Grid Portal Development

Authors: Josh Nedelka & Matthew Reiter
Worcester Polytechnic Institute
The P-GRADE Portal

- Web-based portal allowing access to multiple Grids
- Removes technical barriers for many users
- Provides utilities to easily create and execute workflows
- Recently made open source
Project Statement

- Analysis, design and implementation of two features for the P-GRADE Grid Portal
  - User account creation system
  - Notification system
- Systems serve to enhance a user’s experience with the Portal.
Project Statement

- User Account Creation System
  - Administrator must manually create user accounts
  - Task is to automate account creation
  - Administrator must approve account requests

- Email Notification System
  - Currently users must manually check statuses
  - Task is to integrate system that monitors workflows for status changes
  - When changes occur, send notifications
Account Creation System Implementation

- Several implementations considered
- On web server:
  - Centralized account request form
  - Administration console
- Actual account creation done on portal servers

Impl 1: mostly on web server
Impl 2: combination
Impl 3: entirely on portals
Account Creation System Data Flow

1. User submits an account request form to the Web server.
2. The account request is processed by the Portal server.
3. The Portal server checks the account creator in the Portal database.
4. The user's request is then processed by the Web server.
5. The account creator checks the account in the Common database.
6. The account creator updates the Common database.
7. The account creator updates the Portal server with the account information.
8. The Portal server updates the Portal database.
9. The updated account information is then sent back to the user.
Account Request Form

- Form for user to submit account requests
- Sends requests to admin for approval
- Image confirmation system to prevent spam
Account Request Verification

- Email sent to admin
- Three options:
  - Accept
  - Deny
  - Accept with defaults
- Links to account request dispatcher
Account request dispatcher

- Part of admin console
- Forwards account requests to portals
- Displays account creation results
- On failure, provides option to retry

Account created on SEE-GRID.
The account request has been successfully processed.

Failed to create an account on SEE-GRID!
The account request was not successfully processed.
Try Again
Account Creation Components on Portal

- Account Creation Servlet
  - Handles account creation and verification
- Information Query Servlet
  - Handles requests for user/role/group information
- Account Creator Service
  - Interfaces with built-in services
- Authentication Service
  - Verifies permission to use servlets
Administration Console

- Common login
- Provides access to:
  - User information exporter
  - Settings manager
  - Grid info editor

![Login page](image-url)
User Information Exporter

- Allows admin to download user information from all portals
- Uses CSV format for compatibility
Settings Manager

- Interface for changing settings:
  - Image confirmation
  - Authentication
  - User name
  - Password
  - Email address
Grid Information Editor

- Add/remove grids
- Edit grid information
  - ID
  - Name
  - URL
  - Servlet credentials
  - Account creator credentials
Administrator Manual

- Provides an overview of the system
- Instructions for:
  - Installation
  - Admin console
Security

- Key requirement: prevent unauthorized account creation
- Only administrator can approve account requests
- Administrator must authenticate with web server
- Web server must authenticate with portal servers
- Servlet and account creator credentials stored in encrypted form
- Protection against SQL injection attacks
Notification System Requirements

✓ Sends notification of status changes for normal or parameter study workflows
✓ User can specify when to receive notifications:
  ✓ On workflow end
  ✓ On any status change
✓ Handles aborting, rescuing and resubmitting of workflows
✓ Provides an easy to use interface for implementing new ways of notification
Notification System Analysis

- Considered two main implementations
  - Event Driven – handle notification according to the internal events in the system
  - Polling – constantly query the system for changes
- Event Driven chosen for the system
Notification System Implementation

- Contains three distinct phases:
  
  1. Integrate notification portlet
  2. Duplicate GUI Functionality from WS-PGRADE
  3. Backend Integration
Phase 1: Integrate Notification Portlet

- Rebuilt notification portlet from source
- Added to layout of P-Grade portal under “Workflows”
Phase 2: Duplicate GUI

Functionality from WS-PGRADE

- WS-PGRADE Portal asks the user when they would like to be notified when submitting a workflow.

- Our implementation behaves similarly.

![Image of GUI showing notification options](image-url)
Phase 2: Duplicate GUI
Functionality from WS-PGRADE

- Considerations:
  1. When should the user be allowed to choose this setting?
     - On submit
     - On rescue
     - Anytime
  2. How to handle “Submit All”
     - Choose a setting for each workflow
     - Single setting for all workflows
  3. How to handle Parameter Studies
     - Allow notifications for each workflow
     - Only allow for main workflow
Phase 3: Backend Integration

Workflow started

1. Update Statuses
2. Find Workflow Status
3. Send Email

Workflow

Job A
Status = Run

Job B
Status = Init

Job C
Status = Init

List of Job Statuses

Notification System

Email Plugin

Job Status Changed!
Phase 3: Backend Integration

- Analysis of Design
  - Disadvantages:
    - Has to connect to the existing portal in multiple places
    - Lots of code overlap with existing utilities and classes
  - Advantages:
    - Less overhead and more accurate than using polling to determine statuses
- Should prove capable of handling future improvements and enhancements
Account Creation System Testing

- Functional Testing
  - Part 1: local testing
    - Done continuously throughout implementation
    - Requested/approved accounts and logged in to verify success
    - Changed settings in admin console and verified results
  - Part 2: production testing
    - Sent code to webmaster and portal developer for installation
    - Administrator conducted manual testing to ensure correct operation and functionality
Email Notification Testing

- Necessary to determine how the portal’s performance would be affected by the new system
- Performance was tested in two ways:
  1. Scalability – how well does the system perform as more and more workflows are added?
  2. Versus a baseline portal – what performance impact does the notification system have compared to a portal without it?
Portal Performance: 1 Workflow

With Notification

Without Notification
Portal Performance: 20 Workflows

Java Process Statistics: 20 Workflows with Full Notification

Java Process Statistics: 20 Workflows with No Notification

With Notification

Without Notification
Portal Performance: 50 Workflows

Java Process Statistics: 50 Workflows with Full Notification

Java Process Statistics: 50 Workflows with No Notification

With Notification

Without Notification
Current State

- Both systems integrated into the P-GRADE Portal
  - All features successfully tested by the LPDS
- Can be downloaded now from the P-GRADE SourceForge project
Future Work: Account Creation

- Bulk Account Creator
  - Reduces time to create multiple users
- Improved Database Support
  - Support for more databases
- Account Request Viewer
  - List of pending requests
- Testing of Portal Information
  - Simplifies verifying that information in portal info editor is correct
Future Work: Notification System

• Additional Plugins
  • SMS Messaging
  • Phone Call Messaging

• More Monitoring
  • Finer granularity
  • More types of monitoring
  • Parameter Study Workflows
Acknowledgements

- **MTA SZTAKI**
  - Ákos Balaskó
  - Zoltán Farkas
  - Károly Göschl
  - Gábor Hermann
  - Péter Kacsuk
  - Ádám Kornafeld
  - Miklós Kozlovszky
  - András Schnautigel
  - Gergely Sipos

- **WPI**
  - József Patvarczki
  - Gábor Sárközy
  - Stanley Selkow
Grid Portal Development

Authors: Josh Nedelka & Matthew Reiter