NAREGI PSE with ACS

S.Kawata\textsuperscript{1}, H.Usami\textsuperscript{2}, M.Yamada\textsuperscript{3},
Y.Miyahara\textsuperscript{3}, Y.Hayase\textsuperscript{4}, S.Hwang\textsuperscript{2}, K.Miura\textsuperscript{2}

\textsuperscript{1}Utsunomiya University
\textsuperscript{2}National Institute of Informatics
\textsuperscript{3}FUJITSU Limited
\textsuperscript{4}Toyama College
National Research Grid Initiative (NAREGI)

- What is NAREGI?  
  [http://www.naregi.org/index_e.html](http://www.naregi.org/index_e.html)

- **Research Themes in NaReGI**
  - Lower and Middle-Tier Middleware for Resource Management  
    (Scheduler, Broker, Auditing, Accounting, Grid VM)
  - Grid Programming, Middleware  
    (Grid RPC, Grid MPI)
  - User-Level Grid Tools & PSE  
    (Workflow GUI, Visualization Tools, PSE)
  - Packaging and Configuration Management
  - Networking, Security & User Management  
    (Traffic Measurement, Optimal Routing Algorithms, Robust TCP/IP Protocols)
  - Nanoscience Applications  
    (Parallel Structure, Granularity, Resource Requirement, Coupled Simulation)
What is NAREGI PSE?

- Provide a framework to distribute user applications on grid
  - Users can register, deploy and retrieve applications by using NAREGI PSE.
  - Application developers distribute their applications to research communities without a hard task.
  - Application users do not want to care about grid for using application.

- Focus on a legacy application
  - Deploy application binaries for specific targets
  - Compile from source program, if needed
  - Assume that target system has a grid middleware such as Globus(2.x functions), UNICORE, NAREGI, etc.
Current system structure of NAREGI PSE

- Application Pool
- Register (GUI)
- Deployment (GUI)
- Retrieve (GUI)
- Deployment Service
- Application Info. (App. Schema)
- Resource Info. (cached)
- Get resource information
- File transfer
- Compile
- Register deployment information

Client

WF execution
WF edit
WFT
submit
Application pool

A key component of NAREGI PSE:

- Repository for applications and the information

Application pool contains the followings:

- source files and/or the executables for specific targets
- system requirements for execution
- deployment procedure (transfer files, configure, build, etc.)
- compile procedure (script, makefile, etc.)
- input/output files for execution
- deployment information
- etc.
Our next approach using ACS

- Implement Application pool using Application Repository Interface and Application Archive Format
- Separate an application body and the information as the meta-data stored in the current Application Pool
  - Register an application body to Application Repository of ACS
  - Register an application information to Information Services
- Application retrieve function will be implemented by sending a query to Information Services
  - VO, access control for application, etc. will be supported by using Information Services.
Next NAREGI PSE with ACS

NAREGI PSE

Register-UI

Deploy-UI

Application Contents Service (ACS)

Application Repository

Deployment Service
- compile
- deploy
- un-deploy

Client

Contents (WF)

NAREGI WFT

Register

Register deploy info.

NAREGI IS

NAREGI SS

Register application info.

GetContent

Deploy

Submit
Application Pool using Application Repository

Application Pool

Application Repository

Application Archive (AA)

Signature

Application contents

Source programs, Program binaries, Configuration data, Deployment procedures, etc.

Large initial data, Output data, etc.

Resource requirement (JSDL)

AA Descriptor

refers
NAREGI components and OGSA services

WP3: WFT

WP3: PSE

Application Content Service (ACS)

Deployment

Register / Query

WP1:

Job Manager

Submit

Discover & Select

Execution Planning Services

Reserve

Reservation

Candidate Set Generator

Service Container

Register

Update

Accounting Services

Information Services

Query
Issues

- Suitable for Application Archive?
  - Not only static contents in NAREGI PSE
    - e.g. compiled binaries for multiple specific targets, output files, etc.

- Application structure is not so simple.
  - How do we express a set of single-applications such as coupled-simulations on heterogeneous resources?

- There may be some issues besides these!

An example for A Coupled Simulation:

// Application A on VM1
// Application B on VM2
Application C on VM3

time

-> A hierarchical structure may be required in AA:
Summary

- We have developed NAREGI-PSE without ACS at present.

- We are planning to apply ACS to NAREGI PSE.

- We are looking forward to seeing & having the ACS specification, including Dynamic data:
  - compiled binaries for multiple specific targets, output files, History of executions, etc.