Project no: **RI031844-OMII-Europe**

Project acronym: **OMII-Europe**

Project title: **Open Middleware Infrastructure Institute for Europe**

Instrument: **Integrated Infrastructure Initiative**

Thematic Priority: **Communication network development**

D:JRA1.14 – The Second Yearly Report on the Component Exchange Task

Due date of deliverable: 30 March 2008
Actual submission date: 07 May 2008

Start date of project: **1 May 2006** Duration: **2 years**

University of Southampton

Revision [Final 1.0]

Project co-funded by the European Commission within the Sixth Framework Programme (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

Document Control Sheet

Document	**Title**: The Second Yearly Report on the Component Exchange Task	
	ID: D:JRA1.14	
	Version: 1.0	**Status**: Final
	Available at: http://omii-europe.org	
	Software Tool: Microsoft Word 2003	
	File(s): D.JRA1.14.doc and D.JRA1.14.pdf	
Authorship	**Written by:**	Stephen Crouch (SOTON) Gary Li (SOTON)
	Contributors:	All OMII-Europe partners
	Reviewed by:	Moreno Marzolla (INFN) Fredrik Hedman (KTH)
	Approved by:	Technical committee

Document Status Sheet

Version	Date	Status	Comments
0.1	10 March 2008	Draft	First draft
0.2	9 March 2008	Draft	Revisions
0.3	14 April 2008	Final draft	Revisions to address comments
1.0	25 April 2008	Final version	Minor revisions

Executive Summary

The JRA1 Component Exchange activity was set up to facilitate a two-way exchange of components between OMII-Europe and our project partners in China. This final report provides a complete description of the approach to the work undertaken, activity achievements and experiences for the 24M duration of this activity within OMII-Europe.

Originally, this work entailed a total of 6.0 staff years across the 24M duration, between the University of Southampton (SOTON), the Institute of Computing Technology (ICT), the Computer Network Information Center, China (CNIC) and Tsinghua University (TU), with SOTON supplying the only funded effort. During the course of the collaboration, a restructuring of effort was required to ensure positive outcomes from this activity and results suitable for submission into the OMII-Europe repository.

This work has firstly resulted in the identification of a set of European grid components for potential porting to the Chinese CROWN grid infrastructure, and the porting of two of these: OGSA-DAI in collaboration with the JRA1 Database activity and GridSAM, an OMII-UK job execution component. Secondly the work has resulted in the delivery to the OMII-Europe repository of two components, each from a key Chinese grid infrastructure: DDS from VEGA-GOS developed by the Institute of Computing Technology (ICT), and the CROWN Meta-Scheduler developed by Beihang University (BU). Thirdly, an outcome of the mutual evaluation was that both BU and ICT have adopted the OGSA Basic Execution Service (OGSA-BES) standard within their own Grid infrastructures.

Contents

1. Introduction

This deliverable describes the achievements, development process and experiences of the JRA1 Component Exchange activity within OMII-Europe.

The primary motivation of the JRA1 Component Exchange activity, as described in the Description of Work [DOW] page 85, is to 'greatly increase the user base of components coming from both OMII-Europe and from CNGrid'. To this end, the overall goal of the activity is to facilitate an exchange of components between OMII-Europe and the activity partners in China.

1.1. Description of Work

As described in the Description of Work [DOW] page 85, the following tasks were to be undertaken to achieve this goal:

1. An analysis of the components available in European grid infrastructures to determine suitable candidates for porting to CNGrid and vice versa.
2. Porting Grip from CNGrid to OMII-Europe grid infrastructures for submission to the OMII-Europe repository.
3. Porting Agora from CNGrid to OMII-Europe grid infrastructures for submission to the OMII-Europe repository.

CNGrid represents an amalgam architecture comprising the VEGA-GOS and CROWN grid infrastructures provided by ICT and Beihang respectively, although for Component Exchange, OMII-Europe is only concerned with the components within the constituent VEGA-GOS and CROWN elements. Within VEGA-GOS, Agora is a module for the management of policy and context governing the interaction between grid users and resources, whilst Grip is a grid-level task management module.

However, due to a number of resourcing issues encountered during the activity, and explained in this deliverable, the original goals had to be refactored. This is explained in detail in Section 2.

1.2. Deliverables and Milestones

Each of the above identified tasks has a specific milestone, in addition to two deliverables. These are shown in Table 1.

Milestone:	**Designation:**	**Delivery:**
Components for porting to CNGrid identified	M:JRA1.14	M12
Grip ported to the OMII-Europe repository	M:JRA1.15	M12
12M Report on the Component Exchange Activity	D:JRA1.13	M12
Agora ported to the OMII-Europe repository	M:JRA1.16	M24
24M Report on the Component Exchange Activity	D:JRA1.14	M24

Table 1. Planned deliverables and milestones for the JRA1 Component Exchange activity

1.3. Standards Adopted by this Activity

The standards adopted by this activity are shown in Table 2.

Standard:	Version(s):	Description:	Applied to:
OGSA-BES	26, 33, 1.0 Final	OGSA Basic Execution Service – describes a service interface for job submission	BU CROWN Meta-Scheduler
JSDL	1.0	Job Submission Description Language – specification of a job's computation and data requirements	
HPCProfile	1.0	High Performance Computing Profile – a specification for the HPCProfileApplication extension within JSDL	
OSGi	R4	A specification for modularised, reusable application components	ICT Dynamic Deploy Service (DDS)

Table 2. Standards adopted by the JRA1 Component Exchange activity

2. Overview of Partners and Effort

Originally, the activity was planned to be a collaboration between the following institutions:

- Southampton University, UK (SOTON, activity coordinator)
- Institute of Computing Technology, China (ICT)
- Tsinghua University, China (TU)

The effort from the Chinese partners is entirely unfunded, and therefore the collaboration was conducted on a mutually agreed support and benefit basis.

However, due to a number of issues addressed in this section, the landscape of the collaboration and the partners involved had to be restructured twice over the duration of the activity. Despite this, the activity achieved positive outcomes consistent with the original goal of exchanging components.

2.1 Effort Restructure 1

Following the successful completion of the mutual component analysis task described in Section 4.1, the planned unfunded effort from ICT and TU dropped considerably during this activity in the first year due to scheduling and resourcing issues within their organisations. In addition to creating a resourcing issue, this represented a major concern with the activity's ability to achieve its initial objectives of porting Grip and Agora to the OMII-Europe platforms. A further obstacle to these efforts was that VEGA-GOS, the infrastructure which includes Grip and Agora, was undergoing a major refactorisation as a natural evolution of the product in this phase at ICT.

However, this potential risk in JRA1 activities was identified in the Description of Work [DOW] page 90, which states a restructuring of effort can be applied to ensure positive outcomes and results suitable for placement in the repository. Interest from BU, not originally involved in Component Exchange, to participate in this activity as an unfunded partner enabled this restructuring of effort.

Utilising the results from the analysis of the CROWN grid infrastructure task, the CROWN Meta-Scheduler component was identified as a suitable candidate for exchange. The Meta-Scheduler schedules jobs across multiple CROWN NodeServer job submission systems based on a scheduling policy. This led to a refactoring of the activity goals into the following:

1. An analysis of the components available in European grid infrastructures to determine suitable candidates for porting to CNGrid and vice versa.
2. Porting the Meta-Scheduler from CROWN to OMII-Europe grid infrastructures for submission to the OMII-Europe repository.

Following this agreement of refactored goals, in addition BU and SOTON signed a Memorandum of Understanding to formalise the willingness to foster further collaboration between the two institutions.

2.2 Effort Restructure 2

During the second year, following the resolution of their resourcing issues, ICT expressed an interest to resume their unfunded participation in the activity. A visit to ICT at project M14 by SOTON led to a re-analysis of suitable ICT components for exchange, which is covered in more detail in Section 6.2. The VEGA-GOS component selected for exchange was the Dynamic Deploy Service (DDS). DDS provides a web service that enables OSGi-compliant web services to be deployed and undeployed on a container dynamically, without the need to restart the container. Hence, a second restructuring was undertaken to partially accommodate the initial goal of the activity:

1. An analysis of the components available in European grid infrastructures to determine suitable candidates for porting to CNGrid and vice versa.
2. Porting the Meta-Scheduler from CROWN to OMII-Europe grid infrastructures for submission to the OMII-Europe repository.
3. Porting the Dynamic Deploy Service (DDS) to OMII-Europe grid infrastructures for submission to the OMII-Europe repository.

2.3 Overall Effort Summary

A summary of effort for the second year is given in Table 3.

	Organisation							
	SOTON		**BU**		**ICT**		**TU**	
Effort	**F(0.5)**	**U(0)**	**F(0)**	**U(0)**	**F(0)**	**U(4.0)**	**F(0)**	**U(1.0)**
May 07	0.5	2.0	0	2.2	0	0	0	0
Jun 07	0.5	0.8	0	1.7	0	0	0	0
Jul 07	0.5	0.5	0	1.7	0	0	0	0
Aug 07	0.5	0.25	0	0.1	0	1.5	0	0
Sep 07	0.5	0	0	0.5	0	1.75	0	0
Oct 07	0.5	0	0	0.1	0	2.0	0	0
Nov 07	0.5	0.1	0	1.5	0	1.2	0	0
Dec 07	0.2	0.2	0	2.8	0	2.0	0	0
Jan 07	0.5	0.7	0	2.0	0	0.5	0	0
Feb 07	0.5	0.5	0	1.0	0	0.7	0	0
Mar 07	0.5	0.4	0	0.4	0	1.5	0	0
Apr 07		0	0		0		0	
Total	5.2	5.45	0	14.0	0	11.15	0	0

Table 3. Summary of overall effort in the JRA1 Component Exchange activity

3. Development Process

This section will provide an overview of the overall development process adopted to achieve the activity goals, covering external and internal perspectives. The internal perspective covers the process adopted by Component Exchange partners with the aim to produce the software, whilst the external perspective encompasses the interactions with other OMII-Europe activities necessary to deliver quality components to the OMII-Europe repository.

3.1. Internal Process Perspective

This process was first developed, evolved and enacted with the BU collaboration. This enabled collaboration with ICT to progress more efficiently by following this process, with both partners aware of what is required throughout the activity.

This perspective of the process is given in Figure 1.

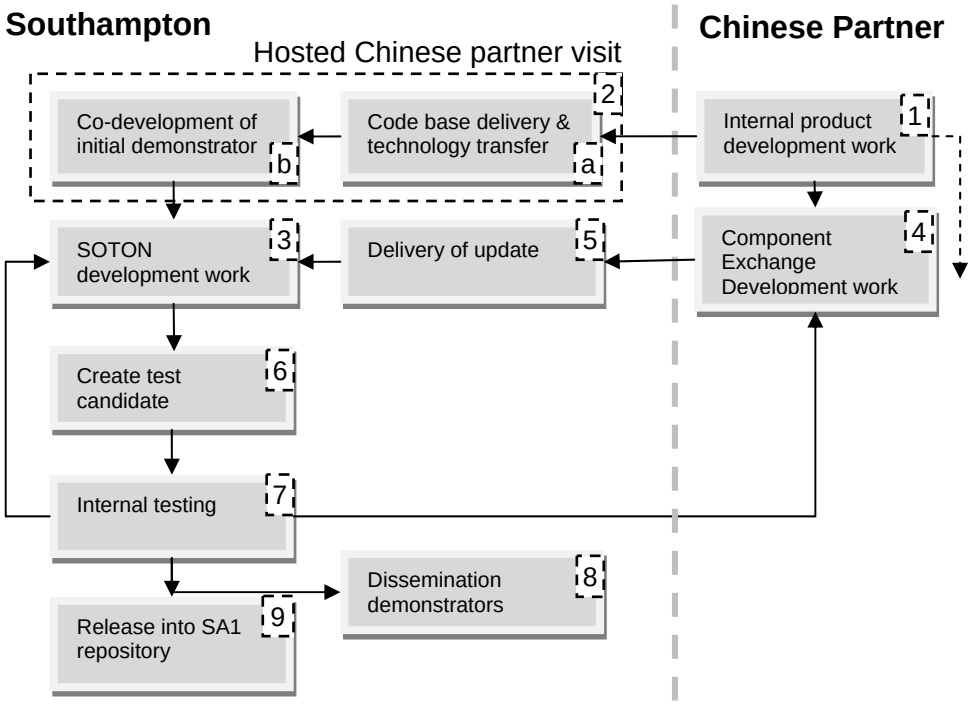


Figure 1. Internal perspective of the software development process adopted by JRA1 Component Exchange

The stages of the process are as follows:

1. Chinese partner conducts normal internal development work on their software.
2. An initial visit of the Chinese partners to Southampton with two key objectives:
    a. Delivery of partner product code base to Southampton.
    b. Co-development of initial proof-of-concept demonstrator for dissemination at an event.
3. Iterative development of the code base into a product by SOTON.
4. Supportive product development by Chinese partner utilising their product expertise.
5. Delivery of partner development work as updates subsumed into SOTON development work.

6. Creation of internal test candidate from development code base. This is motivated by either a formal requirement for a scheduled release, or an informal requirement for a demonstrator for an event.
7. Internal Component Exchange testing of test candidate at SOTON. This stage represents a quality gate, with two possible outcomes:
    a. If the candidate is deemed at the appropriate level of quality for intended purpose (demonstrator and/or release), the process proceeds.
    b. Issues with test candidate are fed into iterative SOTON development work (5) and partner development work (4) for resolution prior to a future test candidate.
8. The candidate is informally prepared for use as a demonstrator for event dissemination purposes.
9. The candidate is formally released into the SA1 repository for testing by SA2, with issues fed back to SOTON development work.

3.2. External Interaction Perspective

This activity interaction perspective is given in Figure 2. For clarity, only those activities that contributed to the development work itself are included.

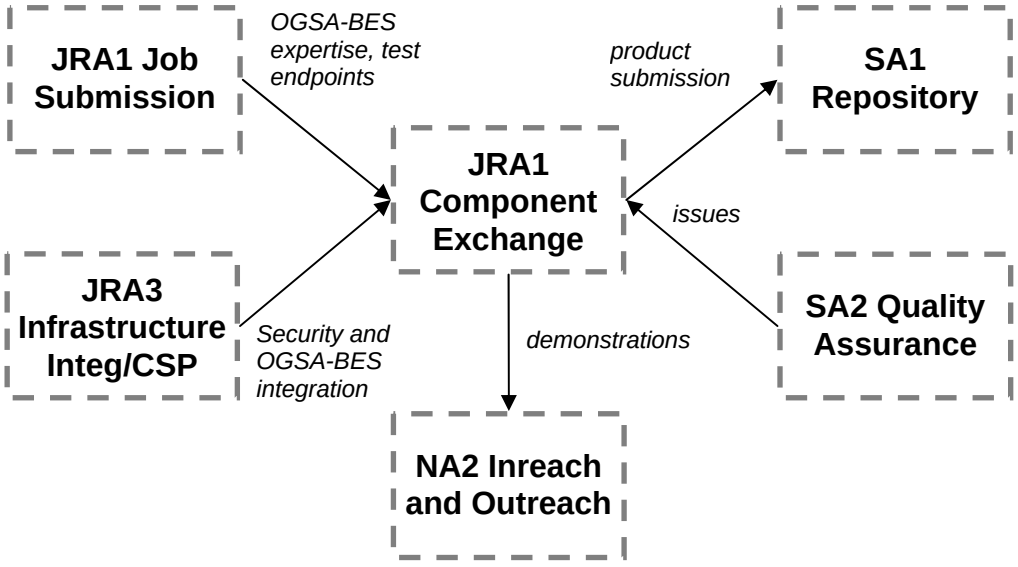


Figure 2. External perspective of the primary interactions between JRA1 Component Exchange and the other OMII-Europe activities

These primary interactions were as follows:

- **JRA1 Job Submission:** this was the major JRA1 contributor to this activity. They supplied test OGSA-BES endpoints for the OMII-Europe platforms for the benefit of the Meta-Scheduler development, and expertise of these implementations required for ensuring interoperability.
- **JRA3 Infrastructure Integration/Security:** provided the development of the Common Security Profile and OGSA-BES integration expertise.
- **SA1 Repository:** hosting of product submissions and SA2 reviews.
- **SA2 Quality Assurance:** the SA2 test and review process provided feedback on issues to Component Exchange on the repository submissions.

4. Summary of First Year

4.1. Identification of Components for Exchange

The first phase of the Component Exchange activity is for SOTON and the Chinese partners to identify mutual components for exchange.

4.1.1. European Component Evaluation by Chinese Partners

The Chinese partners, primarily BU, evaluated the OGSA-DAI, VOMS and GridSphere components available across the OMII-Europe platforms, in addition to the GridSAM job execution component available from OMII-UK [OMII-UK], as described in M:JRA1.14. The target Chinese infrastructure was CROWN for the evaluation.

An in-depth analysis of these components is provided in the M:JRA1.14 milestone, however the conclusions of the analysis of each component were as follows:

- OGSA-DAI (JRA1 Database): ported into CROWN when time permits.
- GridSphere (JRA1 Portal): it should be integrated once time and resources permit.
- VOMS (JRA1 VOMS): very beneficial to CROWN, although uncertainty with effort estimation awards it a lower priority for integration.
- GridSAM: the highest priority to port into CROWN.

As a result, GridSAM was first ported to the CROWN infrastructure, followed by OGSA-DAI (achieved in collaboration with the JRA1 Database task). Due to resource limitations, the other components are still pending porting and/or integration. Additionally, and importantly, ICT also adopted GridSAM into their core VEGA-GOS platform architecture following this activity. Despite the fact GridSAM is not an OMII-Europe component, it supports the OGSA-BES standard adopted by OMII-Europe, and thus this successful component exchange is within both the spirit of the Component Exchange activity and the interoperability goal of OMII-Europe.

4.1.2. CNGrid Component Evaluation by OMII-Europe

A small supervisor and postgraduate team at SOTON were allocated to evaluate the CROWN and VEGA-GOS platforms. This entailed installation of both platforms and an analysis of the components on each infrastructure to gain a broad understanding of their architectures and capability. In addition, a detailed evaluation of CROWN through the ETF [ETF-CROWN] was used to inform this evaluation process.

As previously stated, originally, the Agora and Grip components of VEGA-GOS were identified as desirable by the activity. Agora is a module for the management of policy and context governing the interaction between grid users and resources, whilst Grip is a grid-level task management module. As previously discussed, another component had to be selected from another collaborating partner.

The CROWN Meta-Scheduler from BU was selected for a couple of compelling reasons:

- In general, the Meta-Scheduler acts as a job execution endpoint itself, enabling scheduling policies to be applied to scheduling jobs across ‘real’ job execution endpoints.

- BU's willingness to develop the Meta-Scheduler to adopt the OGSA-BES standard would enable the Meta-Scheduler to act as a BES endpoint itself and schedule jobs across other BES endpoints.

The adoption of the Meta-Scheduler would represent significant value when taken into consideration with the intended outcomes of the JRA1 Job Submission activity, and the focus of OMII-Europe to develop tighter interoperability between different Grid distributions.

Despite the initial resourcing issue at ICT, another component, DDS, was selected in the second year for porting efforts and will be covered later.

4.2. Porting the Meta-Scheduler to OMII-Europe Platforms

In the first year, the majority of the work was conducted in two phases, each phase initiated by a BU visit to SOTON to address a major porting or interoperability goal with the aim to provide a demonstrator for an event.

Initially, the target OMII-Europe platforms identified for porting efforts were Globus, EGEE and UNICORE. However, because of time and resource constraints, and other issues discussed in Section 8 this had to be limited to Globus.

4.2.1. Porting Approach for the Meta-Scheduler

To expedite porting efforts, a well defined porting approach was required. It was decided to initially port each portable component (i.e. the Meta-Scheduler at this stage) to a generic Tomcat/Axis platform instead of direct to the target Grid platform. Apache Tomcat is developed by the Apache Software Foundation [ASF] and is a Java-based web server and servlet container. Apache Axis [AXIS], also developed by the Apache Software Foundation is a Java based SOAP (Simple Object Access Protocol) web service application framework that can be hosted within Tomcat. There were a number of reasons for choosing this strategy:

- Tomcat/Axis is a relatively simple and understood architecture compared to the OMII-Europe Grid infrastructures.
- Tomcat/Axis represented a 'common language' framework very well understood by both SOTON and the Chinese partners.
- Performing a direct port to an OMII-Europe Grid infrastructure would require a tremendous amount of technology transfer from SOTON to the Chinese partners.
- A Tomcat/Axis port of the components would provide an immediate contingency if the unfunded effort from the Chinese partners could not be sustained for the target porting efforts.

The Tomcat/Axis base release selected was from OMII-UK Southampton. The OMII Release provided an easily installed and pre-configured installation of this base infrastructure, with a WS-Security implementation provided by Apache WSS4j, as a platform for other Grid components and web services.

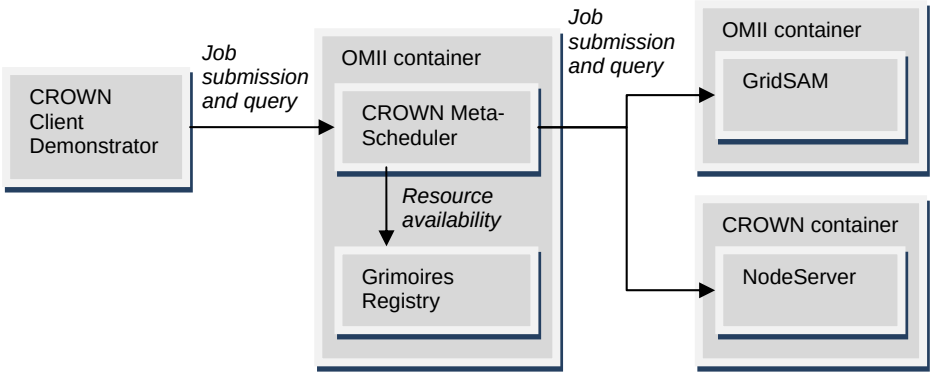
4.2.2. Phase One – Integration and Interoperation

This involved porting the CROWN Meta-Scheduler to the OMII 3.2.0 environment and enabling it to accept CROWN job submission and query requests and forward them to CROWN or GridSAM endpoints.

4.2.2.1. Objectives

The aim was to provide a demonstrable port of the Meta-Scheduler to the OMII 3.2.0 platform for the UK e-Science All Hands Meeting in September 2006 [AHM-2006]. SOTON hosted a BU visit for 4 developers to collaborate to achieve the following working scenario for a demo, diagrammatically outlined in Figure 3:

1. **Hosting:** Meta-Scheduler, hosted in an OMII container, accepts a CROWN client job submission request.
2. **Resource availability:** accesses an OMII Grimoires registry to obtain a list of available CROWN or GridSAM endpoints.
3. **Scheduling policy:** applies its service-specified scheduling policy to determine a suitable endpoint.
4. **Job submission:** submits the job request to the selected endpoint and returns the unique identifier for the job to the client.
5. **Job query:** Client is able to query the Meta-Scheduler service concerning the status of the job.



.

Figure 3. The envisioned architecture for the first phase of the development work

In order to achieve the envisioned demonstrator, a number of technical issues had to be resolved. These efforts were greatly aided by CROWN's easily modified and extended architecture.

4.2.2.2. Summary

The following technical tasks were achieved to attain the objectives:

- **Hosting the Meta-Scheduler in OMII:** the container had to be hosted in the OMII infrastructure, comprising Apache Tomcat 5.0.28 and Axis 1.2.1. Fortunately, both the CROWN hosting environment and OMII both utilised the same version of Axis as their web services implementation, which greatly expedited this porting process and made this a straightforward task.
- **Supporting Resource Availability Determination in OMII:** to achieve the resource availability objective within OMII, an interface to Grimoires (the equivalent registry service

in OMII) was required. This entailed removing the CROWN RLDS implementation and API (Application Programmers Interface) from the Meta-Scheduler, and replacing them with the Grimoires equivalents.

- **Integration with OMII GridSAM:** two distinct issues. Firstly, the scheduler needed to utilise the GridSAM service interface. This was solved in a very similar manner to the Grimoires integration issue; the CROWN endpoint adapter system was extended to include this support and utilise the provided GridSAM client API. The second issue was that the version of JSDL supported by CROWN was an earlier version than that supported by GridSAM. This was solved by simply updating the Meta-Scheduler to support the new version.
- **Supporting the OMII Security Infrastructure:** security had to be addressed for both hosting the Meta-Scheduler within OMII and making GridSAM requests. Basically, these issues represent the same thing, since both GridSAM and the Meta-Scheduler are hosted in the OMII container, so by modifying the Meta-Scheduler to adopt the OMII container security framework (WS-Security provided by WSS4j [WSS4J]), requests made to the scheduler and job submissions to GridSAM are both catered for.
- **Developing a Client Demonstrator:** to provide an effective demonstrator, a graphical Java client application was developed that used the CROWN client API to submit jobs to the Meta-Scheduler and display their progress. The job could be any valid JSDL 1.0 file.

4.2.3. Phase Two – Interoperability through OGSA-BES

This entailed the key aspect of this work – the support for OGSA-BES to facilitate interoperability with the OGSA-BES job submission endpoints from the JRA1 Job Submission activity.

4.2.3.1. Objectives

The aim was to provide a demonstrable OGSA-BES compliant Meta-Scheduler for the OMII platform for the Open Grid Forum (OGF) 20 in May 2007 [OGF20], and also to extend the CROWN NodeServer to support OGSA-BES.

A second BU visit was hosted at SOTON in early May. Essentially, the Meta-Scheduler service interface had to be extended for BES-compliance, and the Meta-Scheduler's job submission client had to be extended to support BES endpoints, as depicted in Figure 4.

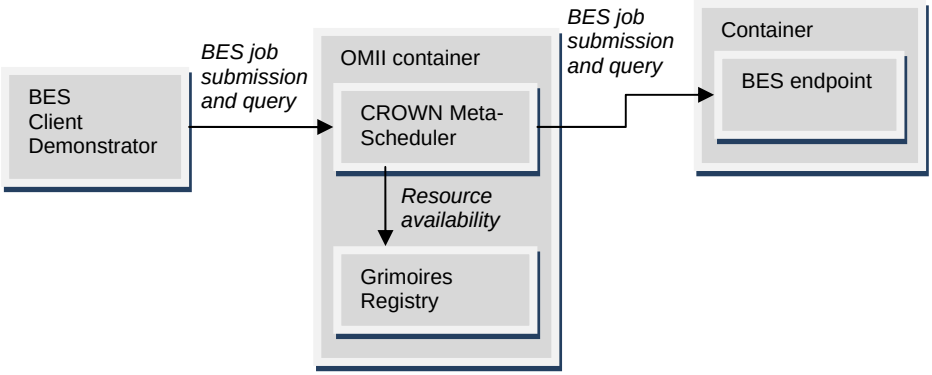


Figure 4. The envisioned architecture for the second phase of the development work

This would require the modification of the demonstrator, Meta-Scheduler and NodeServer interfaces.

4.2.3.2. Summary

The following was achieved to successfully fulfil the objectives:

- **Supporting OGSA-BES within the Architecture:** the work required to support the BES specification (draft version 33 at the time) to support this architecture was quite straightforward. Similar to providing the GridSAM support, an additional BES endpoint adapter was incorporated into the Meta-Scheduler, adopting the same security framework as before. The CROWN NodeServer was modified in a similar manner, and reflects BU's commitment to supporting OGSA-BES within the CROWN Grid infrastructure.
- **Extending the Client Demonstrator:** the original client demonstrator developed for the first demonstration was adapted to support the BES interface, and modified to display the BES state model for scheduling/execution state. This produced the second version of the demonstrator. The updated display for the execution state is shown in Figure 5.

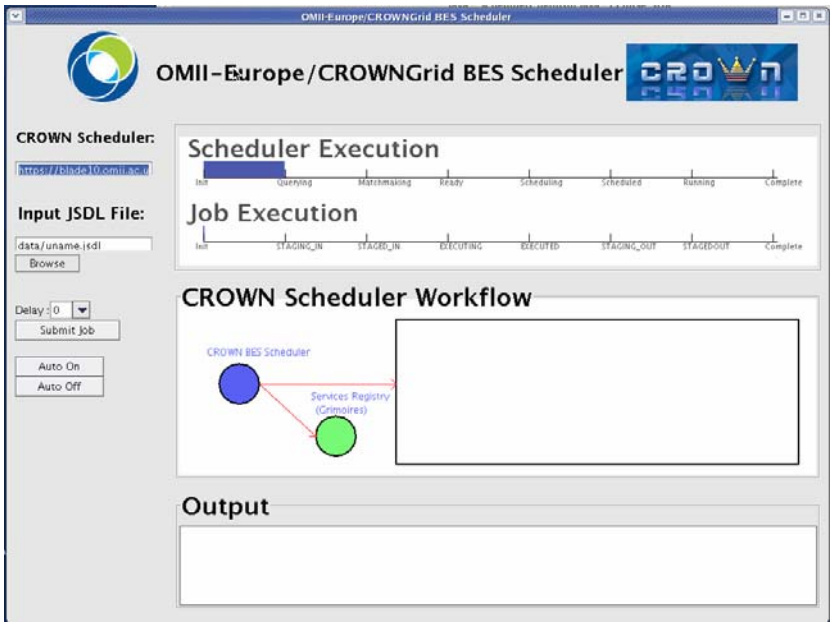


Figure 5. Screenshot of the second demonstrator

The box in the middle shows the available OGSA-BES computation endpoints. The green dot represents the endpoint selected to execute the job submitted by the user. The demonstration system continually updates the jobs status display (from scheduler and OGSA-BES perspectives) in the top panel of the screen, displaying the job status in the scheduling step and the execution step. The lower part of the screen displays a textual update of the job's progress, including suitable output from completed jobs.

5. Review Recommendations for Second Year

5.1. Overall Assessment

27: "Better align the decisions about the selection of versions of external middleware across the project activities. Avoid the use of different versions of a middleware whenever possible."

In line with the Technical Committee recommendations, versions UNICORE 6.x, gLite 3.1 and Globus 4.0.x are accepted and assumed throughout this activity and within this report.

28: "Consider the implementation of a project wide inventory that defines the current version to be used by all OMII-Europe developments."

The project-wide versions of the middlewares have been accepted in accordance with the centralised OMII-Europe middleware version inventory.

29: "Refer clearly in documentation to specific versions not only to the basic name (e.g., UNICORE6 instead of UNICORE)."

This has been done as suggested.

30: "In future versions of the yearly management report of the re-engineering activity, describe the issues faced and how they were addressed with particular emphasis on cross-activity issues."

In addition to the various technical and other issues and resolutions included in this report, Section 8 directly addresses collaborative issues encountered throughout the course of the activity, with approaches taken to address those issues. Additionally, this has been partly addressed in Section 3.1 which indicates the nature of the interactions with the other OMII-Europe activities.

5.2. Workplan and Resources

Work package JRA1 Re-engineering of Services: "the overall progress is seen as satisfactory although in some areas the work deviated from the original plan (e.g., the focus on UNICORE only in an initial phase for job submission or the refocus of work on portals). These modifications are not sufficiently justified or motivated. It is very difficult to assess the progress of work, because in the deliverables the description of a clear baseline where the work started from is lacking."

This has been a core issue with this activity, however, Sections 2 and 6 (and elsewhere in this report) provide justifications for the deviation from the original plan. This report has been specifically structured to clarify the technical baseline and work units required to achieve the activity goals.

DJRA1.13 12M Report on the Component Exchange Activity: "the document does not motivate why this task has been lead by OMII-UK which is not a contractual partner of the project."

This was an error that has been rectified in this report to appropriately state the University of Southampton (SOTON) as the contractual partner instead of OMII-UK.

6. Objectives for Second Year

6.1. Meta-Scheduler - BU

To achieve the revised goal of porting the Meta-Scheduler to the OMII-Europe platforms, the following objectives needed to be met:

1. Port the Meta-Scheduler to Globus, gLite and UNICORE.
2. Include multiple scheduling policy support within the Meta-Scheduler.
3. Interoperate with the BES endpoints from JRA1 job Submission (UNICORE 6.x BES, CREAM-BES and GT 4.0.x BES).
4. Conform to the OMII-Europe Common Security Profile (JRA3 Security).

5. Productise the software and create OMII-Europe-focused user and developer documentation.

Completion of these tasks would ensure goal 2 of the revised goals given in Section 2.2 would be achieved.

6.2. Dynamic Deploy Service (DDS) – ICT

As previously discussed in Section 2, a meeting was held in Beijing a PM14 to discuss re-engagement with ICT. During this discussion it was concluded that the original goals of porting Agora and Grip were not achievable within the remainder of the activity, taking into account the resources available at SOTON and ICT during this timeframe. Porting either of these would require significant effort, since each is very tightly coupled to the VEGA-GOS infrastructure. Instead, it was decided to select a more suitable component from VEGA-GOS that was more self-contained and more tightly scoped. The VEGA-GOS component selected for exchange was the Dynamic Deploy Service (DDS) for a number of reasons:

- DDS enables web services to be packaged and deployed on a remote container, providing the ability of web service provision on an on-demand basis.

Since this goal was adopted late into the activity, the crucial concern was that any porting efforts could not be achieved within the activity timeframe and resources. However, the following technical properties of the component would help mitigate the risk of not achieving the objective:

- This component is self-contained; it is not dependent on additional components that would also need to be included in any porting efforts.
- The component is natively deployed within Apache Tomcat, the container utilised by VEGA-GOS, which is a well understood technology at SOTON.

Therefore, to achieve the revised goal of porting DDS to the OMII-Europe platforms, the following objectives needed to be met:

1. Port DDS to GT 4.0.5.
2. Create a suitable demonstrator for GT 4.0.5.
3. Determine feasibility of porting and estimate of work required for porting to gLite 3.1 and UNICORE 6.
4. Productise the software and create OMII-Europe-focused user and developer documentation.

Completion of these tasks would ensure goal 3 of the revised goals given in Section 2.2 would be achieved.

7. Progress

7.1. The Meta-Scheduler

This work centred on porting the Meta-Scheduler to Globus, solving various practical interoperability issues with the OGSA-BES compliant job submission implementations provided by JRA1 Job Submission, and productising the software for submission to the OMII-Europe repository.

7.1.1. Approach

For this final phase of the work, there were a number of objectives worked towards in parallel. The work was apportioned between SOTON and BU:

1. **BU:** port the Meta-Scheduler to GT 4.0.x, gLite 3.x and UNICORE 6.x.
2. **BU:** include multiple scheduling policy support within the Meta-Scheduler.
3. **SOTON:** ensure interoperation with the BES endpoints from JRA1 Job Submission (UNICORE 6.x BES, CREAM-BES and GT 4.0.x BES).
4. **SOTON:** conform to the OMII-Europe Common Security Profile (JRA3 Security).
5. **BU/SOTON:** productise the software and create OMII-Europe-focused user and developer documentation.

This phase of the work was conducted utilising a shared code repository, provided by and hosted at SOTON, which enabled both institutions to operate on the same code base and prevent any potential conflicts between two separately located code bases that would have to be integrated.

Unfortunately, this phase of the project coincided with an unavoidable, and unexpected, organisational team restructure at BU to allow for a new project. This led to inevitable resourcing issues and reduced Meta-Scheduler development support required by SOTON. This delayed the initiation of this phase and negatively impacted on the progress towards both primary objectives. In particular, support for the Common Security Profile could not be achieved.

7.1.2. Porting to Globus

This work was conducted by BU via the Subversion repository stationed at SOTON. This effectively involved repackaging the Meta-Scheduler for GT4 platforms. All that was required was a high-level GT4-specific build script that reorganised the built Meta-Scheduler components into the GT4 GAR (Grid ARchive) format. Thus, the outcome of the software build process is to provide two deployment packages: a WAR (Web ARchive) for Tomcat, and a GAR for GT4.

7.1.3. Multiple Policy Support

This work was conducted by BU via the Subversion repository stationed at SOTON, and encompassed two development aspects.

Firstly, the multiple policy support within the Meta-Scheduler service had to be included. The original CROWN Meta-Scheduler had a number of supported multiple policies, however when the Meta-Scheduler was integrated with the Grimoires registry service, this support had to be temporarily disabled until full Grimoires-compatible scheduling support could be achieved. Unfortunately, this was not possible within the activity timeframe, so a compromise was reached. Instead of regressing to the previous RLDS CROWN registry system, an intermediate format was adopted: essentially, all endpoints and their hardware and software characteristics are registered within an XML file that can be read by the scheduler service. This file can be independently updated by another process, for example a simple client that continuously updates the information from a live Grimoires, or RLDS, registry elsewhere. This approach enabled registry support, although basic, to be included. This is a fundamental necessity for scheduling, since scheduling policies need to be aware of endpoint capabilities in order to make a decision based on their policy.

The initial policies supported are as follows:

- Random (default) – an endpoint is simply selected at random.
- RoundRobin – endpoints are assigned jobs in a simple cyclic fashion.
- CPUFirst – for computationally-intensive jobs, the endpoint with the most powerful CPU is selected.
- MemoryFirst – for jobs with large memory requirements, the endpoint with the most memory is selected
- Nearest – the geographically closest endpoint is selected.

In addition, the CROWN architecture is largely extensible, and custom policies can be developed in Java and plugged into the service.

Secondly, multiple policy support within the client, so the client is able to request usage of a policy. The client was extended to encompass appropriate options for specifying the request of a policy. Of course, only those that are installed on the service are honoured.

7.1.4. Supporting Multiple Security Models for Practical Interoperability

Support for BES itself does not guarantee interoperability with BES endpoints. In practice, any client to a BES endpoint (the Meta-Scheduler in this case) must comply with the security infrastructure employed by the endpoint containers in order to communicate with them. This was solved by providing an appropriate Axis client configuration to the scheduler's BES resource adapter depending on the endpoint (e.g. a unicore-client-config.wsdd for a UNICORE 6.x BES endpoint). Each individual Axis client configuration would contain the necessary security set up required for that endpoint. This is configurable post-installation.

Whilst interoperability was achieved with CREAM-BES and UNICORE 6.x BES test endpoints provided by JRA1 Job Submission, unfortunately support for GT4.0.x BES could not be completed. To achieve this, it is necessary for the Meta-Scheduler to support the Globus Security Infrastructure (GSI) required by a client to Globus. This was not achievable due to the BU team restructure, and other technical reasons. Firstly, the GSI client code requires a number of Java resources incompatible with the Meta-Scheduler. Addressing this would entail a very complex solution which involved supporting both GT4.0.5 (the version of Globus targeted for Meta-Scheduler porting efforts) and GT4.1.3 security (required by GT4-BES and not an agreed OMII-Europe middleware version), both of which are incompatible with each other. A solution for this was deemed possible but not achievable with the given resources and timeframe.

7.2. Dynamic Deploy Service (DDS)

This work focused on porting DDS to Globus and productising the software for submission to the OMII-Europe repository.

7.2.1. Introduction to DDS

The Dynamic Deploy Service (DDS) provides an interface that enables OSGi-compliant web services to be deployed and undeployed on a container dynamically, without the need to restart the container. It also ensures that the resources of the deployed services do not conflict with the resources from other services resident in the container.

OSGi is an open and unified interface standards framework. With this framework, service providers, developers, software vendors, gateway operators and equipment vendors can join together and coordinate their development, deployment and management of services. OSGi defines an environment where services execute, which includes the management of service components and service registration. In the framework, a service is implemented and deployed as a physical and logical unit, called a Bundle. Physically, a bundle is a Java package, which includes code, resources and configuration. The framework provides a mechanism which can install, invoke, stop, update and clear a bundle. DDS utilises the Equinox [EQUINOX] implementation of the OSGi standard.

7.2.2. Approach

The majority of the work was conducted in a single phase to address the porting goals. This was initiated by an ICT visit to SOTON, with the aim to provide a demonstrator for the OMII-Europe Face-to-Face meeting in November 2007. As previously mentioned, the planned work was comparatively small.

The adopted porting process was the same as for the Meta-Scheduler (as were the reasons; see Section 4.2.1). The objectives for this task and work apportioned were therefore as follows:

1. **ICT:** port DDS to GT 4.0.5 (and OMII 3.4.0).
2. **ICT:** create a suitable demonstrator for GT 4.0.5 (and OMII 3.4.0).
3. **ICT:** determine feasibility of porting and estimate of work required for porting to gLite 3.1 and UNICORE 6.
4. **ICT /SOTON:** productise the software and create OMII-Europe-focused user and developer documentation.

Addressing objective 3, after investigation by ICT it was decided that there was not enough resource for development and testing to support gLite 3.1 and UNICORE 6, given the degree of platform knowledge that would be required for porting to these platforms and the resource available.

7.2.3. Porting DDS to OMII 3.4.0

The first part of this work focused on a port to OMII 3.4.0. This was a straightforward task, since both VEGA-GOS and OMII 3.4.0 are based on the same versions of Apache Tomcat and Apache Axis.

The most noteworthy difference is that the WS-Security implementations employed by DDS and OMII were incompatible, and this represented the largest porting issue. The solution essentially involved supporting the OMII WSS4j WS-Security implementation in the DDS Axis application.

7.2.4. Porting DDS to GT 4.0.5

This involved repackaging DDS as a Grid ARchive (GAR) distributable that is compatible with and supports GT 4.0.5. This proved more problematic due to the complexities involved with supporting the Globus Security Infrastructure (GSI).

To ensure consistency of security model adopted across all Globus-installed services, it was decided that at least minimal GSI support should be included. To this end, support for the following was included and tested:

- Privacy – provided by GSI Secure Conversation
- Integrity – provided by GSI Secure Conversation and GSI Secure Message
- Authentication – via X.509 certificates
- Authorisation – self-authorisation and none modes are supported
- GSI Transport

Delegation, privacy with GSI Secure Message and the other various authorisation methods available in GSI (gridmap, identity authorisation, SAML callout authorisation and host authorisation) are not fully supported at this stage.

8. Issues and Resolutions

During the course of the activity, a number of issues were encountered that had to be addressed. The specific resource issues with ICT and BU have already been covered, but others included:

1. **Unfunded effort:** the Chinese partners contributed effort on an unfunded basis, and despite tremendous effort on their part to contribute, it understandably meant that funded activities within their organisations often had to take priority.
2. **Language barrier:** this impacted on a number of interactive activities between the Chinese partners and SOTON to varying degrees, primarily with transfer of technical knowledge and documentation review.
3. **National holidays:** national holidays between the UK and China very often did not coincide. This was a particular problem between December (UK) and late January/early February (China).
4. **Time difference:** technical issues identified at SOTON that required resolution by the Chinese partners could not often be solved until the next working day. This had a cumulative effect especially noticeable when approaching release dates.

These issues were addressed by the following measures learnt as the collaboration progressed:

1. **Flexibility and constant prioritisation:** taking into account the unfunded nature of the contribution, a flexible and accommodating approach to objectives was fundamental. By constantly assessing objectives and prioritising tasks when a resourcing or scheduling issue presented itself, many potential problems were avoided, and more important tasks could be focused upon.
2. **Alignment of objectives to ensure benefits to unfunded partners:** as previously stated, the unfunded effort was structured around a mutual benefit basis. This was mitigated by ensuring that agreed Component Exchange goals were aligned with those of their own organisation where possible and appropriate.
3. **Communication:** regular teleconferences and email correspondence concerning progress ensured that many potential technical, resourcing and scheduling problems could be ameliorated or avoided altogether.

9. Conclusions and Future Work

The following are the key achievements of the JRA1 Component Exchange activity:

1. **Adoption of OGSA-BES into Chinese partner Grid infrastructures:** CROWN from BU directly included support for the OGSA-BES standard into their computational Grid

infrastructure for their Meta-Scheduler and NodeServer. This was also achieved in VEGA-GOS from ICT, by adopting the GridSAM component in their architecture which supports OGSA-BES.

2. **CROWN Meta-Scheduler on GT 4.0.5 (and OMII 3.x):** the CROWN Meta-Scheduler was ported to Globus, enabling Grid job scheduling accessible from OGSA-BES clients based on a server-based scheduling policy.
3. **Successful interoperability with CREAM-BES and UNICORE 6.x BES:** submission and querying of jobs to either of these OMII-Europe platforms.
4. **Support for multiple scheduling policies within the Meta-Scheduler:** the ability for the Meta-Scheduler to decide where best to submit a job based on a policy defined by a resource administrator, and the architecture necessary to extend this to include custom scheduling policies.
5. **VEGA-GOS DDS on GT 4.0.5 (and OMII 3.x):** the ability for OSGi-compliant services to be deployed and undeployed on Globus remotely when desired without manual reconfiguration or server reboots.
6. **Development of a collaborative process:** based on the experiences with collaborating with BU, a process was developed that helped to ameliorate the issues encountered during the course of the activity which was successfully applied to a similar collaboration with ICT.

Participation in a number of key community and dissemination activities was also achieved:

1. **UK e-Science All Hands Meeting (AHM), September 2006 [AHM2006], SuperComputing 2006 [SC06]:** the first version of the demonstrator was showcased.
2. **Open Grid Forum (OGF) 20, May 2007 [OGF20]:** the second version of the demonstrator was showcased.
3. **UK e-Science All Hands Meeting (AHM), September 2007 [AHM2007]:** presentation of the paper titled ‘Towards Grid Interoperability’ on the promotion of interoperability technologies including the CROWN Meta-Scheduler.
4. **Open Grid Forum (OGF) 21, October 2007 [OGF21]:** presentation on the adoption of OGSA-BES and JSDL within OMII-Europe.

The submissions given in Table 4 were made to the OMII-Europe repository.

Product	Version	Submission Type	Submission Date
OMII-Europe BES Scheduler	0.8.0	Software (inc. docs)	29/02/2008
OMII-Europe BES Scheduler	0.9.5	Software (inc. docs)	02/04/2008
Dynamic Deploy Service	0.0.1-M1.RC1	Software	06/11/2007
Dynamic Deploy Service	0.0.1-M1.RC1	Documentation	06/11/2007
Dynamic Deploy Service	0.0.2	Software (inc. docs)	11/03/2008
Dynamic Deploy Service	0.0.2	Tutorials	13/03/2008
Dynamic Deploy Service GT	0.0.3	Software (inc. docs)	16/04/2008
Dynamic Deploy Service OMII	0.0.3	Software (inc. docs)	16/04/2008

Table 4. JRA1 Component Exchange submissions to the OMII-Europe SA1 Repository

Through this collaboration, the Chinese partners have benefited in a variety of ways. Firstly, their international profile has been increased through this activity which was greatly desired. Secondly, they have benefited through the exchange of European-based components that they have adopted. Through assistance provided by this JRA1 activity (in addition to other JRA1 activities), they have been able to leverage these components to appropriately enhance their own infrastructures by filling perceived functional gaps. They have also embraced and adopted the open standards ethos promoted by OMII-Europe, explicitly extending their own infrastructures to support these open standards where appropriate, and attending standards development conferences to further their understanding of such standards. In addition, through practical collaboration they have developed

an appreciation of European Grid software development processes and techniques, which they can adopt within their own practices.

Given that the entirety of the Chinese partner involvement was unfunded, taking into account the achievements made, this activity has represented tremendous value for money. Despite a number of resource issues encountered throughout the project, the partners were able to refactor their goals and restructure effort to achieve very positive outcomes in accordance with the overall aim of the activity and deliver components to the OMII-Europe repository.

Future work that builds on the progress of this activity includes the following:

1. **Support for the OMII-Europe Common Security Profile:** one unfortunate casualty of the resourcing issues was that the Common Security Profile could not be supported. This would enable the Meta-Scheduler to fully interoperate with the OMII-Europe security infrastructure and support virtual organisations via VOMS.
2. **Meta-Scheduler support for GT 4.1.x BES interoperability:** by satisfying the final interoperability objective with Globus, essentially, a security integration task, a major platform would be supported.
3. **Porting Meta-Scheduler to UNICORE 6.x and gLite 3.x:** this would add the interoperable meta-scheduling capability to these prominent Grid infrastructures.
4. **Support for the GLUE 2.0 specification from JRA2:** this would enable the scheduling engine of the Meta-Scheduler to utilise computational resource information using this standard. The work could be conducted in two phases:
    a. **File-based registry support for GLUE 2.0:** the Meta-Scheduler could be modified to accept its file-based registry in the GLUE 2.0 format.
    b. **Support for a GLUE 2.0-based service registry:** the file-based registry system currently adopted could be replaced with a GLUE 2.0-based registry web service.
5. **Support for OGSA-RUS from JRA1 Accounting:** the job submission engine of the Meta-Scheduler could be adapted to include OGSA-RUS support, by submitting usage information in the Usage Record Format to OMII-Europe OGSA-RUS-compliant accounting services.

10. References

[AHM-2006] UK e-Science All Hands Meeting 2006, September 18th-21st 2006, Nottingham, UK - http://www.allhands.org.uk/2006/

[AHM-2007] UK e-Science All Hands Meeting (AHM), September 10th-13th 2007, Nottingham, UK - http://www.allhands.org.uk/2007/

[ASF] Apache Software Foundation - http://www.apache.org/

[AXIS] Apache Axis - http://ws.apache.org/axis/

[DOW] Evaluation Task Force (ETF) evaluation of the CROWN Grid infrastructure

[EQUINOX] Equinox implementation of the OSGi R4 core framework specification - http://www.eclipse.org/equinox/

[ETF-CROWN] Evaluation Task Force (ETF) evaluation of the CROWN Grid infrastructure

[OGF20] Open Grid Forum (OGF) 20 – May 7th-11th 2007, Manchester, UK - http://www.ogf.org/OGF20/events_ogf20.php

[OGF21] Open Grid Forum (OGF) 21 – October 15th-19th 2007, Seattle, WA, USA - http://www.ogf.org/OGF21/events_ogf21.php

[OMII-UK] OMII-UK - http://www.omii.ac.uk/

[SC06] SuperComputing 2006 – November 11th-17th 2006, Tampa, FL, USA - http://sc06.supercomputing.org/

[SC07] SuperComputing 2007 – November 10th-16th 2007, Reno, NV, USA - http://sc07.supercomputing.org/

[WSS4J] Apache Web Services Security for Java (WSS4j) - http://ws.apache.org/wss4j/