EGI-InSPIRE

SPECIFICATION OF REQUIREMENTS FOR EGI APPLICATION DATABASE

© Members of EGI-InSPIRE collaboration



COPYRIGHT NOTICE

Copyright © Members of the EGI-InSPIRE Collaboration, 2010. See www.egi.eu for details of the EGI-InSPIRE project and the collaboration. EGI-InSPIRE ("European Grid Initiative: Integrated Sustainable Pan-European Infrastructure for Researchers in Europe") is a project co-funded by the European Commission as an Integrated Infrastructure Initiative within the 7th Framework Programme. EGI-InSPIRE began in May 2010 and will run for 4 years. This work is licensed under the Creative Commons Attribution-Noncommercial 3.0 License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc/3.0/ or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, and USA. The work must be attributed by attaching the following reference to the copied elements: "Copyright © Members of the EGI-InSPIRE Collaboration, 2010. See www.egi.eu for details of the EGI-InSPIRE project and the collaboration". Using this document in a way and/or for purposes not foreseen in the license, requires the prior written permission of the copyright holders. The information contained in this document represents the views of the copyright holders as of the date such views are published.

DELIVERY SLIP

	Name	Partner/Activity	Date
From	<<The lead author/editor>>		
Reviewed by	**Moderator:** **Reviewers:** <<To be completed by project office on submission to AMB/PMB>>		
Approved by	**AMB & PMB** <<To be completed by project office on submission to EC>>		

DOCUMENT LOG

Issue	Date	Comment	Author/Partner
1	09/10/2010	First draft v0.1	Karolis Eigelis/EGI.eu
2	18/11/10	Developers' Responses v0.2	William V. Karageorgos/IASA Marios Chatziangelou/IASA Nafsika Zarife/IASA
3			

APPLICATION AREA

This document is for internal purposes inside NA3.

DOCUMENT AMENDMENT PROCEDURE

Amendments, comments and suggestions should be sent to the authors. The procedures documented in the EGI-InSPIRE "Document Management Procedure" will be followed: https://wiki.egi.eu/wiki/Procedures

TERMINOLOGY

A complete project glossary is provided at the following page: http://www.egi.eu/about/glossary/.

<<The authors should check if the acronyms are covered by the glossary page and if the definition is still correct; all the amendments should be communicated to glossary@egi.eu>>



PROJECT SUMMARY

To support science and innovation, a lasting operational model for e-Science is needed – both for coordinating the infrastructure and for delivering integrated services that cross national borders.

The EGI-InSPIRE project will support the transition from a project-based system to a sustainable pan-European e-Infrastructure, by supporting 'grids' of high-performance computing (HPC) and high-throughput computing (HTC) resources. EGI-InSPIRE will also be ideally placed to integrate new Distributed Computing Infrastructures (DCIs) such as clouds, supercomputing networks and desktop grids, to benefit user communities within the European Research Area.

EGI-InSPIRE will collect user requirements and provide support for the current and potential new user communities, for example within the ESFRI projects. Additional support will also be given to the current heavy users of the infrastructure, such as high energy physics, computational chemistry and life sciences, as they move their critical services and tools from a centralised support model to one driven by their own individual communities.

The objectives of the project are:

1. The continued operation and expansion of today's production infrastructure by transitioning to a governance model and operational infrastructure that can be increasingly sustained outside of specific project funding.
2. The continued support of researchers within Europe and their international collaborators that are using the current production infrastructure.
3. The support for current heavy users of the infrastructure in earth science, astronomy and astrophysics, fusion, computational chemistry and materials science technology, life sciences and high energy physics as they move to sustainable support models for their own communities.
4. Interfaces that expand access to new user communities including new potential heavy users of the infrastructure from the ESFRI projects.
5. Mechanisms to integrate existing infrastructure providers in Europe and around the world into the production infrastructure, so as to provide transparent access to all authorised users.
6. Establish processes and procedures to allow the integration of new DCI technologies (e.g. clouds, volunteer desktop grids) and heterogeneous resources (e.g. HTC and HPC) into a seamless production infrastructure as they mature and demonstrate value to the EGI community.

The EGI community is a federation of independent national and community resource providers, whose resources support specific research communities and international collaborators both within Europe and worldwide. EGI.eu, coordinator of EGI-InSPIRE, brings together partner institutions established within the community to provide a set of essential human and technical services that enable secure integrated access to distributed resources on behalf of the community.

The production infrastructure supports Virtual Research Communities (VRCs) – structured international user communities – that are grouped into specific research domains. VRCs are formally represented within EGI at both a technical and strategic level.

TABLE OF CONTENTS

INTRODUCTION

The NA3 activity of the EGI-InSPIRE project provides the following technical services for users:

- Training services
- Database of applications, tools, and their developers (AppDB)
- VO services

A new release of each of these tools is expected from the tool providers every 6 months. The new releases must be prepared according to plans endorsed by the User Community Board. These plans should be prepared by the developers/providers of the tools.
The plans should provide solutions for the issues, bugs and requirements that have been collected from user communities and from NGIs.

The purpose of this document is to provide a list of requirements that have been collected by the UCST about AppDB from VOs, NGIs and HUCs. These requirements should be extended with other requirements that the AppDB developer team collected from users directly. The complete list then must be used as a basis to define AppDB development plan for the next 6 months (until the EGI User Forum).

The requirements can be further discussed with the UCST and with the TNA3.4 partners during the fortnightly teleconferences. The development plan must be discussed at the next User Services Advisory Group teleconference, then at the User Community Board. These events will take place in the second half of November.

For more details about the requirement gathering and processing workflow please refer to https://wiki.egi.eu/wiki/Requirements_gathering_details.

SPECIFICATION OF USE CASES

The specification of use cases is based on the various potential users of applications database (AppDB) service classified in the categories for specific fields of interest.

Users of Applications database service

Classification of the AppDB service potential users:

a) Researchers (end-users)
b) Developers (of grid scientific applications, portals, applications services etc.)
c) Site administrators (supports the Researchers and applications or tools provided by Developers)
d) NGI representatives (the responsible person who takes care of applications which (co-)belong to countries of the NGI.

These types of the users are not the roles within applications database service, these are the group of persons who would use AppDB.

Key actions by Users

Key actions that users classified in previous section would like to perform using AppDB service are grouped in general actions that all classified type of users would like to perform on specific service and to specific ones to be performed just by concrete type of users.

General key actions for AppDB service

General key actions are agreed to be on demand for all types of users for AppDB service.

1. **To get Detailed information(about applications, developers, NGIs, tools)**
    1. **Applications**
        1.1. Published scientific papers
        1.2. General Manual of application (how to use it in general)
        1.3. Where it is available(which NGI/s and grid site/s and VO/s support that application and what versions are supported)
        1.4. What resources are available for that application (number of CPUs, storage space, 32bit/64bit architecture etc.)
        1.5. Specific manual How to use the application on that grid site or within NGI or VO
        1.6. Specific manual How to deploy application on the grid (information is needed by site administrators etc.)
        1.7. Specific manual How to join VO or get started using application (user finds the application he wants to use – he needs further steps to guide him what to do – i.e. contact NGI representative or register to that VO)
        1.8. Developer of Application

1.9. Scientific Contact

2. **Tools**
   2.1. The same as for Applications
3. **Developers**
   3.1. Personal details (name, country, native language, contacts etc.)
   3.2. Experience (which applications or tools he developed, contributed or was involved, what background he has, what is he's specialization etc.)
   3.3. Current involvement/work (on which applications or tools he is working or is involved in etc.)
   3.4. Home Institution
   3.5. Name of NGI he belongs to
   3.6. NGI representative contacts
4. **NGIs**
   4.1. NGI representative personal details (name, country, native language, contacts etc.)
   4.2. Details about NGI (country, resources available, sites, institutions)

2. **Filtering mechanisms (easily to find the information)**
   1. Applications (filter by: scientific field, availability(at which NGI, site, VO application is available), available resources, developer)
   2. Developers (filter by: applications, NGI, country, home institution)
   3. NGIs info (filter by: country, developer, applications, institution, VO)

Specific key actions by user type

Specific key actions are applied only by specific user types on AppDB service. These actions mostly represent the roles and information management.

1. **Profile management by Developer** (information management only within he's own profile)

1. Create (create new profile, only for Developer)
2. Update (update profile's details, only for Developer)
3. Delete (delete profile, only for Developer)

2. **Profile management by NGI representative** (information management across different Developers profiles within the represented NGI)

1. Create (create new profile, also for Developer)
2. Update (update profile details, also for Developer's profile)
3. Delete (delete profile, also delete Developer's profile)
4. Add & Remove profiles (Add/Remove Developer's profile to the NGI, Application)

3. **Applications and Tools profile management by NGI representative** (information management across different Applications profiles within the represented NGI)

1. Create (create new profile, also for Application or Tool)
2. Update (update profile details, also for Application or Tool)
3. Delete (delete profile, also delete Application's or Tool's profile)
4. Add & Remove profiles (Add/Remove Application's profile to the NGI, Developer)

Missing features based on key actions by users

Legend

Status:

- Requested – feature requested to develop.
- In Progress – feature under development. Feature to be available in the next release
- Postponed – feature to be developed in some future release, but not in the next one.
- Rejected – Feature that is rejected by the developer team.

Version:

- This column specifies the release version which will implement the specific feature.

Nr.:

- This section specifies the identification number for the feature request

AppDB service

The list of missing features for the AppDB service is provided in the Table 2. "AppDB service".

Nr.	Version	Status	Description of the requested feature	Use case ref.	Response Developer
1		Requested	Detailed information about Applications	"general actions", 1.1 Applications use case: 1.1.1 – 1.1.9	
2		Requested	Detailed information about Tools	"general actions", 1.2 Tools use case: 1.2.1	
3		Requested	Detailed information about Developers	"general actions", 1.3 Developers use case: 1.3.1 – 1.3.6	

4	Requested	Detailed information about NGIs	“general actions”, 1.4 NGIs use case: 1.4.1 – 1.4.2
5	Requested	Filtering mechanisms	“general actions”, 2. Filtering use case: 2.1 – 2.3
6	Requested	Profile management by Developers	“specific actions by Developers”, 1. profile management use case: 1.1 – 1.3
7	Requested	Profile management by NGI representative	“specific actions by NGI representative”, 2.; 3. profile management use case: 2.1 – 2.4 and 3.1 – 3.4

Table 2. AppDB service

SPECIFICATION OF OTHER, CURRENTLY MISSING FEATURES

The listed features have been identified as crucial elements to realised the use cases of the previous section, and are unavailable in the current AppDB service[R1]. (Time of testing 26/10/2010)

Operational missing features

The operational missing features are the ones those reflect the overall operational processes of the AppDB service and are listed in the Table 1. “Operational missing features”.

Legend

Status:

- Requested – feature requested to develop.
- In Progress – feature under development. Feature to be available in the next release
- Postponed – feature to be developed in some future release, but not in the next one.
- Rejected – Feature that is rejected by the developer team.

Version:

- This column specifies the release version which will implement the specific feature.

Nr.:

- This section specifies the identification number for the feature request

Nr.	Version	Status	Description of the requested feature	reference	Response Developer
1		Requested	**Statistics.** The traffic of AppDB service must be monitored by Google Analytics.		
2		Requested	**Availability.** The availability of the AppDB must be monitored. (e.g. with Nagios)		
3		Requested	**Layout.** The AppDB service website should mimic the main EGI website in order to better integrate into it.		
4		Requested	**Compatibility.** AppDB must be compatible with all the widely used browser types.		
5		Requested	**Roles.** Flexible roles management mechanism should be able to easily create new, remove old and update		

		existing roles.
6	Requested	**Localisation.** NGIs, VRCs or VOs should be able to use a localised version of the AppDB to list and manage their own applications, tools and user profiles. The local db must propagate these into the central AppDB. The localised DBs should support custom look-and-feel. Localised DBs can be either installed by the NGIs/VRCs/VOs, or provided as a custom view of the central AppDB. (See xGUS as example.)
7	Requested	**Integration with sites.** AppDB service should be able to interface and display information from external sources: 1) retrieve information about sites from NGIs, VOs where a particular application is available 2) retrieve information from monitoring tools that test the accessibility and correctness of the installed applications/tool.

Table 1. List of requirements

DEVELOPERS' RESPONSES

AppDB service

Nr.	Response
1	1.1.1: Links to publications already exist 1.1.2: Documentation links already exists (when provided by developers). 1.1.3 - 4: Current monitoring technology is rather sparse (gLite BDII, Globus MDS, Nagios, UNICORE CIS, etc.) and focused on user accounting instead of application accounting, so providing this kind of information would be infeasible at the moment, due to the need for separate implementations and maintenance. However, there exist efforts such as D-MON [R 1] from D-Grid which try to collect monitoring information from all available sources, and provide them in a uniform manner. If such an effort succeeds, this feature should be reconsidered in the future. 1.1.5: If 1.1.3 - 4 is implemented, then this information could be included, if provided by NGI/VO people 1.1.6: This information should be provided by application developers. We could require a link analogous to the one in 1.1.1 1.1.7: This kind of information should be provided by VO people, but VO information in AppDB is limited only to a list of existing VOs. Additional data about VOs could be kept, by extending the data/object model (i.e. create VO detailed listings like it is done for apps,tools,and people), and inviting VO people to join (create user profiles) 1.1.8 - 9: Information currently exists. Additionally, the possibility of providing EGI-reviewed documentation for applications has been discussed. For example, we could store links to entries in a protected area of the EGI wiki, where EGI people can write reviews for all to see, but application developers won't be able to modify.
2	1.2.*: Same as 1.
3	1.3.1: Native language can be added, as well as other such simple data, if specified. 1.3.2: A professional profile can be created. 1.3.3: A "status" field can be added to related applications (active, retired, NULL, ...) 1.3.4: Already exists. 1.3.5 - 6: Currently there is no model for NGIs, only country info is managed. NGI entities could be added to the database, in a similar fashion as VOs in 1.1.7.. Users would then be automatically mapped to the appropriate NGI according to country information.
4	1.4.1: NGI representative profiles needn't differ from other people's profiles. 1.4.2: Should be implemented according to 1.3.5 - 6.
5	2.1 - 3: Filtering already exists for applications, tools, and people. It follows that if stored entities/fields are augmented, filtering will be extended accordingly.
6	1.1 - 3: Already implemented for people holding an EGI SSO account. 2.1 - 4: There has already been some thought about these features, and have been planned for implementation.
7	3.1 - 4: At the moment, NGI representatives. can already manage application/tool info for entries that belong to the same country. Additionally, developers who own an application/tool (i.e. they have registered it themselves) can fully modify said entries, and Scientific Coordinators can do the same for applications they are associated with (in the sense mentioned in 1.1.3)

Table 3: AppDB service response table

Operational Missing Features

Nr.	Response
1	Already implemented.
2	The AppDB is being monitored by our local Nagios server for the marie.hellasgrid.gr domain, (where the AppDB is hosted). If there is a need for monitoring from a centralized Nagios instance, the administrators can contact us for the details.
3	CSS/Styling imformation will be linked to the main EGI site in the next minor release.
4	Currently, fully supported browsers are * Mozilla Firefox 3.0 or higher * Opera 10.0 or higher * Google Chrome 6.0 or higher * Apple Safari 5.0 or higher * Mozilla SeaMonkey 2.0 or higher * Google Android 1.6 or higher Microsoft Internet Explorer 8.0 is also fully supported, but in an experimental state. It will be officially supported by the next minor release. Previous versions are not going to be supported.
5	The mechanism already exists. New roles can be added upon request. The possibility of supporting multiple roles for each user should be discussed. This would introduce considerable complexity, both development and user wise, so the pros and cons should be carefully weighted.
6	At the time being, there exist POST interfaces which provide application, tool, and people information in XML form, with support for filtering. This mechanism can be readily used for deploying localized read-only sites. Information would be fetched on-the-fly, and could be cached to improve performance, without the need for a local DBMS instance. Access details, examples, etc can be provided upon request. A more advanced WebAPI (such as a REST interface) has been considered, but would require significant effort. As an alternative, the MVC based server-side PHP API which we use to interface our database, could be provided to interested parties, should they be willing to install a local DBMS copy.
7	Please refer to Nr. 1, 1.1.3 - 4 entry of the AppDB service response table

Table 4: Operational missing features response table

REFERENCES

R 1	D-MON https://www.egi.eu/indico/contributionDisplay.py?contribId=80&sessionId=141&confId=48