

JISC DEVELOPMENT PROGRAMMES

Project Document Cover Sheet

Final Report

Project

Project Acronym	ShibGrid	Project ID	E9RJUQ0
Project Title	Integrating NGS into the academic framework		
Start Date	1 April 2006	End Date	31 March 2007
Lead Institution	University of Oxford		
Project Director	Dr. Anne Trefethen		
Project Manager & contact details	Dr. Neil Caithness neil.caithness@oerc.ox.ac.uk		
Partner Institutions	Council for the Central Laboratories of the Research Councils (CCLRC)		
Project Web URL	https://wiki.oerc.ox.ac.uk/shibgrid/Documentation		
Programme Name (and number)	core middleware: technology development programme		
Programme Manager	James Farnhill		

Document

Document Title	ShibGrid – Final Report		
Reporting Period			
Author(s) & project role	All project members		
Date	31 March 2007	Filename	shibgrid_final_report.doc
URL			
Access	<input checked="" type="checkbox"/> Project and JISC internal		<input type="checkbox"/> General dissemination

Document History

Version	Date	Comments
1.0	31/03/2007	

ShibGrid – Final Report

funded by the Joint Information Systems Committee (JISC)

Oxford e-Research Centre, University of Oxford
CCLRC e-Science Centre, CCLRC

Contents

1.	Executive Summary	3
2.	Background	4
3.	Aims and Objectives	5
4.	Methodology	6
5.	Implementation	8
6.	Outputs and Results	14
7.	Outcomes	20
8.	Conclusion	22
9.	Recommendations	23

1. Executive Summary

The NGS relies upon X509 digital certificates for user authentication. Users obtain a certificate from a centralised UK certificate authority that allows users access to all core and partner sites, and the services provided by these sites, unless a separate application procedure is required by a site (this is the case with HPCx). This mechanism for secure authentication is tied to the individual and does not allow project or organizationally managed access. This may prove a serious shortcoming for NGS as it develops and the size of the user base increases.

This project has developed prototype software that allows the integration of the standard X509 certificates with Shibboleth a system that allows authorization decisions to be made when a user from an organization requests access to online resources, not necessarily solely as a result of who the person is, but possibly based on other information about the person, such as their role in their organization. Such attributes are maintained by the user's organization and are only disclosed with the user's knowledge, so privacy is preserved. In this way authentication is devolved from the single national entities of the Grid CAs to users' home institutions – this also creates a far more scalable infrastructure where the number of users can increase dramatically.

The project was subdivided into four key stages. The first stage was to complete a user requirements gathering exercise, primarily through our stakeholder users in DIAMOND and Integrative Biology (IB). From these requirements the ShibGrid system was developed with users providing input to the system architecture while the OMII provided guidance for the software development cycle. The software development consisted of taking the existing NGS portal (web interface) and developing key components that enabled the capabilities of both Shibboleth and the certificate based authentication to be provided through a user-friendly interface. This required tools for uploading and downloading certificates, pluggable security modules to allow the transfer of Shibboleth attributes, and portlets to manage user's Shibboleth attributes based low assurance certificate. In order to provide a broader scope for adoption by other projects two versions of the NGS portal were ShibGrid-enabled, one built in Stringbeans and the other in uPortal.

Documentation was developed consisting of a quick-start guide, user guide, administrator guide (containing pre-requisites and full installation instructions) and maintenance documentation, mostly in the form of code comments but also a detailed description of the protocols involved. Users were engaged in testing, including the stakeholder user groups who would test the system against their own user cases. The feed-back provided was then used to improve the code and documentation. The resulting software has been deposited with the OMII code repository.

More information about ShibGrid project is available at <http://www.oerc.ox.ac.uk/shibgrid> .

2. Background

Since its inception the NGS has relied upon X509 certificate authenticated access where every potential user needs to obtain a certificate from a certificate authority accredited by the International Grid Trust Federation. Once an account request is reviewed and approved, access is typically granted to all core, and partner sites and the services provided by these sites, unless a separate application procedure is required by a site (this is the case with HPCx). There is no way to restrict access to some services but not others, depending on who the user is or which project or organization he is from, other than by managing each individual's authentication separately.

This shortcoming can potentially discourage expansion of the NGS to new sites who want to control who can use their resources, especially as the number of users of the NGS increases. This is in addition to the burden of requiring new users to obtain a Grid certificate and to learn how to use it prior to being able to use the NGS.

Shibboleth is a project which allows authorization decisions to be made when a user from an organization requests access to online resources, not necessarily solely as a result of who the person is, but possibly based on other information about the person, such as their role in their organization. Such attributes are maintained by the user's organization and are only disclosed with the user's knowledge, so privacy is preserved.

In this way authentication is devolved from the single national entities of the Grid CAs to users' home institutions – this also creates a far more scalable infrastructure where the number of users can increase dramatically. In removing the need for users to obtain certificates for themselves, the process of obtaining access to the NGS is made far easier as certificates and Public Key Infrastructure (PKI) security in general have traditionally been an obstacle for novice users – this problem is becoming more acute as the NGS attracts more users from non computing disciplines.

The advantages of moving to a devolved security infrastructure such as *ShibGrid* are so significant that the authors believe that it has the potential to bring a major contribution to the future success of the NGS. Shibboleth is already proving to be the most popular solution for integrated and uniform access to multiple resources. When the ShibGrid project was first funded JISC had decided, having considered the various options, to deploy Shibboleth, replacing over time the successful Athens service. During the lifetime of the project the picture has somewhat changed as Shibboleth is now only one of several available technologies that are SAML-based and JISC is rolling out a SAML-based federation, which will support solutions that use Shibboleth.

There are other projects integrating Grid middleware and Shibboleth, notably ShibGrid's sister project, *SHEBANGS*, which also uses Shibboleth to authenticate and authorize users of the NGS. However, ShibGrid generates a Grid credential behind the scenes whereas *SHEBANGS* employs the use of a Credential Translation Service (CTS) that users are required to visit prior to logging onto the NGS. Other related projects include *GridShib* which expects users to have already authenticated themselves with their certificates, thus taking a fundamentally different approach to the problem than that of ShibGrid.

The *MAMS* and *SWITCHaai* projects also make use of Shibboleth to access the Grid, although the work presented here has been influenced by these projects to a lesser extent.

3. Aims and Objectives

The aim of the ShibGrid project was initially agreed “to produce production-level code and documentation that meets the use cases outlined below for the benefit of existing and future NGS users”.

The following four use cases were identified and originally presented in the project plan:

PUC1: User with 12 Month Certificate creates a proxy certificate and stores it on the ShibGrid MyProxy server (accessed via the NGS portal) using only his Home Organisation’s (HO’s) single sign-on (SSO) for authentication. The user can thereafter access and use the proxy certificate using only his HO’s single sign-on for authentication.

This use case is important for existing users of the NGS and necessary for a successful deployment and initial adoption of ShibGrid. Using the prototype, NGS users at Oxford and CCLRC RAL were able to access their proxy certificate having used their home institution to single-sign-on to the ShibGrid portal.

PUC2: User who does not own a 12 Month Certificate is able to perform some Grid functions (for which at least a lower level of assurance certificate, or its associated proxy certificate, is necessary) via the NGS portal and using her HO’s SSO, but without needing to apply for a 12 Month Certificate.

For the large number of potential users of the NGS who do not need the additional assurance that a 12 Month Certificate provides, this use case enables the number of NGS users to scale significantly into the future. As with PUC1, Oxford and CCLRC users without certificates successfully tested the ShibGrid prototype.

PUC3: Users must be able to be registered and approved as NGS Users and to be removed/revoked from the list of NGS Users.

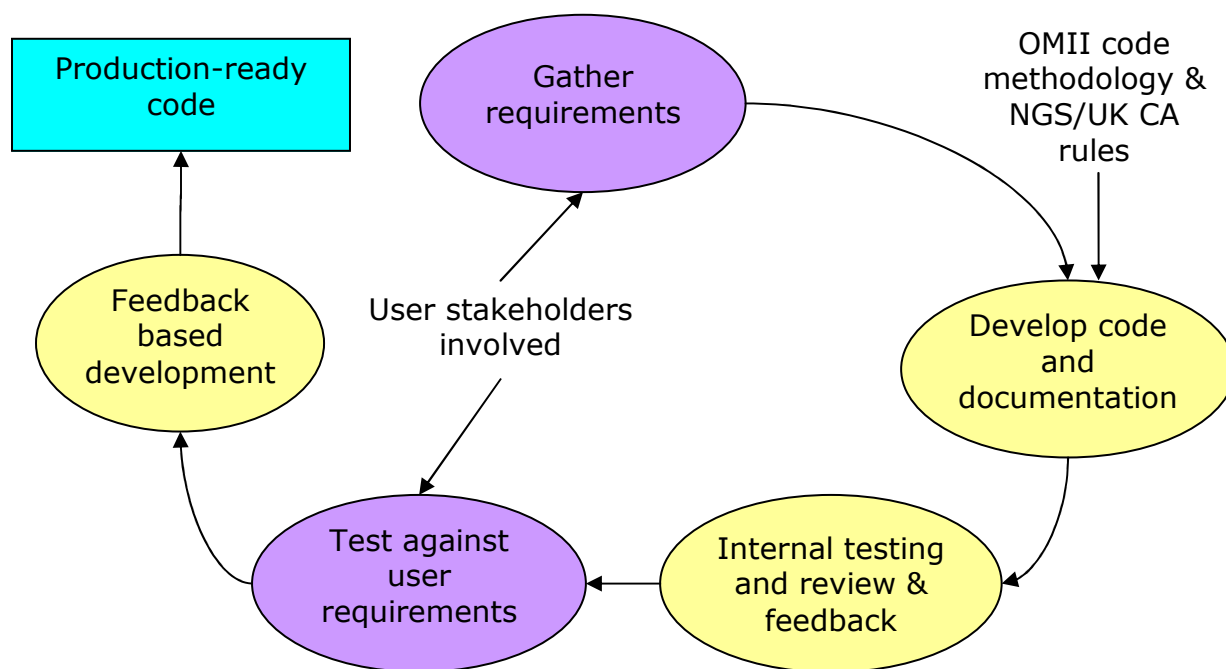
All ShibGrid users ultimately possess an X509 certificates once logged onto the NGS, so the ShibGrid prototype requires no change to Grid middleware regarding the process of authorizing users (unless resource providers wish to implement more finely-grained authorization decisions based on Shibboleth assertions). As a result, existing procedures of registering users and granting or revoking access to NGS resources is unchanged.

PUC4: A user that has generated a Temporary Certificate should be able (if he so wishes) to take control of that certificate and use it in a similar manner to a 12 Month Certificate.

As stated above, ShibGrid users possess standard X509 certificates, to which they have complete access and are free to use them as they wish. However, as with 12 Month Certificates issued by traditional Grid CAs, users with temporary certificates will nevertheless still be bound by the terms of the CP/CPS of the CA which issued the lower level of assurance temporary certificate.

4. Methodology

The methodology for the ShibGrid project largely flowed from the aims and objectives for the project. In particular the aim of the project can be summarized as “the removal of barriers inherent in PKI infrastructures which prevent more users taking advantage of the NGS”. This led to the following development methodology:



The first stage of the project was to investigate what the user requirements for a new security system are; this was to primarily be through our stakeholders DIAMOND and Integrative Biology (IB). At the same time the requirements of other stakeholders was investigated, for example, OMII and the NGS.

From these requirements the ShibGrid system was developed. The requirements from users provided input to the system architecture design while the requirements from OMII provided guidance for the software development cycle. The software development consisted of the following key components:

- Shibboleth-enabling the MyProxy server.
- Enabling the CA component of the MyProxy server and tying this into the other Shibboleth-enabled processes.
- Shibboleth-enabling a clone of the current NGS portal (based on Stringbeans) and, where appropriate, the portlets within it.
- (If possible) Shibboleth-enabled the new, pre-production, version of the NGS portal (based on uPortal).
- Creating tools for uploading and downloading proxy certificates to the MyProxy server using Shibboleth for the authentication/identity mechanism.

The core documentation consisted of a quick-start guide, user guide, administrator guide (pre-requisites, full installation instructions) and maintenance documentation, mostly in the form of code comments but also a detailed description of the protocols involved.

The next stage of the process was user testing, which involved the stakeholder user groups (DIAMOND and IB) who tested the system against their own user cases. The feed-back provided was then used to improve the code and documentation. At this time documentation and code was also reviewed.

In addition the project was active in disseminating the findings of the project through various conferences and providing future directions for ShibGrid and its stakeholders.

Special considerations

The ShibGrid project sits in the middle of two worlds, Shibboleth and Grid/NGS, and must satisfy the standards of both, primarily Shibboleth/SAML and the Grid Security Infrastructure (GSI) and additionally it needs to track changes to the standards and technologies that are employed. In the case of Shibboleth this was the change to SAML 2.0 and Shibboleth 2.x, along with the formulation of the UK Federation policy; and in the NGS this is the adoption of VOMS.

Although it became clear that the release of Shibboleth 2.x would not happen during the life of the project, a roadmap for Shibboleth 2.x development is the subject of a documentation deliverable. ShibGrid also provided input into the formulation of the UK Federation policy to attempt to ensure that it was compatible with ShibGrid.

Running in parallel with the ShibGrid project was the Shibboleth-Enabled Bridge to Access the NGS (SHEBANGS) project. From before the start of these projects the two teams met to ensure that the two solutions were complementary and work was not needlessly duplicated. Therefore while ShibGrid has a focus on authentication, SHEBANGS has an authorization focus and is looking in detail at integration with VOMS. Therefore the ShibGrid project did not look at VOMS in-depth.

Looking across both the Grid and Shibboleth communities it was also important to track and contribute towards any standardization work in the area of Shibboleth and Grids to ensure that the UK and ShibGrid's interests were represented and ShibGrid did not develop in a way that would be incompatible with future standards.

Security was also seen as a key concern in ShibGrid and so it was decided to add tests to ensure the security of the MyProxy server. At times it was considered as to whether to add a full security to the project, but in the end it was decided that the way to engage the most appropriate experts was to present the ShibGrid architecture at venues where Grid and Security experts were present. This would validate the architecture then tests would be run against the implementation to ensure it conformed to the validated architecture.

5. Implementation

User Requirements

Within ShibGrid use case gathering was primarily performed through making use of the requirements already in existence. The DIAMOND and IB projects have already produced user requirements documents. In the case of IB this is through interaction with their users and the ShibGrid project translated these into those requirements that are relevant to ShibGrid; and in the case of DIAMOND this takes the form of high-level user requirements developed for their own internal SSO work.

Other stakeholder requirements (for the NGS, UK e-Science CA and OMII) were derived from the relevant public documents from, and discussions with, the organization.

The results of the User Requirements exercise can be found in the document at <https://wiki.oerc.ox.ac.uk/shibgrid/Documentation>. They are presented here to provide the background for the description of the architecture of ShibGrid.

In summary, the requirements gathered from OMII provided our methodology for code and documentation; the requirements from the NGS and the UK e-Science CA largely defined the rules within which the ShibGrid architecture must operate; and the requirements from DIAMOND and IB provided much backing for the choice of Shibboleth as the authentication framework. Even so the following items were identified as relevant requirements:

1. ShibGrid should allow users with no prior knowledge of PKI and Grid security to use the NGS;
2. ShibGrid should allow the use of users' e-Science certificates (if they have one);
3. There should be a stable mapping from users to the DNs presented to resources;
4. Users who change institution need to still have the same access (if eligible) even though their DN may change;
5. Resource administrators would like access to personal information about users for authorisation and logging purposes;
6. Users are required to apply for access to the NGS via the normal procedures; and
7. ShibGrid must work with NGS operations to provide solutions for the possibility of users having more than one DN.

The use cases given in the project plan link in with these requirements as follows (PUC=Project Use Case):

PUC1: User with 12 month certificate creates a proxy certificate and stores it on the ShibGrid MyProxy server (accessed via the NGS portal) using only his home organisation's single sign-on for authentication. The user can thereafter access and use the proxy certificate using only his home organisation's single sign-on for authentication.

This use case is linked to Requirement 2

PUC2: User who does not own a 12 month certificate is able to perform some Grid functions (for which at least a lower level of assurance certificate, or its associated proxy certificate, is necessary) via the NGS portal and using her home organisation's SSO, but without needing to apply for a 12 month certificate.

This use case is linked to Requirement 1

PUC3: Users must be able to be registered and approved as NGS Users and to be removed/revoked from the list of NGS Users.

This use case is linked to Requirement 6.

PUC4: A user that has generated a Temporary Certificate should be able (if he so wishes) to take control of that certificate and use it in a similar manner to a 12 Month Certificate.

This use case is not linked to any requirement, but instead complements PUC1 and PUC2. The requirements do not talk about access methods where as the use cases do. Therefore PUC4 simply says that what PUC1 and PUC2 say about the use of a portal also applies to other access methods.

The requirements that are not covered by use cases are 3, 4, 5 and 7 and this is mainly because these are more behind-the-scenes requirements on the implementation and not visible to users.

Architecture

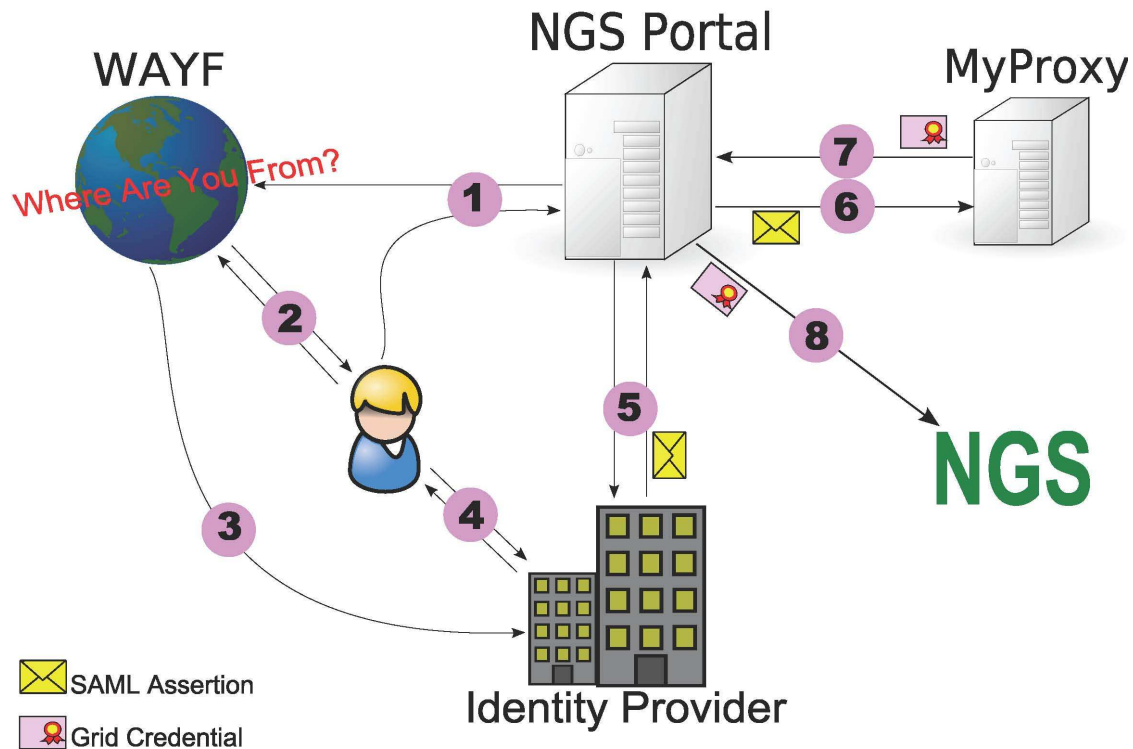
In developing an architecture to satisfy these requirements we did not want to forget the full range of users who use the NGS, from one person working on their own to large collaborative projects; and from Grid beginners to Grid experts. The key to our methodology was to provide a framework in which users from all these backgrounds could gain ease of access.

Project portals were seen as the primary access method to use with ShibGrid because of their aim to provide easy to use access to the Grid, thus meeting requirement 1 above. However, we also wanted users to be easily able to use other methods like the NGS GSI-SSH Terminal and custom project GUIs and other "thick clients" with ShibGrid, hence this required the development of a proxy download tool. These tools are normally outside the reach of Shibboleth because of Shibboleth's dependence on browsers and HTTP redirection. The proxy download tool enables them to use the extracted credentials to authenticate on behalf of the user to Grid and other resources. Conversely, to be able to support advanced users – who will probably already have a certificate – a proxy upload tool is also provided.

In addition, to ShibGrid-enabling the main NGS portal (supporting the needs of most users), we also had to provide instructions for communities that use other portals. Therefore we ShibGrid-enabled two versions of the NGS portal, one built in Stringbeans, the other in uPortal, to give more scope for further adoption by other project portals.

User requirement 6 stipulated that users would still need to apply for access to the NGS through normal means. If nothing was provided to facilitate this, a user would need to discover the DN provided by ShibGrid and then give this on the NGS registration page. This would break user requirement 1 and therefore it was decided that the project should provide some method for registering with the NGS within the ShibGrid project which does not require the handling of DNs.

The following diagram details the architecture of the ShibGrid system, taking the example of the NGS portal:



The steps are as follows:

1. User requests access to the NGS portal, a Shibboleth Service Provider (SP), through a Shibboleth logon, and the user's browser is then redirected to the Where Are You From (WAYF) service.
2. The user chooses the appropriate Identity Provider (IdP) from the form returned by the WAYF.
3. The user's browser is re-directed to the correct IdP.
4. The user is authenticated by the IdP SSO service through the institutions authentication mechanism.
5. The IdP redirects the user's browser back to the portal (the Service Provider or SP in Shibboleth terminology). The signed authentication SAML assertions are passed in this redirect.

The portal calls out to the IdP's attribute authority for attributes about the user.

6. The username is extracted from the (Base 64-encoded) signed attribute assertion. The extracted username and (Base 64-encoded) signed attribute assertions (used as the password) are sent to the MyProxy server. The ShibGrid MyProxy server returns either a proxy of the user's certificate (if the user has an e-Science certificate and has already uploaded a proxy) or returns an automatically-generated low-assurance certificate from MyProxy's built in CA, if they have not uploaded a proxy.
7. The certificate or proxy certificate is returned to the portal.
8. The user can access the NGS via the portal.

Steps 1-5 is standard Shibboleth access, except that the SP requires a signed attribute assertion rather than an un-signed one. This is standard option in the Shibboleth federation metadata. Therefore the Shibboleth federation and the user's IdP do not need to be modified.

The use of MyProxy in steps 6 & 7 does not modify the standard MyProxy protocol and is very similar to what a standard portal would do. Therefore this means that portal integration should be reasonably straightforward and comprehensible to portal developers. This also simplifies the method used to access a portal for a first time user (without a certificate), so it becomes the same as other Shibboleth SPs, there is no need to pre-register anywhere before accessing the portal.

A similar method is used to download and upload proxies to and from the ShibGrid MyProxy server. In these cases the functionality is split between code running on the web server and on the client's machine (an applet in the user's browser) as the system requires access to resources (like files and certificates) on the user's machine.

Code and Documentation

The architecture presented above gave rise to the following split in the required work:

Package	Components	Institution	Technology
MyProxy Server	-MyProxy Server	CCLRC-RAL, drawing on experience of developing against MyProxy in its SSO project	Patch to MyProxy source using Shibboleth/OpenSA ML C++ libraries
ShibGrid tools	-Upload tool -Download tool -Registration Page	CCLRC-RAL, drawing on experience in Grid security and developing easy to use Grid authentication systems.	Java Servlet, deployed in a tomcat container & complementary Java applet
Portals	-Stringbeans portal development -uPortal portal prototype	Oxford University, drawing on experience of Shibboleth-enabling portals in the SPIE project (with assistance from CCLRC-Daresbury who have developed the NGS portals)	Java JASS modules, JSR 168 compliant portlets and patches to the main portal code.
Documentation	-User guide -Administration guide -Maintenance documentation	All	N/A
Testing scripts	Test scripts	All	See "Code and documentation hardening" section

The uPortal development was marked as "if possible" in the project plan (this was because it was dependent on the development cycle of the NGS portal). It was possible, but only towards the end of the project; therefore it could not be put through the same amount of user testing as the Stringbeans portal and is marked as a prototype.

The documentation for the project was written by the developers in parallel with the code. Where feasible, the code and documentation were written in accordance with the OMII standards for software in its managed programme. However, there are noticeable differences between OMII and ShibGrid which make strict adherence infeasible:

- Different users: OMII makes the assumption that the user is a reasonably competent Linux or Windows user (and users and installers are part of the same community). ShibGrid expects some of its users to be complete beginners and so cannot make this assumption and, as ShibGrid is developing one-per-Grid services, we assume that the installers will be Grid operations experts.
- Different Context: Where OMII is developing code from scratch, ShibGrid is largely developing or patching existing code. This means that decisions have already been made about commenting, packaging, software standards, programming language, etc.
- Different Management, Support and Training: As ShibGrid is a project funded for one year it would be impossible to meet OMII's standards in this area.

User testing

User testing started with informal testing of the ShibGrid prototype by members of the project management committee. This testing was mostly informal but ensured incompatibilities between components were ironed out, especially the IdPs – there are several in Oxford – and SPs. This testing also produced useful feedback, for instance the need for a quick start guide.

Our original plan for the main bulk of user testing, as described in the methodology, was to engage a handful of users from the IB and DIAMOND to test the ShibGrid system against their own requirements for a simplified authentication system. Unfortunately there was limited support from the projects due to constraints within the projects. A further problem proved to be the lack of institutions with Shibboleth IdPs and, where there were IdPs, the fact that IdPs are normally not run by the e-Science departments which we would naturally approach for help with testing.

We decided given these considerations to co-ordinate user testing within our own departments (as we set-up IdPs as part of the project). Especially in CCLRC there is a large breadth of people within the e-Science department, including those who do not normally interact with Grids (e.g. Helpdesk staff, Librarians and developers in non-Grid areas).

As the testing by the project management committee had shown that we have fulfilled the requirements from the IB and DIAMOND projects the aim of the user testing became a usability study. Users were simply provided with the quick start guide and a note about which IdP to choose at the WAYF (for CCLRC this is non-obvious) and asked to comment on improvements which could be made. The aim was to enlist the help of people with a broad range of Grid experience (from none to developer). In particular, we also obtained feedback from people whose job is concerned with usability testing.

For the purposes of user testing the ShibGrid credentials were accepted by the Oxford and RAL NGS nodes as well as an internal cluster within Oxford. Note that if a user was using the real UK e-Science CA certificate then they had access to any resource that accepts this CA (e.g. all the NGS sites). The list of available resources in the portal changes to reflect this.

The comments and feedback obtained through this exercise were fed back into the code and documentation.

User testing also covered the registration page, the Stringbeans portal, proxy certificate upload tool and proxy certificate download tool, and, therefore indirectly, the MyProxy server.

Code and documentation hardening

Components of the system which were not reviewed by users were instead reviewed through the project management committee. The components were reviewed as follows:

Component	Review Format
Code quality	Peer review by developers.
System documentation	Review by (non-developer) members of the management committee.
Code functionality (GUI components: portal, upload tool and download tool)	Scripts for human testing with expected outcomes.
MyProxy server security and protocol adherence.	Checked through automated test script which seeks to test that valid requests are processed correctly and no invalid requests are processed.

Future planning/Dissemination

The ShibGrid project was engaged in the community through a number of means:

- Conference papers;
- Engaging in standards discussions in OGF;
- Discussions and visits with other groups active in similar projects, such as *SWITCHaai* and *SHEBANGS*.
- Feedback to the UK Shibboleth Federation on policy.

6. Outputs and Results

User Requirements

As the user requirements were key to the development of the ShibGrid architecture they were presented in the Implementation section.

Code and Documentation

The project's main outputs were in the form of code and its associated documentation. Here we describe each code component in turn.

MyProxy server changes

The central component in the ShibGrid system is the MyProxy server. The modifications to the MyProxy server consisted of a new authentication method and a new type of username->DN mapping. Both these components are semi-modularised within the MyProxy source. The patch to the source consisted of one C++ file and numerous minor changes to the rest of the source to hook in the new methods and pass through configuration options. All the ShibGrid components can be en/disabled at compile time or runtime and are fully configurable, including the format used to convert Shibboleth attributes to DNs.

A significant part of the development of this component involved implementing parts of the Shibboleth library APIs which are not implemented in the actual SP distribution because they are not used.

We considered three choices for DN mapping schemes for low-assurance certificates in ShibGrid. They were as follows:

1. /C=UK /O=eScienceMyProxy /OU=<Institution>/UID=<Site username>/CN=<First name> <Last name> This scheme has the advantage that it is very close to the DN format currently in use for the UK e-Science CA and gives resource administrators a good indication of whose certificate it is. The disadvantage is that within the UK Shibboleth Federation none of the attributes required for this DN format are released as standard by IdPs and with outsourced IdPs many do not have the access to these attributes.
2. /C=UK /O=eScienceMyProxy /L=<IdP entity-id>/CN=<eduPersonTargetedId> This scheme has the main advantage that eduPersonTargetedId is a core UK Shibboleth Federation attribute so there would not have to be any negotiation with sites over data protection issues. Concerns were raised over the traceability of users through this form of DN as the IdP logs will be the only place which will provide a mapping back to the user (these logs might be on the server of a commercial IdP provider). Crucially, because the user can obtain Grid credentials when logging on through many different SPs (e.g. portals, download tools) this scheme cannot be used as each SP will be sent a different eduPersonTargetedId.
3. /C=UK /O=eScienceMyProxy /CN=<eduPersonPrincipleName> This scheme has the advantage that it provides some clues to the resource administrators who is behind the certificate and it is feasible in most cases that traceability would be possible without access to the IdP logs. The eduPersonPrincipleName attribute is a recognised UK Shibboleth Federation attribute, although not a core attribute. This means that some liaison with home institutions over data protection issues may be required. All SPs receive the same eduPersonPrincipleName which means that the problems of eduPersonTargetedId do not affect this scheme.

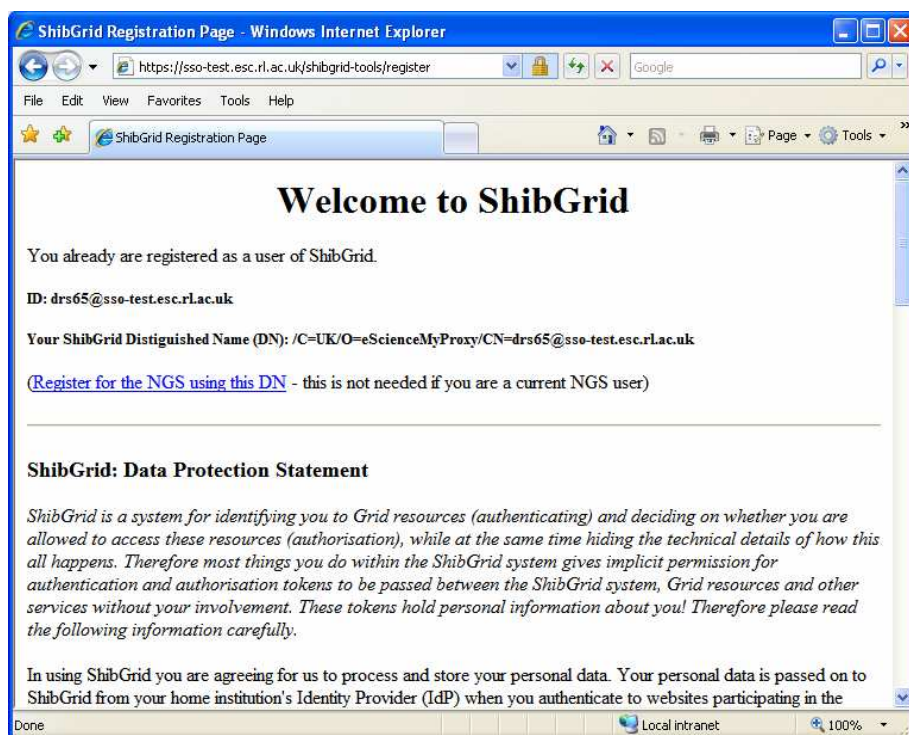
While early testing, both within our own federation and within the SDSS federation, we used Scheme 1. When we later wanted to involve other institutions to use ShibGrid for testing purposes, we had to change over to Scheme 3. This is now our preferred scheme. If a later policy change by the UK federation or other key parties means a different DN scheme becomes the best solution then the ShibGrid system can easily be reconfigured to use a new format as this is all set in configuration files.

Registration Page

The registration page performs three functions:

1. Checks compatibility with a user's IdP.
2. Provides a platform for user to agree to the ShibGrid data protection policy (which is required by the UK Shibboleth Federation due to the way we use attributes) and any other policies (e.g. acceptable use policy).
3. Provides a link to the NGS registration system with the users DN already entered (and a way for advance users to discover their ShibGrid DN).

Once the user has successfully logged in via Shibboleth to the registration page they are asked to agree to the ShibGrid policy. When they agree to the policy they are shown their DN (but they do not need to remember it) and a link to the NGS registration page so they can register without needing to handle their DN. A screenshot is shown below:

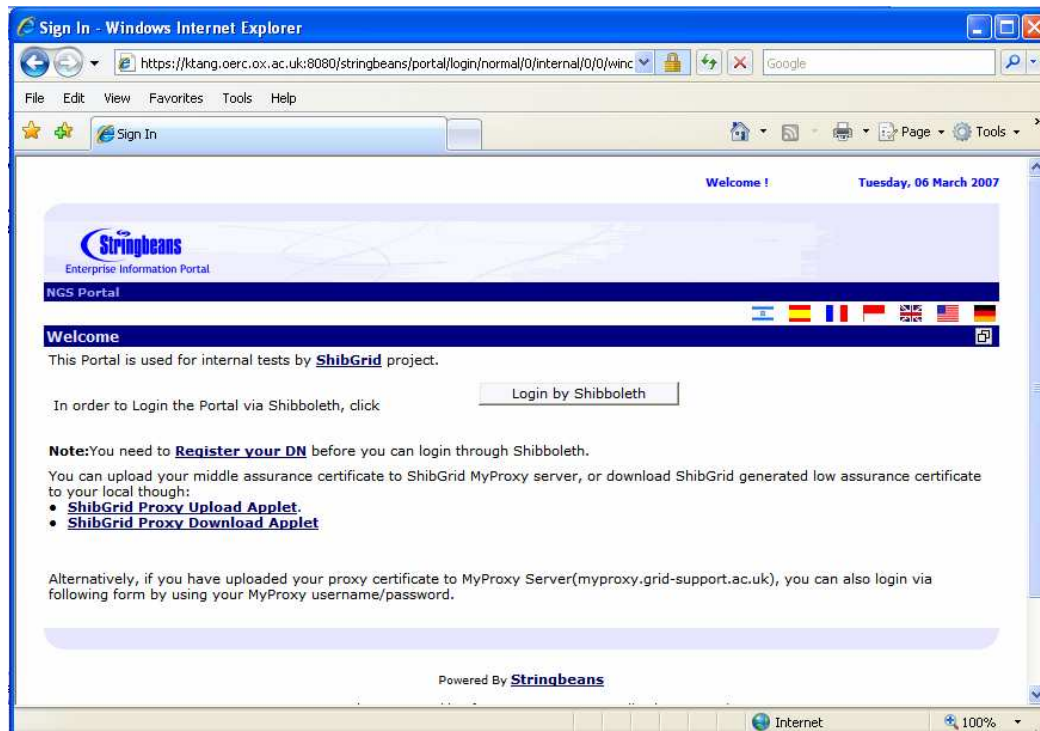


Hardened ShibGrid-enabled Stringbeans Portal

The primary portal to be ShibGrid-enabled in this project was the older Stringbeans portal framework, on which the older NGS portal was based. ShibGrid-enabling this portal built on work done at CCLRC-Daresbury on the NGS portal, in particular work to enable login from a MyProxy server, and work done at Oxford in the SPIE project in Shibboleth-enabling portal frameworks.

The SPIE project has developed JAAS (Java Authentication and Authorisation Standard) modules to provide Shibboleth login to portals. This links in with an Apache Shibboleth SP which is placed in front of the portlet framework. The challenge in the portal work is to obtain the full attribute assertion from this module and then pass it through to a MyProxy JAAS login module as the password. Unlike the login process in SPIE, a ShibGrid login is only successful if both steps pass.

Another important issue which was solved was how to provide users with the ability to renew expired Grid credentials within the portal, as this may require a user to perform another Shibboleth login. So, as with most of the components a simple interface (i.e. a “Login via Shibboleth” button) hides a lot of technology.



Prototype ShibGrid-enabled uPortal portal

Towards the end of the project it became possible to ShibGrid-enable the newer uPortal portal framework on which the newer NGS portal is based. The key issue here is that unlike Stringbeans, uPortal does not support the JAAS module standard for authentication, which is used in the majority of portal frameworks. We adopted a solution from the SPIE project, to implement a ShibbolethSecurityContext to invoke SPIE's Shibboleth JAAS LoginModules. Other than this, the implementation is very similar to that used in Stringbeans.

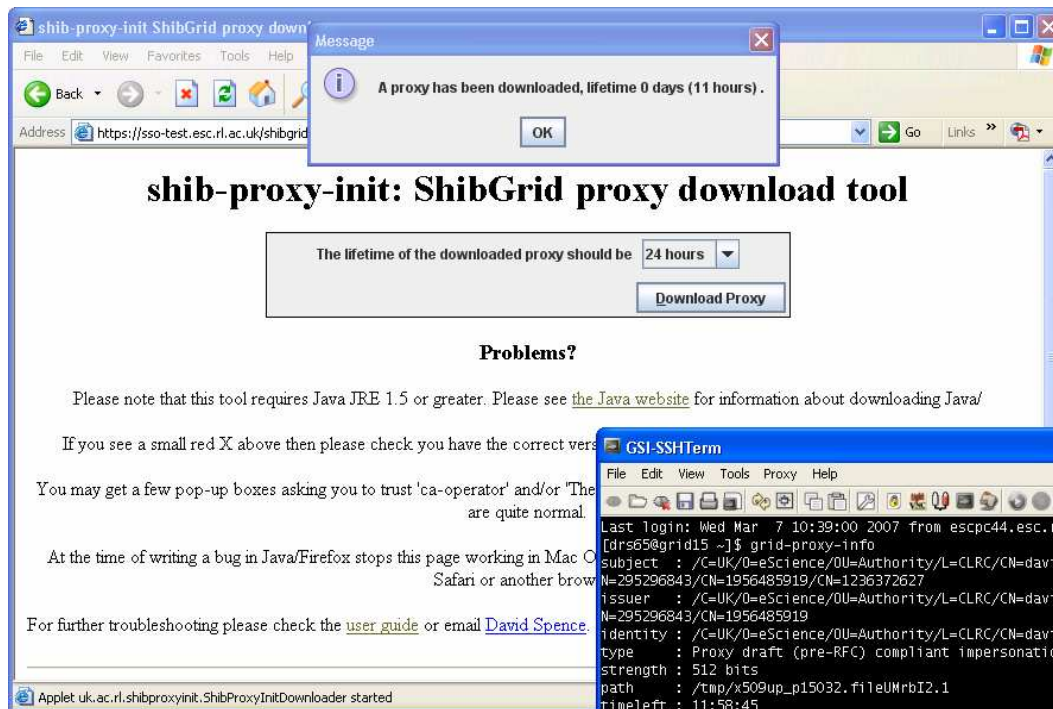
The most important result of achieving uPortal integration is not in tracking developments of the NGS portal, but in providing another ShibGrid-enabled portal framework and another example for current and future Grid portal developers of how to ShibGrid-enable their portal.

Proxy certificate download tool

The proxy download tool needs to have both a presence on the web server (so it can authenticate users using Shibboleth) and on the user's machine (so it can write the downloaded proxy to disk). Therefore the download tools consists of two Java components:

an applet that is displayed within the user's browser and a servlet running in a Tomcat instance, protected through a Apache Shibboleth SP.

The user interface is very simple with a drop-down list to select the desired lifetime of the proxy and a “Download” button. When the user clicks the “Download” button, a proxy request is sent to the servlet to which the user has already logged onto via Shibboleth. The servlet can then go ahead and request a proxy certificate from the MyProxy server, returning it to the applet. The private key for this proxy certificate is never transmitted across the network. The applet then writes the proxy certificate to disk.



Proxy certificate upload tool

The same split between applet and servlet is seen in the proxy certificate upload tool. This time the interface is slightly more complex, but it boils down to specifying where your certificate is to be found (in your browser, in a back-up file, or in a normal Grid setup) and specifying a policy for the uploaded proxy certificate (how long it should persist and the maximum lifetime of proxy certificate that the MyProxy server will generate from it). Users can use this tool to upload proxy certificates or destroy already uploaded proxy certificates

When the user clicks the “Upload” button, then the applet requests the user’s attributes from the servlet, which also delegates its right to use these attributes to the applet. The applet then can upload the proxy certificate to the MyProxy server.



User testing

The following section gives a brief summary of the issues highlighted during user testing.

There were twelve issues reported with the portal, these were mostly peripheral to the ShibGrid system (i.e. errors with the Stringbeans framework or the portal content). There one major issue (reported four times) was that the login screen was confusing. All eight issues were fixed.

There were thirteen issues reported against the ShibGrid tools package (upload tool, download tool, registration page). All were minor changes (e.g. typos/broken links), the look and feel of the tools or with the documentation displayed directly by the package. All were fixed except for one issue (that “registration for ShibGrid is confusing”) which could not be changed due to SDSS/UK Federation policy.

There were four issues reported with the documentation. Two asked for a simpler quick start guide and two requested that they upload tool is mentioned in the quick start guide. All these issues were fixed.

There was one problem with the NGS registration system, that the NGS registration system cannot create SRB usernames for ShibGrid users. This has been reported to the NGS registration system developers and remains outstanding at the time of writing.

The following table shows the coverage of the testing along with the total users:

Component	Number of users giving feedback
Portal	10
Upload tool	7

Download tool	7
Registration tool	10
Documentation	10
NGS registration system	2
Total users giving any feedback	10
Total users (outside the ShibGrid project) who have tested the system	14
Total registered users	22

Future Plans/Dissemination

The dissemination work in ShibGrid covered several areas. The project was actively involved in forums discussing Grid and Shibboleth integration, most notably the beginning of GGF/OGF standards work in this area, presenting twice in BoF sessions. This is highly relevant and effective because the international Shibboleth and MyProxy communities meet in the GGF/OGF. A further paper and presentation about the ShibGrid architecture at the IEEE e-Science 2006 conference also exposed the ShibGrid system to a larger audience.

The ShibGrid project has also been actively involved in the development of the policy for the UK Federation so that the deployment of ShibGrid, and other Shibboleth and Grid integration systems, can be deployed within the policy of the UK Federation. This has taken the form responses to the Community Consultation Exercise and phone conferences with SDSS.

The project has also been following the status of the Shibboleth 2.x development and roadmap. This has culminated in the production of a ShibGrid Shibboleth 2.x roadmap.

7. Outcomes

The aim of ShibGrid was to provide access to the NGS via Shibboleth, both for “expert” users who can be expected already to have certificates, and for novice users who don’t want to know that they have a certificate.

The ShibGrid project has created a Grid authentication solution, using Shibboleth technology, which satisfies the needs of those users, and it enables the use of a variety of Grid access methods (portal, GSI-SSHTerm, local Grid clients, etc.,) out of the box.

The project had to deliver production level code and associated documentation which fulfilled the project use cases and the objectives. This gives two criteria for the project, that the code and documentation is of production-level and that it fulfils the project use cases and objectives.

There were many areas where we tried to ensure that the project’s code was of production quality, the adoption (where possible) of OMII standards, peer review of code, management review of documentation, automated and human testing and user testing. The result is a robust and secure system which the project management team is confident is production-ready.

In the area of security, the ShibGrid architecture was presented at both an international Grid conference where other Shibboleth & Grid developments were presented (IEEE e-Science 2006; a paper focused on architecture and security) and also at GGF/OGF Birds of a Feather sessions on Shibboleth and Grid integration. At neither of these sessions was any questions raised as to the security of the ShibGrid architecture. Automated tests have been developed to check the implementation’s conformance to the security aspects of the architecture.

As the project use cases were foundational, along with the user requirements, to the design of the architecture, the ShibGrid system trivially fulfils these use cases, the use cases being fulfilled by the following components:

Use case	Components
PUC1	MyProxy server, Upload tool, Download tool, Portal
PUC2	MyProxy server, Portal
PUC3	User Registration Page
PUC4	MyProxy server, Download tool

Going back to the Objectives from section 5 (Implementation, under User Requirements):

No	Descr.	Status
1	Allow users with no prior knowledge of PKI to use NGS	Done, by ShibGrid enabling the NGS portal
2	Allow users with e-Science certificates to use the system	Done via MyProxy and associated upload tool
3	Provide stable mapping in DN presented to	Done by implementing appropriate

	resources	naming scheme
4	Users who change institution should have same access	Done by implementing appropriate naming scheme
5	Resource admin should see user information	Met as best we could under constraints imposed by UK Fed. policy
6	Users apply for access using normal procedures	Done – users are forwarded to the normal registration page with their ShibGrid DN
7	Work with NGS operations for users with more than one DN	Handed over to NGS ops. Workaround available.

This latter objective (number 7) is not fully met at the time of writing, but we expect it will become higher priority for NGS operations once ShibGrid-enabled services are wider deployed. The consequences of not meeting this objective are slight, though: few users will ever have more than one DN: we ensure consistent DN allocation by having chosen naming scheme 3, using the eduPersonPrincipalName. Those that do have more than one DN can be managed (renamed and remapped) manually by NGS support.

When ShibGrid is fully deployed, along with a fully deployed UK Federation, it will have a major impact on the take-up of Grid computing by those with no desire to learn the peculiarities of the Grid Security Infrastructure, now there is an easy to use way to get basic and advanced Grid access through a technology which all academics and students will soon become familiar with. This will mean better, faster research results as researchers either use Grid resources where they would not have before or have to spend significantly less time learning more technology freeing up time for core research.

Conversely, ShibGrid will contribute in a large way to attracting large numbers of users from all areas of research to the NGS, as long as their work can be done via the portals, or their applications are otherwise Grid-enabled.

Deployment of ShibGrid could also give some impetus to the adoption of Shibboleth as it provides a compelling application (i.e. free at point of use computing resources) through Shibboleth that was not available in the previous Athens authentication framework.

Along with the SHEBANGS project, the ShibGrid project has also had an impact in ensuring that the UK's interests are represented in the forums discussing standards for Grid and Shibboleth integration and that the needs of Grid systems were represented in the formation of the policy of the new UK Shibboleth federation.

It is always a risk for a project to rely on external projects, and we found the risks materialised with the external user testing. DIAMOND and IB were, for all sorts of good reasons, not able to provide the time for testing, especially since they do not have many, if any, end users yet who can actually do this. Even so, it was still invaluable to engage these (longer-term) projects in the area of user requirements, as obtaining meaningful user requirements independently in a one year project would have been a difficult challenge.

8. Conclusion

The ShibGrid project was a success. Building on existing components, Shibboleth, MyProxy, and standard portals, it has produced components to combine the UK Shibboleth and NGS infrastructures to enable users, from novice to expert, to access the NGS via portals. It meets the requirements of initial target users, DIAMOND and Integrative Biology, for single sign-on and Grid access. The work was deployed for the NGS portal but is not tied to a single portal - we managed to ShibGrid-enable both uPortal and StringBeans-based portals. The application areas are not yet exhausted: via credential upload and download tools, we have demonstrated that not just portals but also "thick" clients (desktop applications, normally outside the each of Shibboleth) and "thin" clients (e.g. browsers) can benefit from the credential management built in this project.

Security was considered throughout, and the product is able to generate lower assurance certificates, and manage higher assurance certificates for people who have these already. Other security issues that were addressed include passing signed assertions between components, addressing identity and anonymity/pseudonymity issues, and resource logging and access auditability.

The impact of this project spreads over many areas: first and foremost, it succeeded in meeting its primary goal, to provide Shibboleth access to the Grid. It will enable the NGS to grant access to users who do not have, and do not want, an e-Science certificate, thus lowering the barrier for beginners, and widening the user base. Furthermore, using standard components and protocols ensures the product is easily deployable, maintainable, and interoperable. The project has thus gained visibility in international Shibboleth and MyProxy communities, primarily via the Open Grid Forum where such people usually meet, but also via publications and other forms of dissemination.

9. Recommendations

With the culmination of the ShibGrid project it is important that we make these recommendations, using the experience in the use of Shibboleth within the NGS, on the continued and further use of the outputs from this project. This project has been successful in creating a prototype/pre-production infrastructure that allows access to NGS resources by both those users who have a certificate and those that do not; JISC would best leverage the investment in this work by ensuring that any further Shibboleth project makes full use of the outputs of the ShibGrid project.

To continue the uptake of Shibboleth within the NGS it is also important that Virtual Organisation user attributes are recognised and as such the current NGS solution, using VOMS, is not unnecessarily broken. The solution can be provided using the SHEBANGS software and as such any follow on projects should be steered in the strongest of ways towards the use of both of these projects outputs.

Further to the purely technical aspects of the project there are also policy recommendations that must be passed from JISC to the appropriate bodies. The UK Shibboleth federation should be asked to ensure that there are not unnecessary restrictions made on user attributes made available from site identity providers that may block or constrain the take up of Shibboleth. This is primarily due to the grid giving access to resources far beyond the passive repository type (e.g. online journal access), to the type of resources for whom it is essential that accurate and easy user identification by the systems owners are essential. It was also noted that in user testing (by resource providers) that identification of users could be confusing, a principal component of which was again due to the UK Federation policies and as such an issue that should be further investigated as the system moves towards production.