EBA Procurement Procedure for the Supply of Website Services 2016: Annex 1 System Architecture Document

SYSTEM ARCHITECTURE DOCUMENT

Contents

1. Introduction

The EBA website will be developed using the Liferay portal. A portal is generally defined as a software platform for building websites and web applications. Some common uses for a portal include websites that require the presentation of different pages depending on user's login status or role, websites that allow quick and easy maintenance of their content through web content management systems and websites that allow groups of individuals to collaborate through applications, on content or with documents. Furthermore portal platforms assist users with building web pages and websites by assembling portlets (independent functional modules) or gadgets onto a portal page. Existing portlets and gadgets can be reused in multiple pages. Finally a portal provides a simplified method to manage the content in various languages and from multiple platforms.
The EBA website will be based on the Liferay portal 6.1.1 CE GA2 version. Liferay portal is written in Java. The scope of this document is to present the most significant parts of the Liferay's architecture but mainly focus in the deployment view of the actual EBA website and the various environments (Development, Test, Pre Production and Production) that will be used.

1.1 Purpose

This document provides a comprehensive architectural overview of the EBA website, using different architectural views to depict different aspects of the system. It intends to capture and convey the significant architectural decisions which have been made on the system.
The audience for the Software Architecture document includes people that need to understand the software architecture of the system:

a. Developers, testers, analysts, part of the project's technical team.

b. Staff from EBA, in charge of the technical aspects of the system and its future technical support and maintenance.

c. Contractor's ICT team, as hosting centre, in particular in terms of compliance with their architecture and quality requirements.

2. Liferay Architecture

2.1 Liferay Versions

Liferay is available in two main versions: Liferay Portal Enterprise Edition (currently 6.1) and the Liferay Portal Community Edition (currently 6.1.1 GA2). While the Enterprise version is an enterprise-tested and refined version with a long-term support, both versions share the same architecture and features. It has been decided that the EBA Website will be based on the Liferay CE edition. Therefore, this document will present the Liferay solution specialising in the Liferay 6.1.1. CE GA2 version.

2.2 Liferay Software Architecture

Liferay is a an enterprise web platform that contains among others a robust collaboration suite. Being a part of Liferay's development platform the collaboration suite inherits the strengths that Liferay offers in the areas of user management, security management and integration with other systems.

Liferay consists of two main layers, the Enterprise and the Service layer. The enterprise layer forms the top layer of services and components that are grouped into taxonomies which support and realize enterprise functions such as Portal Management, Content Management, Workflow Management, Document Management, User Management and Security Management. The Enterprise Layer is also comprised of inter-related service components and features such as Personalization, Collaboration, Social Networking, Delivery Channels, Virtualization and Tunnelling Servlets. These components form the basic backbone or core enterprise features of Liferay.
In the Service Layer, Liferay follows a Model Driven Architecture approach. In the traditional sense it requires a Platform Independent Model (PIM) and Platform Specific Model (PSM). Liferay however starts with a Domain Specific Model (DSM) since the root model is only specific to Liferay domain and it defines all the nouns of the system with their interactions and is translated to a Platform Specific Model (PSM) which constitutes the EJB's, Spring Framework, Hibernate Layer, Web Services and then generates the implementation classes by extending the appropriate services. This is made possible due to the implementation of Service Builder, which is the most integral tool provided by Liferay, and enforces the same standards throughout the portal framework.
Each layer has specific responsibilities.
The Enterprise layer contains the following parts:

- Presentation & User interaction: deals with the presentation logic and page rendering. (Struts 1.2.9 technology & Liferay MVC as UI Controller, JSF with Facelets & Icefaces components also available for custom portlets)
- WEB services interaction: deals with the web services available to external applications (both existing Liferay services & custom web services) (Axis 1.4 based Web services)
- Remote services: they handle the remote services available to applications/AJAX clients/portlets (Spring beans 3.0.5 / Axis 1.4 / remote tunnel servlets)

The Service layer is related to the business logic and manages the accesses to the resource layer (Spring 3.0.5 with POJO implementation as business service component, Liferay Service builder generated implementation classes).
Additional components are connected to the service layer to provide data access and integration with other J2EE components:

- The data access integration component is responsible for the access to the enterprise information system (databases) (Hibernate 3.5.3 for the implementation of the data access layer).
- The messaging component handles the interaction with the Message Bus (Java Message Services 1.1)
- The mail interaction component is responsible to provide email (JavaMail 1.4)
- Other components are available to handle specific interactions such as document repository handling, indexing and searching, scheduling etc. (Jackrabbit, Lucene, Quartz etc.)

The following diagram depicts the layers and the components described above.

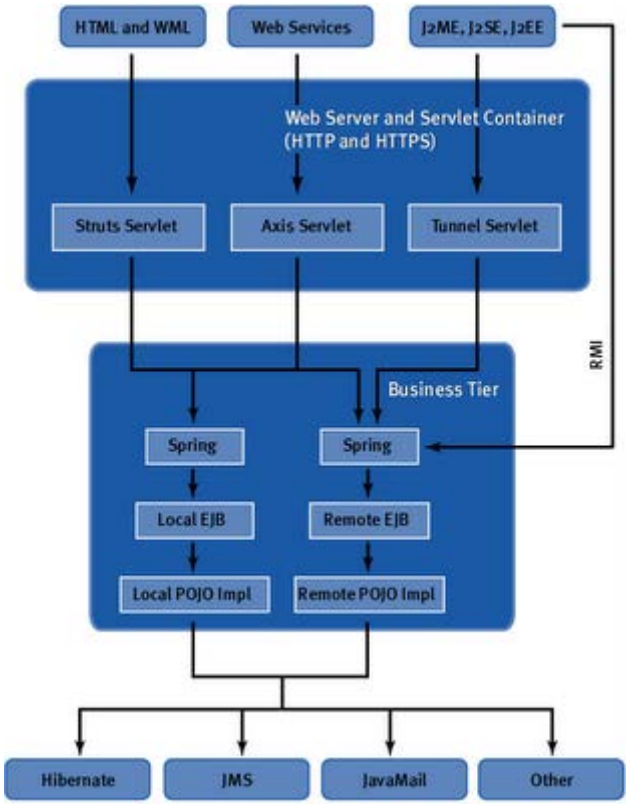

Figure 1: Liferay Software Architecture

2.3 Logical architecture of Liferay

In the case of the EBA website, Liferay is installed on a Red Hat Linux Operating System. Java Runtime Environment is installed on the OS to host the JVM. An application server (Tomcat) is required to host the Liferay instance. More accurately, Liferay is bundled with Tomcat application server.

The server provides connectivity and interoperability using an Enterprise Service Bus (ESB), and there are multiple services offered by the server which are leveraged by Liferay. Some of the services, which are primarily used by Liferay on the Application server, include the following: JNDI, JDBC, JTS, JMS, JAAS, JDO, JWS, JSP/Servlets, JavaMail. Applications can be deployed on the server like SOLR Search engine, or Tunneling Servlets which can further provide extension or integration of external applications with Liferay. Liferay uses a number of technologies at its core to offer the various services. These technologies include EJB, Hibernate, Spring and JBPM. Liferay implements Lucene Search Engine by default and can be configured to extend the Solr Search Engine (which is based on Lucene) to provide capabilities such as clustering, faceted search, filtering with additional enhancements and scalability.

The Administration Kernel provides the base framework for integration and support of all modules, with tooling support, wizards, service providers, listeners and runtime configuration parameters to tweak the application server in runtime mode. The Liferay Service builder provides the basic framework to construct and deploy the services using a Model Driven Development (MDD) approach. The portlet plug-in leverages on the portlet bridge to provide dynamically generated portlets to the end users and enhanced RIA (Rich Internet Application) integration. The hooks plug-in provide convenient access to intercept and alter the services and functionality of the Liferay instance in a standardized approach. A robust Enterprise Services layer leverages all the services and features provided by Liferay core, this providing complete solutions ranging from Portal Management, Web Content Management, Enterprise Content Management, Document Management, User Management, Workflow Management, Security Management. These in turn provide features such as Personalization, Collaboration, Virtualization, Social Networking and integrates Dynamic Delivery Channels, and Tunneling Services.

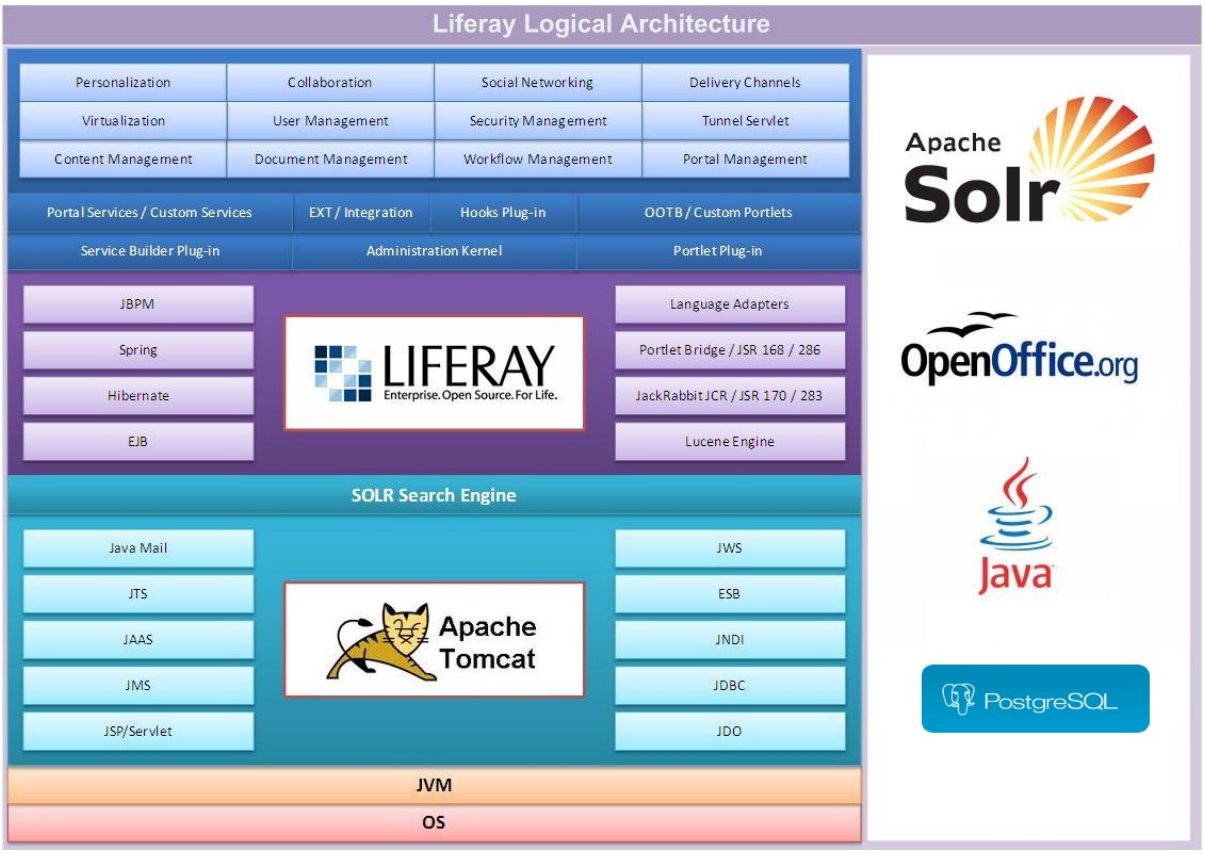

Figure 2: Liferay Logical Architecture

3. PROCESS VIEW

The process architecture takes into account some non-functional requirements, such as performance and availability. It addresses issues like concurrency and distribution, system's integrity and fault-tolerance. The selected Liferay platform deals natively with these threading aspects.

4. EBA ENVIRONMENTS&DEPLOYMENTS

4.1 Environments

The EBA website is deployed in four different environments:

- **Production environment (PROD):** Production environment is the publicly accessible EBA environment from the Internet. The latest stable release is deployed in the production environment according to EBA's requests. Both extranet and public websites are available on the production environment.

- **Pre-Production environment (PREPROD):** Pre-production environment is an exact replica of the production environment in terms of application deployments and servers topology. It contains the same release as the production environment. It is used for issue replication/testing; for issues tracked in the production environment. Hotfixes are first deployed in the pre-production environment.

- **Test environment (TEST):** Test environment is an exact replica of the production environment in terms of application deployments and servers topology. The distribution of application servers in VMs may differ from the production environment. The Test environment contains the next release that is going to be deployed in the production environment. The test environment is open to EBA staff, in order to test the new functionalities, fixes etc.

- **Development environment (DEV):** Development environment is an exact replica of the production environment in terms of application deployments and servers topology. The distribution of application servers in VMs may differ from the production environment. The development environment is continuous integrated with the EBA development repository and contains the latest working branch.

All environments will be hosted in contractor's premises.
Moreover, in the Contractor’s premises, several development environments will also be maintained. Development environments are hosted in personal computers with a non clustered application server running Liferay portal CE for developing and testing new modules.

4.2 Releases deployment workflow

While the development team is working on a new release, the changes are immediately deployed and tested in the DEV environment. Once the changes are finished, a release file is produced and deployed to the TEST environment. EBA is notified that the upcoming release is available for testing in the TEST environment. Once the release is tested and accepted by the Contractor and/or by EBA, it is then deployed in the PREPROD and PROD environments.

5. Deployment View

The EBA Website supports high availability through failover clustering. High-availability clusters provide continuous availability of services by eliminating single points of failure and by failing over services from one cluster node to another in case a node becomes inoperative. The chosen architecture addresses the critical requirements of High-availability. Figure 3 depicts the architectural overview of the EBA website (R2).

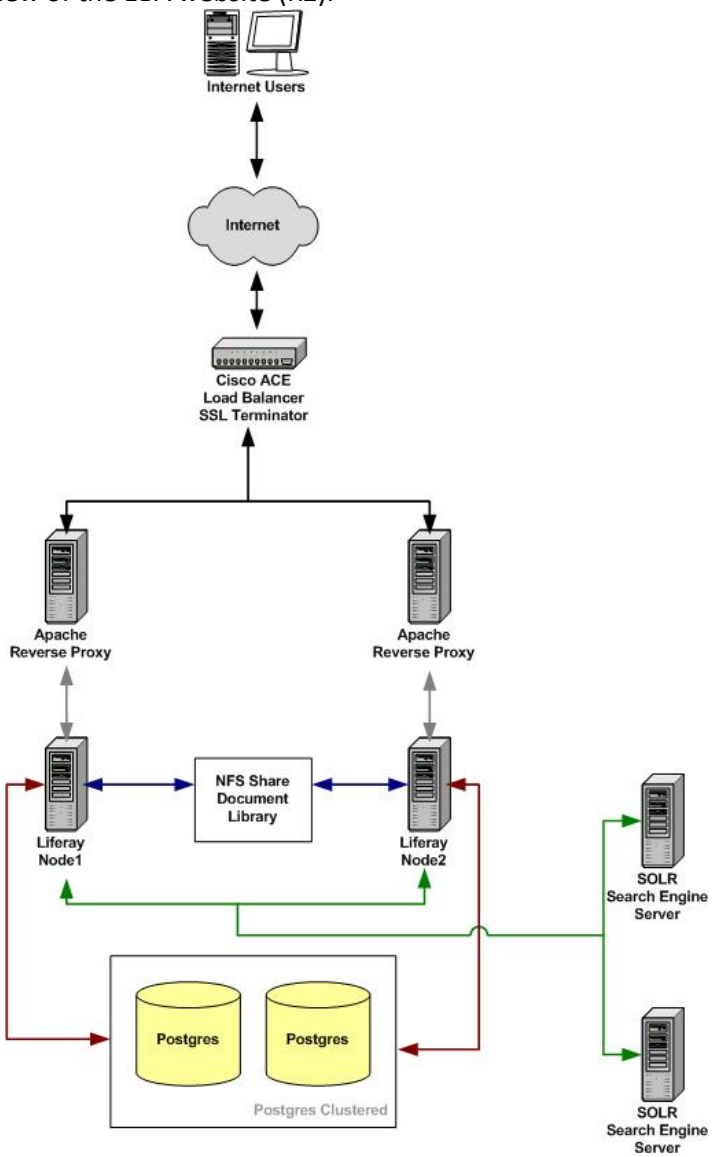

Figure 3: Liferay topology

Liferay is installed on two nodes functioning as a cluster; each one of the nodes resides behind an Apache reverse proxy. Both Liferay nodes communicate with the clustered database of the EBA website, the document library and the SOLR search servers. SOLR Search engine is also clustered for High-availability and performance.

Public/Extranet users access the production environment using the appropriate URL (http://eba.europa.eu). Their requests are passed through the ACE load balancer and the Apache reverse proxy. Then, the load balancer sends the request to one of the two Liferay nodes. The

Liferay nodes may contact the document library and/or the database and/or the search engine in order to serve the request.
Each Liferay node, SOLR server node, Postgres node will reside in different virtual machines (VMs). The described topology and distribution of the servers in VMs, guaranties the availability of the EBA website in case of any server or VM failure.

5.1 Load Balancer

European Dyanmics uses hardware (Cisco ACE) for load balancing and SSL termination purposes. Cisco ACE is used to increase application availability, using best-in-class application-switching algorithms for highly available system software and hardware. It achieves these goals through intelligent load-balancing and content-switching technologies integrated with acceleration and security capabilities.

5.2 Apache Reverse Proxy

Apache HTTP server with mod_proxy_ajp is used for the reverse proxy of Liferay server.

5.3 Liferay clustered Application server

5.3.1 Production environment

Two identical cluster nodes with Liferay 6.1.1 CE GA2 (bundled with Tomcat) constitute the portal in the production environment:
Production Tomcat Liferay Node 1:

VM Specifications	
Architecture	x86_64
Operating System	Red Hat Enterprise Linux Server release 6.3 (Santiago)
CPU	At least 2x 64bit Xeon Dual Core
RAM	At least 8GB
JVM	Java 6.0_25
Application server	Tomcat-7.0.279 (bundled with Liferay CE)
Liferay version	Liferay portal tomcat CE 6.1.1 CE GA2

Table 1: Production Tomcat Liferay Node 1

Production Tomcat Liferay Node 2:

VM Specifications	
Architecture	x86_64
Operating System	Red Hat Enterprise Linux Server release 6.3 (Santiago)
CPU	At least 2x 64bit Xeon Dual Core
RAM	At least 8GB
JVM	Java 6.0_25
Application server	Tomcat-7.0.279 (bundled with Liferay CE)
Liferay version	Liferay portal tomcat CE 6.1.1 CE GA2

Table 2: Production Tomcat Liferay Node 2

5.4 SOLR clustered Search Engine

SOLR is an open source enterprise search platform from the Apache Lucene project and is used as a search engine. SOLR has proven to be a reliable search engine, proving advanced searching

capabilities in low response times. All searching operation will be directed to SOLR minimizing the overhead on the Liferay side. Moreover, having two SOLR servers, both containing the same index replica, provides high availability in case of SOLR or VM failure.
SOLR is a standalone Java web-application that can be deployed on any application server. The Apache Tomcat is a tested and reliable choice for hosting SOLR and will be adopted for the EBA Website.

5.4.1 Production part

Two identical cluster nodes with SOLR 3.6.1 and Tomcat 6.0.32 constitute the Search engine in the production part:

Production Solr Node 1:

VM Specifications	
Architecture	x86_64
Operating System	Red Hat Enterprise Linux Server release 6.3 (Santiago)
CPU	At least 2x 64bit Xeon Dual Core
RAM	At least 8GB
JVM	Java Sun 1.6.0_25
Search Engine	Solr 3.6.1 integrated with solr-web plugin in Liferay side
Application Server	Tomcat 6.0.32

Table 3: Production Tomcat SOLR Node 1

Production Solr Node 2:

VM Specifications	
Architecture	x86_64
Operating System	Red Hat Enterprise Linux Server release 6.3 (Santiago)
CPU	At least 2x 64bit Xeon Dual Core
RAM	At least 8GB
JVM	Java Sun 1.6.0_25
Search Engine	Solr 3.6.1 integrated with solr-web plugin in Liferay side
Application Server	Tomcat 6.0.32

Table 4: Production Tomcat SOLR Node 2

5.5 Document Library

The Liferay Advanced file system hook is utilised. This will be provided by a Network File System (NFS) share.

5.6 Database

PostgreSQL along with RedHat Cluster Suite (RHCS) and shared SAN storage are used as DBMS solution, thus ensuring that the EBA CMS will enjoy high availability through failover clustering.

Configuring a PostgreSQL database cluster using RHCS provides:

- Database management system redundancy
- Data redundancy
- Network redundancy
- Server and power redundancy
- Maximum uptime
- Service failover
- Data integrity

The infrastructure that hosts the high availability EBA Website database consists of a VMWare HA cluster running on three high performance HP BL460 G7 blade servers with two 6-core XEON X5640 CPUs and 96GB RAM. A shared storage resource provided by a highly available HP StorageWorks EVA 6400 Storage Area Network (SAN) fitted with 40TB of storage capacity.

With VMware HA, a set of ESX Server hosts is combined into a cluster with a shared pool of resources. VMware HA monitors all hosts in the cluster. If one of the hosts fails, VMware HA immediately responds by restarting each affected virtual machine on a different host.

6. Implementation View

6.1 Overview

The following diagram provides a visual containment overview of the components deployed on the Application Server (Tomcat) and on the EBA Database Server (Postgres).

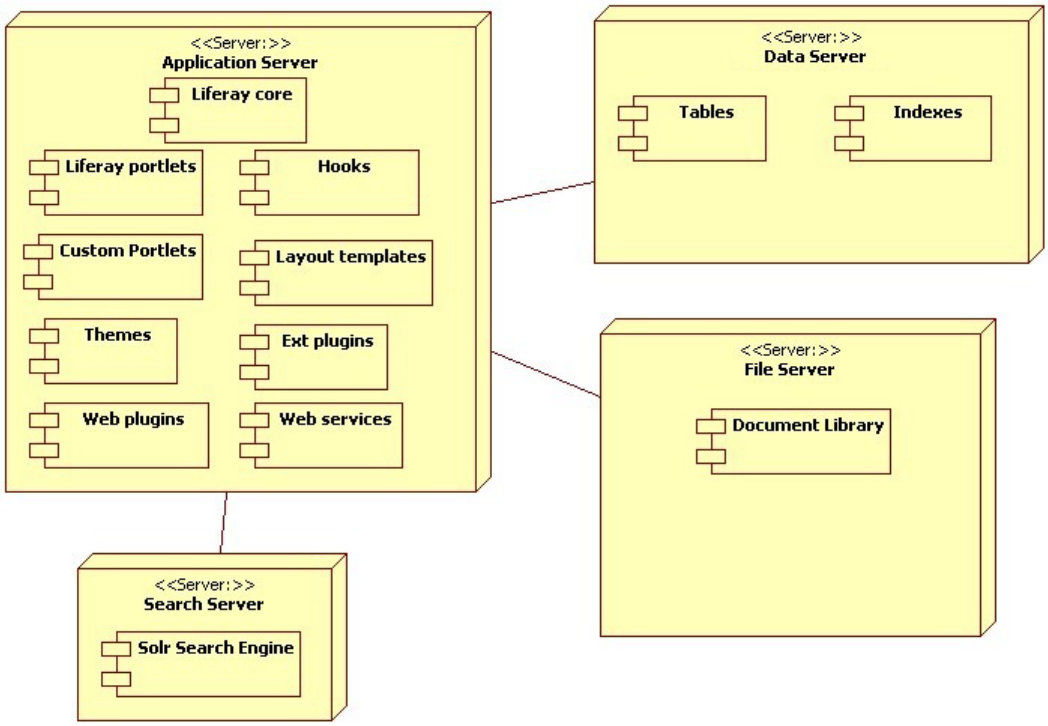

Figure 4: Deployed components

The EBA website will comprise the following modules: Liferay portlets, custom portlets, hooks, themes, tunnel-web services etc. Custom modules will be packaged inside war files.

The Contractor will create packages and deploy war files for custom portlets, themes, hooks, layouts. The war files will be deployed in the application server while the server is running using a hot-deployment technique. The newly deployed applications will be available to be used shortly after the deployment procedure.
The Ext-plugin, which is used to extend the core of Liferay portal is not hot-deployable, requiring server restarts in order to complete its deployment. During the deployment of the ext plugin, the server is excluded from the cluster and the website functions with one node. The same procedure is applied to the second node. Required deployment time of the ext plugin is ~3-10 minutes. The EBA website will not have any downtime during the deployment of ext or regular plugins.

6.2 Development Methodology and Tools

6.2.1 Liferay Plugins SDK

Liferay provides a set of tools called Plugins SDK for developing new plugins or extending/modifying the existing Liferay functionalities. The use of the Plugins SDK, results in a cleaner development approach rather than intervening directly to the source of Liferay. Moreover, Plugins SDK can be quickly installed in a new development environment. Finally, Plugins SDK is integrated with Liferay IDE.

6.2.2 Integrated Development Environment (IDE)

Liferay IDE is an extension for the Eclipse platform that supports development of plug-in projects for the Liferay Portal platform. It supports developing five Liferay plug-in types: portlets, hooks, layout templates, themes, and EXT-style plug-ins. Liferay IDE requires the Eclipse Java EE developer package using the Juno version.

6.2.3 Version Control system

SVN is used as a version control system for the development of the EBA website. The server part is the Subversion 1.5.0 by Collabnet (now called Apache Subversion). The TortoiseSVN 1.7.11 is used as the subversion client.

END OF DOCUMENT