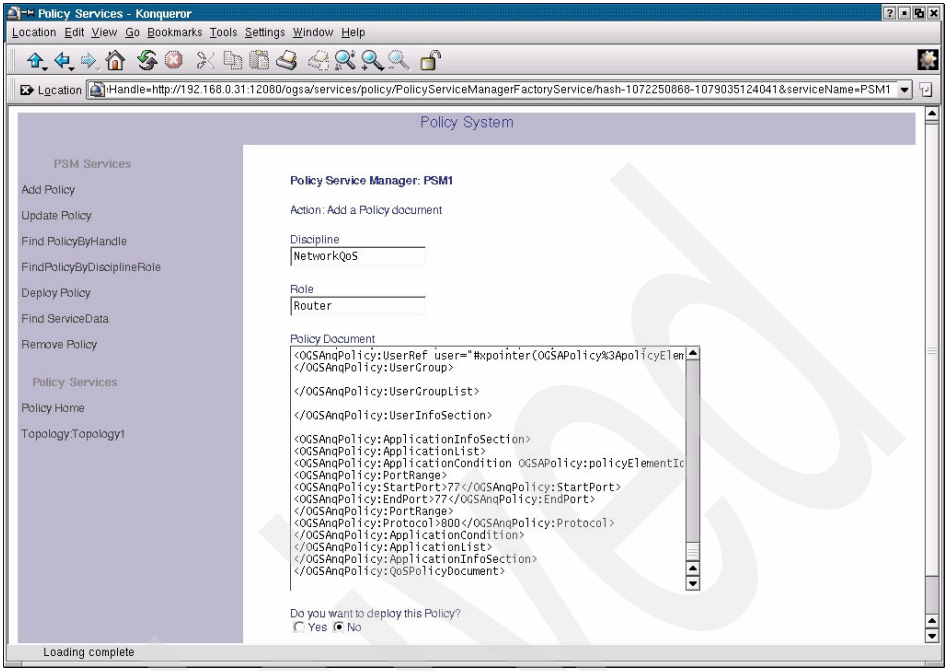


Figure 6-21 Add Policy page

2. Fill in the required policy information. Table 6-4 lists the fields, their descriptions, and the values used in the sample.

Table 6-4 Policy information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router
Policy Document	XML document describing the policy	Content of policy file

The policy application provides a sample discipline-level policy document for this example. Copy the entire contents of this file to the Policy Document field:

```
/opt/IBMGrid/AppServer/installedApps/DefaultNode/PolicySample.ear/policy-sample.war/data/networkQoS_sample.xml
```

3. Select **No** to avoid deploying the policy.
4. Click **Submit** to add the new policy. You should receive a confirmation for the new policy.

Update a policy

To update a policy in the policy application sample follow these steps:

1. Click **Update Policy** on the left navigation frame. This opens the Update Policy page shown in Figure 6-22.

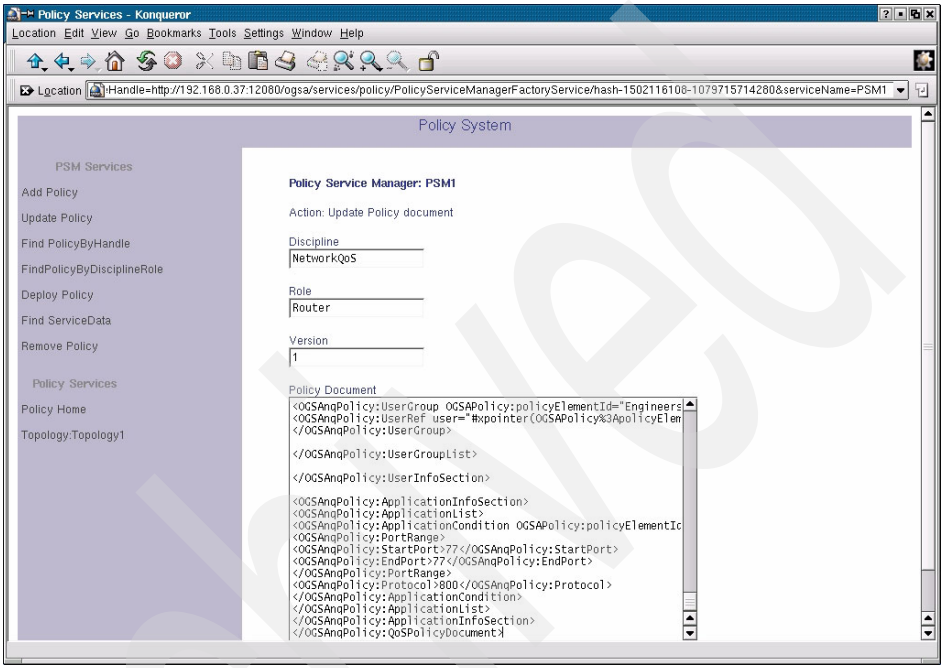


Figure 6-22 Update Policy page

2. Fill in the required policy information. Table 6-5 lists the fields, their descriptions, and the values used in the sample.

Table 6-5 Update policy information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Role
Version	Version identifier for a policy	1
Policy Document	XML document describing the policy	Content of policy file

The policy application provides a sample discipline-level policy document, which can be found in the following directory:

```
/opt/IBMGrid/AppServer/installedApps/DefaultNode/PolicySample.ear/policy
-sample.war/data/networkQoS_sample.xml
```

3. Click **Submit** to update the policy. You should receive a confirmation for the updated policy.

Find a policy by handle

Each policy in the Policy Service Manager is given a unique handle that includes information about the discipline, role, version, and level of a given policy. To query the PSM for these handles:

1. Click **Find PolicyByHandle** on the left navigation frame. This opens the page shown in Figure 6-23.

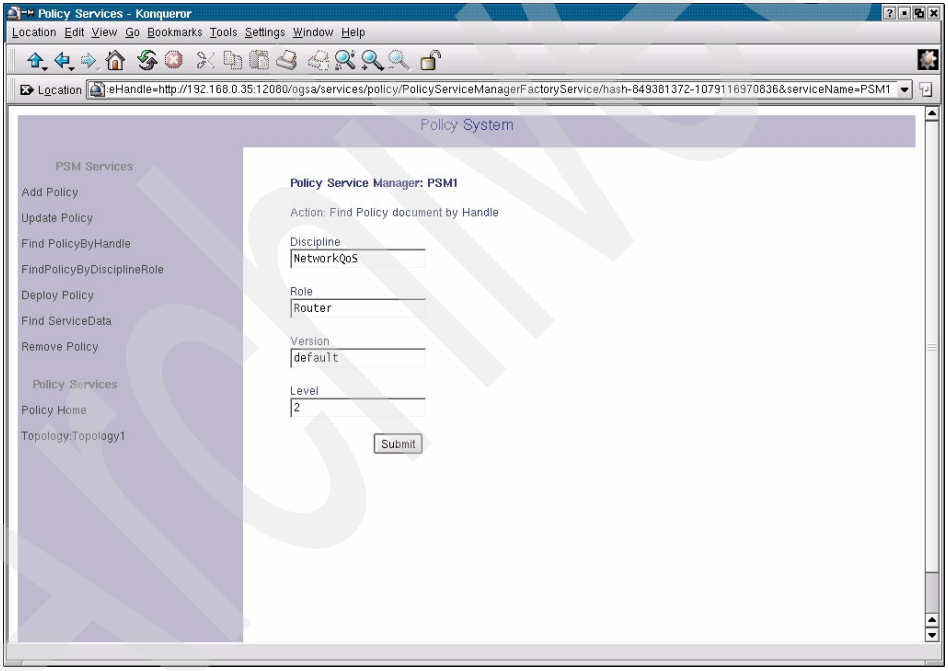


Figure 6-23 Find Policy By Handle page

2. Fill in the required policy information. Table 6-6 on page 138 lists the fields, their descriptions, and the values used in the sample.

Table 6-6 Find Policy By Name information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router
Version	Version identifier for a policy	1
Level	Level of the policy to retrieve: - Level 1 is for business level policies. - Level 2 for discipline level policies. - Level 3 for device level policies.	2

3. Click **Submit** to run the query. The policy application retrieves the information and displays it on the page.

Find a policy by discipline and role

The Policy Service Manager also enables for the querying of policies by specifying a discipline and role. To do this, use these steps:

1. Click **FindPolicyByDisciplineRole** on the left navigation frame. This opens the Find policy by discipline and role page in Figure 6-24.

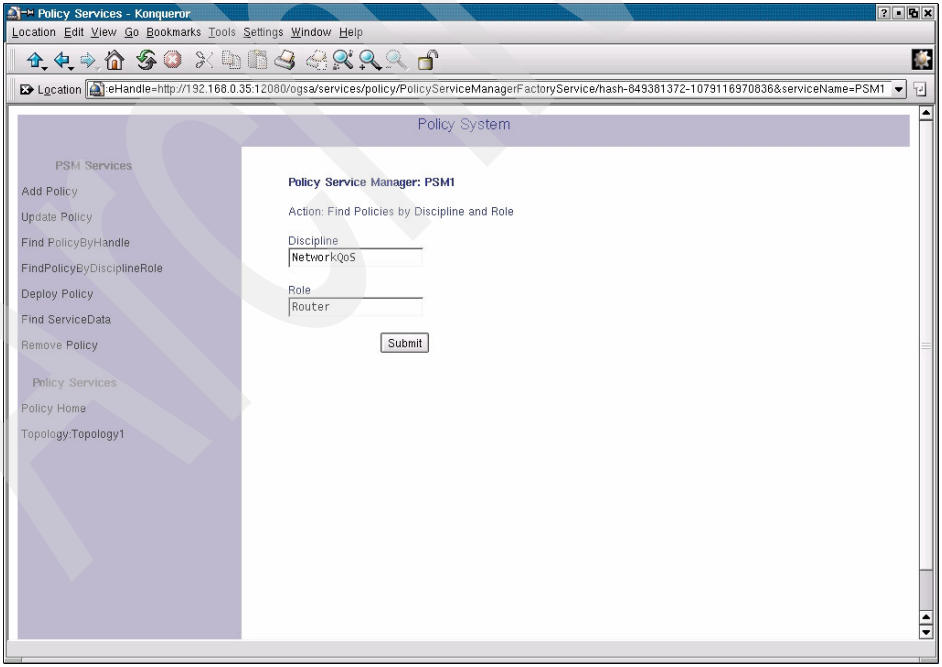


Figure 6-24 Find policy by discipline and role page

2. Fill in the required policy information. Table 6-7 lists the fields, their descriptions, and the values used in the sample.

Table 6-7 Find policy by discipline and role information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router

3. Click **Submit** to run the query. The policy application retrieves the information and displays it on the page.

Deploy a policy

The Policy Service Manager provides an operation to help deploy a new or updated policy. The operation sends a notification to the associated PSAs with the desired discipline and role in the policy. To deploy a new or updated policy:

1. Click **Deploy Policy** on the left navigation frame. This opens the Deploy policy page in Figure 6-25.

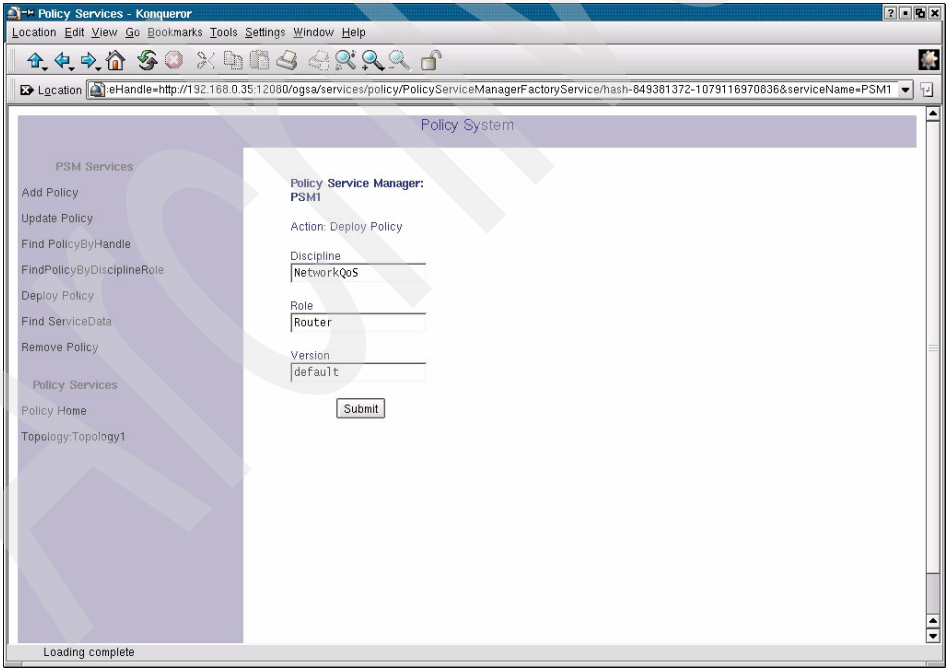


Figure 6-25 Deploy policy page

2. Fill in the required policy information. Table 6-8 lists the fields, their descriptions, and the values used in the sample.

Table 6-8 Deploy policy information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router
Version	Version identifier for a policy	1

3. Click **Submit** to deploy the new or updated policy.

Query the PSM service data

The Policy Service Manager stores several service data elements that the policy application sample and grid clients can query. To query the service data elements:

1. Click **Find ServiceData** on the left navigation frame. This opens the Find Service Data page in Figure 6-26.

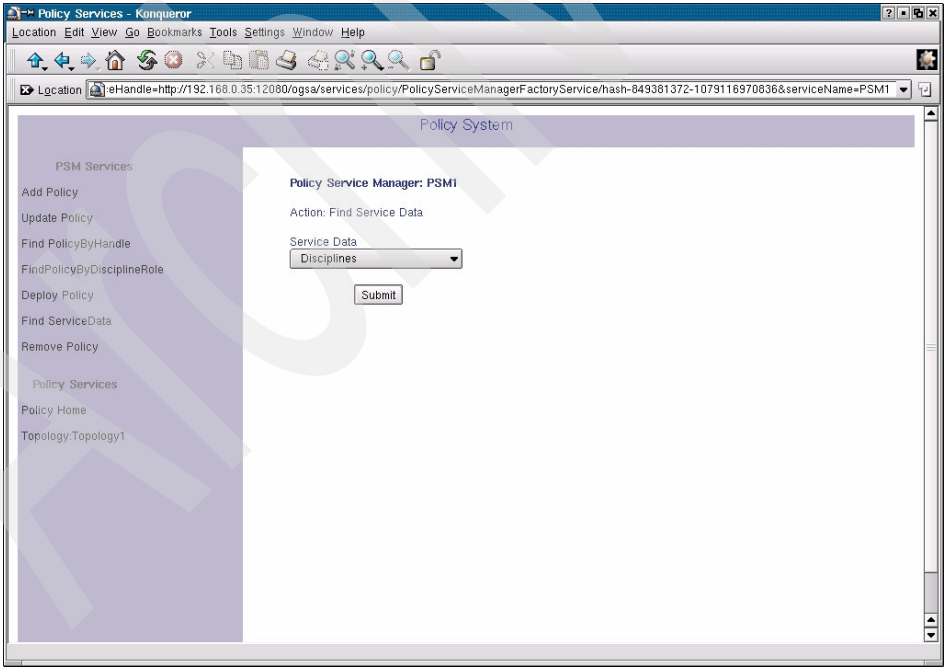


Figure 6-26 Query PSM service data page

2. In the Find Service Data page, enter the name of the service data element that you would like to see.
3. Click **Submit** to run the query. The Web application finds the desired service data element and displays it in the browser.

Remove a policy

The Policy Service Manager provides operations to remove any given policy. To remove a policy:

1. Click **Remove Policy** on the left navigation frame. This opens the Remove Policy Document page in Figure 6-27.

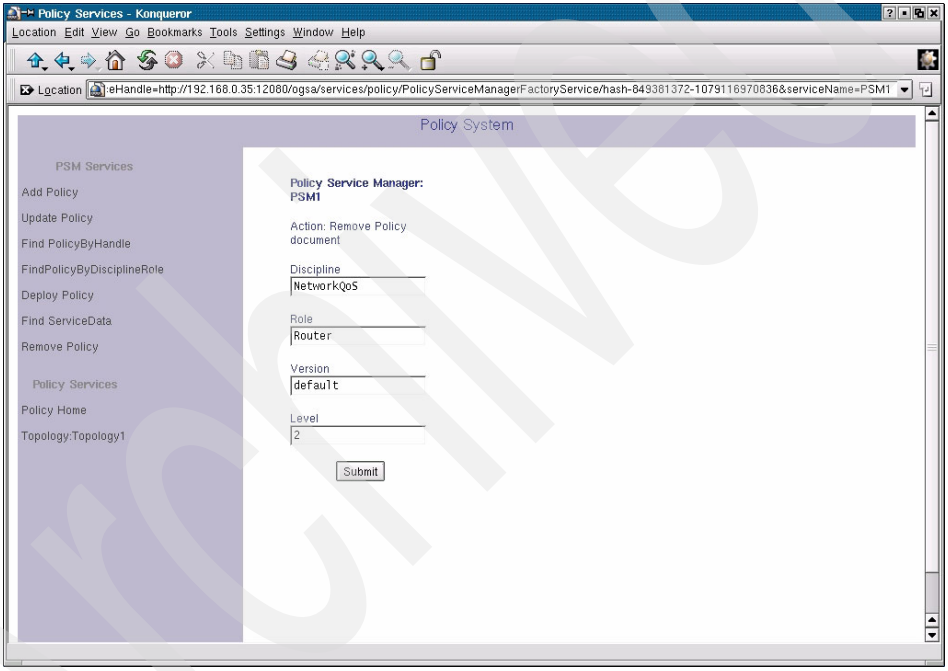


Figure 6-27 Remove policy page

2. Fill in the required policy information. Table 6-9 lists the fields, their descriptions, and the values used in the sample.

Table 6-9 Remove policy information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router

Field name	Description	Value
Version	Version identifier for a policy	1
Level	Level of the policy to retrieve. Level 1 is for business level policies. Level 2 for discipline level policies. Level 3 for device level policies.	2

3. Click **Submit** to remove the desired policy.

Using the Policy Service Agent

The Policy Service Agent stores multiple service data elements that can be queried by grid applications. To query service data elements from the PSA:

1. Return to the Manage topology page by clicking the **Topology Details** link on the left navigational frame.
2. Click the PSA you would like to query from the Policy Service Agent section.
3. Click the **Find ServiceData** link on the left navigational frame. The Find Service data page should look similar to Figure 6-28.

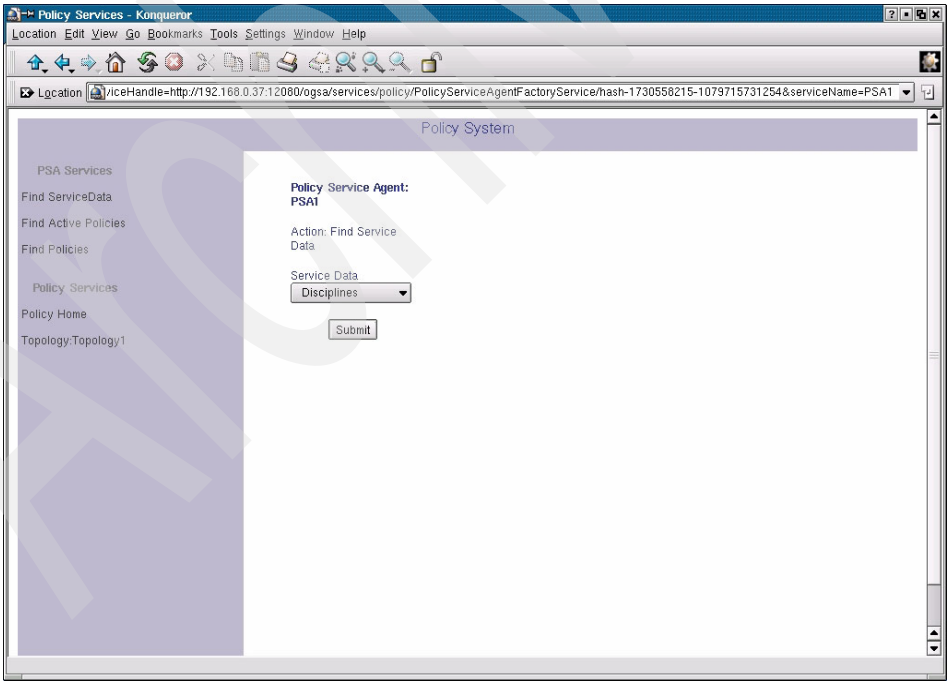


Figure 6-28 Find Service data page

4. In the Find Service Data page, designate the name of the Service Data associated with the PSA that you want to query.
5. Click **Submit** to run the query. The Web application returns the results of its search to the browser.

Using the Policy Enforcement Point

The Policy Enforcement Point (PEP) gives grid administrators a variety of operations for applying policies to resources associated with a given PEP. In addition, it provides the tools to query the service data for the PEP.

To access the PEP tools:

1. Return to the Manage topology page by clicking the **Topology Details** link on the left navigational frame.
2. Click the **PEP** you would like to query from the Policy Enforcement Point section.

Query PEP service data

The policy application enables the querying of service data elements associated with a particular PEP. To query the service data:

1. Click the **Find ServiceData** link on the left navigational frame. The Find Service data page should look similar to Figure 6-29 on page 144.

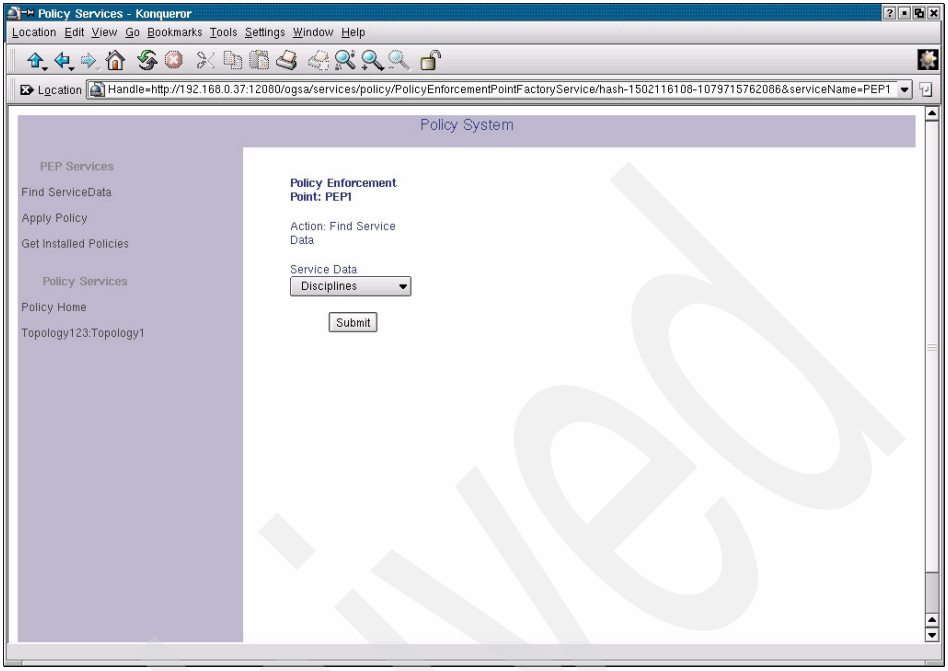


Figure 6-29 Find service data page

2. In the Find Service Data page enter the name of the service data element you would like to see.
3. Click **Submit** to run the query. The Web application will find the desired service data element and display it on the browser.

Apply a policy

The PEP gives grid administrators the ability to apply policies on a resource represented by the PEP. To do this follow these steps:

1. Click on the **Apply Policy** link on the left navigational frame. The find service data page should look similar to that in Figure 6-30 on page 145.

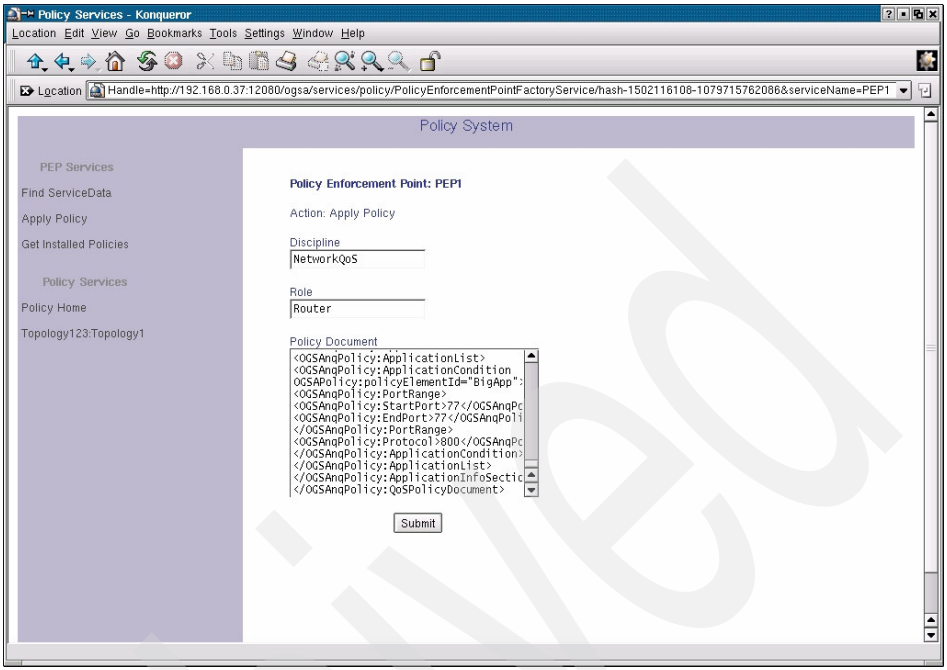


Figure 6-30 Apply policy page

2. Fill in the required policy information. Table 6-10 lists the fields, their descriptions, and the values used in the sample.

Table 6-10 Apply policy information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router
Policy Document	An XML document describing the policy	Content of policy file

3. The policy application provides a sample discipline-level policy document for this sample. Copy the entire content of this file to the Policy Document field:

```
/opt/IBMGrid/AppServer/installedApps/DefaultNode/PolicySample.ear/policy
-sample.war/data/networkQoS_sample.xml
```

4. Click **Submit**. The Web application applies the specified policy and provides confirmation in the browser.

Get installed policies

To query the PEP for installed policies:

1. Click the **Get Installed Policies** link on the left navigational frame. The Get Installed Policies page should look similar to Figure 6-31.

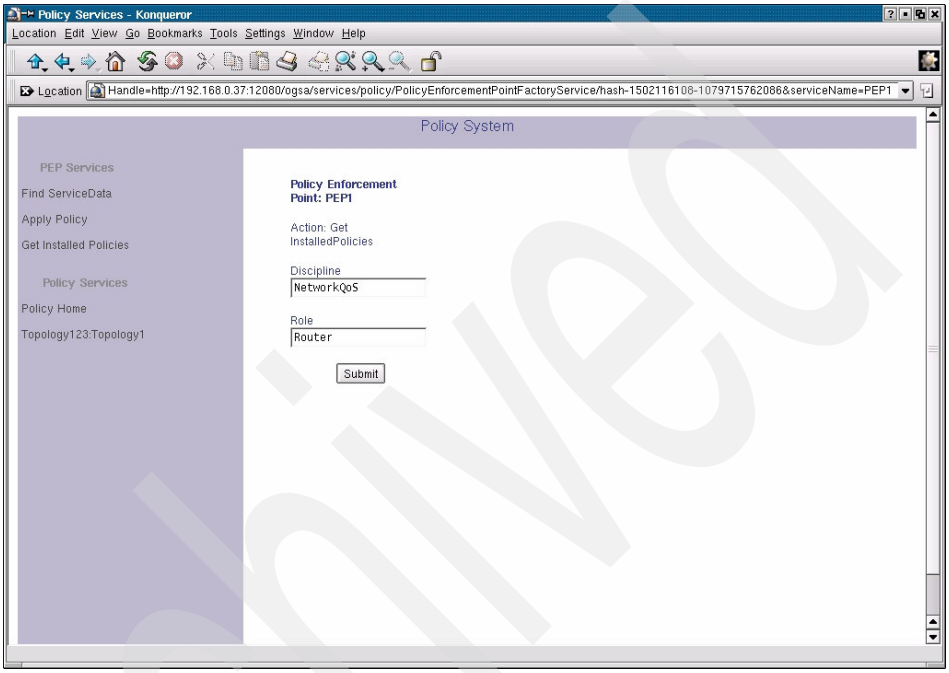


Figure 6-31 Get installed policies page

2. Fill in the required policy information. Table 6-11 lists the fields, their descriptions, and the values used in the sample.

Table 6-11 Get installed policies information

Field name	Description	Value
Discipline	Discipline for which a policy will apply	NetworkQoS
Role	Role for which a policy will apply	Router

3. Click **Submit**. The Web application retrieves the desired installed policies and displays them in the browser.

6.7 Reliable File Transfer

Reliable File Transfer (RFT) is a grid service that builds on GridFTP functionality by implementing recovery services and enhanced reliability though the use of the included Cloudscape™ database.

6.7.1 Installing RFT

Reliable File Transfer is deployed as part of the Data Management Services component in the IBM Grid Toolbox installation. If you did not deploy Data Management when you installed the IBM Grid Toolbox, you can still deploy it by running this command:

```
igt-deploy-gar /opt/IBMGrid/gars/multirft.gar
```

Note: For additional information about deploying services, refer to 5.2, “Deploying and undeploying grid services” on page 94.

6.7.2 Running RFT

Before running RFT, you must log on as *root* to start an instance of GridFTP on each host that is involved in the transfer. For more information about GridFTP, refer to 4.3, “GridFTP” on page 66.

Note: If GridFTP is automated using inetd or xinetd, it is not necessary to manually start a GridFTP server.

Important: Make sure that you have a valid proxy on each host involved with the transfer. To create a proxy as ibmgrid, run the **`grid-proxy-init`** command.

Next, create a transfer file. The RFT grid service reads in the arguments supplied by the transfer file and executes the transfer described in the file.

Note: The transfer file only has to be created on the machine from which you will be executing the RFT command.

Your RFT transfer file should contain the following elements in this order.

1. Type of transfer
   a. true = binary
   b. false = ascii
2. Block size in bytes

3. TCP buffer size in bytes
4. Not Parallel Transfer
   a. true = do not allow parallel
   b. false = allow parallel
5. Number of parallel streams
6. Data Channel Authent
   a. true = no authentication
   b. false = authentication
7. Number of concurrent file transfers
8. Source URL
9. Destination URL

Example 6-15 shows the transfer file that is used in our sample.

Example 6-15 Sample transfer file

```
false
16000
16000
false
1
true
1
gsiftp://y1.itsolab-yanomani.com:5678/tmp/ftpfile
gsiftp://y2.itsolab-yanomani.com:5678/tmp/y1remotecopy1
```

Important: It is important that source and destination URLs contain the following information:

```
gsiftp://<hostname>:<port>/<location of file to transfer>
```

Tip: To test RFT with a single host, make sure that the destination URL points to your local host.

Now, with a valid transfer file, we can issue the RFT command:

```
java org.globus.ogsa.gui.RFTClient <URL of the host providing the MultiRFT
Factory Service> <location of transfer file>
```

Tip: You can find out the URL for the RFT Factory Service by using the IBM Grid Services Manager (GSM). Refer to 5.1.5, “Managing a grid service” on page 88 for more information.

Example 6-16 shows the RFT command that we used in the lab environment and its results.

Example 6-16 RFT command and output

```
[ibmgrid@y1 tmp]$ java org.globus.ogsa.gui.RFTClient
http://y1.itsolab-yanomani.com:12080/ogsa/services/base/multirft/MultiFileRFTFa
ctoryService /opt/IBMGrid/etc/transfer.xfr

Multifile RFT command line client
Request Data Size 10 1
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
Created an instance of Multi-RFT
Request id: 16
[ibmgrid@y1 tmp]$
```

The file should now appear on the destination host.

Tip: If the transfer did not work, make sure that the GridFTP server was started with the -a option. You can also try checking your firewall settings.

6.8 Managed-job-globusrun sample

The following sample uses the command line client for Globus Resource Allocation Manager (GRAM), managed-job-globusrun. It uses the test.xml file found in the /opt/IBMGrid/etc directory. The format of the command used here is:

```
managed-job-globusrun -factory http://<host>:<port>/<service> -file <rsl>
```

The host option can be any host in your environment on which the IBM Grid Toolbox has been installed and the **igt-start-container** command has been issued. The Generic Persistent Grid Service must also be active. In our lab, Example 6-17 on page 149 shows the command that we used and the expected response.

Example 6-17 Managed-job-globusrun command

```
ibmgrid@x3:/opt/IBMGrid/etc> managed-job-globusrun -factory
http://192.168.0.11:12080/ogsa/services/base/gram/MasterForkManagedJobFactorySe
rvice -file /opt/IBMGrid/etc/test.xml
```

```
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
WAITING FOR JOB TO FINISH
========== Status Notification ==========
Job Status: Done
=========================================
DESTROYING SERVICE
SERVICE DESTROYED
```

A

Directory Tree

This appendix discusses the directory structure and content of the IBM Grid Toolbox. Specifically, the following directories are described:

- /opt/IBMGrid
- /opt/IBMGrid/AppServer
- /opt/IBMGrid/Database
- /opt/IBMGrid/OpenJMS

/opt/IBMGrid directory

This is the principle directory for the IBM Grid Toolbox. Example A-1 lists the subdirectories contained within /opt/IBMGrid.

Example: A-1 /opt/IBMGrid subdirectory listing

```
- /opt/IBMGrid/AppServer
- /opt/IBMGrid/DataBase
- /opt/IBMGrid/OpenJMS
- /opt/IBMGrid/_jvm
- /opt/IBMGrid/_uninst
- /opt/IBMGrid/bin
- /opt/IBMGrid/build
- /opt/IBMGrid/deploy
- /opt/IBMGrid/docs
- /opt/IBMGrid/endorsed
- /opt/IBMGrid/etc
- /opt/IBMGrid/gars
- /opt/IBMGrid/guide
- /opt/IBMGrid/include
- /opt/IBMGrid/lib
- /opt/IBMGrid/libexec
- /opt/IBMGrid/license
- /opt/IBMGrid/logs
- /opt/IBMGrid/man
- /opt/IBMGrid/samples
- /opt/IBMGrid/sbin
- /opt/IBMGrid/schema
- /opt/IBMGrid/scripts
- /opt/IBMGrid/setup
- /opt/IBMGrid/share
- /opt/IBMGrid/tomcat
- /opt/IBMGrid/undeploy
- /opt/IBMGrid/var
- /opt/IBMGrid/webapps
```

Some of the more common directories are:

- AppServer

  This directory contains most of the files that are associated with the IBM WebSphere Application Server - Express, V5.0, that is provided with the IBM Grid Toolbox.

- bin

  The bin directory contains a majority of the scripts and commands that are used to perform the functions that are provided with the IBM Grid Toolbox.

- Database

  This directory contains most of the files that are associated with the IBM CloudScape Database that is provided with the IBM Grid Toolbox.

- deploy

  The deploy directory contains files that are associated with grid services that have been deployed in an instance of the IBM Grid Toolbox.

- gars

  The gars directory contains all of the .gar files used by the IBM Grid Toolbox. These files include the .gar files that are used in all of the samples in this book.

- logs

  The logs directory contains the log files that are used to store information about some of the processes that are executed by the IBM Grid Toolbox (such as igt-start-container).

  Note: Application Server logs are in /opt/IBMGrid/AppServer/logs/server1/. SystemOut.log file contains the most useful information.

- OpenJMS

  This directory contains most of the files that are associated with the OpenJMS that is provided with the IBM Grid Toolbox.

- sbin

  The sbin directory contains a variety of the scripts and commands that are used to perform the functions that are provided with the IBM Grid Toolbox.

- undeploy

  The undeploy directory provides the uninstall files that are used when you need to undeploy a grid service. The directory is also an easy place to find the gar IDs for specific services.

- _uninst

  The _uninst directory contains the uninstall application that is used to uninstall an instance of the IBM Grid Toolbox.

/opt/IBMGrid/AppServer directory

This is the principle directory for the IBM WebSphere Application Server - Express V5.0 that is included with the IBM Grid Toolbox. Example A-2 lists the subdirectories that are contained within /opt/IBMGrid/AppServer.

Example: A-2 /opt/IBMGrid/AppServer subdirectory listing

```
- /opt/IBMGrid/AppServer
- - /opt/IBMGrid/AppServer/bin
- - /opt/IBMGrid/AppServer/config
- - /opt/IBMGrid/AppServer/etc
- - /opt/IBMGrid/AppServer/installableApps
- - /opt/IBMGrid/AppServer/installedApps
- - /opt/IBMGrid/AppServer/installedConnectors
- - /opt/IBMGrid/AppServer/java
- - /opt/IBMGrid/AppServer/lib
- - /opt/IBMGrid/AppServer/logs
- - /opt/IBMGrid/AppServer/properties
- - /opt/IBMGrid/AppServer/temp
- - /opt/IBMGrid/AppServer/tranlog
- - /opt/IBMGrid/AppServer/wstemp
```

Some of the more common directories are:

- bin

  The bin directory contains a majority of the scripts and commands that are used to perform the functions that are provided with IBM WebSphere Application Server - Express, V5.0.

- config

  The config directory contains some of the configuration files and settings that are associated with IBM WebSphere Application Server - Express, V5.0.

- installedApps

  The installedApps directory contains the files that are associated with the Web applications that have been installed on an instance of the IBM Grid Toolbox.

Note:
AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/schema/ contains most of the schema files that are deployed.

AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF is the actual deployment directory of most of the toolkit files.

- java

  The java directory contains files for a copy of Java that is included with IBM WebSphere Application Server - Express, V5.0.

- logs

  The logs directory contains the log files that are used to store information about some of the processes that are executed by IBM WebSphere Application Server - Express, V5.0.

/opt/IBMGrid/DataBase directory

This is the principle directory for the IBM CloudScape Database that is included with the IBM Grid Toolbox. Example A-3 lists the subdirectories that are contained within /opt/IBMGrid/Database.

Example: A-3 /opt/IBMGrid/DataBase subdirectory listing

```
- /opt/IBMGrid/DataBase
- - /opt/IBMGrid/DataBase/lib
- - /opt/IBMGrid/DataBase/system
```

Some of the more common directories are:

- lib

  The lib directory contains some of the .jar files that are used by the IBM CloudScape Database.

- system

  The system directory contains the system files that are used by the IBM CloudScape Database.

/opt/IBMGrid/OpenJMS directory

This is the principle directory for the OpenJMS environment that is included with the IBM Grid Toolbox. Example A-4 lists the subdirectories that are contained within /opt/IBMGrid/OpenJMS.

Example: A-4 /opt/IBMGrid/OpenJMS subdirectory listing

```
- /opt/IBMGrid/OpenJMS
- - /opt/IBMGrid/OpenJMS/bin
- - /opt/IBMGrid/OpenJMS/config
- - /opt/IBMGrid/OpenJMS/lib
- - /opt/IBMGrid/OpenJMS/src
```

Some of the more common directories are:

- bin

  The bin directory contains a majority of the scripts and commands that are used to perform the functions for the OpenJMS environment that are provided with the IBM Grid Toolbox.

- config

  The config directory contains some of the configuration files and scripts that are associated with OpenJMS.

- lib

  The lib directory contains some of the .jar files that are used by the OpenJMS environment that are included with the IBM Grid Toolbox.

- src

  The source directory contains some of the source information and files that are related to the copy of OpenJMS that is included with the IBM Grid Toolbox.

B

Commands

This appendix discusses some of the commands that are employed in the daily use and management of a grid instance.

The big picture

The IBM Grid Toolbox contains a collection of commands to help in managing most of the tasks of the grid. The following contains a description of the commands frequently used and the common output for most of them; these outputs can vary depending on the particular configuration of a grid instance. The commands are expected to be run by the IBM Grid Toolbox administrator (ibmgrid). When a command has a specific user ID requirement, it is noted.

Note: Some commands do not have example output.

igt-add-cmmconnectionfactory

The **igt-add-cmmconnectionfactory** command adds a connection to the CIMOM server.

Note: All values are required.

Synopsis

```
usage: igt-add-cmmconnectionfactory -cfName <cf_name> -jndiName <jndi_name>
-serverName <host_name> -portNumber <port> -userName <user_name> -password
<password> -nameSpace <name_space>
 cfName - Connection Factory Name (required)
 jndiName - JNDI Name of the Connection Factory (required)
 serverName - IP address of CIMOM (required)
 portNumber - CIMOM port (required)
 userName - user name to connect to CIMOM (required)
 password - password to connect to CIMOM (required)
 nameSpace - name space to connect to within the CIMOM (required)
```

Example

Example B-1 provides an example of this command.

Example: B-1 igt-add-cmmconnectionfactory command

```
[ibmgrid@y1 ibmgrid]$ igt-add-cmmconnectionfactory -cfName gridtool -jndiName
ibmgrid -serverName 192.168.0.21 -portNumber 5988 -userName ibmgrid -password
passwOrd -nameSpace root/cimv2
option=-cfName
option=gridtool
option=-jndiName
option=ibmgrid
option=-serverName
option=192.168.0.21
option=-portNumber
```

```
option=5988
option=-userName
option=ibmgrid
option=-password
option=passw0rd
option=-nameSpace
option=root/cimv2
PROPS=-javaoption -Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer
option=-c
option=ra-connection-factory
option=-rarName
option=CimRA
option=-cfName
option=gridtool
option=-jndiName
option=ibmgrid
option=-serverName
option=192.168.0.21
option=-portNumber
option=5988
option=-userName
option=ibmgrid
option=-passw
option=passw0rd
option=-nameSpace
option=root/cimv2
option=-adapterType
option=cim
/opt/IBMGrid/AppServer/bin/wsadmin.sh -connytpe NONE -javaoption
-Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer -f
/opt/IBMGrid/scripts/AppServer/ra-connection-factory.jacl -rarName CimRA
-cfName gridtool -jndiName ibmgrid -serverName 192.168.0.21 -portNumber 5988
-userName ibmgrid -passw passw0rd -nameSpace root/cimv2 -adapterType cim
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7357I: By request, this scripting client is not connected to any server
process. Certain configuration and application operations will be available in
local mode.
createCF: Retrieving the Cim Resource Adapter Details
ConnectionFactory gridtool has been created
```

Associated files:

- /opt/IBMGrid/AppServer/installedApps/DefaultNode/crm-sample_war.ear/crm-sample.war/WEB-INF/CRMResourceBinding.xml

igt-add-user

The **igt-add-user** command adds a new user to the local IBM Grid Services Manager registry. If a password is not included as a parameter, you will be prompted for one during execution.

Note: Executing this command does not add the user to the operating system.

Syntax

```
igt-add-user [-help] USER_NAME [-n "Full Name"]
```

Example

Example B-2 provides an example of this command.

Example: B-2 igt-add-user command

```
[ibmgrid@y1 ibmgrid]$ igt-add-user fcastro
Password:
[ibmgrid@y1 ibmgrid]$
```

igt-change-port

The **igt-change-port** command changes which port is used by the grid container.

Note: The grid container should be stopped with the **igt-stop-container** command before executing a port change.

Synopsis

```
Usage:igt-change-port <port_value>
Changes the HTTP port for the application server.Server must be restarted for
change to take effect.
Arguements:
port_value:new value for the HTTP port (must be an integer)
```

Examples

Example B-3 provides an example of this command changing the port from 12080 (default) to 12090.

Example: B-3 igt-change-port command

```
[ibmgrid@y1 ibmgrid]$ igt-change-port 12090
```

igt-change-timeout

The **igt-change-timeout** command changes the timeout value for the Apache Axis component. The timeout value should be in seconds.

Note: After executing this command, you have to stop and restart the grid container with the **igt-stop-container** and **igt-start-container** commands.

Synopsis

```
Usage:igt-change-timeout <time value>

Changes the Axis client timeout

Arguments:
<time value>The new timeout value.
```

Examples

Example B-4 provides an example of this command, changing the timeout to 5 seconds.

Example: B-4 igt-change-timeout command

```
[ibmgrid@y1 ibmgrid]$ igt-change-timeout 5
```

igt-container-status

The **igt-container-status** command provides the status of a local grid container.

Synopsis:

```
igt-container-status
```

Example

Example B-5 provides an example of this command.

Example: B-5 igt-container-status command

```
[ibmgrid@y1 ibmgrid]$ igt-container-status
/opt/IBMGrid/AppServer/bin/serverStatus.sh: ulimit: cannot modify limit:
Operation not permitted
IBMGrid container is started
```

igt-delete-ca

The **igt-delete-ca** command provides a means with which to remove a CA from a host.

Note: This command should be executed as *root*.

Note: The CA to be removed cannot be set as the default CA. If you want to remove a default CA first use the **igt-grid-default-ca** command to change the default.

Synopsis

```
igt-delete-ca [-help] <options>
Options:
-help             : Display this message
-ca <ca hash>     : remove the specified CA non-interactively
```

Example

Example B-6 provides an example of this command.

Example: B-6 igt-delete-ca command

```
[ibmgrid@y1 ibmgrid]$ igt-delete-ca
****************************************************************
* The available CA configurations installed on this host are:
****************************************************************
** WARNING: This script will delete files on your system.
** Use this command with caution

1) e82f5117 -  /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca
2) e82f5118 -  /C=SP/ST=Madrid/L=Madrid/O=ITSOLAB/CN=ca/Email=cd@ca
The default CA is:  e82f5117

Enter the index number of the CA to be deleted: e82f5118
```

Associated files

- /etc/grid-security/certificates/<CA hash>.0
- /etc/grid-security/certificates/<CA hash>.signing_policy
- /etc/grid-security/certificates/globus-host-ssl.conf.<CA hash>
- /etc/grid-security/certificates/globus-user-ssl.conf.<CA hash>
- /etc/grid-security/certificates/grid-security.conf.<CA hash>

igt-delete-cmmconnectionfactory

The **igt-add-cmmconnectionfactory** command removes a connection to the CIMOM server.

Synopsis

```
usage: igt-delete-cmmconnectionfactory -cfName <cf_name>
cfName - Connection Factory Name (required)
```

Example

Example B-7 provides an example of this command.

Example: B-7 igt-delete-cmmconnectionfactory command

```
[ibmgrid@y1 ibmgrid]$ igt-delete-cmmconnectionfactory -cfName gridtool
option=-cfName
option=gridtool
PROPS=-javaoption -Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer
option=-c
option=remove-ra-connection-factory
option=-raName
option=CimRA
option=-cfName
option=gridtool
/opt/IBMGrid/AppServer/bin/wsadmin.sh -connytpe NONE -javaoption
-Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer -f
/opt/IBMGrid/scripts/AppServer/remove-ra-connection-factory.jacl -raName CimRA
-cfName gridtool
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7357I: By request, this scripting client is not connected to any server
process. Certain configuration and application operations will be available in
local mode.
ConnectionFactory gridtool has been deleted
```

Associated files

- /opt/IBMGrid/AppServer/installedApps/DefaultNode/crm-sample_war.ear/crm-sample.war/WEB-INF/CRMResourceBinding.xml

igt-delete-user

The **igt-delete-user** command removes a user from the local IBM Grid Services Manager registry.

Note: Execution of this command does will not remove the user from the operating system.

Attention: The command will not ask you to confirm the deletion of a user.

Syntax

```
igt-delete-user -help USER_NAME
Options:
-help: display this message.
USER_NAME: name of the grid user to be deleted.
```

Example:

Example B-8 provides an example of this command.

Example: B-8 igt-delete-user command

```
[ibmgrid@y1 ibmgrid]$ igt-delete-user fcastro
[ibmgrid@y1 ibmgrid]$
```

igt-deploy-gar

The **igt-deploy-gar** command provides a means to deploys .gar files for use in the grid container. In addition **igt-deploy-gar** will execute any scripts associated with the deployment of a grid service.

Synopsis

```
Usage: igt-deploy-gar <gar file name> {<gar file name>}
                      [-predeploy    <pre-deploy script name>]
                      [-postdeploy   <post-deploy script name>]
                      [-preundeploy  <pre-undeploy script name>]
                      [-postundeploy <post-undeploy script name>]
                      [-garListFile  <file containing list of gars>]
                      [-help]

Arguments:
  <gar file name>  A file containing the Grid service ARchive to be deployed.
The file name has the format <gar id>.gar

Options:
   -help           Displays help
   -predeploy <pre-deploy script name>
           Run the given script name, instead of the default
           <gar id>-preDeploy.sh
```

```
   -postdeploy <post-deploy script name>
          Run the given script name, instead of the default
          <gar id>-postDeploy.sh
   -preundeploy <pre-undeploy script name>
          Run the given script name, instead of the default
          <gar id>-preUndeploy.sh
   -postundeploy <post-undeploy script name>
          Run the given script name, instead of the default
          <gar id>-postUndeploy.sh
   -garListFile <file containing list of gars>
         Deploys all of the gars listed in the file.  When
         this option is chosen, any pre/post [un]deploy
         scripts specified on the command line are
         disregarded.
```

Example:

Example B-9 provides an example of this command when deploying a new .gar file.

Example: B-9 igt-deploy-gar command

```
[ibmgrid@y2 gars]$ igt-deploy-gar /opt/IBMGrid/gars/crm-core.gar
2004-03-15 11:12:39: Starting
Number of gars to deploy: 1
   /opt/IBMGrid/gars/crm-core.gar
2004-03-15 11:12:39: Expanding all gars ..
extracted: crm-core.jar
extracted: crm-stubs.jar
  created: deploy/
extracted: deploy/crm-core-postDeploy.sh
  created: schema/
  created: schema/base/
  created: schema/base/crm/
  created: schema/base/crm/security/
extracted: schema/base/crm/security/AuthSchema.xsd
----- ----- -----
extracted: schema/base/crm/FileSystem_PT.wsdl
deploying /opt/IBMGrid/gars/crm-core.gar ...
Copying the schema files ...
Copying the jars ...
Generating the undeployment information ...
2004-03-15 11:12:43: Running
/opt/IBMGrid/deploy/crm-core/crm-core-postDeploy.sh
PROPS=-javaoption -Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer
option=-connect
option=-c
option=resource-adapter
option=-rarName
option=CimRA
```

```
option=-rarFile
option=/opt/IBMGrid/AppServer/installableApps/cim-connector.rar
/opt/IBMGrid/AppServer/bin/serverStatus.sh: ulimit: cannot modify limit:
Operation not permitted

container state = 1
/opt/IBMGrid/AppServer/bin/wsadmin.sh -connytpe SOAP -javaoption
-Dcom.ibm.grid.wsadmin.lib=/opt/IBMGrid/scripts/AppServer -f
/opt/IBMGrid/scripts/AppServer/resource-adapter.jacl -rarName CimRA -rarFile
/opt/IBMGrid/AppServer/installableApps/cim-connector.rar
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7209I: Connected to process "server1" on node DefaultNode using SOAP
connector;  The type of process is: UnManagedProcess
2004-03-15 11:13:53: Done processing
```

Example B-10 provides an example of this command when you specify the garListFile option that takes as its argument a file, which contains a list of gars to deploy (ex: /opt/IBMGrid/gars/deploy.lst). In this particular case, we are trying to deploy gars that have already been deployed. So, the informational messages you see are **igt-deploy-gar** telling the user that it refuses to deploy those gars because they already have been deployed.

Example: B-10 igt-deploy-gar command

```
[ibmgrid@y2 gars]$ igt-deploy-gar -garListFile /opt/IBMGrid/gars/deploy.lst
2004-03-15 10:36:43: Starting
'mds-index.gar' has already been deployed.
'mds-db.gar' has already been deployed.
'mds-providers.gar' has already been deployed.
'mds-aggregator.gar' has already been deployed.
'multirft.gar' has already been deployed.
'filestreaming.gar' has already been deployed.
'gram-rips.gar' has already been deployed.
'mmjfs.gar' has already been deployed.
'mds-aggregator.gar' has already been deployed.
'crm-core.gar' has already been deployed.
'servicegroup.gar' has already been deployed.
'policy-core.gar' has already been deployed.
Number of gars to deploy: 0
```

Associated files:

- .gar in /opt/IBMGrid/gars
- /opt/IBMGrid/gars/deploy.lst

igt-grid-cert-request

The **igt-grid-cert-request** command creates host and user certificate requests in the supplied directory. The request files can then be sent to a certificate authority.

Note: If a directory is not supplied, the default value is:

```
/home/<username>/.globus
```

Synopsis

```
igt-grid-cert-request [-help ][options ...]

Example Usage:

Creating a user certificate:
igt-grid-cert-request

Creating a host or gatekeeper certificate:
igt-grid-cert-request -host [my.host.fqdn ]

Options:
    -version          : Display version
    -?, -h, -help,    : Display usage
    -usage
    -cn <name>,       : Common name of the user
    -commonname <name>
    -host <FQDN>      : Create certificate for a host named <FQDN>
    -dir <dir_name>   : Changes the directory the private key and certificate
                        request will be placed in. By default user
                        certificates are placed in /home/ibmgrid/.globus, host
                        certificates are placed in /etc/grid-security and
                        service certificates are place in
                        /etc/grid-security/<service>.
    -prefix <prefix>  : Causes the generated files to be named
                        <prefix>cert.pem, <prefix>key.pem and
                        <prefix>cert_request.pem
    -nodes,
    -verbose          : Don't clear the screen
    -int[eractive]    : Prompt user for each component of the DN
    -force            : Overwrites preexisting certifictes
    -ca               : Will ask which CA is to be used (interactive)
    -ca <hash>        : Will use the CA with hash value <hash>
```

Examples

Example B-11 on page 168 provides an example of this command creating user certificates in /tmp/certtest/.

Example: B-11 igt-grid-cert-request command

```
[ibmgrid@y2 IBMGrid]$ igt-grid-cert-request -dir /tmp/certtest
Using user id: ibmgrid
Common Name: IBM Grid Toolbox V3 for Multiplatforms

A private key and a certificate request has been generated with the subject:

/C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=IBM Grid Toolbox V3 for Multiplatforms

If the CN=IBM Grid Toolbox V3 for Multiplatforms is not appropriate, rerun this
script with the -force -cn "Common Name" options.

Your private key is stored in /tmp/certtest/userkey.pem
Your request is stored in /tmp/certtest/usercert_request.pem

This certificate request (/tmp/certtest/usercert_request.pem) should be
processed with your trusted Certificate Authority, consistent with your site
security policy.
```

Associated files

- userkey.pem
- usercert_request.pem
- usercert.pem

igt-grid-default-ca

The **igt-grid-default-ca** command provides a means with which to set a CA as the default for a host.

Note: This command should be executed as *root*.

Synopsis

```
igt-grid-default-ca [-help] [options ....]
Options:
-help           : Display this message
-dir <dir_name>  : The security config directory (defaults to
/etc/grid-security/)
-list           : List the available CAs to use and the current default
-ca <ca hash>    : Set the default CA non-interactively
```

Example

Example B-12 on page 169 provides an example of this command.

Example: B-12 igt-grid-default-ca command

```
[ibmgrid@y1 ibmgrid]$ igt-grid-default-ca -list
The available CA configurations installed on this host are:
1) e82f5117 -  /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca
The default CA is:  e82f5117
```

Example B-13 provides another example of this command.

Example: B-13 igt-grid-default-ca command

```
[ibmgrid@y1 ibmgrid]$ igt-grid-default-ca -ca e82f5117
setting the default CA to: /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca
linking /etc/grid-security//certificates//grid-security.conf.e82f5117 to
        /etc/grid-security//grid-security.conf
linking /etc/grid-security//certificates//globus-host-ssl.conf.e82f5117 to
        /etc/grid-security//globus-host-ssl.conf
linking /etc/grid-security//certificates//globus-user-ssl.conf.e82f5117 to
        /etc/grid-security//globus-user-ssl.conf
```

Associated files

- grid-security.conf
- globus-host-ssl.conf
- globus-user-ssl.conf

igt-import-ca

The **igt-import-ca** command imports a CA certificate provided by a trusted CA.

Note: This command should be executed as *root*.

Synopsis:

```
igt-import-ca [-help] -type [CA TYPE][options]
Import a certificate authority file/cert.
Options:
    -help     : Display this message
    -default  : Make installed certificate a default host CA
CA types are:
    gt3:    imports a simpleCA install tar ball
            Required options: -tar [TARBALL_FILE]
    pem:    imports PEM encoded CA certificate.
            Required options: -cert [CERT_FILENAME]
    der:    imports DER encoded CA certificate.
            Required options: -cert [CERT_FILENAME]
```

```
Samples:
    1. Import an entrust CA cert & make it default
        igt-import-ca -type PEM -cert /tmp/entrust/cacert.pem -default
    2. Import a Microsoft der encoded CA certificate and make it default
        igt-import-ca -type der -cert /tmp/ms/cacert.der -default
    3. Import the Globus simple CA export package.
        igt-import-ca -type GT3 -tar
globus_simple_ca_3719641b_setup-0.12.tar.gz
```

Examples

Example B-14 provides an example of this command, importing a CA certificate and setting it as the default.

Example: B-14 igt-import-ca command

```
[root@y1 tmp]# igt-import-ca -type PEM -cert /tmp/y1-usercert.pem -default
**********************************************************
grid security folder: /etc/grid-security/
has been backed up to:
/etc/grid-security//backup-gsi-Wed-Mar-10-16-11-59-CST-2004.tar
**********************************************************
Configuring GSI security...
Cert Hash: 7e11311b

Creating /etc/grid-security//certificates/grid-security.conf.7e11311b
Creating /etc/grid-security//certificates/globus-host-ssl.conf.7e11311b
Creating /etc/grid-security//certificates/globus-user-ssl.conf.7e11311b

Saving policy file in: /etc/grid-security//certificates/7e11311b.signing_policy
Certificate DN: /C=US/ST=Texas/O=ITSOLAB/CN=usery1

Copying /tmp/y1-usercert.pem to /etc/grid-security//certificates/7e11311b.0

Setting CA 7e11311b, as default...
linking /etc/grid-security//certificates/grid-security.conf.7e11311b to
       /etc/grid-security//grid-security.conf

linking /etc/grid-security//certificates/globus-host-ssl.conf.7e11311b to
       /etc/grid-security//globus-host-ssl.conf

linking /etc/grid-security//certificates/globus-user-ssl.conf.7e11311b to
       /etc/grid-security//globus-user-ssl.conf

Done.
```

Associated files

- /etc/grid-security/certificates

- /etc/grid-security/globus-host-ssl.conf
- /etc/grid-security/globus-user-ssl.conf
- /etc/grid-security/grid-security.conf

igt-install-certs

The **igt-install-certs** command provides a means to install both host and user certificates.

Note: The installation of host certificates must be done as *root*.

Synopsis

```
igt-install-certs [-help] [ -user <userid>  | -host ] -cert <cert_file>
    Options:
      -help           : Display this message
      -user <userid>  : Install a certificate for the specified user (optional)
      -host           : Install a specified host cert
      -cert <cert_file> : Cert to be installed as user or host
      -force          : force overwriting destination certificates
```

Example

Example B-15 provides an example of this command when installing a user certificate.

Example: B-15 igt-install-certs command

```
[root@y1 etc]# igt-install-certs -user ibmgrid -cert /tmp/y1-usercert.pem
******************************************************
Installing user cert: /tmp/y1-usercert.pem
User: ibmgrid
Destination: /home/ibmgrid/.globus/usercert.pem
Cert DN: /C=US/ST=Texas/O=ITSOLAB/CN=usery1
******************************************************
Copying /tmp/y1-usercert.pem to /home/ibmgrid/.globus/usercert.pem...
Setting permissions...

Adding user ibmgrid to gridmapfile...
(1) entry added

Added grid-mapfile entry for:
       User: ibmgrid
       DN: /C=US/ST=Texas/O=ITSOLAB/CN=usery1
```

```
INFO: Copying user key: /tmp/userkey.pem to
/home/ibmgrid/.globus/userkey.pem...

Done.
```

Associated files

- /home/ibmgrid/.globus/usercert.pem
- /etc/grid-security/grid-mapfile

igt-list-users

The **igt-list-users** command lists all users registered with the local IBM Grid Services Manager registry.

Note: Users listed with this command may not have accounts with the operating system.

Synopsis

```
List users from the grid registry: igt-list-users [-help]
```

Examples

Example B-16 provides an example of this command.

Example: B-16 igt-list-users command

```
[ibmgrid@y1 ibmgrid]$ igt-list-users
ibmgrid
dmmosby
Mosby
acastro
```

Associated files

- /opt/IBMGrid/AppServer/etc/registry/users.props
- /opt/IBMGrid/AppServer/etc/registry/groups.props

igt-set-admin-user

The **igt-set-admin-user** command sets a user as the administrator with the local IBM Grid Services Manager registry.

Synopsis

```
Usage:

igt-set-admin-user users

Where users is a space delimited list of users to add.
You can specify a maximum of 9 users.
Users must containe valid users in the user registry. Use igt-useradd to add,
igt-userdel to delete and igt-userlist to list users in the registry.
```

Example

Example B-17 provides an example of this command.

Example: B-17 igt-set-admin-user command

```
[ibmgrid@y1 IBMGrid]$ igt-add-user ibmtest -p passw0rd -n "Ibmtest Fullname"
[ibmgrid@y1 IBMGrid]$ igt-set-admin-user ibmtest
/opt/IBMGrid/AppServer/bin/wsadmin.sh: ulimit: cannot modify limit: Operation
not permitted
WASX7357I: By request, this scripting client is not connected to any server
process. Certain configuration and application operations will be
available in local mode.

Starting processing
Processing user: ibmtest
ibmtest
Finished processing users
ibmtest
{Administrator No No ibmtest {}}
Added users, ibmtest , to administrator role

Saving configuration
[ibmgrid@y1 IBMGrid]$
```

Associated files

- /opt/IBMGrid/AppServer/etc/registry/users.props
- /opt/IBMGrid/AppServer/etc/registry/groups.props

igt-setenv.sh

This shell script simply exports all of the necessary variables so that the IBM Grid Toolbox environment will work properly. After booting or rebooting the host, if this is not run by default, it should be run as follows from a shell window.

Example: B-18 Sourcing igt-setenv.sh

```
[ibmgrid@y1 ibmgrid]$ source /opt/IBMGrid/igt-setenv.sh
[ibmgrid@y1 ibmgrid]$
```

Some Unix operating systems do not have the source command available; in this case, just execute the command as shown in Example B-19.

Note: Take notice of the space between the dot '.' and the command itself.

Example: B-19 Executing igt-setenv.sh

```
[ibmgrid@y1 ibmgrid]$ . /opt/IBMGrid/igt-setenv.sh
[ibmgrid@y1 ibmgrid]$
```

Example B-20 provides the content of the shell script.

Example: B-20 igt-setenv.sh content

```
##############################################################################
#  Licensed Materials - Property of IBM
####  5765-G29 #  5765-G22
##
##  (C) Copyright IBM Corp. 2003. All Rights Reserved.
##
##  US Government Users Restricted Rights - Use, duplication or
##  disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
##
##  %Z%%M%        %I%  %W% %G% %U%
##############################################################################
export GLOBUS_LOCATION=/opt/IBMGrid
export WAS_INSTALL=$GLOBUS_LOCATION/AppServer
export JAVA_HOME=$WAS_INSTALL/java
export IGT_CONTAINER_STARTED=1
export IGT_CONTAINER_STOPPED=2
export INSTALLABLE_APPS=$WAS_INSTALL/installableApps
export IGT_CONFIGURE=$GLOBUS_LOCATION/scripts/AppServer/igt-configure.sh
export IGT_APP_NAME=IBMGrid
export DB_DIR=$GLOBUS_LOCATION/DataBase
export DB_LIB=$DB_DIR/lib
export DB_SYSTEM=$DB_DIR/system
export CLASSPATH=$DB_LIB/db2j.jar:$DB_LIB/db2jtools.jar
export IGT_DEFAULT_TIMEOUT=600000
. ${GLOBUS_LOCATION}/setenv.sh
. ${GLOBUS_LOCATION}/etc/globus-user-env.sh
export PATH=${JAVA_HOME}/bin:${PATH}
```

igt-start-container

The **igt-start-container** command starts the local grid container.

Synopsis

```
Usage: igt-start-container [-help]
Starts the IBMGrid container (Websphere Application Server)
Arguments: none
```

Example

Example B-21 provides an example of this command.

Example: B-21 igt-start-container command

```
[ibmgrid@y2 ibmgrid]$ igt-start-container
/opt/IBMGrid/AppServer/bin/startServer.sh: ulimit: cannot modify limit:
Operation not permitted
ADMU0116I: Tool information is being logged in file
           /opt/IBMGrid/AppServer/logs/server1/startServer.log
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is 11921
```

igt-stop-container

The **igt-stop-container** command stops the local grid container.

Synopsis

```
Usage: igt-stop-container [-help]
Stop the IBMGrid container (Websphere Application Server)
Arguments: none
```

Example

Example B-22 provides an example of this command.

Example: B-22 igt-stop-container command

```
[ibmgrid@y2 ibmgrid]$ igt-stop-container
/opt/IBMGrid/AppServer/bin/stopServer.sh: ulimit: cannot modify limit:
Operation not permitted
ADMU0116I: Tool information is being logged in file
           /opt/IBMGrid/AppServer/logs/server1/stopServer.log
ADMU3100I: Reading configuration for server: server1
ADMU3201I: Server stop request issued. Waiting for stop status.
ADMU4000I: Server server1 stop completed.
```

igt-undeploy-gar

The **igt-undeploy-gar** command provides a means to undeploy .gar files from a grid container. In addition, **igt-deploy-gar** executes any scripts that are associated with the undeployment of a grid service.

Note: For information about obtaining a gar ID, refer to 5.2.2, “Undeploying” on page 96.

Synopsis

```
Usage: igt-undeploy-gar [gar id] [-help]
Reverses the effect of a deploy operation for a particular GAR.
Arguments:
  [gar id]  Name of the Grid service ARchive. (The corresponding GAR deployed
was <gar id>.gar)
Options:
  -help     Displays help
```

Example

Example B-23 provides an example of this command when deploying a new .gar file.

Example: B-23 igt-undeploy-gar command

```
[ibmgrid@y2 ibmgrid]$ igt-undeploy-gar mds-db
2004-03-11 14:08:19: Starting
2004-03-11 14:08:19: Running /opt/IBMGrid/undeploy/mds-db-undeploy.sh
Undeploying mds-db
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
...(author omits lines)
2004-03-11 14:08:36: Done processing
```

Associated files

- .gar in /opt/IBMGrid/gars
- /opt/IBMGrid/gars/deploy.lst

globus-domainname

The **globus-domainname** command returns the system’s domain name.

Synopsis

```
globus-domainname [-help] [-version]

    globus-domainname tries to return the system domainname
    Setting the environment variable GLOBUS_HOSTNAME will cause
    globus-domainname to return a valule based on this variable.
    This is useful for specifying the use of certain network interfaces when
    communicating etc.
```

Example

Example B-24 provides an example of this command.

Example: B-24 globus-domainname command

```
[ibmgrid@y1 ibmgrid]$ globus-domainname
itsolab-yanomani.com
```

globus-hostname

The **globus-hostname** command returns the system's host name.

Synopsis

```
globus-hostname [-help] [-version]

    globus-hostname returns the system hostname and makes some
    additional checks to ensure a fully qualified hostname.
    Setting the environment variable GLOBUS_HOSTNAME will cause
    globus-hostname to return a valule based on this variable.
    This is useful for specifying the use of certain network interfaces when
    communicating etc.
```

Example

Example B-25 provides an example of this command.

Example: B-25 globus-hostname command

```
[ibmgrid@y1 ibmgrid]$ globus-hostname
y1.itsolab-yanomani.com
```

globus-url-copy

The **globus-url-copy** command provides a command line interface for using GridFTP.

Synopsis

```
globus-url-copy [options] sourceURL destURL
OPTIONS
         -help | -usage
              Print help
         -version
              Print the version of this program
         -versions
              Print the versions of all modules that this program uses
         -a | -ascii
              convert the file to/from netASCII format to/from local file
format
         -vb | -verbose
              during the transfer, display the number of bytes transferred
              and the transfer rate per second
         -dbg |-debugftp
              Debug ftp connections.  Prints control channel communication
              to stderr
         -b | -binary
              Do not apply any conversion to the files. *default*
         -s  <subject> | -subject <subject>
              Use this subject to match with both the source and dest servers
         -ss <subject> | -source-subject <subject>
              Use this subject to match with the source server
         -ds <subject> | -dest-subject <subject>
              Use this subject to match with the destionation server
         -tcp-bs <size> | -tcp-buffer-size <size>
              specify the size (in bytes) of the buffer to be used by the
              underlying ftp data channels
         -bs <block size> | -block-size <block size>
              specify the size (in bytes) of the buffer to be used by the
              underlying transfer methods
         -p <parallelism> | -parallel <parallelism>
              specify the number of streams to be used in the ftp transfer
         -notpt | -no-third-party-transfers
              turn third-party transfers off (on by default)
         -nodcau | -no-data-channel-authentication
              turn off data channel authentication for ftp transfers
```

Example

Example B-26 provides an example of this command when transferring a file from the local file system to a GridFTP server on a second host.

Example: B-26 globus-url-copy command

```
[ibmgrid@y1 ibmgrid]$ globus-url-copy file:///home/ibmgrid/transfertest
gsiftp://y2.itsolab-yanomani.com/tmp/transfertest
```

grid-cert-info

The **grid-cert-info** command provides information about a specified certificate.

Synopsis

```
grid-cert-info [-help] [-file certfile] [-all] [-subject] [...]

    Displays certificate information. Unless the optional -file argument is
given, the default location of the file containing the certficate is assumed:

      -- The location pointed to by the .
      -- If X509_USER_CERT not set, /home/ibmgrid/.globus/usercert.pem.

    Several options can be given: The output of
        "grid-cert-info -subject -issuer"
    is equivalent to that of
        "grid-cert-info -subject ; grid-cert-info -issuer"

    Options
      -help, -usage              Display usage
      -version                   Display version
      -file certfile     |-f      Use 'certfile' at non-default location

    Options determining what to print from certificate
      -all                     Whole certificate
      -subject           |-s    Subject string of the cert
      -issuer            |-i    Issuer
      -startdate         |-sd   Validity of cert: start date
      -enddate           |-ed   Validity of cert: end date
```

Example

Example B-27 provides an example of this command.

Example: B-27 grid-cert-info command

```
[ibmgrid@y2 gars]$ grid-cert-info
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 18 (0x12)
        Signature Algorithm: md5WithRSAEncryption
        Issuer: C=US, ST=Texas, L=Austin, O=ITSOLAB, CN=ca/Email=cd@ca
        Validity
            Not Before: Mar  3 15:34:31 2004 GMT
            Not After : Mar  3 15:34:31 2005 GMT
        Subject: C=US, ST=Texas, O=ITSOLAB, CN=usery2
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
```

```
            RSA Public Key: (1024 bit)
                Modulus (1024 bit):
                    00:bc:5a:96:2c:eb:3d:67:52:47:eb:98:5c:93:52:
                    2d:19:16:94:42:d5:3f:a9:3c:71:3b:d0:e2:c2:64:
                    1b:4e:bd:4d:8b:d0:48:cd:24:76:3c:ed:02:60:cf:
                    ca:01:ad:1c:a0:f6:aa:da:75:76:7b:e0:b5:b0:db:
                    56:03:95:4b:48:71:35:51:3e:bc:73:f7:7d:f5:3d:
                    70:83:71:f4:9c:29:04:dc:6e:a5:ff:ed:cb:b1:11:
                    5e:ca:d3:3e:b3:b9:1b:e0:31:ee:20:b4:06:77:53:
                    4c:2b:c6:da:e8:b3:e5:3d:4a:c4:4a:d8:29:82:d7:
                    d8:97:f7:90:41:75:49:b1:65
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints:
                CA:FALSE
            Netscape Comment:
                OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
                B6:50:68:8E:B9:6D:41:EF:D8:D9:D9:3F:3F:4D:2D:76:EC:AE:F1:5A
            X509v3 Authority Key Identifier:

keyid:FE:86:77:C5:F7:A4:AB:3A:0B:7C:DA:A5:0F:C6:62:22:F0:DD:5A:ED
                DirName:/C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/Email=cd@ca
                serial:00

    Signature Algorithm: md5WithRSAEncryption
        5b:53:9c:45:18:0e:f6:2e:06:95:50:d0:fb:b6:63:ee:33:15:
        1b:db:b8:94:84:1e:4d:3d:36:d3:fe:8a:08:2c:0b:2b:83:a8:
        13:1d:76:48:ab:47:b2:70:6a:57:2e:fa:39:b5:9e:d7:c5:1f:
        34:77:c8:4d:f7:ca:eb:56:59:c6:7a:1b:87:75:3f:e4:b0:4d:
        c5:ad:e0:7e:90:85:f4:1e:5c:d6:56:a0:78:3a:cd:02:28:2e:
        2f:40:51:2e:f8:61:5c:ed:64:c5:b3:35:71:e3:aa:6d:b5:8e:
        74:8c:ae:18:8b:64:52:d7:4c:87:09:d4:ad:ee:b8:56:f6:ef:
        81:5a
```

Associated files:

- /home/ibmgrid/.global/usercert.pem

grid-change-pass-phrase

The **grid-change-pass-phrase** command changes the pass phrase that is used with a particular private key.

Synopsis

```
grid-change-pass-phrase [-help] [-version] [-file private_key_file]

   Changes the passphrase that protects the private key. Note that
   this command will work even if the original key is not password
   protected. If the -file argument is not given, the default location
   of the file containing the private key is assumed:

     -- The location pointed to by X509_USER_KEY
     -- If X509_USER_KEY not set, /home/ibmgrid/.globus/userkey.pem

   Options:
      -help, -usage    Displays usage
      -version         Displays version
      -file location   Change passphrase on key stored in the file at
                       the non-standard location 'location'.
```

Example

Example B-28 provides an example of this command when the change is successful.

Example: B-28 grid-change-pass-phrase command

```
[ibmgrid@y2 gars]$ grid-change-pass-phrase
read RSA key
Enter PEM pass phrase:
writing RSA key
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
[ibmgrid@y2 gars]$
```

Example B-29 provides an example of this command when the change is unsuccessful.

Example: B-29 grid-change-pass-phrase command

```
[ibmgrid@y2 gars]$ grid-change-pass-phrase
read RSA key
Enter PEM pass phrase:
unable to load key
14873:error:06065064:digital envelope routines:EVP_DecryptFinal:bad
decrypt:evp_enc.c:277:
14873:error:0906A065:PEM routines:PEM_do_header:bad decrypt:pem_lib.c:455:
Failed to change passphrase
[ibmgrid@y2 gars]$
```

Associated files

- /home/ibmgrid/.globus/userkey.pem

grid-mapfile-add-entry

The **grid-mapfile-add-entry** command adds an entry to the /etc/grid-security/grid-mapfile file.

Note: This command should be executed as *root*.

Synopsis

```
grid-mapfile-add-entry -dn DN -ln LN
[-help] [-d] [-f mapfile FILE]

grid-mapfile-add-entry adds an entry to a Grid mapfile. The mapfile should be
owned by user running this command, otherwise it has to be run like root.

Options:
      -help, -usage          Displays help
      -version               Displays version
      -dn DN                 Distinguished Name (DN) to add. Remember to quote
theDN if it contains spaces.
      -ln LN1 [LN2...]       Local login name(s) to map DN to
      -dryrun, -d            Shows what would be done but will not add the
entry
      -mapfile FILE, -f FILE  Path of Grid map file to be used
```

Examples

Example B-30 provides an example of this command when executed by a non-root user.

Example: B-30 grid-mapfile-add-entry command

```
[ibmgrid@y2 ibmgrid]$ grid-mapfile-add-entry -dn
/C=SP/ST=Madrid/O=ITSOLAB/CN=usery2 -ln ibmgrid

Error, argument  : "/etc/grid-security/grid-mapfile" is not writeable.

Syntax : grid-mapfile-add-entry -dn DN -ln LN
[-help] [-d] [-f mapfile FILE]

Use -help to display full usage.

[ibmgrid@y2 ibmgrid]$
```

Example B-31 on page 183 provides an example of this command when executed as root.

Example: B-31 grid-mapfile-add-entry command

```
[root@y2 ibmgrid]# grid-mapfile-add-entry -dn
/C=SP/ST/Madrid/O=ITSOLAB/CN=usery2 -ln ibmgrid -mapfile
/etc/grid-security/grid-mapfile
(1) entry added
[root@y2 ibmgrid]#
```

After executing the command, the file /opt/grid-security/grid-mapfile should appear similar to Example B-32.

Example: B-32 Sample grid-mapfile file

```
[root@y2 grid-security]# more /etc/grid-security/grid-mapfile
"/C=SP/ST/Madrid/O=ITSOLAB/CN=usery2" ibmgrid
[root@y2 grid-security]#
```

Associated files

- /etc/grid-security/grid-mapfile

grid-mapfile-check-consistency

The **grid-mapfile-check-consistency** command checks a grid mapfile for legal logon names, duplicate entries, and for C=, O=, and CN= fields. The command provides output only when there are errors in the grid mapfile.

Synopsis

```
grid-mapfile-check-consistency [-help] [-mapfile FILE]
grid-mapfile-check-consistency checks the consistency of the Grid mapfile.
Options:
      -help, -usage           Displays help
      -version                Displays version
      -mapfile FILE, -f FILE  Path of gridmap to be used
```

Examples

Example B-33 provides an example of this command when there are no errors.

Example: B-33 grid-mapfile-check-consistency command

```
[ibmgrid@y2 ibmgrid]$ grid-mapfile-check-consistency
[ibmgrid@y2 ibmgrid]$
```

Example B-34 on page 184 provides an example of this command when there are errors.

Example: B-34 grid-mapfile-check-consistency command

```
[ibmgrid@y2 ibmgrid]$ grid-mapfile-check-consistency
The following entry contains an invalid login name
"/C=SP/ST=Texas/O=ITSOLAB/CN=usery15" ibmgrod
The login name not found is: ibmgrod
[ibmgrid@y2 ibmgrid]$
```

Associated files

- /etc/grid-security/grid/mapfile

grid-mapfile-delete-entry

The **grid-mapfile-delete-entry** command removes an entry from the /etc/grid-security/grid-mapfile file.

Note: This command should be executed as root.

Synopsis

```
grid-mapfile-delete-entry [-help] [-dn <DN>] [-ln <local name>] [-d] [-f file]

    grid-mapfile-delete-entry deletes one or more matching entries from the
Grid mapfile.

Options:
      -help, -usage          Displays help
      -version               Displays version
      -dn <DN>               Distinguished Name (DN) to delete
      -ln <local name>       Local Login Name (LN) to delete
      -dryrun, -d            Shows what would be done but will not
                             delete the entry
      -mapfile file, -f file  Path of gridmap file to be used
```

Example

Example B-35 provides an example of this command when the entry exists.

Example: B-35 grid-mapfile-delete-entry command

```
[root@y2 ibmgrid]# grid-mapfile-delete-entry -dn
/C=SP/ST/Madrid/O=ITSOLAB/CN=usery2 -ln ibmgrid -mapfile
/etc/grid-security/grid-mapfile
(1) entry deleted
```

Example B-36 on page 185 provides an example of this command when the entry does not exist.

Example: B-36 grid-mapfile-delete-entry command

```
[root@y2 ibmgrid]# grid-mapfile-delete-entry -dn
/C=SP/ST/Madrid/O=ITSOLAB/CN=usery2 -ln ibmgrid -mapfile
/etc/grid-security/grid-mapfile
ERROR: No such entry exists
```

Associated files

- /etc/grid-security/grid-mapfile

grid-proxy-destroy

The **grid-proxy-destroy** command destroys the current grid proxy.

Synopsis

```
Syntax: grid-proxy-destroy [-help][-dryrun][-default][-all][--] [file1...]

    Options
    -help, -usage           Displays usage
    -version                Displays version
    -debug                  Display debugging information
    -dryrun                 Prints what files would have been destroyed
    -default                Destroys file at default proxy location
    -all                    Destroys any user (default) and delegated
                            proxies that are found
    --                      End processing of options
    file1 file2 ...         Destroys files listed
```

Example

Example B-37 provides an example of this command when using the **-dryrun** flag.

Example: B-37 grid-proxy-destroy command

```
[ibmgrid@y1 ibmgrid]$ grid-proxy-destroy -dryrun
Would remove /tmp/x509up_u500
```

Example B-38 provides an example of this command when actually destroying a proxy.

Example: B-38 grid-proxy-destroy command

```
[ibmgrid@y1 ibmgrid]$ grid-proxy-destroy /tmp/x509up_u500
```

```
[ibmgrid@y1 ibmgrid]$
```

Associated files

- /tmp/x509up_u500

grid-proxy-info

The **grid-proxy-info** command provides information about the current grid proxy.

Synopsis

```
Syntax: grid-proxy-info [-help][-f proxyfile][-subject][...][-e [-h H][-b B]]

    Options
    -help, -usage             Displays usage
    -version                  Displays version
    -debug                    Displays debugging output
    -file <proxyfile>  (-f)   Non-standard location of proxy
    [printoptions]            Prints information about proxy
    -exists [options]  (-e)   Returns 0 if valid proxy exists, 1 otherwise

    [printoptions]
        -subject       (-s)   Distinguished name (DN) of subject
        -issuer        (-i)   DN of issuer (certificate signer)
        -identity             DN of the identity represented by the proxy
        -type                 Type of proxy (full or limited)
        -timeleft             Time (in seconds) until proxy expires
        -strength             Key size (in bits)
        -all                  All above options in a human readable format
        -text                 All of the certificate
        -path                 Pathname of proxy file

    [options to -exists]      (if none are given, H = B = 0 are assumed)
        -valid H:M     (-v)   time requirement for proxy to be valid
        -hours H       (-h)   time requirement for proxy to be valid
                              (deprecated, use -valid instead)
        -bits  B       (-b)   strength requirement for proxy to be valid
```

Example

Example B-39 provides an example of this command.

Example: B-39 grid-proxy-info command

```
[ibmgrid@y1 ibmgrid]$ grid-proxy-info
subject  : /C=US/ST=Texas/O=ITSOLAB/CN=userz2/CN=1621841649
issuer   : /C=US/ST=Texas/O=ITSOLAB/CN=userz2
```

```
identity : /C=US/ST=Texas/O=ITSOLAB/CN=userz2
type     : Proxy draft compliant impersonation proxy
strength : 512 bits
path     : /tmp/x509up_u500
timeleft : 11:29:00
```

Associated files

- /tmp/x509up_u500

grid-proxy-init

The **grid-proxy-init** command creates a new grid proxy.

Note: By default a grid proxy lasts for 12 hours.

Synopsis

```
Syntax: grid-proxy-init-bin [-help][-pwstdin][-limited][-valid H:M] ...
Option -help will display usage.
```

Example

Example B-40 provides an example of this command.

Example: B-40 grid-proxy-init command

```
[ibmgrid@y1 bin]$ grid-proxy-init
Your identity: /C=US/ST=Texas/O=ITSOLAB/CN=userz2
Enter GRID pass phrase for this identity:
Creating proxy.....................................................Done
Your proxy is valid until: Thu Mar 18 03:50:29
2004
```

Associated files:

- /tmp/x509up_u500

managed-job-globusrun

The **managed-job-globusrun** command provides the ability to submit jobs to globus resources.

Note: The existence of a valid proxy is required.

Synopsis

```
ARGUMENTS:
         [options] [<factory>] <RSL>
         -p <RSL>
         -list [<factory>]
         (-status | -kill) <job service URL>
         -help | -usage | -version

with
     <RSL>     = -file <RSL filename> | <RSL string>
     <factory> = -factory <contact> [-type <type>]
     <contact> = [<protocol>://]<host>[:[port]][/<service>]
     [options] = [-s] [-w] [-o] [-q] [-n]
                 [-b] [-duration] [-terminate-at]
                 [-auth <auth>] [-xmlsec <sec>] [-nogrim] [-personal]


DESCRIPTION:
This command is used to submit jobs to globus resources. The job startup is
done using the GRAM services. Also, the GASS service can be used to provide
access to remote files and for redirecting standard output streams.
In addition to starting jobs, it is possible to list previously started jobs,
query status of previously started jobs, parse RSL request strings and/or
files.
The existence of a valid proxy is required for essentially all supported
operations but RSL parsing (-p).
OPTIONS:
Help:
 -help                 display help.
 -usage                display usage.
 -v, -version          display version.
Job Factory Contact:
 -factory <contact>    specify the URL of the Job Factory Service
                       to contact when submitting or listing jobs.
                       A factory contact string can be specified in
                       the following ways:
                       host
                       host:
                       host:port
                       host:port/service
                       host/service
                       host:/service
                       It is also possible to specify the protocol
                       by prepending   protocol://  to each of the
                       previous possibilities, bringing the total
                       number of supported syntaxes to 12.
                       For those factory contacts which omit the
                       protocol, port or service field, properties
```

```
                        in the file ogsa.properties are used as
                        defaults, but if the property file cannot be
                        found or read, the following default values
                        are used, as the following table explains:
                        URL part | $property        | default value
                        port     | $service.port    | 8080
                        protocol | $binding.protocol| http
                        service  | none             | ogsa/services/base/gram/
                                 MasterForkManagedJobFactoryService

                        Omitting altogether the -factory option is
                        equivalent to specifying the local host as
                        the contact string.
 -type <factory type>   specify the job factory service as
                        a shortname instead of specifying a full
                        service path with -factory or using the
                        default service path. This is equivalent to
                        specifying the service with the -factory
                        option as:
                        ogsa/services/base/gram/
                                Master<factory type>ManagedJobFactoryService
                        Examples:   -factory myHost -type Fork
                                    -factory myHost -type Pbs
                        Default: Fork

Job Specification:
 <RSL string>           read RSL from the string <RSL string>.
 -file <RSL filename>   read RSL from the local file <RSL filename>.
                        The RSL must be a single job request.
 -p                     only parse the RSL, and then print either a
                        success message or a parser failure. No job
                        will be submitted to any factory service.
                        The RSL must be a single job request.

Internal GASS Server:
 -s, -server            start GASS server with read access to local
                        files, providing read-only service to the
                        local filesystem.
 -w, -write-allow       start GASS server with read/write access to
                        local files. Similar to -server, except the
                        GASS server URL will allow writing to the
                        local filesystem as well as reading to it.
                        Implies -server.
 -o, -output            start GASS server and display the job's
                        standard output and error streams on the
                        standard output and error of the command's
                        process. No other read/write access will be
                        provided by this option alone.
                        Implies -quiet.
```

```
 The substitution variable GLOBUSRUN_GASS_URL can be used in RSL
 to access files local to the submission machine via GASS.

Batch Operations:
 -b, -batch            do not wait for started job to complete (and
                       do not destroy started job service on exit).
                       The handle of the job service will be
                       printed on the standard output.
                       Incompatible with internal GASS options
                       (-server, -write-allow, and -output).
                       The job must use an external GASS server if
                       it needs to access local files.
                       incompatible with multi-request jobs.
                       Implies -quiet.
 -l, -list             list previously started and not destroyed
                       job services. The output of this command
                       consists of the job service URLs, and the
                       job RSL string. Requires the -factory <URL>
                       argument.
 -status <URL>         printout the status of the specified job.
                       For a list of valid states, see the GRAM
                       documentation; the current valid states are
                       Pending, Active, Done, Suspended, and Failed.
                       The <URL> argument should be one printed out
                       when executing in batch mode or when using
                       the -list option.
 -k, -kill <URL>       kill the specified job.
                       The <URL> argument should be one printed out
                       when executing in batch mode or when using
                       the -list option.

Job Service Termination Time:
 -duration <duration>  specify duration of job service. The job
                       service will destroy itself automatically
                       after the specified duration starting from
                       service creation.
                       Format: HH:mm
                       Default job service duration is 24 hours.
                       Incompatible with -date-time.
                       Useful with -batch.
 -terminate-at <date>  specify termination date/time of service.
                       Same as -duration but with an absolute
                       date/time value.
                       Format: MM/dd/yyyy HH:mm
                       The date expression may need to be quoted,
                       as in:     -terminate-at '08/15/2005 11:30'
                       Incompatible with -duration.
                       Useful with -batch.
```

```
Security:
 -auth <auth>          set authorization type. <auth> can be:
                           'host' for host authorization,
                           'self' for self authorization (default).
                       Otherwise identity authorization is
                       performed.
 -xmlsec <sec>         set xml security type to use. <sec> can be:
                           'sig' for XML Signature (default),
                           'enc' for XML Encryption.
 -nogrim               disable grim checks (enabled by default).
 -personal             shortcut for -nogrim and -auth self.

Miscellaneous:
 -q, -quiet            set quiet mode on (do not print diagnostic
                       messages when job status changes, in
                       non-batch mode). Useful when job output is
                       redirected to the local process and parsed.
 -n, -no-interrupt     disable interrupt handling. By default,
                       interrupt signals (typically generated by
                       Ctrl + C) cause the program to terminate the
                       currently submitted job. This flag disables
                       that behavior.
 -timeout <integer>    set timeout for HTTP socket, in milliseconds.
                       Applies to job submission only.
                       Default is 120000.

GT2 globusrun options not functional (yet):
 -dryrun               NOT IMPLEMENTED ON SERVER SIDE YET.
                       augment the RSL in order to mark this job as
                       a dry run, if the RSL does not already say
                       so. This causes the job manager to stop
                       short of starting the job, but still detect
                       other RSL errors (such as bad directory,
                       bad executable, etc). An error message will
                       be displayed if the dry run fails.
                       Otherwise, a message will be displayed
                       indicating that the dryrun was successful.
 -authenticate-only    NOT IMPLEMENTED ON SERVER SIDE YET.
 -interactive          DUROC not supported yet.
 -stop-manager         doesn't apply in GT3 (yet).
```

ogsi-add-service

The **ogsi-add-service** command takes a service instance and adds it to a Service Group.

Synopsis

```
Usage:  AddService [options] <member handle> <service group handle> <timeout>
[content file]

Where [options] are:
 -gsiSecConv <type>
        Enables GSI Secure Conversation. <type> one of:
         'sig' - for XML Signature
         'enc' - for XML Encryption
 -gsiSecConvActor <actor>
        Sets actor name for GSI Secure Conversation
 -gsiXmlSig
        Enables GSI XML Signature
        (can be used together with -gssxml)
 -gsiXmlSigActor <actor>
        Sets actor name for GSI XML Signature
 -deleg <mode>
        Performs delegation. <mode> one of:
         'limited' - performs limited delegation
         'full'    - performs full delegation
 -auth <type>
        Performs authorization type. If <type> is:
         'host' - performs host authorization
         'self' - performs self authorization
         'none' - disables authorization
        Otherwise, identity authorization is performed
        with <type> identity.
 -grimPolicyHandler <policyClass>
        Sets GRIM policy handler
 -debug
        Enables debug mode
```

Example

Example B-41 provides an example of this command when adding a service instance to a service group.

Example: B-41 ogsi-add-service command

```
[ibmgrid@y1 /]$ ogsi-add-service
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService/hash-190937948-1079725562128
http://192.168.0.21:12080/ogsa/services/base/xr/servicegroup/ServiceGroupFactor
y/hash-1070857053-1079725516000 1000
```

```
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties

Entry Handle returned =
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService/hash-190937948-1079725562128
```

ogsi-create-service

The **ogsi-create-service** command creates an instance of a grid service from a service factory.

Synopsis

```
Usage: CreateService [options] <factory handle> [id]
Where [options] are:
 -gsiSecConv <type>
        Enables GSI Secure Conversation. <type> one of:
         'sig' - for XML Signature
         'enc' - for XML Encryption
 -gsiSecConvActor <actor>
        Sets actor name for GSI Secure Conversation
 -gsiXmlSig
        Enables GSI XML Signature
        (can be used together with -gssxml)
 -gsiXmlSigActor <actor>
        Sets actor name for GSI XML Signature
 -deleg <mode>
        Performs delegation. <mode> one of:
         'limited' - performs limited delegation
         'full'    - performs full delegation
 -auth <type>
        Performs authorization type. If <type> is:
         'host' - performs host authorization
         'self' - performs self authorization
         'none' - disables authorization
        Otherwise, identity authorization is performed
        with <type> identity.
 -grimPolicyHandler <policyClass>
        Sets GRIM policy handler
 -debug
        Enables debug mode
```

Example

Example B-42 provides an example of this command.

Example: B-42 ogsi-create-service command

```
[ibmgrid@y1 ibmgrid]$ ogsi-create-service
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService

OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
Service successfully created:
  Handle:
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService/hash-1009046413-1079724052316
  Termination Time: infinity
```

ogsi-destroy-service

The **ogsi-destroy-service** command takes an existing instance of a grid service and destroy it.

Synopsis

```
Usage: DestroyService [options] <handle>

Where [options] are:
 -gsiSecConv <type>
        Enables GSI Secure Conversation. <type> one of:
         'sig' - for XML Signature
         'enc' - for XML Encryption
 -gsiSecConvActor <actor>
        Sets actor name for GSI Secure Conversation
 -gsiXmlSig
        Enables GSI XML Signature
        (can be used together with -gssxml)
 -gsiXmlSigActor <actor>
        Sets actor name for GSI XML Signature
 -deleg <mode>
        Performs delegation. <mode> one of:
         'limited' - performs limited delegation
         'full'    - performs full delegation
 -auth <type>
        Performs authorization type. If <type> is:
```

```
        'host' - performs host authorization
        'self' - performs self authorization
        'none' - disables authorization
       Otherwise, identity authorization is performed
       with <type> identity.
 -grimPolicyHandler <policyClass>
       Sets GRIM policy handler
 -debug
       Enables debug mode
```

Example

Example B-43 provides an example of this command.

Example: B-43 ogsi-destroy-service command

```
[ibmgrid@y1 ibmgrid]$ ogsi-destroy-service
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService/hash-1009046413-1079724052316

OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
Destroyed service:
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService/hash-1009046413-1079724052316
```

ogsi-find-service-data-by-xpath

The **ogsi-find-service-data-by-xpath** command returns the service data of a grid service with the desired XPath.

Synopsis

```
Usage: FindServiceDataByXPath [options] <service data name namespace> <service
data name> <handle> <XPath expression> <namespace-mapping-1;
namespace-mapping-2; etc... >

Where [options] are:
 -gsiSecConv <type>
       Enables GSI Secure Conversation. <type> one of:
        'sig' - for XML Signature
        'enc' - for XML Encryption
 -gsiSecConvActor <actor>
       Sets actor name for GSI Secure Conversation
 -gsiXmlSig
```

```
       Enables GSI XML Signature
       (can be used together with -gssxml)
-gsiXmlSigActor <actor>
       Sets actor name for GSI XML Signature
-deleg <mode>
       Performs delegation. <mode> one of:
        'limited' - performs limited delegation
        'full'    - performs full delegation
-auth <type>
       Performs authorization type. If <type> is:
        'host' - performs host authorization
        'self' - performs self authorization
        'none' - disables authorization
       Otherwise, identity authorization is performed
       with <type> identity.
-grimPolicyHandler <policyClass>
       Sets GRIM policy handler
-debug
       Enables debug mode
```

ogsi-get-gwsdl-port-types

The **ogsi-get-gwsdl-port-types** command parses the WSDL for a grid service to display its service data and port types.

Synopsis

```
Usage:ogsi-get-gswdl-port-types <gsr url>
Parses WSDL file and prints out GWSDL port types and
the contained service data.
```

ogsi-notification-sink

The **ogsi-notification-sink** command starts a notification sink, logging the handle of the sink to be used to send messages. Messages are logged to STDOUT.

Synopsis

```
Starts up a stand-alone notification sink and logs the handle
of the sink that can be used to send it messages. Messages are
logged to stdout.
```

ogsi-notification-sink-notifyer

The **ogsi-notification sink-notifyer** command sends a test message to a notification sink.

Synopsis

```
NotificationSinkNotifyer <handle><message>
Sends a test message to notification sink.
```

ogsi-notification-topic-listener

The **ogsi-notification-topic-listener** command starts up a notification sink and subscribes it to a service data element in a service.

```
Synopsis
NotificationTopicListener <topic><source>
Starts up a sink and subscribes it to a service data element (topic)in a source
service.
Messages are logged to stdout.
```

ogsi-remove-service

The **ogsi-remove-service** command takes a grid service instance and removes it from a given service group.

Synopsis

```
Usage: RemoveService [options] <member handle> <service group handle>

Where [options] are:
 -gsiSecConv <type>
        Enables GSI Secure Conversation. <type> one of:
         'sig' - for XML Signature
         'enc' - for XML Encryption
 -gsiSecConvActor <actor>
        Sets actor name for GSI Secure Conversation
 -gsiXmlSig
        Enables GSI XML Signature
        (can be used together with -gssxml)
 -gsiXmlSigActor <actor>
        Sets actor name for GSI XML Signature
 -deleg <mode>
        Performs delegation. <mode> one of:
         'limited' - performs limited delegation
         'full'    - performs full delegation
 -auth <type>
        Performs authorization type. If <type> is:
         'host' - performs host authorization
```

```
        'self' - performs self authorization
        'none' - disables authorization
       Otherwise, identity authorization is performed
       with <type> identity.
 -grimPolicyHandler <policyClass>
       Sets GRIM policy handler
 -debug
       Enables debug mode
```

Example

Example B-44 provides an example of this command.

Example: B-44 ogsi-remove-service command

```
[ibmgrid@y1 /]$ ogsi-remove-service
http://192.168.0.21:12080/ogsa/services/services/com/ibm/grid/ogsa/samples/coun
ters/basic/BasicCounterProviderFactoryService/hash-190937948-1079725562128
http://192.168.0.21:12080/ogsa/services/base/xr/servicegroup/ServiceGroupFactor
y/hash-1070857053-1079725516000

OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties

OGSALogFactory Looking for file :
/opt/IBMGrid/AppServer/installedApps/DefaultNode/IBMGrid.ear/ogsa.war/WEB-INF/o
gsilogging.properties
```

ogsi-request-termination

The **ogsi-request-termination** command sends a request to terminate a service.

Synopsis

```
Usage: RequestTermination [options] <seconds|infinity> <handle>

Where [options] are:
 -gsiSecConv <type>
       Enables GSI Secure Conversation. <type> one of:
        'sig' - for XML Signature
        'enc' - for XML Encryption
 -gsiSecConvActor <actor>
       Sets actor name for GSI Secure Conversation
 -gsiXmlSig
       Enables GSI XML Signature
       (can be used together with -gssxml)
```

```
-gsiXmlSigActor <actor>
       Sets actor name for GSI XML Signature
-deleg <mode>
       Performs delegation. <mode> one of:
        'limited' - performs limited delegation
        'full'    - performs full delegation
-auth <type>
       Performs authorization type. If <type> is:
        'host' - performs host authorization
        'self' - performs self authorization
        'none' - disables authorization
       Otherwise, identity authorization is performed
       with <type> identity.
-grimPolicyHandler <policyClass>
       Sets GRIM policy handler
-debug
       Enables debug mode
```

ogsi-resolve-handle

The **ogsi-notification-topic-listener** command resolves a grid service handle (GSH) to a grid service reference (GSR).

Synopsis

```
Usage:ogsi-resolve-handle <handle>
Resolves a Grid Service Handle (GSH) to a Grid Service Reference (GSR).
```

ogsi-set-service-data-by-name

The **ogsi-ser-service-data-by-name** command adds service data element values to a service.

Synopsis

```
Usage: SetServiceDataByName [options] <service data name file> <handle>

Where [options] are:
 -gsiSecConv <type>
       Enables GSI Secure Conversation. <type> one of:
        'sig' - for XML Signature
        'enc' - for XML Encryption
 -gsiSecConvActor <actor>
       Sets actor name for GSI Secure Conversation
 -gsiXmlSig
       Enables GSI XML Signature
       (can be used together with -gssxml)
 -gsiXmlSigActor <actor>
```

```
       Sets actor name for GSI XML Signature
-deleg <mode>
       Performs delegation. <mode> one of:
        'limited' - performs limited delegation
        'full'    - performs full delegation
-auth <type>
       Performs authorization type. If <type> is:
        'host' - performs host authorization
        'self' - performs self authorization
        'none' - disables authorization
       Otherwise, identity authorization is performed
       with <type> identity.
-grimPolicyHandler <policyClass>
       Sets GRIM policy handler
-debug
       Enables debug mode
```

Other Globus commands

Additional commands that could be of interest include:

- globus-gass-cache
- globus-gass-cache-destroy
- globus-gass-server
- globus-gass-server-shutdown
- globus-grim
- globus-makefile-header
- globus-jms-adapter-client
- globus-personal-gatekeeper
- globus-sdb
- globus-service-browser
- globus-sh-exec
- globus-start-container
- globus-stop-container

For more information about these commands visit:

http://www.globus.org/

C

Script the installation

This appendix discusses some of the possibilities for scripting segments of the IBM Grid Toolbox installation. This appendix covers:

- Basics for scripting
- Scripting the IBM Grid Toolbox installation
- Scripting the Apache Ant installation
- Scripting the GridFTP installation
- Scripting the installation of additional files

Basics for scripting

Certain aspects of the complete IBM Grid Toolbox installation can be scripted. The advantage to creating scripts to automate an installation comes when you need to install large numbers of instances and would prefer to do so without operator intervention.

Attention: The scripts are provided as-is. IBM has no obligation to provide any error correction or enhancements. Further, IBM makes no representations or warranties, expressed or implied, including the implied warranties of merchantability and fitness for a particular purpose with respect to the scripts or their use, nor shall IBM have any liability in respect to any infringement of any intelectual property rights of third parties due to customer operation under the licenses or rights herein granted.

Important: The scripts in this appendix provide no verification that installations were really successful. To verify the installations, check the installation directories and make sure that the files were installed.

Scripting the IBM Grid Toolbox installation

In order to script the installation of the actual IBM Grid Toolbox product, we recommend that you use the silent installation method along with a response file. For more information about the silent installation method, refer to 3.3.3, “Silent installation method” on page 46. For information about response files, refer to Appendix D, “Response file” on page 215.

The script in Example C-1 was written in Perl and takes in a single command line argument, which is the address of the target NFS server and the desired directory on that server. The command is executed using the following syntax:

```
perl install-igt.pl <address of NFS server>:/<directory>
```

Example C-1 provides the code used in the lab environment.

Example: C-1 install-igt.pl code

```
#!/usr/bin/perl

#Title : IBM Grid Toolbox installation script
#Author: Lee B Wilson
#Email : wilsol@us.ibm.com
#Date  : March 13, 2004
```

```
#Purpose: Automate elements of the IBM Grid Toolbox Installation.
#       After running the script you will still need to set up
#       security and take any necessary post install actions.


#Read in the command line argument providing the location of the NFS server.
#If no location is provided exit and give error.

$nfsServer = $ARGV[0];

#If a NFS server was not supplied terminate script.
if ($nfsServer eq "") {
   print ("NFS server location missing\nSyntax : perl install-igt.pl <NFS
server address>:/<directory>\n");
   exit(0);
}

system ("clear");

print ("---------------------------------------\n");
print ("ITSO Lab IGT setup script\n");
print ("---------------------------------------\n");
print ("\n");
print ("This script will install the IBM Grid Toolbox on\n");
print ("this host using the $nfsServer source.\n");
print ("\n");
print ("To begin the installation hit ENTER.\n\n");

#Temporary variable to catch the ENTER key.
$temp = <STDIN>;

print ("Setting up installation:\n\n");

#Make the /etc/grid-security directory

print ("Creating /etc/grid-security directory\t\t\t\t");
system ("mkdir /etc/grid-security");
print ("DONE\n");

#Mount the infrastructure machine.

print ("Creating mounting point /mnt/infrastructure\t\t\t");
system ("mkdir /mnt/infrastructure");
print ("DONE\n");

print ("Mounting infrastructure on $nfsServer\t\t");
system ("mount $nfsServer /mnt/infrastructure");
print ("DONE\n");
```

```
#Copy installation files

print ("Copying IBM Grid Toolbox installation files\t\t\t");
system ("cp /mnt/infrastructure/IGT10/setuplx.bin /tmp/");
system ("cp /mnt/infrastructure/response-file.txt /tmp/");
print ("DONE\n");

print ("\nInstalling IBM Grid Toolbox:\n\n");

#Begin a silent installation using response file

print ("Installing IBM Grid Toolbox (This may take some time)\t\t");
system ("/tmp/setuplx.bin -options /tmp/response-file.txt -silent");
print ("DONE\n");

#Copy /mnt/infrastructure/igt-setenv.sh to profile.d

print ("Copying igt-setenv.sh to profile.d\t\t\t\t");
system ("cp /opt/IBMGrid/igt-setenv.sh /etc/profile.d/");
print ("DONE\n");

print ("\nCleaning up:\n\n");

print ("Unounting NFS server\t\t\t\t\t");
system ("umount /mnt/infrastructure");
print ("DONE\n");

print ("Removing mounting point /mnt/infrastructure\t\t\t");
system ("rmdir /mnt/infrastructure");
print ("DONE\n");

print ("Removing IBM Grid Toolbox installation files\t\t\t");
system ("rm /tmp/setuplx.bin");
system ("rm /tmp/response-file.txt ");
print ("DONE\n");

print ("\nInstallation of IBM Grid Toolbox is COMPLETE.\n\n");

print ("REMEMBER: Before continuing you must set up security for this IBM Grid
Toolbox instance.\n");
```

Example C-2 illustrates the way that the output of this script should look.

Example: C-2 install-igt.pl output

```
[root@y2 /]perl /tmp/install-igt.pl 192.168.0.111:/gr1105

---------------------------------------
ITSO Lab IGT setup script
```

```
----------------------------------------

This script will install the IBM Grid Toolbox on
this host using the 192.168.0.111:/gr1105 source.

To begin the installation hit ENTER.


Setting up installation:

Creating /etc/grid-security directory                          DONE
Creating mounting point /mnt/infrastructure                    DONE
Mounting infrastructure on 192.168.0.111:/gr1105               DONE
Copying IBM Grid Toolbox installation files                    DONE

Installing IBM Grid Toolbox:

Installing IBM Grid Toolbox (This may take some time)          DONE
Copying igt-setenv.sh to profile.d                             DONE

Cleaning up:

Unounting NFS server                                           DONE
Removing mounting point /mnt/infrastructure                    DONE
Removing IBM Grid Toolbox installation files                   DONE

Installation of IBM Grid Toolbox is COMPLETE.

REMEMBER: Before continuing you must set up security for this IBM Grid Toolbox
instance.
```

After the IBM Grid Toolbox is installed, you must perform all security and configuration tasks that are necessary to completing the installation.

Scripting the Apache Ant installation

The script in Example C-3 on page 206 shows one possibility when scripting the installation of Apache Ant. The script was written in Perl and takes in a single command line argument, which is the address of the target NFS server and the desired directory on that server. The command is executed using this syntax:

```
perl install-ant.pl <address of NFS server>:/<directory>
```

Example C-3 on page 206 provides the code used in the lab environment.

Example: C-3 install-ant.pl code

```
#!/usr/bin/perl

#Title : Apache Ant installation script
#Author: Lee B Wilson
#Email : wilsol@us.ibm.com
#Date  : March 13, 2004

#Purpose: Automate Apache Ant installation.


#Read in the command line argument providing the location of the NFS server.
#If no location is provided exit and give error.

$nfsServer = $ARGV[0];

#If a NFS server was not supplied terminate script.
if ($nfsServer eq "") {
   print ("NFS server location missing\nSyntax : perl install-igt.pl <NFS
server address>:/<directory>\n");
   exit(0);
}

system ("clear");

print ("----------------------------------------\n");
print ("ITSO Lab IGT setup script\n");
print ("----------------------------------------\n");
print ("\n");
print ("This script will install Apache Ant on\n");
print ("this host using the $nfsServer source.\n");
print ("\n");
print ("To begin the installation hit ENTER.\n\n");

#Temporary variable to catch the ENTER key.
$temp = <STDIN>;

print ("Setting up installation:\n\n");

#Mount the infrastructure machine.

print ("Creating mounting point /mnt/infrastructure\t\t\t");
system ("mkdir /mnt/infrastructure");
print ("DONE\n");

print ("Mounting infrastructure on $nfsServer\t\t");
system ("mount $nfsServer /mnt/infrastructure");
print ("DONE\n");
```

```
#Copy installation files

print ("Copying Apache Ant installation files\t\t\t\t");
system ("cp /mnt/infrastructure/IGT10/apache-ant-1.6.1-bin.tar.gz /tmp/");
print ("DONE\n");

print ("\nInstalling Apache Ant:\n\n");

#Install Apache Ant in /opt/apache-ant

print ("Installing Apache Ant\t\t\t\t\t\t");
system ("mkdir /opt/apache-ant");
system ("tar -zxf /tmp/apache-ant-1.6.1-bin.tar.gz -C /opt/apache-ant");
print ("DONE\n");

print ("\nCleaning up:\n\n");
print ("Unounting NFS server\t\t\t\t\t\t");
system ("umount /mnt/infrastructure");
print ("DONE\n");

print ("Removing mounting point /mnt/infrastructure\t\t\t");
system ("rmdir /mnt/infrastructure");
print ("DONE\n");

print ("Removing Apache Ant installation files\t\t\t\t");
system ("rm /tmp/apache-ant-1.6.1-bin.tar.gz");
print ("DONE\n");

print ("\nInstallation of Apache Ant is COMPLETE.\n");
```

Example C-4 illustrates the way the output of this script should look.

Example: C-4 install-ant.pl output

```
[root@y2 /]perl /tmp/install-ant.pl 192.168.0.111:/gr1105

----------------------------------------
ITSO Lab IGT setup script
----------------------------------------

This script will install Apache Ant on
this host using the 192.168.0.111:/gr1105 source.

To begin the installation hit ENTER.


Setting up installation:
```

```
Creating mounting point /mnt/infrastructure                    DONE
Mounting infrastructure on 192.168.0.111:/gr1105               DONE
Copying Apache Ant installation files                          DONE

Installing Apache Ant:

Installing Apache Ant                                          DONE

Cleaning up:

Unounting NFS server                                           DONE
Removing mounting point /mnt/infrastructure                    DONE
Removing Apache Ant installation files                         DONE

Installation of Apache Ant is COMPLETE.
```

Scripting the GridFTP installation

The following script shows one possibility when scripting the installation of GridFTP. It is also possible to modify the installation script to make the necessary modifications to inetd or xinetd in order to create GridFTP servers dynamically when the host receives a request.

The script was written in Perl and takes in a single command line argument, which is the address of the target NFS server and the desired directory on that server. The command is executed using the following syntax:

```
perl install-gridftp.pl <address of NFS server>:/<directory>
```

Example C-5 provides the code used in the lab environment.

Example: C-5 install-gridftp.pl code

```
#!/usr/bin/perl

#Title : GridFTP Installation script
#Author: Lee B Wilson
#Email : wilsol@us.ibm.com
#Date  : March 13, 2004

#Purpose: Automate GridFTP installation. This script does not modify inetd or
xinetd.


#Read in the command line argument providing the location of the NFS server.
#If no location is provided exit and give error.
```

```
$nfsServer = $ARGV[0];

#If a NFS server was not supplied terminate script.
if ($nfsServer eq "") {
   print ("NFS server location missing\nSyntax : perl install-igt.pl <NFS
server address>:/<directory>\n");
   exit(0);
}

system ("clear");

print ("----------------------------------------\n");
print ("ITSO Lab IGT setup script\n");
print ("----------------------------------------\n");
print ("\n");
print ("This script will install GridFTP on\n");
print ("this host using the $nfsServer source.\n");
print ("\n");
print ("To begin the installation hit ENTER.\n\n");

#Temporary variable to catch the ENTER key.
$temp = <STDIN>;

print ("Setting up installation:\n\n");

#Mount the infrastructure machine.

print ("Creating mounting point /mnt/infrastructure\t\t\t");
system ("mkdir /mnt/infrastructure");
print ("DONE\n");

print ("Mounting infrastructure on $nfsServer\t\t");
system ("mount $nfsServer /mnt/infrastructure");
print ("DONE\n");

#Copy installation files

print ("Copying GridFTP installation files\t\t\t\t");
system ("cp /mnt/infrastructure/IGT10/gridftp-1.10-xl.tar.gz /tmp/");
print ("DONE\n");

print ("\nInstalling GridFTP:\n\n");

#Install GridFTP in /opt/gridftp

print ("Installing GridFTP\t\t\t\t\t\t");
system ("mkdir /opt/gridftp");
system ("tar -zxf /tmp/gridftp-1.10-xl.tar.gz -C /opt/gridftp");
print ("DONE\n");
```

```
print ("\nCleaning up:\n\n");
print ("Unounting NFS server\t\t\t\t\t\t");
system ("umount /mnt/infrastructure");
print ("DONE\n");

print ("Removing mounting point /mnt/infrastructure\t\t\t");
system ("rmdir /mnt/infrastructure");
print ("DONE\n");

print ("Removing GridFTP installation files\t\t\t\t");
system ("rm /tmp/gridftp-1.10-xl.tar.gz");
print ("DONE\n");

print ("\nInstallation of GridFTP is COMPLETE.\n");
```

Example C-6 illustrates the way the output of this script should appear.

Example: C-6 install-gridftp.pl output

```
[root@y2 /]perl /tmp/install-gridftp.pl 192.168.0.111:/gr1105

----------------------------------------
ITSO Lab IGT setup script
----------------------------------------

This script will install GridFTP on
this host using the 192.168.0.111:/gr1105 source.

To begin the installation hit ENTER.


Setting up installation:

Creating mounting point /mnt/infrastructure                    DONE
Mounting infrastructure on 192.168.0.111:/gr1105               DONE
Copying GridFTP installation files                             DONE

Installing GridFTP:

Installing GridFTP                                             DONE

Cleaning up:

Unounting NFS server                                           DONE
Removing mounting point /mnt/infrastructure                    DONE
Removing GridFTP installation files                            DONE

Installation of GridFTP is COMPLETE.
```

Scripting the installation of additional files

During the course of installing the IBM Grid Toolbox, certain files must be updated manually to properly configure the grid. When installing a large number of instances of the IBM Grid Toolbox, it may be beneficial to store these altered files in one central location and distribute them to each node. The following script was written to copy the following files from the lab NFS server on to a given instance of the IBM Grid Toolbox.

- /etc/grid-security/grid-mapfile
- /etc/hosts
- /etc/ntp.conf

The script was written in Perl and takes in a single command line argument, which is the address of the target NFS server and the desired directory on that server. The command is executed using the following syntax:

```
perl install-files.pl <address of NFS server>:/<directory>
```

Example C-7 provides the code used in the lab environment.

Example: C-7 install-files.pl code

```
#!/usr/bin/perl

#Title : Additional files installation script
#Author: Lee B Wilson
#Email : wilsol@us.ibm.com
#Date  : March 13, 2004

#Purpose: Automate the installation of additional grid files
#       These are files such as /etc/hosts which will be identical
#       on all ITSO lab instances.


#Read in the command line argument providing the location of the NFS server.
#If no location is provided exit and give error.

$nfsServer = $ARGV[0];

#If a NFS server was not supplied terminate script.
if ($nfsServer eq "") {
   print ("NFS server location missing\nSyntax : perl install-igt.pl <NFS
server address>:/<directory>\n");
   exit(0);
}

system ("clear");
```

```
print ("----------------------------------------\n");
print ("ITSO Lab IGT setup script\n");
print ("----------------------------------------\n");
print ("\n");
print ("This script will install additional files on\n");
print ("this host using the $nfsServer source.\n");
print ("\n");
print ("To begin the installation hit ENTER.\n\n");

#Temporary variable to catch the ENTER key.
$temp = <STDIN>;

print ("Setting up installation:\n\n");

#Mount the infrastructure machine.

print ("Creating mounting point /mnt/infrastructure\t\t\t");
system ("mkdir /mnt/infrastructure");
print ("DONE\n");

print ("Mounting infrastructure on $nfsServer\t\t");
system ("mount $nfsServer /mnt/infrastructure");
print ("DONE\n");

#Copy additional files

print ("\nInstalling additional files:\n\n");

#Copy lab grid-mapfile
print ("Copying ITSO lab grid-mapfile\t\t\t\t");
system ("cp /mnt/infrastructure/grid-mapfile /etc/grid-security/grid-mapfile");
print ("DONE\n");

print ("Copying ITSO lab /etc/hosts to /etc/hosts\t\t\t");
system ("mv /etc/hosts /etc/hosts-old");
system ("cp /mnt/infrastructure/hosts /etc/hosts");
print ("DONE\n");

print ("Copying ITSO lab ntp.conf\t\t\t\t\t");
system ("mv /etc/ntp.conf /etc/ntp.conf-old");
system ("cp /mnt/infrastructure/ntp.conf /etc/ntp.conf");
print ("DONE\n");

print ("\nCleaning up:\n\n");
print ("Unounting NFS server\t\t\t\t\t");
system ("umount /mnt/infrastructure");
print ("DONE\n");

print ("Removing mounting point /mnt/infrastructure\t\t\t");
```

```
system ("rmdir /mnt/infrastructure");
print ("DONE\n");

print ("\nInstallation of additional files is COMPLETE.\n");
```

Example C-8 illustrates the way the output of this script should appear.

Example: C-8 install-files.pl output

```
[root@y2 /]perl /tmp/install-files.pl 192.168.0.111:/gr1105

----------------------------------------
ITSO Lab IGT setup script
----------------------------------------

This script will install additional files on
this host using the 192.168.0.111:/gr1105 source.

To begin the installation hit ENTER.


Setting up installation:

Creating mounting point /mnt/infrastructure                    DONE
Mounting infrastructure on 192.168.0.111:/gr1105               DONE

Installing additional files:

Copying ITSO lab grid-mapfile                                  DONE
Copying ITSO lab /etc/hosts to /etc/hosts                      DONE
Copying ITSO lab ntp.conf                                      DONE

Cleaning up:

Unounting NFS server                                           DONE
Removing mounting point /mnt/infrastructure                    DONE

Installation of additional files is COMPLETE.
```

D

Response file

This appendix presents a sample of a response file.

Sample response file

Example D-1 provides a sample of a response file that was created by using the -options-record flag when running a graphical installation of the IBM Grid Toolbox. For more information about creating and using response files, refer to 3.3.3, “Silent installation method” on page 46.

Example: D-1 Sample response file

```
############################################################################
#
#
# InstallShield Options File
#
# Wizard name: Install
# Wizard source: setup.jar
# Created on: Thu Feb 26 14:46:44 CST 2004
# Created by: InstallShield Options File Generator
#
# This file contains values that were specified during a recent execution of
# Install. It can be used to configure Install with the options specified below
# when the wizard is run with the "-options" command line option. Read each
# setting's documentation for information on how to change its value.
#
# A common use of an options file is to run the wizard in silent mode. This
lets
# the options file author specify wizard settings without having to run the
# wizard in graphical or console mode. To use this options file for silent mode
# execution, use the following command line arguments when running the wizard:
#
#    -options "/mnt/gr1105/CAPTURES/RedHat/response-file.rh" -silent
#
############################################################################
#


############################################################################
#
#
# User Input Field - password
#
#


-W passwordPanel.password="passwOrd"


############################################################################
#
```

```
#
# User Input Field - verifyPassword
#
#


-W passwordPanel.verifyPassword="passwOrd"


################################################################################
#
#
# "Base IBM" Feature
#
# The selection state of the "Base IBM" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Base IBM" for installation, use
#
#    -P Base.active=true
#


-P Base.active=true


################################################################################
#
#
# "IBM GT3" Feature
#
# The selection state of the "IBM GT3" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "IBM GT3" for installation, use
#
#    -P IBMGT3.active=true
#


-P IBMGT3.active=true


################################################################################
#
```

```
#
# "Application Server" Feature
#
# The selection state of the "Application Server" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Application Server" for installation, use
#
#    -P AppServer.active=true
#


-P AppServer.active=true


################################################################################
#
#
# "DataBase" Feature
#
# The selection state of the "DataBase" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "DataBase" for installation, use
#
#    -P IBMDatabase.active=true
#


-P IBMDatabase.active=true


################################################################################
#
#
# "IBM Grid Application" Feature
#
# The selection state of the "IBM Grid Application" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "IBM Grid Application" for installation, use
#
#    -P WebApp.active=true
```

```
#


-P WebApp.active=true


################################################################################
#
#
# "OpenJMS" Feature
#
# The selection state of the "OpenJMS" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "OpenJMS" for installation, use
#
#    -P OpenJMS.active=true
#


-P OpenJMS.active=true


################################################################################
#
#
# "Service Group" Feature
#
# The selection state of the "Service Group" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Service Group" for installation, use
#
#    -P ServiceGroup.active=true
#


-P ServiceGroup.active=true


################################################################################
#
#
# "CMM" Feature
#
```

```
# The selection state of the "CMM" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "CMM" for installation, use
#
#    -P CMM.active=true
#


-P CMM.active=true


################################################################################
#
#
# "Policy" Feature
#
# The selection state of the "Policy" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Policy" for installation, use
#
#    -P Policy.active=true
#


-P Policy.active=true


################################################################################
#
#
# "Information Services" Feature
#
# The selection state of the "Information Services" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Information Services" for installation, use
#
#    -P DeployIS.active=true
#
```

```
-P DeployIS.active=true


###############################################################################
#
#
# "Data Management Services" Feature
#
# The selection state of the "Data Management Services" feature. Legal values
# are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Data Management Services" for installation, use
#
#    -P DeployDM.active=true
#


-P DeployDM.active=true


###############################################################################
#
#
# "Program Management Services" Feature
#
# The selection state of the "Program Management Services" feature. Legal
values
# are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Program Management Services" for installation, use
#
#    -P DeployPM.active=true
#


-P DeployPM.active=true


###############################################################################
#
#
# "Common Management Model Services" Feature
#
```

```
# The selection state of the "Common Management Model Services" feature. Legal
# values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Common Management Model Services" for installation,
# use
#
#    -P DeployCMM.active=true
#


-P DeployCMM.active=true


################################################################################
#
#
# "Service Group Services" Feature
#
# The selection state of the "Service Group Services" feature. Legal values
are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Service Group Services" for installation, use
#
#    -P DeployServiceGroup.active=true
#


-P DeployServiceGroup.active=true


################################################################################
#
#
# "Policy Services" Feature
#
# The selection state of the "Policy Services" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Policy Services" for installation, use
#
#    -P DeployPolicy.active=true
```

```
#


-P DeployPolicy.active=true


################################################################################
#
#
# "Post-Install 2" Feature
#
# The selection state of the "Post-Install 2" feature. Legal values are:
#
#    true  - Indicates that the feature is selected for installation
#    false - Indicates that the feature is not selected for installation
#
# For example, to select "Post-Install 2" for installation, use
#
#    -P postInstall2.active=true
#


-P postInstall2.active=true
```

E

Certificate authority

This appendix describes how we established and used a certificate authority in our lab. The following topics are discussed:

- Certificate authority environment
- Setting up the CA used in our lab
- Managing certificates

Certificate Authority environment

In order to establish a grid environment, certificates that were created by an official Certificate Authority (CA) are necessary. In a habitual production environment, a pre-established and reputable CA should be used. For the lab environment that was established to write this book, we created our own CA to introduce the certificate authority topic to the readers, in case they are not familiar with it.

Figure E-1 depicts our lab environment with a separate server that functions as the certificate authority.

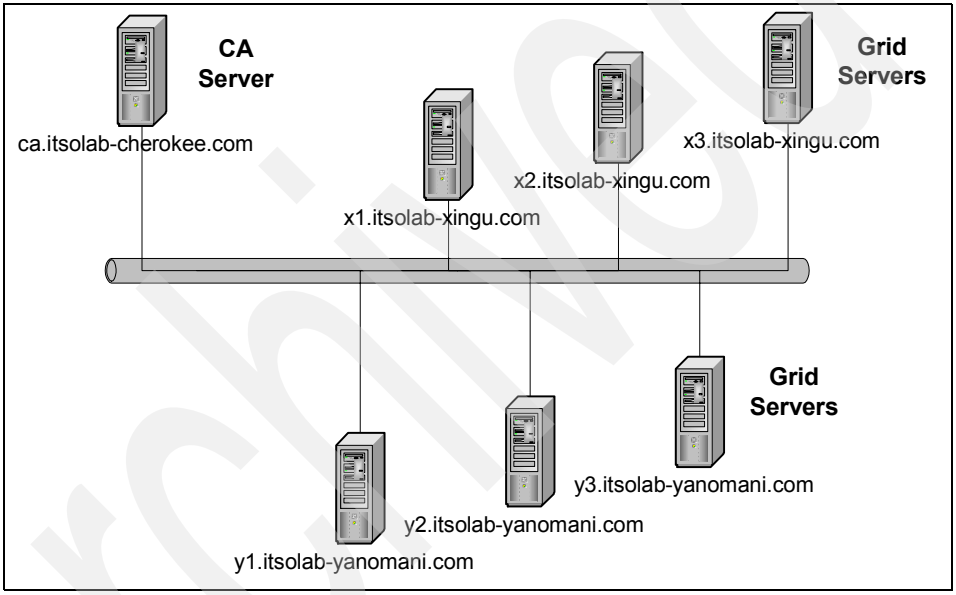


Figure E-1 Lab environment

Hardware requirements

The minimum hardware configuration that is needed to build a CA machine is:

- A Pentium® 1 GHz processor or compatible
- 256 MB of memory
- A hard disk of 10 GB

Software installed

The software that is needed to build the CA machine is:

- Any Linux distribution

Naming and addressing schemes

Table E-1. Table E-2, and Table E-3 describe the naming convention and addressing schemes used in our lab.

Table E-1 IDs and password

User ID	Group ID	Password
root	root	<password>

Table E-2 Host names and IP addressing

Host name	IP	Description
x1.itsolab-xingu.com	192.168.0.11	Grid Server
x2.itsolab-xingu.com	192.168.0.12	Grid Server
x3.itsolab-xingu.com	192.168.0.13	Grid Server
y1.itsolab-yanomani.com	192.168.0.21	Grid Server
y2.itsolab-yanomani.com	192.168.0.22	Grid Server
y3.itsolab-yanomani.com	192.168.0.23	Grid Server
ca.itsolab-cherokee.com	192.168.0.100	CA Server

Table E-3 Certificate naming convention

Generic name	Role	Name in the CA
usercert_request.pem	non-signed user certificate	<user>usercert_request.pem
usercert_cert.pem	signed user certificate	<user>usercert_request.pem
hostcert_request.pem	non-signed host certificate	<machine>hostcert_request.pem
hostcert_cert.pem	signed host certificate	<machine>hostcert_request.pem

Setting up the CA used in our lab environment

We chose to install the OpenSSL package provided by Red Hat Linux during the installation. More information about OpenSSL can be found at:

http://www.openssl.org/

These are the necessary macro activities to create the CA:

- Create CA directory structure.
- Copy the CA configuration file.
- Set up CA.
- Copy the certificate/public key to /CA directory.

CA directory structure

1. Create the directory structure as defined in Table E-4.

Table E-4 CA directory structure

Directory name	Role	Permission
/CA	Top level directory of the CA. This directory must be exported via NFS. Also, contains the OpenSSL demoCA directory and the openssl.cnf configuration file.	drwxr-xr-x
/CA/IN	Used to store incoming certificate requests. The non-signed certificate must be copied to this directory before signing it.	drwxrw-rw-
/CA/OUT	Used to store signed certificates. The signed certificate is placed in this directory by CA after signing it.	drwxr--r--

2. As root, issue the commands in Example E-1 to create the CA directory structure.

Example: E-1 CA directory structure

```
# mkdir /CA
# mkdir /CA/IN
# mkdir /CA/OUT
# chmod 766 /CA/IN
# chmod 744 /CA/OUT
```

CA configuration file

Copy the OpenSSL /usr/share/ssl/opensl.cnf configuration file to our CA directory. This configuration file contains options for OpenSSL, in case you want to customize it.

1. As root, issue the command in Example E-2 to copy the file.

Example: E-2 Copy openssl.cnf file

```
# cp /usr/share/ssl/openssl.cnf /CA
```

2. Be sure to include the definition of the SSLEAY_CONFIG environment variable in /etc/profile, as presented in Table E-5. This variable is used by the internal scripts that are executed by the **openssl** command.

Table E-5 SSLEAY_CONFIG variable

Variable	Description	Value
SSLEAY_CONFIG	openSSL configuration file	“-config /CA/openssl.cnf”

3. As root, use a text editor to add this line into /etc/profile, as shown in Example E-3:

```
export SSLEAY_CONFIG="-config /CA/openssl.cnf"
```

Example: E-3 Export SSLEAY_CONFIG in /etc/profile

```
...(author omits lines)

HOSTNAME=`/bin/hostname`
HISTSIZE=1000

if [ -z "$INPUTRC" -a ! -f "$HOME/.inputrc" ]; then
    INPUTRC=/etc/inputrc
fi

export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE INPUTRC

export SSLEAY_CONFIG="-config /CA/openssl.cnf"

...(author omits lines)
```

4. After this line is added to /etc/profile file, you must source /etc/profile so that the necessary variables are set: As root, issue the command in Example E-4.

Example: E-4 Sourcing /etc/profile

```
# . /etc/profile
```

CA setup

The **CA** script command is used to set up a new CA. This script is provided by OpenSSL and is located in the /usr/share/ssl/misc directory. It displays a setup screen. Enter the information presented in Table E-6 and Table E-7.

Table E-6 CA Setup screen prompt

Setup screen prompt	To be entered
CA certificate filename (or Enter to create)	Click Enter key.
Enter PEM pass phrase.	<passphrase>
Verifying - Enter PEM pass phrase.	<passphrase>
Distinguished Name or a DN	Information that will be part of the CA DN, Distinguished Name. Table E-7 presented the DN used in our lab.

Attention: Be sure to record the PEM pass phrase (password) where you will not forget it. You must uninstall and reinstall the CA if you forget the password.

Table E-7 CA Distinguished name used in our lab

DN component	Value entered
Country Name (2 letter code) [GB]:	US
State or Province Name (full name) [Berkshire]:	Texas
Locality Name (eg, city) [Newbury]:	Austin
Organization Name (eg, company) [My Company Ltd]:	ITSOLAB
Organizational Unit Name (eg, section) []:	
Common Name (eg, your name or your server's hostname) []:	ca
Email Address []:	cd@ca

As *root*, issue the command in Example E-5 on page 231 to set up a new CA.

Attention: Make sure to run the CA setup command from within the /CA directory. The installation creates a demoCA directory within the directory where you run the setup.

Example: E-5 CA setup

```
[root@ca CA]# /usr/share/ssl/misc/CA -newca
CA certificate filename (or enter to create)

Making CA certificate ...
Generating a 1024 bit RSA private key
..................................................++++++
........++++++
writing new private key to './demoCA/private/./cakey.pem'
Enter PEM pass phrase:
Verifying - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:US
State or Province Name (full name) [Berkshire]:Texas
Locality Name (eg, city) [Newbury]:Austin
Organization Name (eg, company) [My Company Ltd]:ITSOLAB
Organizational Unit Name (eg, section) []:
Common Name (eg, your name or your server's hostname) []:ca
Email Address []:ca@ca
```

Public key

The certificate/public key (cacert.pem) is created during the CA setup and posted in the /CA/demoCA directory. This certificate/public key must be distributed later to all grid machines with a different file name that is formed by the hash number of the certificate plus .0. The **c_hash** script command can be used to get the hash of your CA.

- As root, issue the command in Example E-6 to get the hash number of the CA certificate and copy it to /CA directory.

Example: E-6 Get the hash number of the CA

```
[root@ca CA]#  cd /CA/demoCA
[root@ca demoCA]# HASH=`/usr/share/ssl/misc/c_hash cacert.pem | cut -c 1-10`
[root@ca demoCA]# echo $HASH
5243259c.0
[root@ca demoCA]#  cp cacert.pem /CA/$HASH
[root@ca CA]# ls /CA
5243259c.0  demoCA  IN  openssl.cnf  OUT  PROCESSED
```

Note: In the previous commands, the symbol ` represents the open single quotation mark (backtick) that is used to expand shell commands to stdout.

Managing certificates

Now that the CA is installed, it will be possible to sign certificates and remove them from the CA database.

Signing certificates

To sign a certificate, execute the **openssl ca** command using this syntax:

```
openssl ca -in /CA/IN/<filename> -out /CA/OUT/<filename>
```

This command generates the output shown in Example E-7.

Example: E-7 Output when signing a certificate

```
[root@ca /]# openssl ca -in /ca/in/x1-hostcert_request.pem -out
/ca/out/x1-user.pem

Using configuration from /usr/share/ssl/openssl.cnf
Enter pass phrase for ./demoCA/private/cakey.pem:
Check that the request matches the signature
Signature ok
Certificate Details:
        Serial Number: 8 (0x8)
        Validity
            Not Before: Feb 27 16:18:55 2004 GMT
            Not After : Feb 26 16:18:55 2005 GMT
        Subject:
            countryName               = US
            stateOrProvinceName       = Texas
            organizationName          = ITSOLAB
            commonName                = host/x1.itsolab-xingu.com
        X509v3 extensions:
            X509v3 Basic Constraints:
            CA:FALSE
            Netscape Comment:
            OpenSSL Generated Certificate
            X509v3 Subject Key Identifier:
            8A:6D:51:77:06:20:88:10:E5:7F:B2:43:C3:81:67:19:C5:5F:4A:7A
            X509v3 Authority Key Identifier:
            keyid:FE:86:77:C5:F7:A4:AB:3A:0B:7C:DA:A5:0F:C6:62:22:F0:DD:5A:ED
            DirName:/C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=ca/emailAddress=cd@ca
            serial:00
```

```
Certificate is to be certified until Feb 26 16:20:00 2005 GMT (365 days)
Sign the certificate? [y/n]:y

1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated
[root@ca /]#
```

After a certificate is signed, it can be sent back to the user who requested it.

Note: There are more options available for this command, including batch signing of certificates. For more information about these options, run the **man openssl** command.

Removing certificates

To remove a certificate from the CA:

1. Go to the /CA/demoCA directory.
2. Edit the index.txt file with a text editor and remove the line entry for the certificate being removed. Make sure to take note of the certificate's number. In Example E-8, the certificate number is 04.

Example: E-8 Sample line entry in index.txt

```
V   050225223922Z   04    unknown   /C=US/ST=Texas/L=Austin/O=ITSOLAB/CN=userx
```

3. Remove the file in /CA/newcerts with a matching certificate number.

Example: E-9 Removing a certificate

```
[root@ca /]# cd /CA/demoCA/newcerts/
[root@ca newcerts]# ls -al
total 16
drwxr-xr-x    2 root     root         4096 Feb 27 10:22 .
drwxr-xr-x    6 root     root         4096 Feb 27 10:22 ..
-rw-r--r--    1 root     root         3313 Feb 27 10:21 04.pem
-rw-r--r--    1 root     root         3307 Feb 27 10:22 05.pem
[root@ca newcerts]# rm 04.pem
rm: remove regular file `04.pem'? y
[root@ca newcerts]#
```

F

Uninstalling the IBM Grid Toolbox

This appendix describes the following topics related to the uninstall process:

- How to uninstall the IBM Grid Toolbox using the graphical user interface
- How to uninstall the IBM Grid Toolbox from the command line
- How to uninstall the IBM Grid Toolbox in silent mode
- Post-uninstall actions

Uninstalling the IBM Grid Toolbox

As with its installation, the IBM Grid Toolbox can be uninstalled using any of three methods:

- Graphical user interface
- From a command line
- In silent mode without operator interaction

Before beginning any of these methods, stop the IBM Grid Toolbox container to ensure that no grid services are running. The command to stop the container is:

```
/opt/IBMGrid/bin/igt-stop-container
```

The output of this command should be similar to that in Example F-1.

Example: F-1 igt-stop-container command

```
[root@y2 /]# /opt/IBMGrid/bin/igt-stop-container
ADMU0116I: Tool information is being logged in file
           /opt/IBMGrid/AppServer/logs/server1/stopServer.log
ADMU3100I: Reading configuration for server: server1
ADMU3201I: Server stop request issued. Waiting for stop status.
ADMU4000I: Server server1 stop completed.
[root@y2 /]#
```

Graphical user interface uninstall method

This method consists of a graphical wizard that guides the user through the uninstall process.

To begin the uninstall process, enter the following command as root:

```
/opt/IBMGrid/_uninst/uninstaller.bin
```

Tip: To speed up the process for subsequent uninstalls, we recommend generating a response file during the uninstall. To record the responses, begin the uninstall with the following command:

```
/opt/IBMGrid/_uninst/uninstaller.bin -options-record
/<directory>/<response-file-name>
```

Important: Enter this command as a single line.

Example F-2 on page 237 shows how we started the uninstall binary executable in the lab environment.

Example: F-2 Starting the graphical uninstall binary executable

```
[root@y2 /]# /opt/IBMGrid/_unist/uninstaller.bin -options-record
/tmp/uninstall-response-file.txt
```

When initialized, the window shown in Figure F-1 appears.

Figure F-1 Initial uninstall window

Click **Next** to proceed. The uninstaller presents a summary of the items to be uninstalled, as shown in Figure F-2. Review the summary and click **Next** to continue.

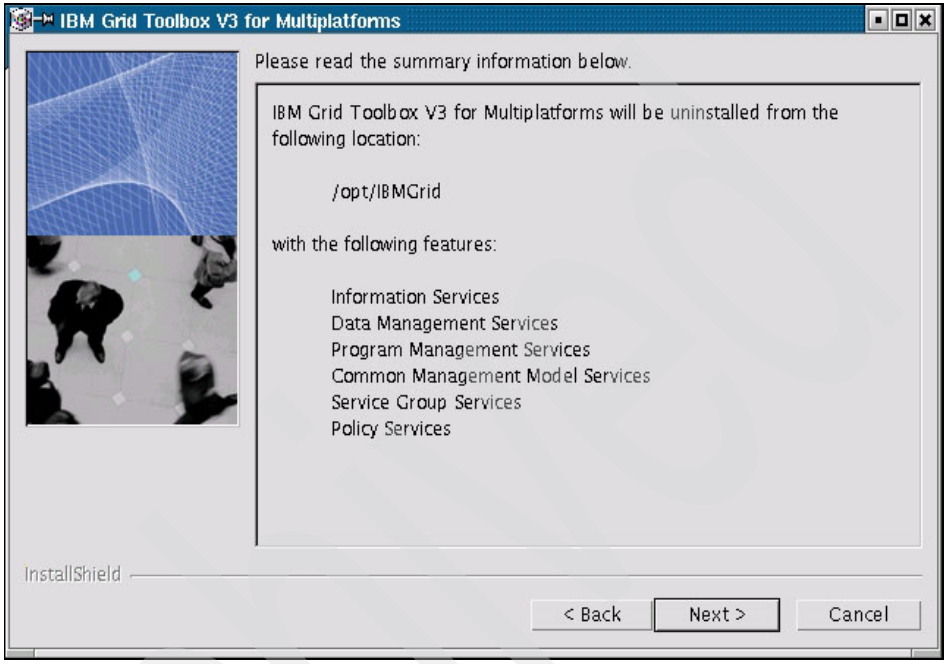


Figure F-2 Summary of uninstall

When complete, the uninstaller opens the window shown in Figure F-3. Click **Finish** to conclude the uninstall.

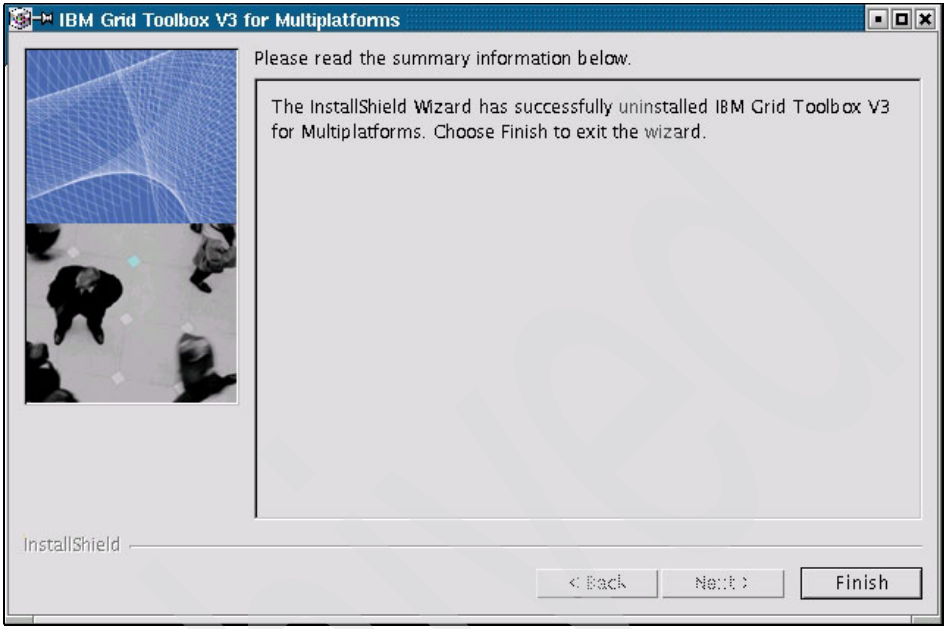


Figure F-3 Final uninstall window

Command line uninstall method

This method consists of a text-based wizard that guides the user through the uninstall process.

1. To begin the uninstall process, enter the following command as root:

```
/opt/IBMGrid/_uninst/uninstaller.bin -console
```

> **Tip:** To speed up the process for subsequent uninstalls, we recommend generating a response file during the uninstall. To record the responses, start the uninstall with the following command:
>
> ```
> /opt/IBMGrid/_uninst/uninstaller.bin
> -options-record/<directory>/<response-file-name> -console
> ```
>
> **Important:** Enter this command as a *single* line.

Example F-3 on page 240 shows how we started the uninstall binary executable in our lab environment.

Example: F-3 Start command line uninstall

```
[root@y2 /]# /opt/IBMGrid/_unist/uninstaller.bin -options-record
/tmp/uninstall-response-file.txt -console
```

2. After the process has begun you should see output similar to Example F-4.

Example: F-4 Initial command line screen

```
[root@y2 /]# /opt/IBMGrid/_uninst/uninstaller.bin -options-record
/tmp/uninstall-response-file.txt -console

InstallShield Wizard

Initializing InstallShield Wizard...
Options record mode enabled - run the wizard to completion to create the
options file /tmp/uninstall-response-file.txt

-------------------------------------------------------------------------------
Welcome to the InstallShield Wizard for IBM Grid Toolbox V3 for Multiplatforms

The InstallShield Wizard will uninstall IBM Grid Toolbox V3 for Multiplatforms
from your computer.
To continue, choose Next.

IBM Grid Toolbox V3 for Multiplatforms
IBM Corporation
www.ibm.com


Press 1 for Next, 3 to Cancel or 4 to Redisplay [1] 1
```

3. Type 1 and press Enter to continue to the uninstall summary screen. It should look similar to Example F-5.

Example: F-5 Uninstall summary

```
-------------------------------------------------------------------------------
IBM Grid Toolbox V3 for Multiplatforms will be uninstalled from the following
location:

/opt/IBMGrid

with the following features:

Information Services
Data Management Services
Program Management Services
Common Management Model Services
Service Group Services
```

```
Policy Services

Press 1 for Next, 2 for Previous, 3 to Cancel or 4 to Redisplay [1] 1
```

4. Review the uninstall summary and type `1` and press Enter to continue. The uninstall begins and the output shown in Example F-6 appears.

Example: F-6 Command line uninstall

```
Uninstalling IBM Grid Toolbox V3 for Multiplatforms...

Uninstalling RPM Package

Uninstalling RPM Package - Uninstalling RPM PackageIBMGrid-Policy...

Uninstalling RPM Package

...(author omits text)...

-------------------------------------------------------------------------------
The InstallShield Wizard has successfully uninstalled IBM Grid Toolbox V3 for
Multiplatforms. Choose Finish to exit the wizard.

Press 3 to Finish or 4 to Redisplay [3] 3
Options file /tmp/uninstall-response-file.txt was successfully created
WARNING: could not delete locked file /opt/IBMGrid
```

5. When the output is complete, type `3` and press Enter to complete the uninstall.

Silent uninstall method

The silent uninstall option enables the grid administrator to uninstall the IBM Grid Toolbox without operator interaction through the use of a response file.

Response files can be generated either as part of a graphical or command line installation, or they can be generated using the following command:

```
/opt/IBMGrid/_uninst/uninstaller.bin
-options-template/<directory>/<response-file-name>
```

Important: This command should appear as a *single* line.

The resulting response file can be edited with a text editor. Example F-7 on page 242 shows a sample uninstall response file that was generated as part of a graphical installation.

Example: F-7 Uninstall response file

```
##############################################################################
#
#
# InstallShield Options File
#
# Wizard name: Uninstall
# Wizard source: uninstall.jar
# Created on: Mon Mar 01 09:32:48 CST 2004
# Created by: InstallShield Options File Generator
#
# This file contains values that were specified during a recent execution of
# Uninstall. It can be used to configure Uninstall with the options specified
# below when the wizard is run with the "-options" command line option. Read
# each setting's documentation for information on how to change its value.
#
# A common use of an options file is to run the wizard in silent mode. This
lets
# the options file author specify wizard settings without having to run the
# wizard in graphical or console mode. To use this options file for silent mode
# execution, use the following command line arguments when running the wizard:
#
#    -options "/mnt/gr1105/CAPTURES/x1-uninstall-response-file.txt" -silent
#
##############################################################################


##############################################################################
#
# This wizard does not have user-modifiable options.
#
```

To begin the silent uninstall, run the following command as root:

```
/opt/IBMGrid/_uninst/uninstaller.bin -options/<directory>/<response-file-na
me> -silent
```

Important: This command should appear as a *single* line.

Example F-8 shows how we started the uninstall binary executable file in our lab environment.

Example: F-8 Silent uninstall routine

```
[root@y2 /]# /opt/IBMGrid/_unist/uninstaller.bin -options
/tmp/uninstall-response-file.txt -silent
```

The uninstall procedure begins, and a prompt is returned when the uninstall is complete.

Note: During this procedure, no output is generated.

Tip: To regain control immediately, run the uninstall in the background by appending an ampersand (&) to the end of the uninstall command.

Post-uninstall actions

The IBM Grid Toolbox uninstall routine does not completely remove the IBM Grid Toolbox. To clean up any files left by the uninstaller, enter the following command from the /opt directory as root:

```
rm -rf IBMGrid
```

Example F-9 illustrates how to remove any leftover files.

Example: F-9 Remove remaining IBMGrid files

```
[root@y2 opt]# rm -rf IBMGrid
```

Uninstalling related software

For more information about how to uninstall related software, refer to the uninstall section for the product you wish to uninstall in Chapter 4, “Installing related software” on page 59.

G

Logging & Error Messages

This appendix provides some information about the various log files that are associated with the IBM Grid Toolbox.

Log files in the IBM Grid Toolbox

The IBM Grid Toolbox includes various log files that provide useful information about the running of IGT, which can also be used when troubleshooting errors.

For deploy issues, refer to these logs in /opt/IBMGrid/logs:

deploy.txt	Services deployment log messages
deploye.txt	Error messages during the services deployment

For messages related to starting or stopping the container, refer to these logs in /opt/IBMGrid/logs.

start.log	Shows server launching messages.
starte.log	Shows server launching error messages.
stop.log	Shows server stop request messages.
stope.log	Shows server stop request error messages.

Application server–related message logs are located in /opt/IBMGrid/AppServer/logs/<server1>:

startServer.log	Shows environment of the started server.
stopServer.log	Shows environment of the stopped server.
serverStatus.log	Shows the status (stopped/started) of the server.
SystemOut.log	Shows server environment and services execution/start.
SystemErr.log	Shows error messages during services execution/start.
<server1>.pid	Process-id of the server.
native_stderr.log	Logs server stdout messages.
native_stdout.log	Logs server stderr messages.

Problems with the Grid Services Manager (GSM) or the inability to open the GSM page could be from the installation, and messages relating to this are found in these logs in /opt/IBMGrid/logs:

omui.log	Shows management starting messages.
omuie.log	Shows error messages related to management.

IBM Grid Toolbox permission changes during the install and related messages regarding this are found in these logs in /opt/IBMGrid/logs:

setperms.log	Shows setup and post-install messages.
setpermse.log	Shows error messages during setup and post-install.

For all other error messages refer to these logs /opt/IBMGrid/logs.

stderr.log	Shows stdout messages.
stdout.log	Shows stderr messages.

When contacting Service, include all of the files in these directories, depending on the type of problem you are reporting.

For other error messages and troubleshooting, refer to:

http://publib.boulder.ibm.com/eserver/v1r1/en_US/index.html?info/ogsainfo/trouble.htm

H

WSRF

This appendix gives an overview of the WS-Resource framework, known as WSRF, an open framework for modeling and accessing stateful resources using Web services. WSRF defines where Web service standards are evolving to meet grid services.

WS-Resource Framework

The WS-Resource Framework is a set of six Web Services specifications that define terms such as WS-Resource approach to modeling and managing state in a Web services context. Three drafts of specifications have been released as we write this book, and an architecture document that describes the WS-Resource approach to modeling stateful resources with Web Services. There are also plans for other related documents, one of them comparing the WS-Resource Framework with the Open Grid Services Infrastructure.

WSRF is the the natural convengence of the grid services, as defined in OGSA, and the Web services framework.

WS-Resource Framework specifications

We include basic specifications for WSRF here. Specification authors plan to submit them to an appropiate standards body in the near future. These drafts have been made avaliable to the GGF OGSI working group for comments.

WS-ResourceLifetime

Defines the mechanisms for WS-Resource destruction, including message exchanges that enable a requestor to destroy a resource, either immediately or by using a time-based scheduled resource termination mechanism.

WS-ResourceProperties

Defines how the type of definition of a WS-Resource can be associated with the interface description of a Web service, and message exchanges for retrieving, changing, and deleting WS-Resource properties. This relationship is the implied resource pattern.

WS-Notification

Defines mechanisms for event subscription and notification using a topic-based publish-subscribe pattern.

WS-RenewableReferences

Defines a conventional decoration of a WS-Addressing endpoint reference with policy information that is needed to retrieve an updated version of an endpoint reference when it becomes invalid.

WS-ServiceGroup

Defines an interface to heterogeneous by-reference collections of Web services. ServicesGroup can be used to form a wide variety of collections of services or resources, including registries of services and associated resources.

WS-BaseFault

Defines an XML Schema type for base faults, along with rules for how this base fault type is used and extended by Web services. This simplifies problem determination by standarizing a base set of information that would appear in fault messages.

WS-Resource Framework, some definitions

In this section, some definitions are included regarding WSRF. This definitions are extracted from work-in-progress documents. Refer to the pointers at the end of this appendix for further updates.

Web service

The term *Web services* emerged in the year 2000 with the introduction of technologies such as SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language), and UDDI (Universal Description Discovery and Integration). Later, the term SOA (Service Oriented Architecture) was coined to describe the overall approach of building loosely coupled distributed systems with minimal shared understanding among system components.

Web services are basically Web-based applications, but they are different in the fact that they are designed to support application-to-application communication.

> "A Web Service is a software system designed to support interoperable machine-to-machine interaction over a network."

A Web service is a component, deployed within some runtime environment, that is responsible for executing the code of the Web service and for dispatching messages to the Web Service. IBM WebSphere and JBoss are two examples of runtime environment.

A Web service, as defined in the WSRF, is stateless, meaning that a service whose implementation *maintains no dynamic data*, but it acts upon stateful resources (documents) based on messages it sends and receives. When a Web service is stateless, it will not maintain dynamic state (a state for which the service is responsible between message exchanges with its requestors). This also brings some advantages: A stateless Web service can be restarted following failure without concern for its history or prior interactions, and more copies can be created or destroyed in response to changing load.

Any Web service should be designed with the Web service interface in mind. This Web service interface is described by using WSDL, *and defines the Web service capabilities in terms of a collection of operations that may be invoked by other entities, known as service requestor.* Any other Web service can be, at some point in time, a service requestor.

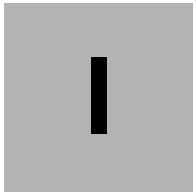
WS-Resource

When Web services are supposed to be stateless, considered as a stateless message processor, in many casese these message exchanges are supposed to enable access/update to state maintained by other system components such as those file systems, databases, or other entities that can also be considered as stateful resources. The link between one or more stateful resources and a Web service is the Implied Resource Pattern, *a set of conventions on Web services technologies, in particular XML, WSDL, and WS-Addressing.* Implicit means that the requestor does not provide the identity if the resource as an explicit parameter in the body of the request message, the context used to designate the implied stateful resource is encapsulated in the WS-Addressing endpoint reference used to address the target Web service at its endpoint. The term patter indicates that the relationship between Web services and stateful resources is codified by a set of conventions on existing Web services technologies: XML, WSDL, and WS-Addressing. Note that when the Web service itself is stateless, when it participates with these other two elements the result can be a stateful operation.

> "WS-Resource is the result of the participation of a stateful resource in the implied resource pattern."

References

http://www.globus.org/
http://www.ggf.org/
http://forge.gridforum.org/
http://www.oasis-open.org/
http://www.globusworld.org/
http://www.ibm.com/developerworks/library/ws-resource/



Checklist and worksheet

This appendix provides a checklist that describes the activities that are employed in planning and installing the IBM Grid Toolbox and where that activity is defined in this book. There is also a configuration worksheet to assist in planning and installing the IBM Grid Toolbox.

IBM Grid Toolbox checklist

Table I-1 can be used to identify activities for planning and installing the IBM Grid Toolbox, to find where the activity is discussed in this book, and to keep track of the status of the activity.

Table I-1 IBM Grid Toolbox checklistt

Activity description	Location	Completion date	Status
Planning			
1. Acquire the IBM Grid Toolbox	2.1, “IBM Grid Toolbox packaging” on page 12		
2. Plan hardware and software requirements	2.2, “IBM Grid Toolbox requirements” on page 13		
3. Plan which IBM Grid Toolbox service to install	2.3, “Planning for installation” on page 17		
4. Plan grid security	2.4, “Planning for security” on page 23		
5. Plan production environment	2.6, “Planning a production environment” on page 25		
6. Plan development environment	2.7, “Planning a development environment” on page 26		
7. Plan product support	Appendix J, “Software support for the IBM Grid Toolbox” on page 259		
Installation			
1. Install Linux	3.2.1, “Install Linux” on page 30		
2. Configure network	3.2.2, “Configure network” on page 31		

Activity description	Location	Completion date	Status
3. Configure NTP	3.2.3, “Configure Network Time Protocol (NTP)” on page 31		
4. Install IBM Grid Toolbox	3.3, “Installing the IBM Grid Toolbox” on page 33		
5. Post-install tasks	3.3.4, “Post installation” on page 46		
6. Secure the grid			
a. Obtain CA certificate	“Obtain a CA certificate” on page 47		
b. Set up host certificates	“Request a host certificate” on page 49		
c. Set up user certificates	“Request a user certificate” on page 50		
d. Update gridmap file	“Adding a grid mapfile entry” on page 53		
7. Installation verification	3.3.6, “Verifying the installation” on page 54		
8. Install Apache Ant	4.1, “Apache Ant” on page 60		
9. Install Pegasus and SBLIM	4.2, “Pegasus and SBLIM” on page 61		
10. Install GridFTP	4.3, “GridFTP” on page 66		

Configuration worksheet

The following worksheet can assist in recording information about the grid environment.

Servers

Table I-2 and Table I-3 provide information about the server machines.

Table I-2 Grid nodes

Host name	Short name	IP address	User certification common name

Table I-3 Support servers

Function	Host name	Short name	IP address
Infrastructure			
Certificate Authority			
NTP server			

Installed grid services

Table I-4 records what grid services are installed on each grid node.

Table I-4 Grid services installed (check each column that applies)

Host name	IS	DMS	PMS	CMM	PS	SGS

IS = Information Services

DMS = Data Management Services

PMS = Program Management Services

CMM = Common Management Model Services

PS = Policy Services

SGS = IBM Service Group Services

User IDs

Table I-5 records what user IDs are created and their associated group and password.

Table I-5 User IDs

ID	Group ID	Password
ibmgrid	ibmgrid	

J

Software support for the IBM Grid Toolbox

IBM offers support for basic usage, installation, and defect problems on the IBM Grid Toolbox V3 through a special support offering. The terms and pricing of this offering are variable, based on individual customer needs. To obtain more information, and to obtain a support contract, contact the Support Sales team.

Preventive maintenance service packs are available for each of the platforms on which the IBM Grid Toolbox is supported.

IBM Grid Toolbox Web Page

To get IBM support for the IBM Grid Toolbox (Figure J-1), go to:

http://www.ibm.com/grid/solutions/grid_toolbox.shtml

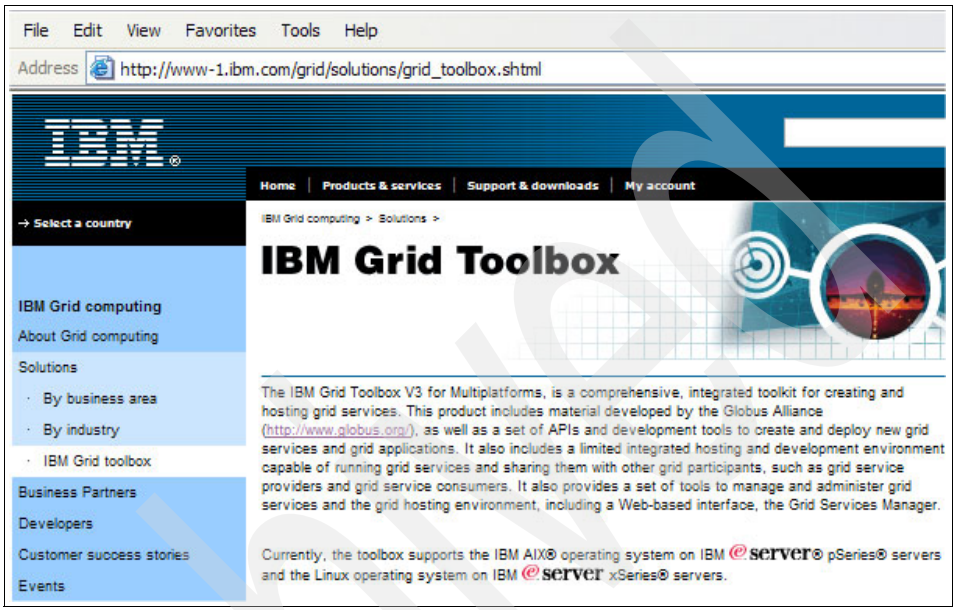


Figure J-1 IBM Grid Toolbox Web site

Choose **Support** and navigate to your country. Contact the number for Software Support Sales to reach a sales representative who will assist you with the contracts for receiving IBM Grid Toolbox product support.

Glossary

Ant Apache Ant; a Java-based build tool for developing software across multiple platforms.

API Application Programming Interface.

Automation. The capability to dynamically deploy, monitor, manage, and protect an IT infrastructure to meet business needs with little or no human intervention. One of the three properties of the On Demand Operating Environment as depicted in the IBM On Demand Blueprint.

Autonomic computing An approach to self-managed computing systems with a minimum of human interference. The term derives from the body's autonomic nervous system, which controls key functions without conscious awareness or involvement.

Axis Apache Extensible Interaction System; an Open Source SOAP server and client.

CA Certificate Authority; (**1**) an instance or external institute to issue authority certificates to identify the certificate holder to use certain services; (**2**) in e-commerce, an organization that issues certificates, authenticates the certificate owner's identity and the services that the owner is authorized to use, renews existing certificates, and revokes certificates belonging to users who no longer exist.

Common Management Model (CMM) Services CMM services provide the infrastructure required to represent and instrument resources as a grid service to enable it to be queried and managed.

Data Management Services Data Management Services enables data transfer throughout the grid. This service gives grid applications the ability to move data from one node to another.

Data placement manager Data placement encompasses data transfer, staging, replication, data positioning, and space allocation and de-allocation. Placement manager automatically handles failure and network characteristics changes, and performs tuning whenever characteristics change.

Data policy manager Grids are distributed and heterogeneous environments that handle different data types, different data access policy. Cache, replica, and security are concerns of the data policy manager.

Factory A special class of service that is responsible for creating grid service instances.

GASS Global Access to Secondary Storage; used for file staging and cache management.

GGF The Global Grid Forum; founded in 2001 when the merger of regional grid organizations created a single worldwide group.

GIIS Grid Index Information Service; the database that contains indexes of resource information registered by the GRIS and other GIISs. It can be seen as a grid-wide information server. GIIS has a hierarchical mechanism, such as DNS, and each GIIS has its own name. This means that client users can specify the name of a GIIS node to search for information.

Globalnamespace One file structure subdivided into parts called filesets, which is available simultaneously to all clients.

Globus A collaborative project based at Argonne National Laboratory that is focused on enabling the application of grid concepts to computing.

GRAM The Grid Resource Allocation and Management API; a means for enabling programs to be started on remote resources.

Grid Computing A type of distributed computing in which a wide-ranging network connects multiple computers whose resources can then be shared by all end users. Includes what is often called *peer-to-peer* computing.

Grid Services Services provided by a grid; Web services that conform to a set of conventions (interfaces and behaviors).

GridFTP The Grid File Transfer Program; provides high-performance and reliant data transfer.

GRIS The Grid Resource Information Service; the repository of local resource information derived from information providers. GRIS can register its information with a GIIS.

GSI Grid Security Infrastructure; contains components to secure your grid network.

GT3 Globus Toolkit 3.0, a reference implementation of OGSA developed by Globus.

IBM Service Group Services A grid service that maintains information about a group of grid services. Grid applications can query the service group to get information about the grid services that are associated with it, including a description of the included services, and references to the services themselves.

IEEE Institute of Electrical and Electronics Engineers, a professional society accredited by the American National Standards Institute (ANSI) to issue standards for the electronics industry.

IESG The Internet Engineering Steering Group; the executive committee of the Internet Engineering Task Force (IETF). The IESG reviews and oversees the work that is produced by individual IETF working groups and charters all new working groups.

IETF The Internet Engineering Task Force; the task force of the Internet Architecture Board (IAB) that is responsible for solving the short-term engineering needs of the Internet. The IETF consists of numerous working groups, each focused on a particular problem. Internet standards are typically developed or reviewed by individual working groups before they can become standards.

Information Integration A collection of technologies that combines database management systems, Web services, replication, federated systems, and warehousing functions into a common platform.

Information Services Information services provide information about grid resources for use in resource discovery, selection, and optimization and maintain knowledge about resource availability, capacity, and current utilization.

ISV Independent Software Vendor; an enterprise that offers software solutions.

ITS Integrated Technology Services, a subdivision of IBM Global Services that provides technical support services for IBM and non-IBM platforms and products.

J2EE Java 2 Enterprise Edition; the second version of the platform definition for advanced Java applications built on Enterprise Java Beans (EJB).

LDAP Lightweight Directory Access Protocol; builds on TCP/IP to define a query-response protocol for querying the state of remote databases.

Life Cycle The states between the creation and destruction of grid service instances.

Metadata In storage management terminology, the information about files when the control information flows in a different path than the data.

MPI Message Passing Interface; an application interface that enables formatted messages to enter and leave an application.

NCSA National Center for Supercomputing Applications; a US government institution for supercomputing.

NFS Network File System; a distributed file system protocol that supports traditional file access while integrating support for file locking and the mount protocol.

Notification A mechanism by which a party sends (notification source) a change of state to the party who has requested (notification sink) to be notified.

NTP Network Time Protocol server; keeps the time on the network.

OASIS Organization for the Advancement of Structure Information Standards; an industry consortium promoting XML, TCP/IP, and UDDI.

OGSA Open Grid Services Architecture; a standard setting the base for communication in grids across virtual organizations. OGSA marries open standards and grid computing protocols with Web Services, bringing together the ability to share computing resources with the ability to provide application interoperability over the Internet.

OGSA-DAI Open Grid Services Architecture - Data Access and Integration; its objective is to define an open standard and open source based in uniform service interfaces for accessing heterogeneous data sources

OGSI Open Grid Services Infrastructure; defines how to create, destroy, manage, and exchange message among Grid services.

Pegasus An open-source implementation of the Distributed Management Task Force (DMTF) Common Information Model (CIM) and Web Based Enterprise Management (WBEM) standards.

Policy Services Policy services in the IBM Grid Toolbox enable administrators to define a set of business goals and to enforce a set of rules that enable their grid to meet those goals.

Program Management Services The IBM implementation of resource management grid services in the IBM Grid Toolbox V3 for Multiplatforms. Resource management grid services simplify the use of remote systems by providing a standard interface for requesting and using remote system resources for the submission and control of jobs.

QoS Quality of Service; a term used in a Service Level Agreement (SLA) that denotes a guaranteed level of performance (for example, response times less than 1 second).

Registry The interface that enables a set of grid services to periodically register their GSHs (Grid Service Handle) into a registry service, to allow for discovery of services in that set.

Reliable Invocation Techniques for ensuring the accuracy of method invocations in case multiple instances have been created and there are redundant grid services in the space.

SBLIM The Standards Based Linux Instrumentation for Manageability; an Open Source project for enhancing the manageability of GNU/Linux systems.

Secure Interface The physical layer connection between a gateway and a secure network.

Secure Network A set of nodes that are controlled by a single administrative party.

Service Data A structured collection of information that is associated with an instance of a grid service.

SLA Service Level Agreement; a contract in which a service provider agrees to deliver a minimum level of service.

SOA Service Oriented Architecture; an architectural approach where an application is composed of independent and cooperating components called Services.

SOAP Simple Object Access Protocol; an XML-based messaging protocol.

SSL/TLS Secure Socket Layer / Transport Layer Security; a security protocol that provides communication privacy. SSL enables client/server applications to communicate in a way that is designed to prevent eavesdropping, tampering, and message forgery. SSL was developed by Netscape Communications Corp. and RSA Data Security.

TLS Transport Layer Security; a protocol that provides privacy and data integrity between two communicating applications, layered on top of TCP/IP.

Tomcat A servlet container developed by Apache. The official reference implementation for JSP/Servlet technologies.

UDDI Universal Description, Discovery, and Integration; a specification that defines the mechanisms for storing and searching the Web services definitions information defined in the WSDL documents.

URI Uniform Resource Identifier. A web service is a software application identified by an URI.

URL Universal Resource Locator; the address that translates to a TCP/IP address in order to access resources on the Internet.

Utility-based computing A computing environment on which services are available and used as in utility enterprises such as electricity or water, and paid according to consumption.

Virtual Organization A virtual entity whose users and servers are geographically apart but share their resources collectively as a larger grid. The users of the grid can be organized dynamically into a number of virtual organizations, each with different policy requirements.

W3C The World Wide Web Consortium; an independent work group formed in 1994 that provides standards like XML.

Web Services A way of providing computational capabilities using standard Internet protocols and architectural elements. For example, a database Web Service would use Web browser interactions to retrieve and update remotely located data. Web Services use UDDI to make their presence known.

Wrappers Mechanisms by which the federated server interacts with data sources. The federated server uses routines that are stored in a library called a wrapper module to implement a wrapper. These routines enable the federated server to perform operations such as connecting to a data source and retrieving data from it iteratively.

WSA Web Services Architecture; a standardized approach based on SOAP and WSDL to develop Web Service solutions.

WSDD Web Service Deployment Descriptor; an XML-based deployment descriptor.

WSDL Web Services Description Language; a mechanism for representing the services that a provider has and is making available and the specifics for accessing that service.

WSFL Web Services Flow Language; addresses flows and dependencies between services.

WSIL Web Services Inspection Language; a means for discovering the WSDL-described services that a provider has made available.

WSRF Web Services Resource Framework; proposed as a refactoring and evolution of OGSI.

XML Extensible Markup Language; a data representation method.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 268. Note that some of the documents referenced here may be available only in softcopy.

- *e-business On Demand Operating Environment*, REDP-3673
- *Enabling Applications for Grid Computing with Globus,* SG24-6936
- *Fundamentals of Grid Computing,* REDP-3613
- *Globus Toolkit 3.0 Quick Start*, REDP-3697
- *How to Organize a Localization Pack*, TIPS0130
- *Introduction to Grid Computing with Globus,* SG24-6895
- *Using a Callback Mechanism with Globus*, TIPS0190

Other publications

These publications are also relevant as further information sources:

- Foster, et al, *The Grid: Blueprint for a New Computing Infrastructure,* Morgan Kaufmann, 1999, ISBN 1558604758
- Foster, Kesselman, Tuecke, *The Anatomy of the Grid: Enabling Scalable Virtual Organizations,* 2001

  http://www.globus.org/research/papers/anatomy.pdf
- Jacob, *Design an application for grid*, 2003

  http://www-106.ibm.com/developerworks/grid/library/gr-design.html
- Kreger, *Web Services Conceptual Architecture,* 2001

  http://www-306.ibm.com/software/solutions/webservices/pdf/WSCA.pdf

Online resources

These Web sites and URLs are also relevant as further information sources:

- Globus Java Programmer's Guide Core Framework

  http://www-unix.globus.org/toolkit/3.0/ogsa/docs/java_programmers_guide.html
- Grid Service Development Tools Guide

  http://www-unix.globus.org/toolkit/3.0/ogsa/docs/tools_guide.html
- The Globus Toolkit 3 Programmer's Tutorial

  http://www.casa-sotomayor.net/gt3-tutorial/
- How to build a Grid Service using GT3

  http://www-unix.mcs.anl.gov/~bacon/tutorial/
- WS-Resource Framework Documents

  http://www.globus.org/wsrf/#relevant
- Globus Alliance

  http://www.globus.org
- Global Grid Forum (GGF)

  http://www.ggf.org/
- GridForge - working respository for GGF Working and Research Groups

  http://forge.gridforum.org/
- OASIS Technical Committee

  http://www.oasis-open.org/
- GlobusWORLD

  http://www.globusworld.org/
- IBM developerWorks Grid Computing Web site

  http://www.ibm.com/developerworks/grid
- OGSI Version 1.0 (Draft)

  http://www.gridforum.org/ogsi-wg/drafts/draft-ggf-ogsi-gridservice-29_2003-04-05.pdf,
- Grid Service Specification (Draft 3)

  http://www.globus.org/research/papers/gsspec.pdf
- Apache Software Foundation

  http://www.apache.org

- Apache Ant Project

  http://ant.apache.org
- Apache Axis Project

  http://ws.apache.org/axis/
- Apache Jakarta Project

  http://jakarta.apache.org
- Apache Software license

  http://www.opensource.org/licenses/apachepl.php
- Condor

  http://www.cs.wisc.edu/condor/
- World Wide Web Consortium (W3C)

  http://www.w3.org/
- SOAP 1.1 specification

  http://www.w3.org/TR/SOAP/
- Web Service Description Language (WSDL) 1.1 specification

  http://www.w3.org/TR/wsdl
- Open Source Initiative license information

  http://opensource.org/docs/certification_mark.php
- Open JMS

  http://openjms.sourceforge.net/
- GNU General Public License

  http://www.gnu.org/copyleft/gpl.html
- GNU Lesser General Public License

  http://www.opensource.org/licenses/lgpl-license.php
- IBM @server Information Center

  http://publib.boulder.ibm.com/eserver/v1r1/en_US/index.html
- IBM LoadLeveler®

  http://www.ibm.com/servers/eserver/pseries/library/sp_books/loadleveler.html
- IBM Public License

  http://www.opensource.org/licenses/ibmpl.php
- Application Development Environments

  http://www.globus.org/research/development-environments.html

- IBM Grid Toolbox

  http://www-1.ibm.com/grid/solutions/grid_toolbox.shtml?Open&ca=daw-prod-gridtoolbox
- Grid Application Framework for Java

  http://www.alphaworks.ibm.com/tech/GAF4J
- Globus Java Programmer's Guide Core Framework

  http://www-unix.globus.org/toolkit/3.0/ogsa/docs/java_programmers_guide.html
- Grid Service Development Tools Guide

  http://www-unix.globus.org/toolkit/3.0/ogsa/docs/tools_guide.html
- Portable Batch System (PBS)

  http://pbs.mrj.com/
- OpenPegasus

  http://www.openpegasus.org/
- Distributed Management Task Force (DMTF)

  http://www.dmtf.org/about
- Standards Based Linux Instrumentation for Manageability (SBLIM)

  http://www-124.ibm.com/developerworks/projects/sblim/

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications, and Additional materials, as well as order hardcopy Redbooks and CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

D

F

G

H

I

J

L

M

N

O

P

R

S

Grid Computing with IBM Grid Toolbox

(0.5” spine)
0.475”<->0.875”
250 <-> 459 pages

Grid Computing with the IBM Grid Toolbox

Redbooks

Introduces IBM Grid Toolbox V3 for Multiplatforms V1.1

Demonstrates installation and use

Outlines sample services

The IBM Grid Toolbox can assist enterprises that deploy, manage, and control grid computing, as well as developers who create products that assist in managing and deploying grids. This grid-enabling toolkit contains standardized development code, much of which was harvested from the open source community, plus an added database and run-time environment.

This IBM Redbook is designed to give the reader a comprehensive view of the IBM Grid Toolbox.

As the IBM Grid Toolbox is designed in a layered approach, we describe the product by introducing each underlying layer until the whole ecosystem is revealed. The product significantly leverages open standards in the grid computing world, so we show how the IBM Grid Toolbox complements and enhances these standards for the development and deployment of grid services and applications.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6332-00 ISBN 0738498041