Implementing HPC Basic Profile
SMOA Computing approach

Mariusz Mamoński
Krzysztof Kurowski

{krzysztof.kurowski,mamonski}@man.poznan.pl
OpenDSP - Retrospection

In the mid 2005 at Poznan Supercomputing and Networking Center OpenDSP project was launched aiming to:

• give consistent, remote, multi-user access to various DRM systems using standardized DRMAA interface
• add Authentication, Authorization and Accounting layers, which are out of scope DRMAA specification
• use JSDL as the job description language
• provide high performance: C + gSOAP + DRMAA C binding (which usually relays on internal DRMS API)
OpenDSP - Example Performance Tests

Concurrent submissions response time

Time (s)

Concurrent clients

POZNAŃSKIE CENTRUM
SUPERKOMPUTEROWO-SIECIOWE
OpenDSP

- Open DRMAA Service Provider (OpenDSP) successfully demonstrated in the inteliGrid EU project
- More than 300 downloads all together (versions 0.1, 1.0 and 1.1) from sourceforge.net
- Many users and deployments, e.g.:
- Tele-HVEM Seoul National University project
- Successfully tested with SGE, Condor, PBS/Torque, Condor and Xgrid DRMAA libraries
OpenDSP

• Open DRMAA Service Provider (OpenDSP) started in 2006 back in the inteliGrid EU project
• More than 300 downloads all together (versions 0.1, 1.0 and 1.1) from sourceforge.net
• Many users and deployments, e.g.:
  • Tele-HVEM Seoul National University project
  • Successfully tested with SGE, Condor, PBS/Torque, Condor and Xgrid DRMAA libraries
From OpenDSP to SMOA Computing

• migration from OpenDSP specific Web Service interface to the open OGSA-BES (HPC BES Basic Profile) OGF standard

• refined architecture:
  • privilege separation instead of setuid binaries
  • JSDL used as the only format for internal job representation

• support for modules written in Python

• more extensions points added

• exposed Advance Reservation capability of underlying DRMS (LSF, SGE)

• file staging support as the part of job life cycle

• separate interface for lightweight, direct file staging (SOAP attachments)
Authentication and Authorization

- Transport level security:
  - none (plain http)
  - SSL (with or without client authentication)
  - GSI

- Message level security:
  - WS-Security Username Token Profile
  - WS-Security SAML Token Profile (Bearer)
  - WS-Security X509 Token Profile

- Authorization modules:
  - Globus like mapfile
  - Anonymous (fixed mapping/jsdl-hpc:UserName)
  - Connection rate limit (per user/IP)
SMOA Computing - Architecture
Accounting

• SMOA Computing stores all vital information in database:
  • submitted/post processed JSDL documents (getActivityDocuments)
  • local user account name (getActivityDocuments)
  • exit status (getActivityStatuses)
  • job resource usage information
JSDL to DRMAA mapping

- `<jsdl:JobName>` ➞ DRMAA_JOB_NAME
- `<jsdl-hpcpa:Executable>` ➞ DRMAA_REMOTE_COMMAND
- `<jsdl-hpcpa:Argument>)*` ➞ DRMAA_V_ARGV
- `<jsdl-hpcpa:Environment>)*` ➞ DRMAA_V_ENV
- `<jsdl-hpcpa:WorkingDirectory>` ➞ DRMAA_WD
- `<jsdl-hpcpa:Input>` ➞ DRMAA_INPUT_PATH
- `<jsdl-hpcpa:Output>` ➞ DRMAA_OUTPUT_PATH
- `<jsdl-hpcpa:Error>` ➞ DRMAA_ERROR_PATH
- other JSDL elements (e.g. TotalCpuCount) are mapped to DRMAA_NATIVE_SPECIFICATION by dedicated SMOA Computing JSDL Filter module
State model

- **Cancelled**
- **Pending**
  - Stage-in
  - Queued
  - Held
- **Running**
  - Executing
  - Stage-out
  - Suspended
- **Finished**
- **Failed**
Advance REServation Factory

- CreateReservation
- GetReservationStatus
- GetReservationDocument
- TerminateReservation
- GetActiveReservations

```
<ardl:ReservationDefinition xmlns:ardl="http://schemas.smoa-project.com/comp/2009/01/ardl">
  <ardl:TimeWindow>
    <ardl:StartTime>2009-03-21T21:00:00+02:00</ardl:StartTime>
    <ardl:EndTime>2009-03-21T22:00:00+02:00</ardl:EndTime>
  </ardl:TimeWindow>
  <ardl:Resources>
    <ardl:ReservedSlotsCount>1</ardl:ReservedSlotsCount>
  </ardl:Resources>
</ardl:ReservationDescription>
```
Users

- Developed in some EU projects, e.g. QosCosGrid, BREIN, BeinGRID to support specific user requirements
- Deployed and tested by many users in ongoing infrastructure projects, such as PL-Grid and MAPPER
- External users and independent resource providers interested in productive SMOA Computing deployments
- More meetings and workshops are planned
SMOA Computing - Summary

- **elastic** - dynamically loadable modules
- **efficient** - preliminary tests (with SSL authentication) shown ability to handle 30 job submit requests per **second**
- **standard based** - support for OGF DRMAA and BES makes the maintenance and 3rd party integration easier
- **simple and easy to install** - guides and docs
- **useful interfaces** - distributed together with various client tools and interfaces
SMOA Project

Other SMOA Project components:
• SMOA Core - common library
• SMOA Notification - open implementation of the WS-Notification 1.3 OASIS specification
• SMOA Devices - remote management and monitoring of devices

http://gforge.man.poznan.pl/gf/project/smocomp/
http://gforge.man.poznan.pl/gf/project/smoa-ntf/
http://gforge.man.poznan.pl/gf/project/smoadvices/
Thank you