



GridSphere: A Portal Framework for Building Collaborations

Jason Novotny novotny@aei.mpg.de
Michael Russell russell@aei.mpg.de
Oliver Wehrens wehrens@aei.mpg.de

Albert Einstein Institute

Portals Team



Introduction



- Grid Portals build upon the familiar Web portal model, such as Yahoo or Amazon, to deliver the benefits of Grid computing to virtual communities of researchers and scientists, providing customizable, easy-to-use, singular access points to Grids.



Grid Middleware Congress



The State of Grid Portals



- A Portal is only as good as the underlying deployed infrastructure
- Portlet development often involves debugging underlying middleware
- Often difficult and hard to maintain glue code must be written connecting the portal to Grid services, due to lack of/evolving standards.
- Most portals are stovepipe solutions that provide a complete solution with very little customization capabilities.
 - Separation of presentation and login
 - Generally hard coding of underlying Grid infrastructure details and the codebase
- Lack of real usability has made it difficult to test and evaluate user interfaces.
- Web application development still remains a tedious task with little in the way of reusable components, forcing developers to constantly “re-invent” the wheel.

Grid Middleware Congress



GridLab Project



- Funded by the EU (5+ M€), January 2002 – December 2004
- Application and Testbed oriented
 - Cactus Code, Triana Workflow, all the other applications that want to be Grid-enabled
- Main goal: to develop a Grid Application Toolkit (GAT) and set of grid services and tools...:
 - resource management (GRMS),
 - data management,
 - monitoring,
 - adaptive components,
 - mobile user support,
 - security services,
 - portals,
- ... and test them on a real testbed with real applications

Grid Middleware Congress



GridLab VO



- PSNC (Poznan) - coordination
 - AEI (Potsdam)
 - ZIB (Berlin)
 - Univ. of Lecce
 - Cardiff University
 - Vrije Univ. (Amsterdam)
 - SZTAKI (Budapest)
 - Masaryk Univ. (Brno)
 - NTUA (Athens)
 - Sun Microsystems
 - Compaq (HP)
 - ANL (Chicago, I. Foster)
 - ISI (LA, C.Kesselman)
 - UoWisconsin (M. Livny)
- Users!
 - EU Astrophysics Network,
 - DFN TiKSL/GriKSL
 - NSF ASC Project
 - other Grid projects
 - Globus, Condor,
 - GrADS,
 - PROGRESS,
 - GriPhyn/iVDGL,
 - CrossGrid and all the other European Grid Projects (GRIDSTART)
 - other...

Grid Middleware Congress



GridLab Services

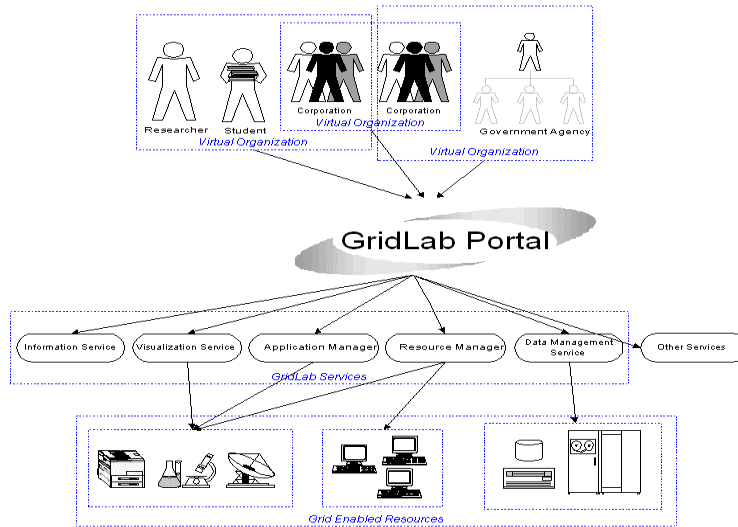


- Resource Management Services
 - Resource Brokering
 - Job Submission
 - Usage Policy Management
- Data Management Services
 - Data Replication
 - Date Movement
 - Visualization Services
- Monitoring and Adaptive Services
 - Application Performance Monitoring
 - Resource Utilization
 - The ability to alter applications at runtime
- Information Services
 - GIIS/GRIS Based
- Testbed Management Services
 - Form the basis of the GridLab Administrative Portlets
- Notification Services... and many more!

Grid Middleware Congress



Portal Bridges Users and Services



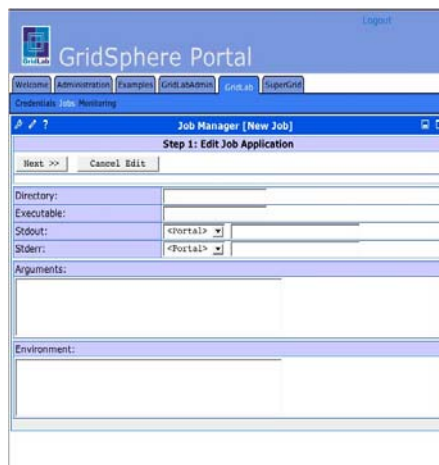
Grid Middleware Congress



GridLab Portal



- The GridLab Portal plays a critical role within the GridLab architecture as the front-end to the resources and services GridLab provides.
- Provide a well-integrated, cohesive environment for accessing GridLab resources and services.



Grid Middleware Congress



Early Grid Portal Projects



- **Grid-Port:**
 - Perl-based framework developed by Mary Thomas and Steve Mock at San-Diego Supercomputing Center (SDSC)
- **Grid Portal Development Toolkit (GPDK):**
 - Developed by Jason Novotny at Lawrence Berkeley National Laboratories (LBNL)
 - Aimed at providing a generalized reusable portal that can be tailored for application specific requirements
- **Astrophysics Simulation Collaboratory (ASC):**
 - Developed by Michael Russell at University of Chicago, now lead by Greg Daues of National Super-Computing Alliance (NCSA)
 - Provides an application specific portal to meet the needs of the computational astrophysics community

Grid Middleware Congress



GridSphere Portal Framework



- GridLab Portal is developed using the GridSphere Portal Framework
- GridSphere combines the lessons learned in the development of the **Astrophysics Simulation Collaboratory** and the **Grid Portal Development Toolkit**, one of the earliest and most widely used research projects within the Grid portal community.
- The GridSphere portal framework provides an architecture for building “pluggable” web applications using the Portlet model.
- Allows developers to concentrate on delivering end-user functionality encapsulated by portlets which act as “mini-applications” displayed in a portal
- Framework is responsible for presentation and management of portlets.
- Makes use of many well understood design patterns for separating logic from presentation

Grid Middleware Congress



GridSphere Feature List



- Portlet API implementation nearly fully compatible with IBM's WebSphere 4.2.
- Support for the easy development and integration of "third-party portlets"
- Higher-level model for building complex portlets using visual beans and the GridSphere User Interface (UI) tag library.
- Flexible XML based portal presentation description can be easily modified to create customized portal layouts.
- Built-in support for Role Based Access Control (RBAC) separating users into guests, users, admins and super users.
- Sophisticated portlet service model that allows for creation of "user services", where service methods can be limited according to user rights.

Grid Middleware Congress



GridSphere Feature List (cont.)



- Persistence of data provided using Castor JDO for RDMS database support
- Integrated Junit/Cactus unit tests for complete server side testing of portlet services including the generation of test reports.
- Documentation uses DocBook for HTML & PDF output of guides and tutorials
- GridSphere core portlets offer base functionality including login, logout, user and access control management.
- Localization support in the Portlet API implementation and GridSphere core portlets support English, German, Czech, Polish, Hungarian and Greek.
- Open-source and 100% free! :-)

Grid Middleware Congress



Portlets



- The Portlet Java Specification Request (JSR-168) lays the foundation for a new open-standard for Web portal development frameworks.
- Portlets define an API for building atomic, composable visual interfaces to Web content or service providers
- A portlet provides a “mini-window” within a portal page. Multiple portlets can be composed in a portal page.
- Portlets extend servlets. The idea being to reuse common method signatures.

Grid Middleware Congress



Portlet Implementations



- Still waiting for standardized Portlet API but implementations already exist:
 - Jakarta Jetspeed
 - IBM WebSphere
 - Oracle i9AS Portal
 - BEA WebLogic Portal 7.0 (?)
 - GridSphere...
- Why not Use Jetspeed?
 - Performed an evaluation of Jetspeed 6/02/2002
 - Code not very stable, too many dependencies that are all changing e.g. Turbine, ECS, etc.
 - Large open-source projects have disadvantage that it can be difficult to provide changes
 - Wanted to build Portlet implementation that could meet the needs of the Grid community

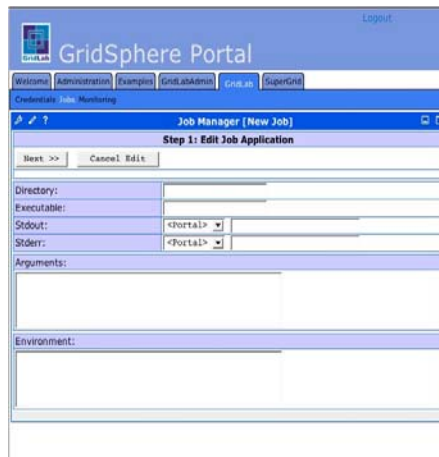
Grid Middleware Congress



Layout & Presentation



- Portlets define how to construct and deliver Web content as modular components within a Web page.
- Portlets can be “maximized” or “minimized” within a Web page.
- Portlets support various modes
 - View, Edit, Help, Configure
- Users can choose to which portlets they want to be “subscribed”.
- Portlets are organized within tabbed panes



Grid Middleware Congress



Portlet Modes



- **View**
 - The standard view of a portlet on a Web page
- **Edit**
 - Allows portlet to capture user-specific parameterization, which leads to personalized view of the portlet
- **Help**
 - A portlet should provide online-help with this mode
- **Configure**
 - Allows a portlet to display its own configuration mode if required
 - Generally, you want to restrict access to this mode

Grid Middleware Congress



Portlet Life-Cycle



- The GridSphere *portlet container* loads and instantiates portlet classes.
- A *application portlet* is initialized with the `init()` method
- A *concrete portlet* is created and destroyed with `initConcrete()` and `destroyConcrete()` methods.
- If a user logs in or logs out, the `login()` and `logout()` methods are invoked
- If a portlet is taken of service temporarily, say while administering it, the portlet container may finish the lifecycle before taking it out of service. Once the administering is done, it may be re-initialized.

Grid Middleware Congress



Portlet Deployment Descriptor



- A Portlet Deployment Descriptor provides the portlet container with portlet configuration information
- Defines a collection of portlet definitions as an XML schema e.g. `portlets.xml`
- Each portlet definition must contain one application portlet definition and one or more concrete portlet definitions
- Application portlet definition defines configuration information valid for a single portlet/servlet for all users
- Concrete portlet definition provides parametrization information for application portlets.

Grid Middleware Congress



Deployment Descriptor (cont)



• Application Portlet Definition:

```
<portlet-app id= portlet class name>
  <portlet-name>portlet name</portlet-name>
  <servlet-name>servlet name</servlet-name>
  <portlet-config>
    <param-name>name</param-name>
    <param-value>value</param-value>
  </portlet-config>
  <allows>
    <maximized/>
    <minimized/>
    <resizing/>
  </allows>
</portlet-app>
```

Grid Middleware Congress



Deployment Descriptor (cont)



• Concrete Portlet Definition:

```
<concrete-portlet-app id= portlet class . # >
  <context-param>
    <param-name>name</param-name>
    <param-value>value</param-value>
  </context-param>
  <concrete-portlet>
    <portlet-name>Portlet Name</portlet-name>
    <default-locale>en</default-locale>
    <language locale="en_US">
      <title>Portlet Title</title>
      <description> Portlet Description</description>
    </language>
    <allowed-access visibility="PUBLIC">
      <role>USER</role>
    </allowed-access>
  </concrete-portlet>
</concrete-portlet-app>
```

Grid Middleware Congress



Portal Presentation



- Portal uses header and double layer tabbed pane to organize content
- Portal layout specified as XML schema:

```
<portlet-tab>
  <title>Examples</title>
  <portlet-tabbed-pane style="sub-menu">
    <portlet-tab>
      <title>Hello</title>
      <portlet-panel>
        <grid-layout>
          <portlet-frame>
            <portlet-class>org.gridlab.gridsphere.portlets.examples.HelloWelt.1</portlet-class>
          </portlet-frame>
        </grid-layout>
      </portlet-panel>
    </portlet-tab>
  </portlet-tabbed-pane>
</portlet-tab>
```

Grid Middleware Congress



Presentation Components



- Layout component library similar to Java AWT or Swing
- Basic layout components:
 - PortletTabbedPane
 - PortletTab
 - PortletFrame
 - PortletTitleBar
 - PortletPanel
 - PortletGridLayout
- Some components act as containers for other components e.g. PortletPanel
- Follows the Composite Design Pattern
- Components are marshalled/unmarshalled to XML using Castor libraries

Grid Middleware Congress



Presentation Tag Library



- GridSphere provides value-added UI JSP tag library
- Goal is to minimize HTML usage
 - UI tags can provide platform independence e.g. support HTML and WML
- Example tag usage in JSP:

```
<%@ taglib uri="/portletUI" prefix="ui" %>
<%@ taglib uri="/portletAPI" prefix="portletAPI" %>
<portletAPI:init/>
<ui:form action="login">
<ui:inputfield name="username" size="8" maxlength="20"/>
<ui:passwordfield name="password" size="8" maxlength="20"/>
</ui:form>
```

Grid Middleware Congress



Presentation tags continued



- Additional “container” tags make it possible to quickly create interfaces that hide CSS/HTML from presentation design

```
<%@ taglib uri="/portletUI" prefix="ui" %>
<%@ taglib uri="/portletAPI" prefix="portletAPI" %>
<portletAPI:init/>

<ui:panel>
  <ui:errorframe beanId="editError"/>
  <ui:frame>
    <ui:tablerow>
      <ui:tablecell width="50%">
        <ui:actionlink action="doSomething"/>
      </ui:tablecell>
      <ui:tablecell/>
    </ui:tablerow>
  </ui:frame>
</ui:panel>
```

Grid Middleware Congress



Visual Bean Model



- For each visual tag, there is a visual bean counterpart that provides access to the visual component from the portlet.
- A visual bean is obtained from a FormEvent object

```
public void doViewUserFiles(FormEvent event) throws PortletException {
    log.debug("in LoginPortlet: doViewUser");
    PortletRequest request = event.getPortletRequest();
    User user = request.getUser();
    ListBoxBean lb = event.getListBoxBean("filelist");
    lb.clear();
    String[] list = userStorage.getUserFileList(user);
    ...
}
```

Grid Middleware Congress



Portlet Services



- **PortletService** is the base interface for all portlet services.
- PortletService classes define “reusable services” within the context of a Portal.
- Portlets can obtain portlet service instances by calling the method `PortletContext.getService`.
- PortletService instances are created by a **PortletServiceFactory**.
- Similar to Portlets, PortletService objects are configured at initialization with a **PortletServiceConfig** object.

Grid Middleware Congress



Portlet Service Security



- Extended the service model to support “**user services**”
- User service model allows services to provide capabilities that are specific to a users role
- Portal security uses role based access control (RBAC) model
- Groups contain Users which have Roles
- Roles can be Guest, User, Admin, and Super
- Currently, Users can have only one predefined role within a group but this can be extended in future releases
- Group and role information available from the AccessControlManagerService

Grid Middleware Congress



Portlet Services Descriptor



```
<portlet-services>
  <service>
    <name>Portlet Manager Service</name>
    <user-required>true</user-required>
    <description>Provides Administration Capabilities for Portlet Web Applications</description>
    <interface>org.gridlab.gridisphere.services.core.registry.PortletManagerService</interface>
    <implementation>org.gridlab.gridisphere.services.core.registry.impl.PortletManagerServiceImpl</implementation>
    <service-config>
      <param-name>coreContext</param-name>
      <param-value>gsmanager, coreportlets, gridportlets, exampleportlets</param-value>
    </service-config>
  </service>

  <service>
    <name>Login Service</name>
    <user-required>true</user-required>
    <description>Provides Login Capabilities</description>
    <interface>org.gridlab.gridisphere.services.core.user.LoginService</interface>
    <implementation>org.gridlab.gridisphere.services.core.user.impl.LoginServiceImpl</implementation>
  </service>
```

Grid Middleware Congress



Core Services



- **Portlet Manager Service**
 - Provides lifecycle methods to allow portlets to be installed, removed, initialized and destroyed by authorized users
- **Login Service**
 - Allows a User to be retrieved from a username and password
- **User Manager Service**
 - Add/Remove User Accounts
 - Edit User Profiles
- **Access Control Service**
 - Add/Remove User Groups
 - Add/Remove User Roles

Grid Middleware Congress



Core Portlets



- **Login Portlet**
 - Enables user to logon
 - Pluggable authentication modules with which we provide support for database password and credential based logins.
- **Account Request Portlet**
 - Enables new users to request an account and existing users to modify their accounts
 - Users may request to join one or more portlet groups.
- **Account Management Portlet**
 - Enables admins to manage user accounts.
- **Access Control Manager Portlet**
 - Enables admins to define new groups (and potentially roles) and to specify which users belong to which groups and roles.

Grid Middleware Congress



Core Portlets (continued)



- Portlet Configuration Portlet
 - Enables admins to configure portlets
- Portlet Subscription Portlet
 - Enables users to subscribe to portlets
- Portlet Layout Portlet
 - Enables users to configure the presentation of portlets

Grid Middleware Congress



Grid Services (the beginning)



- Credential Manager Service
 - Add/Remove allowed User Credentials
 - Configure use of Credential Retrieval Service
- Job Manager Service
 - For listing, starting, migrating, stopping jobs.
- Job Monitoring Service
 - Specify what to monitor for any given job and archive related information.
- File Transfer Service
 - For managing and scheduling file transfers.
- Data Manager Service
 - Access to data replica catalogues.
 - Describe data with meta-data.
- Notification Service
 - Define events to be notified about.
 - Specify how to be notified about those events.

Grid Middleware Congress



Grid Portlets



- **Credential Administrative Portlets**
 - Admins can specify what credentials are permitted for use.
 - Admins can specify mappings between credential subjects and user accounts, as well as mappings to particular resources.
 - Admins can view active credentials and their usage online.
- **Credential User Portlets**
 - Users may request new credential mappings to their accounts.
 - Users may retrieve and refresh credentials for later use.
- **Resource Management Portlets**
 - Admins can specify and describe Grid resources.
 - Provides tools for discovering resources on the Grid.
 - Provides tools for tracking requests made to site admins for configuring or updating resources with software, etc.

Grid Middleware Congress



Grid Portlets (continued)



- **Job Management Portlets**
 - Users can submit or migrate jobs, get job status and so forth.
 - Provide access to job monitoring services where available.
 - Online reservation for resources for job submission.
- **Data Management Portlets**
 - Provides users with tools for transferring and managing files.
 - Online data catalogue and replica interface.
- **Notification Portlets**
 - Users can specify where and how they want to be notified about particular events, such as when a job completes.
 - Generally, however, most portlets will provide notification options...

Grid Middleware Congress



Groupware Portlets



- We are pursuing a number of ideas for enabling users to collaborate online, some of those ideas include...
- Messaging portlets for using Email or SMS.
- Chat portlets (Jabber-based, requiring Java perhaps)
- Calendar and scheduling portlets, including ability to publish future downtime of Grid machines.

Grid Middleware Congress



Third-party portlet development



- GridSphere supports the notion of “plug-and-play” portlets
- Portlets are packaged as WAR files
- WAR contains:
 - Portlet.xml describing portlet capabilities
 - Layout.xml describing layout configuration
 - Portlet class files and associated JSP pages for presentation
- Portlets can be deployed dynamically at run-time by authorized users
- Portlet access can be dynamically administered by authorized users
- When Portlet API is standardized, communities can trade portlets
- Portlet registry to be hosted at GridSphere project web site www.gridisphere.org
- Developers can focus on building and re-using existing portlets to meet specific portal community requirements

Grid Middleware Congress



External Collaborations



- Working to support GridSphere for the general Grid portal community
- Collaborations with folks from Canadian National research Council (NRC) to develop grid data transfer portlets using GridFTP
- Collaborations with SZTAKI research center in Budapest to use GridSphere to support users of SuperGrid and ClusterGrid projects
- Collaboration with physicists at Albert-Einstein-institute in Berlin to develop portlets for studying numerical relativity.
- Collaborating with the Global Grid Forum (GGF) on enhancing the usefulness and awareness of portlets in the Grid Computing Environments WG.
- Interested in promoting GridSphere and Portlets to more communities! Please e-mail us if interested!

Grid Middleware Congress



Conclusion



- Portals provide a usable high level environment for end users
- Do not write a portal from scratch!
- The GridSphere portlet model supports the concept of Virtual Organizations and allows VO's to plug into the portal framework
- Portlets provide reusable code that can be shared between various communities
- Portlet developers can focus on delivering end functionality instead of low level details
- GridSphere Project site <http://www.gridisphere.org>

Grid Middleware Congress



References



- GridSphere Project
 - <http://www.gridsphere.org/>
- Astrophysics Simulation Collaboratory Web site:
 - <http://www.ascportal.org>
- Grid Portal Development Toolkit:
 - <http://www.doesciencegrid.org/Projects/GPDK>
- Portlet Specification:
 - <http://www.icp.org/en/jsr/detail?id=168>
- Jakarta JetSpeed:
 - <http://jakarta.apache.org/jetspeed>
- Castor Project:
 - <http://castor.exolab.org>
- Cactus Project:
 - <http://www.cactuscode.org>
- Globus Project:
 - <http://www.globus.org>
- Jakarta JetSpeed Portlet Tutorial:
 - <http://www.bluesunrise.com/jetspeed-docs/JetspeedTutorial.htm>
- Oracle i9AS Portal:
 - <http://otn.oracle.com/products/portal/>
- WebSphere Portal API:
 - <http://www7b.software.ibm.com/wsd/d/zones/portal/portlet/4.1api/>