



# NextData System of Systems Infrastructure (ND-SoS-Ina)

**DELIVERABLE D2.3 (CINECA, CNR-IIA) - Web Portal Architecture**

**DELIVERABLE D4.1 (CINECA, CNR-IIA) - Test Infrastructure**

Document identifier:	D2.3 D4.1
Date:	Jan 2014
Editors:	Marco Rorro, Michele Carpené, Stefano Nativi
Activity:	WP3 - Task 2.3 WP4 - Task 4.1
Lead Partner:	CINECA,CNR
Document status:	draft
Document link:	
Diffusion:	Internal

Document version	Date	Authors
------------------	------	---------

1.0	Jan 2014	Michele Carpené, Marco Rorro - CINECA
-----	----------	---------------------------------------

**Abstract:**

This document presents the **NextData SoSIna Web Portal general architecture** and provides a common foundation for further portal solutions.

Furthermore it describes the **test infrastructure** where a first portal prototype operate.

## TABLE OF CONTENTS

- [1. Introduction](#)
- [2. NextData SoSIna Web Portal development - General Requirements](#)
  - [Requirements and Use Cases](#)
    - [Functional Requirements](#)
    - [Non-functional Requirements](#)
    - [Prioritizing the Functionalities](#)
- [3. Use Cases](#)
- [4. NextData SoSIna Web Portal development - General Architecture](#)
  - [Requirements mapping](#)
  - [Web Portal design](#)
    - [Layout](#)
    - [Technologies](#)
      - [Liferay](#)
      - [GI-cat](#)
      - [GI-axe](#)
      - [GI-API](#)
      - [OpenLayers](#)
      - [Geoserver](#)
  - [Architecture](#)
    - [General](#)
    - [Broker Configuration](#)
    - [Authentication and Security aspects](#)
    - [Logging and messages](#)
    - [Resources](#)
    - [Recommendations](#)
  - [Development plan](#)
  - [Conclusions and future extensions](#)
  - [References](#)

## 1. Introduction

The WP2 Task 2.3 pertains with the development of the NextData SoSIna Web Portal. The SoSIna Web Portal will make available environmental and climate data provided by the NextData project to support decision-makers, stakeholders, and the scientific community in the development of environmental analyses.

The global strategy defined to develop the SoSIna Web Portal is focused on three axes:

- Identification of users requirements and use cases;
- Web Portal design and technology evaluation;
- Interaction with the brokering middleware that provide discovery, evaluation and access services across heterogeneous capacities.

The Web Portal implementation will reflect the requirements elicited by user communities and collected in WP2 (Requirements and System Architecture).

## 2. NextData SoSIna Web Portal development - General Requirements

### Requirements and Use Cases

Functional and non-functional requirements have been collected and prioritized in order to foster the continue delivery of well-defined and small developments to guarantee they satisfy users expectation. More details on requirements description can be found in deliverable D2.1 (General Requirements) and D2.2 (General System and Interoperability Architecture).

### Functional Requirements

The following functional requirements have been recognized as important for the design and development of the NextData Web Portal Architecture and in general for the infrastructure architecture (i.e. SoSIna).

#### 1. Basic:

1. **Authentication and Authorization functions** (From the NextData policy: in order to download data the users are required to register at the Portal, providing their true identity, affiliation and contact information)

#### 2. Discovery

- i. Simple search (Users add a keyword and click on the search button)
- ii. Advanced search (Users add spatial and temporal constraints)
- iii. Semantic-enriched search (Semantic options can be specified)
- iv. Results paging, ranking and clustering

#### 3. Access

- i. Download function to access data
- ii. Transformation functions to access datasets according to a common (spatial and temporal) domain and format
- iii. Implementation of a basket table to download datasets with a common grid environment

#### 4. Evaluation

- i. Access to metadata, quality and provenance information. Add data quality evaluation options based on user preferences

#### 5. Visualization

- i. Portrayal function on a reference map in latitude and longitude to get a preview of data.

### 2. General (to be defined and implemented in a second version):

1. Statistics function providing a link to the most popular searches
2. Help function providing a simple way to get information about the Portal widgets and gadgets
3. Portal-feedback functions to keep Users informed about the status and the outcome of their functional requests.
4. User-feedback function providing a simple Web form to submit comments and or queries dealing with the Portal functionalities and/or results
5. Credits information and contact points.

### Non-functional Requirements

According **The NextData e-infrastructure System Requirements** the Portal shall implement also a set of non-functional requirements:

- (a) User Interface and Human Factors
- (b) Documentation (i.e. User guide, Installation guide, etc.)
- (c) Quality indicators (i.e. to guarantee a certain level of performance, reliability and availability of the entire system)
- (d) Security (i.e. Security of the access) - Generally some credential management policies should be defined (periodic renewal of the credentials, certificate based authentication, credential revocation, etc.)

Now we are going to define a priority on functionalities to better guide the Web Portal development process.

### Prioritizing the Functionalities

SoSIna considers two versions of the NextData Portal, based on a prioritization of the functional requirements:

- First version of the Portal
  - Simple search
  - Advanced search
  - Access
  - Semantic-enriched search
  - Authentication and Authorization functions
  - Evaluation

- Visualization
- Second version of the Portal - all the functionalities implemented in the first version plus:
  - Statistics function
  - Help function
  - User-feedback function
  - Documentation
  - Performance Improvement
  - Portal-feedback
  - Credits

From requirements listed above several use cases have been derived according to those described in deliverable D2.1 (General Requirements).

### 3. Use Cases

The following use cases have been identified, these use cases are intended to cover the most of the requirements collected.

1. NextData User issues a discovery request clicking on a search button;
2. The NextData e-infrastructure answers with a set of matching results coming from all the brokered resources in the form of a paginated catalog;
3. NextData User browses the result set and adds elements to a basket, in order to further download dataset with a common grid environment;
4. NextData User adds more fine parameters clicking on the Advanced Search button, which shows the Advanced Search panel. From the Advanced panel is possible to specify spatial and temporal domain, data format, etc..
5. NextData User asks the system to visualize dataset information on a reference map.

In the next section, we are going to explore how the higher priority functional requirements and use cases have been translated and mapped as logical elements compounding the SoSIna Web Portal. Then we deepen the Web Portal design and we explain technologies adopted in details.

### 4. NextData SoSIna Web Portal development - General Architecture

The aim of this part is:

- To explain how the high priority requirements and use cases have been mapped to the portal as a set of logical elements;
- To briefly describe the SoSIna Web Portal proposed architecture and design.

## Requirements mapping

As result of a mapping process requirements and related use cases have been traduced into logical elements to be integrated into the Web Portal interface.

In order to implement the use cases a technology assessment has been conducted to evaluate the most promising technologies available today for shaping a lightweight and agile Web Portal. Because of such assessment, Liferay emerged as the right tool to address this challenge.

Liferay Portal [R1] consists in a JSR-286 (Java Specification Request) compliant web platform for the development of websites and portals.

Thanks to Liferay developers are provided with a sophisticated and extensible tool, moreover all the requested functionalities can be mapped within Liferay into graphical elements (like pages, portlets, etc.).

Portlets can be seen as a portion of a web page where input fields, buttons, images and outputs can be represented. Composing a web page with portlets allows differentiating functionalities and it avoids to reload the whole page when just a portion of it changes.

The first basic functionalities (Discovery and Access) have been implemented inside the “Search” portlet. Introducing the “Search” portlet we are able to cover the most part of the high priority requirements and functionalities expressed in the above paragraphs.

Because of this reason in the next section, we will explore more in details this portlet, including graphical user interface elements and input fields.

The “Search” portlet is built upon services and interfaces made available by the brokering middleware Geo DAB (Discover and Access Broker) and by the GI-API, which are in charge of interconnecting heterogeneous and distributed resources.

Through the GI-API the “Search” portlet can exploit protocols published by the DAB.

Liferay portal as well as the brokering middleware will be described more deeply in the further sections.

## Web Portal design

In this section, a general overview of the Web Portal architecture is presented.

First, the Web Portal simplified layout is presented with particular reference to the most important expected functionalities. Then the technologies adopted are described, and finally connections between the main logical components are shown in a diagram.

### Layout

This section contains a brief description of the Web Portal layer structure. Here we show also a first image of the Web Portal interface including main components.

Each menu on the left side (Figure 1) is linked to a page, so, for example, the Home menu points to the Home page where general information is reported.

The Search menu is linked to the Search page where the “Search” portlet has been placed.

The “Search” portlet implements the most important functionalities and contains:

- A **Search** button to start querying the Geo DAB;
- The **Input** search field where users write research keywords;

- **Advanced Search** button to show/hide additional research options.

The Advanced Search button shows an additional area containing three more panels: Sources, Constraints and Options:

- The **Sources** panel includes a checkbox tree that allows to include/exclude resources in the search;
- The **Constraints** panel contains a map created with OpenLayers (<http://openlayers.org/>), the map is used in order to specify latitude and longitude coordinates to be included as research constraints. The panel includes also a couple of fields to select Start/End dates and set proper temporal constraints;
- The **Options** panel includes two more elements: **Search** and **Extension**. Search is a select list used to specify a proper field subset; for example, users can restrict search by keyword between titles or descriptions. Extension allows selecting **semantic options**. From Extension users can add keywords and related concepts from a remote Controlled Vocabulary, expanding the research according to one or more relations.

Figure 2 shows a screenshot of the “Search” portlet prototype implementation.

The results returned from the brokering middleware are formatted in a paginated catalog represented on the bottom of the page as shown in Figure 3. Each result in the output is directly linked to the associated data or metadata file.

From here, users will be able to add each single result to a Basket, then all the files will be downloaded all at once. Download Options can be also specified in a Dialog Box that will appear as the user clicks on the chosen element.

Other pages, as for example “Statistics”, will be introduced in future documents when the second portal version will be released.

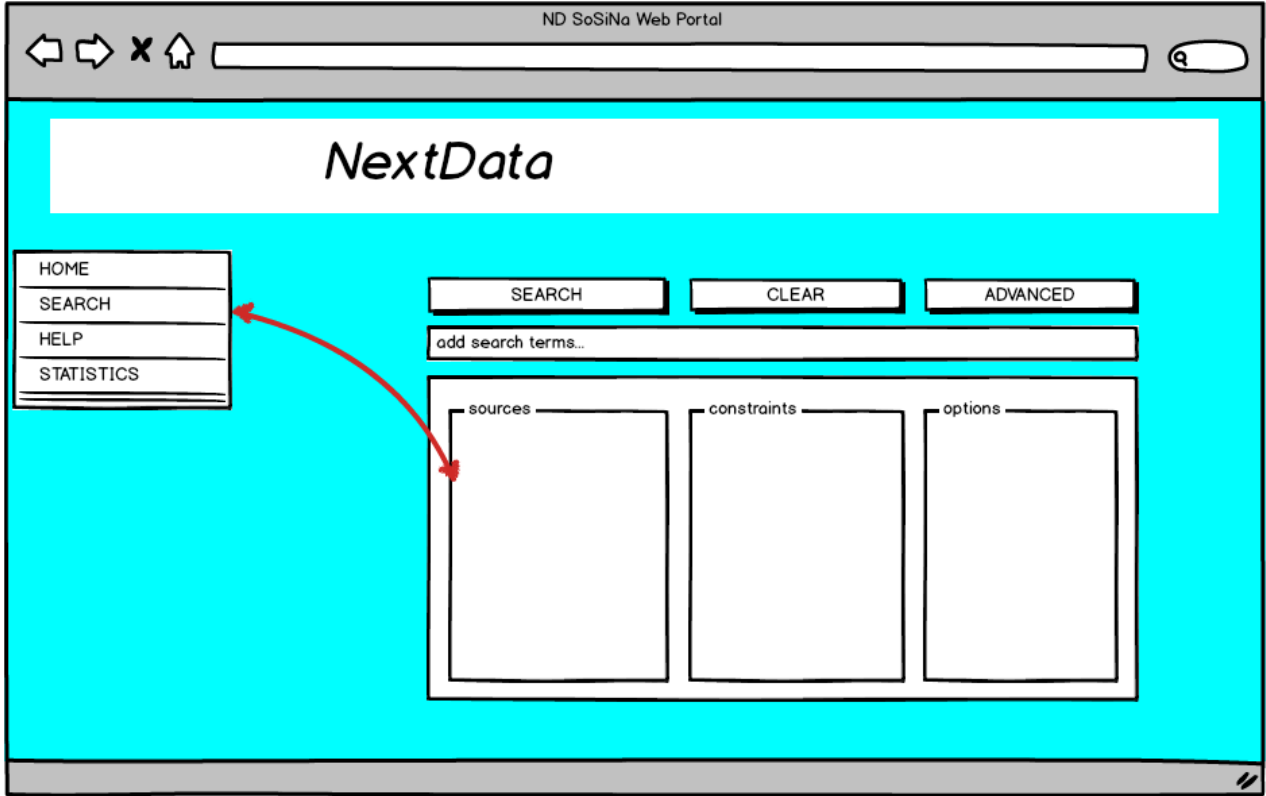


Figure 1. SoSiNa Web Portal Frontend Layout





NextData SoSIna

Sign In

Home Search Help Statistics

Search

Search   Advanced Search

Sources

-  Data

- ICTP@CINECA
- CMCC
- ENEA
- CNR-ISAC
- CRN-ISAC@CINECA
- EVK2CNR

Constraints

Where



North  South  West  East

When

From  To

Options

Search

Fields

Operator

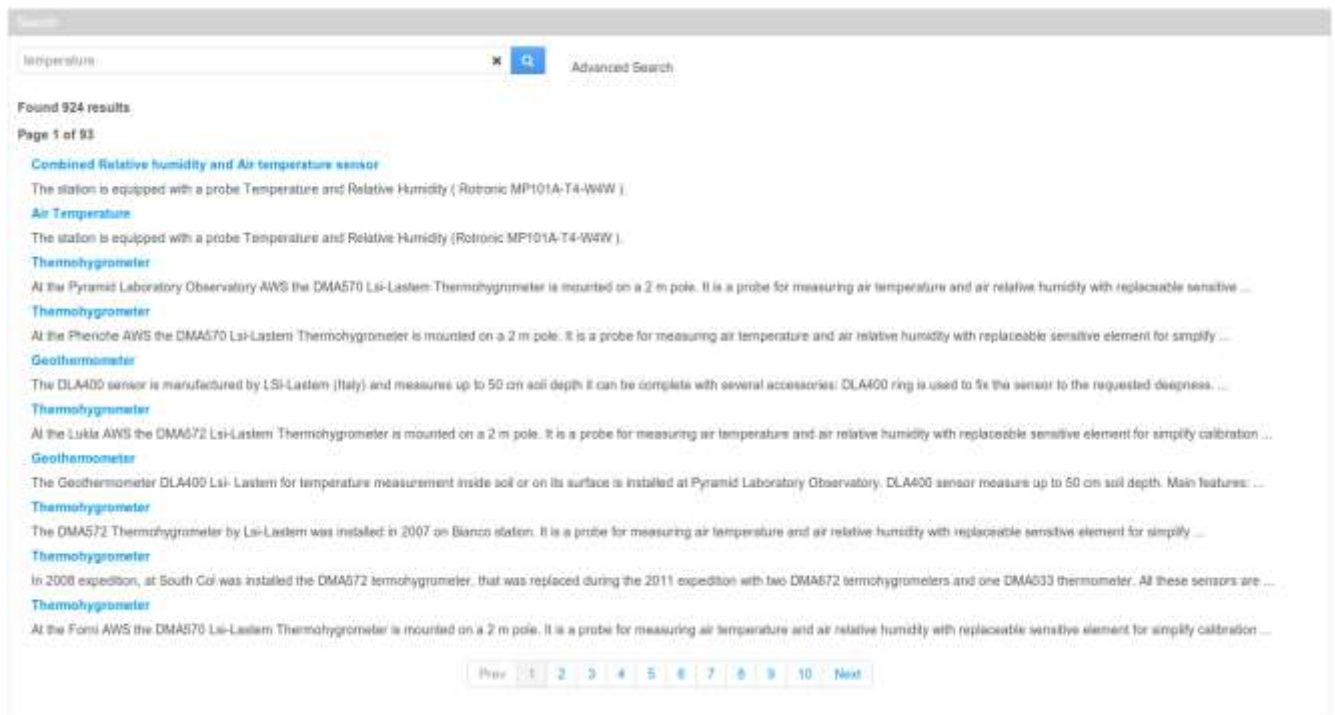
Extension

Relation

Keyword

Concept

Figure 2. SoSIna Web Portal Prototype



**Figure 4. Output results as they are presented on the Web Portal**

The next section describes tools and technologies used to realize the SoSIna Web Portal and to implement all the features previously described. The Technologies section forms the basis to subsequently understand the architecture part.

## Technologies

In this section, we present a set of tools and software libraries that have been adopted. Each software component has been previously approved as result of a common agreement between partners involved in the project.

### Liferay

Liferay portal includes a built-in web content management system allowing users to build their own portals as an assembly of themes, pages and portlets.

We adopted the last release of Liferay, version 6.2. In order to obtain a responsive Web Portal we exploited some libraries provided along with Liferay, including the javascript library AlloyUI 2.0 built on top of YUI3 for the user interface components and the CSS/HTML library Bootstrap 2.3.2.

### GI-cat

GI-cat is an implementation of a broker catalog service made by **ESSI-Lab [R2]**.

GI-cat permits grouping disparate data sources and in particular:

- It allows clients to discover and evaluate geo-information resources over a federation of

data sources addressing also the semantic heterogeneity among terminology through a semantic-enriched search

- It publishes different catalog interfaces, allowing different clients to use the service

#### GI-axe

GI-axe **[R5]** implements Access Broker functionalities, providing transparent access to normalized data, through the invocation of an appropriate sequence of data pre-processing services, including sub-setting (i.e. trimming, slicing); format conversion; CRS transformation; data interpolation.

#### GI-API

The high level client-side Open APIs have been designed in order to simplify the development of applications and clients making use of the **DAB (GEO Discovery and Access Broker)** Gi-cat and Gi-axe.

The DAB is a middleware component which is in charge of interconnecting the heterogeneous and distributed capacities contributing to **GEOS [R3]**;

The DAB provides three main functionalities:

1. Discovery of resources from brokered sources
2. Semantics-enriched discovery
3. Access of resources

The APIs have been developed in **JavaScript** and published together with some examples and documentation. Further information can be found here **[R4]**.

#### OpenLayers

OpenLayers **[R7]** is a free, Open Source JavaScript Framework that makes easy to put a dynamic map in any web page. It can display map tiles and markers loaded from any source. OpenLayers has been developed to further the use of geographic information of all kinds. OpenLayers APIs are released under GNU General Public License.

#### Geoserver

Geoserver **[R6]** is a Java based software server that allows viewing and editing geospatial data. We used it as a map server for OpenLayers. GeoServer is distributed under the GNU General Public License.

In the next section we will see how technologies, tools and software components have been put together in order to implement the Web Portal architecture.

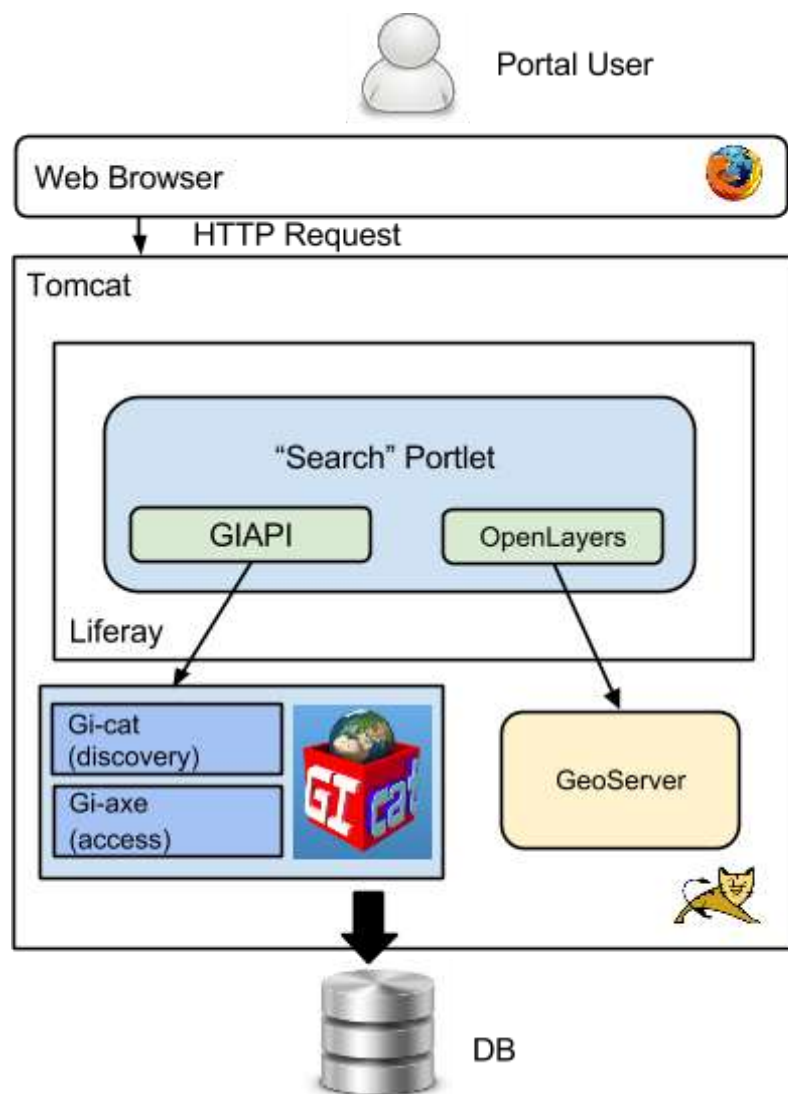
## Architecture

## General

Figure 4 shows a general view of the portal architecture including the main components involved. The Liferay Web Container includes portlets and contents and it runs as a Tomcat web application. The Liferay "Search" Portlet implements functionalities for broker discovery and access as a set of JavaScript functions. The "Search" Portlet interacts with the DAB through the GI-API and forwards parameters as well as access options to the DAB.

In order to allow users to specify latitude and longitude input values an OpenLayers map has been included exploiting the geoserver functionalities.

In the next paragraphs, we will add some information about: how GI-cat has been configured, authentication and security aspects to be considered, logging, resources and finally some recommendations.



**Figure 4. SoSIna Web Portal Architecture**

## Broker Configuration

The Gi-cat broker has been initially configured to search through the following resources:

- <http://nextdataproject.hpc.cineca.it/thredds/catalog/NextData/EC-Earth/catalog.html>
- [http://sansone.to.isac.cnr.it:8080/thredds/catalog/EC-Earth\\_CMIP5/catalog.html](http://sansone.to.isac.cnr.it:8080/thredds/catalog/EC-Earth_CMIP5/catalog.html)
- <http://geonetwork.evk2cnr.org:8080/geonetwork/srv/en/csw>
- [http://nextdata01.cmcc.it:8080/thredds/catalog/CMIP5\\_for\\_nextdata/catalog.html](http://nextdata01.cmcc.it:8080/thredds/catalog/CMIP5_for_nextdata/catalog.html)
- <http://utmea.enea.it:8080/nextdata/catalog/CIRCE/catalog.html>
- <http://fec03.cineca.it:8009/thredds/catalog/NextData/ICTP/catalog.html>

and to publish the following interfaces:

- GI-CAT
- CSW/ISO 2.0.2
- OPENSEARCH-GEO
- CSW/ISO-GEO 2.0.2
- OPENSEARCH-SEMANTIC-ENHANCED

The resources list will be updated depending on the outcome of the other Work Packages of the NextData project as well as the required interfaces.

## Authentication and Security aspects

Liferay supports several authentication mechanisms: user email and password, LDAP, OpenID. Actually only SoSIna Web Portal administrators can login and modify the Portal, users simply go to the Web Portal main page and use the “Search” portlet to discover and access data.

The plain HTTP protocol is adopted (no HTTPS) and the Web Application has been actually deployed on a development node. In the future more restrictions will be applied for the production site such as authentication and subscription of the Nextdata policy to access data.

Authentication and authorization to access federated resources will be agreed with NextData project data providers.

## Logging and messages

Log messages are collected through the Tomcat basic logging mechanism (Catalina log file).

## Resources

The first development instance prototype of the SoSIna Web Portal Architecture has been deployed on the CINECA Front End Cluster (FEC). It is composed by four nodes:

- node01 and node02 host production services and are configured for high-availability
- node03 and node04 host development services

In particular the first development instance prototype of the SoSIna Web Portal has been deployed on the node04 (Intel Xeon E5420 2.5 Ghz quad-core processors with 32 GB of RAM - OS: Debian GNU/Linux 6.0 x86\_64) and is available at:

<http://nextdataproject.hpc.cineca.it>

The same node host the brokering and access services that are available at:

<http://fec04.cineca.it:8009/cineca-demo/>

## Recommendations

Generally, the whole system must be designed paying attention to memory consumption, reliability, availability and scalability issues.

GI-cat and GI-axe should be optimized in the future in order to avoid bottlenecks and improve responsiveness.

Liferay can be configured following the recommendations of the “Liferay Portal Tuning User Guide” [R8]. Web Portal performance improvements can be achieved changing the JVM memory settings (i.e. increasing the memory available), optimizing the garbage collector and eventually disabling unused servlet filters.

## Development plan

Next deadlines:

**January 2014** - Access functionalities

**February 2014** - Semantic-enriched search

**March 2014** - Authentication functionalities

**April 2014** - Evaluation and visualization

**May 2014** - First portal version released

## Conclusions and future extensions

In this document, we have presented the general architecture of the NextData SoSIIna Web Portal and the test infrastructure where services have been deployed. A first version of users requirements has been produced, technologies adopted have been summarized and a general view of the whole architecture has been shown. The Web Portal Design section contains an abstract view of the portal Layout and shows the components which implement the most important functionalities. The interaction between the Broker and the Web application as well as the connection between logical elements are described. In the future, more functional and non-functional aspects will be covered, i.e. regarding Visualization, FAQ, Help functions and Documentation.

## References

R1	Liferay Web Site: <a href="http://www.liferay.com">http://www.liferay.com</a>
----	---

R2	<b>GI-cat Web Site:</b> <a href="http://essi-lab.eu/do/view/GIcat">http://essi-lab.eu/do/view/GIcat</a>
R3	<b>GEOSS Web Site:</b> <a href="http://www.earthobservations.org/geoss.shtml">http://www.earthobservations.org/geoss.shtml</a>
R4	<b>GI-API Web Site:</b> <a href="http://api.eurogeoss-brokerer.eu">http://api.eurogeoss-brokerer.eu</a>
R5	<b>GI-axe Web Site:</b> <a href="http://www.essi-lab.eu/do/view/GIaxe/WebHome">http://www.essi-lab.eu/do/view/GIaxe/WebHome</a>
R6	<b>GeoServer Web Site:</b> <a href="http://geoserver.org/display/GEOS/Welcome">http://geoserver.org/display/GEOS/Welcome</a>
R7	<b>OpenLayers Web Site:</b> <a href="http://openlayers.org/">http://openlayers.org/</a>
R8	<b>Liferay Tuning User Guide:</b> <a href="http://www.liferay.com/documentation/liferay-portal/6.2/user-guide/-/ai/performance-tuning-liferay-portal-6-2-user-guide-20-en">http://www.liferay.com/documentation/liferay-portal/6.2/user-guide/-/ai/performance-tuning-liferay-portal-6-2-user-guide-20-en</a>