

Reference Guide / Knowledge Functions

Table of Contents

Prerequisites	v
1. Applications	1
1.1. Portlets	1
1.1.1. Forum Portlet	1
1.1.2. Answers Portlet	6
1.1.3. FAQ Portlet	7
1.1.4. Polls Portlet	8
1.2. Gadgets	8
2. Configuration	11
2.1. Components	11
2.2. External component plugins	12
2.2.1. Init data configuration	13
2.2.2. Roles Configuration	26
2.2.3. ProfileProvider Configuration	28
2.2.4. Forum Configuration	30
2.2.5. Answers Configuration	53
2.2.6. Poll Configuration	55
2.3. Data Injectors	58
2.3.1. Configuration	59
2.3.2. How to use	64
3. Developer reference	65
3.1. Extension points	65
3.1.1. ForumEventLifeCycle	66
3.1.2. AnswerEventLifeCycle	67
3.1.3. BBCodeRenderer	69
3.2. Internal API	69
3.3. Wiki Service API	71
3.3.1. DiffService	72
3.3.2. LinkService	72
3.3.3. PageRenderingCacheService	73
3.3.4. ResizeImageService	73
3.3.5. RenderingService	74
3.3.6. WikiRestService	75
3.3.7. WikiService	77
3.4. FAQ Template Configuration	81
3.4.1. Configuration plug-in	82
3.4.2. How to change look and feel	83
3.4.3. API provided by the UIComponent (UIViewer.java)	83
3.5. Actions over a wiki page from external jars	84
3.5.1. Create a new project for action extension	85
3.5.2. Create new actions and their corresponding listeners	86
3.5.3. Register new actions with UIExtensionManager	87
3.5.4. Deploy new action extension	87
3.6. JCR structure	88
3.6.1. Forum JCR structure	89
3.6.2. FAQ JCR structure	101
3.6.3. Poll JCR structure	107
3.6.4. Wiki JCR structure	108

Prerequisites

There are 2 prerequisites you may need to learn before thoroughly going into Knowledge as follows:

- [GateIn Reference Guide](#)
- [OpenSocial Core Gadget Specification v1.1](#)
- [WebService](#)

Applications

This chapter provides you with a comprehensive view about applications in Knowledge, including:

- **Portlets**

Details (package, portlet.xml, and portlet preferences) of all portlets used in Knowledge, including Forum, Answers, FAQ and Polls.

- **Gadgets**

Introduction to Poll Gadget, its preferences and links to used REST services.

These applications are packaged as Web application archives (WARs).

Also, you can specify the package of each portlet and gadget and its available preferences that allow you to extend the configuration choices for standard preferences.

1.1. Portlets

- **Forum Portlet**

Details (including package, portlet.xml, preferences and events) of the Forum application which allows posting and reading messages on different topics.

- **Answers Portlet**

Details (including package, portlet.xml, preferences) of the Answers application that allows creating answers, replying and managing questions.

- **FAQ Portlet**

Details (including package, portlet.xml, preferences) of the FAQ application which allows showing questions and answers.

- **Polls Portlet**

Details (including package, portlet.xml, preferences) of the Poll application which allows voting any ideas, or activities.

See also

- **Gadgets**

1.1.1. Forum Portlet

The Forum portlet is the application for users to post and read messages on different topics.

Package

This portlet is packaged in the *forum.war* file.

Portlet.xml

- See the *portlet.xml* file in the project following this path: *forum/WEB-INF/portlet.xml*.

Preferences

Preference name	Possible value	Default value	Description
useAjax	true, false	true	

Preference name	Possible value	Default value	Description
			Define if links in the Forum will be plain hrefs or javascript ajax (better for SEO) or not.
showForumActionBar	true, false	true	This is the UIForumActionBar. If the value is set to "true", the UIForumActionBar will be shown. If false, the UIForumActionBar will be hidden.
forumNewPost	day number	1	Specify if a post is new. If the post is created within the set period, it is new in the Forum.
enableIPLogging	true, false	true	Enable the IP logging function in the Forum. IP addresses of all posts will be collected.
enableIPFiltering	true, false	true	Enable the IP filter function in Forum, enabling IP addresses to be blocked in the Forum.
invisibleCategories	id categories	empty	Hide some categories. If the value is set empty, all categories of the Forum will be shown.
invisibleForums	id forums	empty	Hide some Forums. All Forums will be shown if the value is set empty.
uploadFileSizeLimitMB	integer	20	Limit the size of uploaded files in MB in the Forum.
isShowForumJump	true, false	true	Specify if the Forum jump panel is shown or not.
isShowIconsLegend	true, false	true	Specify if the icon legends panel is shown or not.
isShowModerators	true, false	true	Specify if the moderators panel is shown or not.
isShowPoll	true, false	true	Specify if the poll panel is shown or not.
isShowQuickReply	true, false	true	Specify if the quick reply panel is shown or not.
isShowRules	true, false	true	Specify if the forum rules panel is shown or not.
isShowStatistics	true, false	true	Specify if the statistics panel is shown or not.

Events

Name	Description
ForumLinkEvent	Set the render for UIForumLinkPortlet and set values for UIForumLinks.
ReloadPortletEvent	Reload UIForumPortlet.
OpenLink	Update values for UIForumLinks.
ForumPollEvent	Set the render for UIForumPollPortlet.
ForumModerateEvent	Set the render for UIForumModeratorPortlet.
ForumRuleEvent	Set the render for UIForumRulePortlet.
QuickReplyEvent	Set the render for UIForumQuickReplyPortlet.

ForumLinkEvent This event is fired through UIForumLinkPortlet.

To receive ForumLinkEvent, you must use the *ForumParameter* class with two properties:

Name	Type	Possible value	Description
isRenderForumLink	boolean	true/false	If the value is set to true or false, the Forum link is rendered or not respectively.
path	string	The absolute path of the class node (including types: category, forum, topic) defined by JCR.	Set data for the UIForumLinkPortlet.

For example:

```
PortletRequestContext pcontext = (PortletRequestContext) WebuiRequestContext.getCurrentInstance();
ActionResponse actionRes = (ActionResponse) pcontext.getResponse();
ForumParameter param = new ForumParameter();
String path = forum.getPath();
if ( ...condition to render the UIForumLinkPortlet... ) {
    param.setRenderForumLink(true);
    param.setPath(path);
} else {
    param.setRenderForumLink(false);
}
actionRes.setEvent(new QName("ForumLinkEvent"), param);
```

ReloadPortletEvent This event is fired through UIForumPortlet.

To receive ReloadPortletEvent, you must use the *ForumParameter* class with two properties:

Name	Type	Possible value	Description
topicId	string	Id of topic.	Return the id of topic for UIForumPortlet
isRenderPoll	boolean	true/false	If the value is set to true or false, the <i>UITopicPoll</i> component is rendered or not respectively.

For example:

```
....
ActionResponse actionRes = pcontext.getResponse() ;
ForumParameter param = new ForumParameter() ;
param.setRenderPoll(true);
param.setTopicId(topic.getId());
actionRes.setEvent(new QName("ReloadPortletEvent"), param) ;
....
```

OpenLink This event is fired through UIForumPortlet.

To receive OpenLink, you must use the *ForumParameter* class with one property:

Name	Type	Possible value	Description
path	string	The absolute path of the node defined by JCR.	Set data for the UIForumPortlet.

For example:

```
....
ActionResponse actionRes = pcontext.getResponse() ;
ForumParameter param = new ForumParameter() ;
param.setPath(path);
actionRes.setEvent(new QName("OpenLink"), param) ;
....
```

ForumPollEvent This event is fired through UIForumPollPortlet.

To receive ForumPollEvent, you must use the *ForumParameter* class with four properties:

Name	Type	Possible value	Description
isRenderPoll	boolean	True/false	If the value is set to true or false, the <i>UIForumPollPortlet</i> portlet is rendered or not respectively.
categoryId	string	Id of category	Return the Id of category for UIForumPollPortlet.
forumId	string	Id of forum	Return the Id of forum for UIForumPollPortlet.
topicId	string	Id of topic	Return the Id of topic for UIForumPollPortlet.

For example:

```
....
ActionResponse actionRes = pcontext.getResponse() ;
ForumParameter param = new ForumParameter() ;
param.setCategoryId(categoryId) ;
param.setForumId(forumId);
param.setTopicId(topicId);
param.setRenderPoll(topic.getIsPoll());
actionRes.setEvent(new QName("ForumPollEvent"), param);
```

....

ForumModerateEvent This event is fired through UIForumModeratePortlet.

To receive ForumModerateEvent, you must use the *ForumParameter* class with two properties:

Name	Type	Possible value	Description
isRenderModerator	boolean	True/false	If the value is set to true or false, the <i>UIForumModeratePortlet</i> portlet is rendered or not respectively.
moderator	list of strings	List of user name	Set data for UIForumModeratePortlet.

For example:

```
....
List<String> moderators = Arrays.asList(forum.getModerators());
ActionResponse actionRes = pcontext.getResponse() ;
ForumParameter param = new ForumParameter() ;
param.setModerators(moderators);
param.setRenderModerator(true);
actionRes.setEvent(new QName("ForumPollEvent"), param);
....
```

ForumRuleEvent This event is fired through UIForumRulePortlet.

To receive ForumRuleEvent, you must use the *ForumParameter* class with two properties:

Name	Type	Possible value	Description
isRenderRule	boolean	True/false	If the value is set to true or false, the <i>UIForumRulePortlet</i> portlet is rendered or not respectively.
infoRules	list of strings	The list of states: can create topic, can add post and topic has lock.	Set permissions for users in UIForumRulePortlet.

For example:

```
....
ActionResponse actionRes = pcontext.getResponse() ;
ForumParameter param = new ForumParameter() ;
List<String> list = param.getInfoRules();
if(forum.getIsClosed() || forum.getIsLock()) {
    list.set(0, "true");
} else {
    list.set(0, "false");
}
list.set(1, String.valueOf(canCreateTopic));
list.set(2, String.valueOf(isCanPost));
param.setInfoRules(list);
```

```
param.setRenderRule(true);
actionRes.setEvent(new QName("ForumRuleEvent"), param) ;
....
```

QuickReplyEvent This event is fired through UIQuickReplyPortlet.

To receive QuickReplyEvent, you must use the *ForumParameter* class with five properties:

Name	Type	Possible value	Description
isRenderQuickReply	boolean	True/false	If the value is set to true or false, the <i>UIQuickReplyPortlet</i> portlet is rendered or not respectively.
isModerator	boolean	True/false	Specify if the user is moderator of forum containing the topic with quick reply or not.
categoryId	string	Id of category	Return the Id of category for <i>UIQuickReplyPortlet</i> .
forumId	string	Id of forum	Return the Id of forum for <i>UIQuickReplyPortlet</i> .
topicId	string	Id of topic	Return the Id of topic for <i>UIQuickReplyPortlet</i> .

For example:

```
....
ActionResponse actionRes = pcontext.getResponse() ;
ForumParameter param = new ForumParameter() ;
param.setRenderQuickReply(isCanPost);
param.setModerator(isMod);
param.setCategoryId(categoryId) ;
param.setForumId(forumId);
param.setTopicId(topicId);
actionRes.setEvent(new QName("QuickReplyEvent"), param) ;;
....
```

1.1.2. Answers Portlet

The Answers portlet is the application to create answers, reply and manage questions.

Package

This portlet is packaged in the *faq.war* file.

Portlet.xml

- See the *portlet.xml* file in the project following this path: */webapps/faq/WEB-INF/portlet.xml*.

Portlet Preferences

The Answers portlet consists of preferences as follows:

Preference name	Possible value	Default value	Description
enableViewAvatar	true, false	true	

Preference name	Possible value	Default value	Description
enableAutomaticRSS	true, false	true	Enable users to view the avatar of owner posting the question.
enableVotes AndComments	true, false	true	Enable users to get RSS automatically.
enableAnonymous SubmitQuestion	true, false	true	Enable users to give votes and comments for the question.
display	approved, both	both	Enable administrators to submit questions.
SendMailAdd NewQuestion	string	empty	Enable administrators to view unapproved questions in the questions list in UIQuestions.
SendMailEdit ResponseQuestion	string	empty	Display the content of sent email when a new question is added.
emailMoveQuestion	string	empty	Display the email content when a response is edited.
orderBy	alphabet, created	alphabet	Display the email content when a question is moved.
orderType	asc, desc	asc	Arrange questions in the alphabet or created date order.
isDiscussForum	true, false	false	Display questions in the ascending or descending order.
idNameCategoryForum	CategoryName, ForumName	empty	Enable the DiscussQuestions function.
uploadFileSizeLimitMB	integer	20	Select categories and forums for the DiscussionQuestions function.
			Set the maximum size of uploaded files in MB.

1.1.3. FAQ Portlet

The FAQ portlet is the application to show questions and answers.

Package

This portlet is packaged in the *faq.war* file.

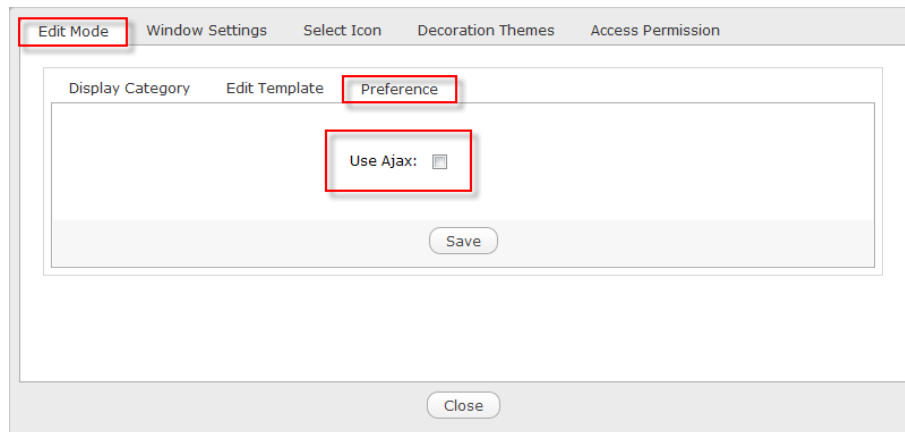
Portlet.xml

- See the *portlet.xml* file in the project following this path: */webapps/faq/WEB-INF/portlet.xml*.

Portlet Preferences

The FAQ portlet includes some portlet preferences that you can configure to alter the behavior.

At runtime, you can use the EDIT mode portlet to set the preferences:



Alternatively, you can configure the portlet in the *portlet-preferences.xml* file.

Preference name	Possible value	Default value	Description
useAjax	true, false	false	Specify if the AJAX load or HREF load is used.

1.1.4. Polls Portlet

The Poll portlet is the application for users to vote any ideas, or activities.

Package

This portlet is packaged in the *poll.war* file.

Portlet.xml

- See the *portlet.xml* file in the project following this path: *poll/WEB-INF/portlet.xml*.

```
<portlet-preferences>
  <preference>
    <name>pollIdShow</name>
    <value/> <!-- PollId -->
    <read-only>false</read-only>
  </preference>
</portlet-preferences>
```

Portlet Preferences

Preference name	Possible value	Default value	Description
pollIdShow	string	empty	The Id of poll which is displayed in the Polls portlet.

1.2. Gadgets

eXo Platform provides a gadget which enables users to see a poll. The Poll Gadget is developed on the combination of Gadget by GateIn and Polls Service. The Poll Gadget allows users to apply functions of Polls, such as viewing and voting Polls.

Gadget name	War name	Description
pollslist	poll.war	The list of Polls.

Preferences

Preference name	Description
pollId	The Id of Polls which is displayed in the Polls gadget.

Links to used REST services

- [portal/rest/private/ks/poll/viewpoll/pollId](#)
- [portal/rest/private/ks/poll/votepoll/pollId/indexVote](#)

See also

- [Portlets](#)

Configuration

This chapter describes configurations used in Knowledge. It consists of the following main sections:

- **Components**

Basic information about 3 types of components which are used to implement the Knowledge applications: Knowledge, Forum, Answers and Polls.

- **External component plugins**

Description of the main external component plugins used in Knowledge: Init data, Roles, ProfileProvider, Forum, Answer, and Poll. Each part supplies an example configuration with the explanation about ini-params so you can know how to use these plugins.

- **Data Injectors**

Introduction to the data injectors which are used to create data for performance test in Knowledge, their configurations and how to use them.

2.1. Components

This section shows some main components used to implement applications of Knowledge.

Components of Knowledge

Key	Data type	Description
org.exoplatform.ks.common.jcr.KSDataLocation	org.exoplatform.ks.common.jcr.KSDataLocation	Hold the JCR storage location for the Knowledge data.
org.exoplatform.services.scheduler.JobSchedulerService	org.exoplatform.services.scheduler.impl.JobSchedulerServiceImpl	Define a job to execute a given number of times during a given period. It is used to monitor jobs automatically and continuously, to schedule event-driven jobs and reports, and to control performance.

- Init-params of org.exoplatform.ks.common.jcr.KSDataLocation:

Name	Possible value	Default value
repository	string	repository
workspace	string	knowledge

Components of Forum

Key	Data type	Description
org.exoplatform.ks.bbcode.api.BBCodeService	org.exoplatform.ks.bbcode.core.BBCodeServiceImpl	Manage CRUD operations on BBcodes.
org.exoplatform.forum.service.DataStorage	org.exoplatform.forum.service.impl.JCRDataStorage	Store data of Forum via the JCR system.
org.exoplatform.forum.service.ForumService	org.exoplatform.forum.service.impl.ForumServiceImpl	Include all public APIs to interact with the UI component and database.

Key	Data type	Description
org.exoplatform.forum.service. ForumStatisticsService	org.exoplatform.forum.service. impl.ForumStatisticsServiceImpl	Include all public APIs to interact with the database of Statistics system.

Components of Answers

Key	Data type	Description
org.exoplatform.faq. service.FAQService	org.exoplatform.faq.service. impl.FAQServiceImpl	Include all public APIs to interact with the UI component and database.
org.exoplatform.faq. service.DataStorage	org.exoplatform.faq.service. impl.JCRDataStorage	Store data of FAQ via the JCR system.

Components of Polls

Key	Data type	Description
org.exoplatform.poll. service.DataStorage	org.exoplatform.poll.service. impl.JCRDataStorage	Include all public APIs to interact with the UI component and database.
org.exoplatform.poll. service.PollService	org.exoplatform.poll.service. impl.PollServiceImpl	Store data of Polls via the JCR system.

See also

- [External Component Plugins](#)
- [Data Injectors](#)

2.2. External component plugins

• [Init data configuration](#)

Introduction to the `Init` data plug-in that defines the default data in the `.xml` files, and instructions on how to initialize the `conf-part` for loading `repository-configuration.xml/workspace` name and repository name in `storage-configuration.xml/data`.

• [Roles Configuration](#)

Details of the `RoleRulePlugin` plugin that defines roles of users in the Forum application of eXo Platform.

• [ProfileProvider Configuration](#)

Introduction to the `ProfileProvider` plugin that allows retrieving personal information of users, how to configure the `profile-configuration.xml` file and to get the `ContactProvider`.

• [Forum Configuration](#)

Information and detailed configurations of plugin components that define data of the Forum application, including: `BBCode`, `Forums Initializer`, `Auto-prune`, `User Statistics`, `Statistic Data`, and `Default User Profile`.

• [Answers Configuration](#)

Information about the plugin components that define data of the Answers application, including: `Answers Initializer` and `Answers Email Templates`.

• [Poll Configuration \[54\]](#)

Details of the configurations of default Poll data.

See also

- [Components](#)
- [Data Injectors](#)

2.2.1. Init data configuration

The Init data plug-in is used to define the default data in the `.xml` file. It includes nodes (node of jcr). When the `org.exoplatform.services.jcr.config.RepositoryServiceConfiguration` component is initialized, the `org.exoplatform.services.jcr.impl.config.RepositoryServiceConfigurationPlugin` component will be get and the function `addConfig` is called. Then, the `/ks-extension/jcr/repository-configuration.xml` file is loaded and the component `org.exoplatform.ks.common.jcr.KSDataLocation` will be initialized. Next, the `setLocation` function is called, setting up the workspace and repository for KS. After that, the `addPlugin` function will be run, generating the `DataLocation` (some parent nodes) for Knowledge.

The following is the list of applications in Knowledge and the corresponding components used to initialize the default data.

Application	Component	Description
Forum	<code>KSDataLocation</code> , <code>ForumServiceImpl</code>	Initialize default data of the Forum portlet.
Answers	<code>KSDataLocation</code> , <code>AnswerServiceImpl</code>	Initialize default data of the Answers portlet.
Polls	<code>KSDataLocation</code> , <code>PollServiceImpl</code>	Initialize default data of the Polls portlet.

In this section, you will understand how to initialize data via the sample configurations later.

2.2.1.1. Initialize the conf-part for loading repository-configuration.xml

When the server starts, the `jcr-configuration.xml` file is initialized. The component-plugin named `addConfig` will be referred to `org.exoplatform.services.jcr.impl.config.RepositoryServiceConfigurationPlugin` to load the `war:/ks-extension/jcr/repository-configuration.xml` file.

```
<component-plugin>
  <!-- The name of the plugin -->
  <name>Sample RepositoryServiceConfiguration Plugin</name>
  <!-- The name of the method to call on the RepositoryServiceConfiguration
  in order to add the RepositoryServiceConfigurations -->
  <set-method>addConfig</set-method>
  <!-- The full qualified name of the RepositoryServiceConfigurationPlugin -->
  <type>org.exoplatform.services.jcr.impl.config.RepositoryServiceConfigurationPlugin</type>
  <init-params>
    <value-param>
      <name>conf-path</name>
      <description>JCR configuration file</description>
      <value>war:/ks-extension/jcr/repository-configuration.xml</value>
    </value-param>
  </init-params>
</component-plugin>
```

- In which:

Name	Set-method	Type	Description
RepositoryServiceConfigurationPlugin	<code>addConfig</code>	<code>org.exoplatform.services.jcr.impl.config.RepositoryServiceConfigurationPlugin</code>	Read the configuration of JCR data to initialize data.

- Init-params

Name	Possible value	Default value	Description
conf-path	<code>string</code>	<code>war:/ks-extension/jcr/repository-configuration.xml</code>	The path to the <code>repository-configuration.xml</code> file.

2.2.1.2. Initialize workspace name and repository name in storage-configuration.xml

In details:

Once the `war:/ks-extension/jcr/repository-configuration.xml` file has been initialized, the server will load the `storage-configuration.xml` file, and the `setLocation` function in the `org.exoplatform.ks.common.conf.DataLocationPlugin` component will run.

```
<external-component-plugins>
  <target-component>org.exoplatform.ks.common.jcr.KSDataLocation</target-component>
  <component-plugin>
    <name>ks.data.location</name>
    <set-method>setLocation</set-method>
    <type>org.exoplatform.ks.common.conf.DataLocationPlugin</type>
    <init-params>
      <value-param>
        <name>repository</name>
        <description>JCR repository for KS data</description>
        <value>repository</value>
      </value-param>
      <value-param>
        <name>workspace</name>
        <description>workspace for KS data</description>
        <value>knowledge</value>
      </value-param>
    </init-params>
  </component-plugin>
</external-component-plugins>
```

In which,

Value-param	Possible value	Default value	Description
repository	string	repository	The JCR repository for KS data.
workspace	string	knowledge	The workspace for KS data.

2.2.1.3. Initialize data

Once the workspace name and repository name are registered, the server will load `org.exoplatform.services.jcr.ext.hierarchy.NodeHierarchyCreator` and the `addPaths` function in `org.exoplatform.services.jcr.ext.hierarchy.impl.AddPathPlugin` is called. Then, the data location will be built.

```
<component-plugin>
  <name>addPaths</name>
  <set-method>addPlugin</set-method>
  <type>org.exoplatform.services.jcr.ext.hierarchy.impl.AddPathPlugin</type>
  <init-params>
    <object-param>
      <name>ks.storage</name>
      <description>ks data storage tree</description>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig">
        <field name="repository">
          <string>repository</string>
        </field>
        <field name="workspaces">
          <collection type="java.util.ArrayList">
            <value>
              <string>knowledge</string>
            </value>
          </collection>
        </field>
      </object>
    </object-param>
  </init-params>
</component-plugin>
```

```

    </collection>
  </field>
  <field name="jcrPaths">
    <collection type="java.util.ArrayList">
      <value>
        <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
          <field name="alias">
            <string>eXoApplications</string>
          </field>
          <field name="path">
            <string>/exo:applications</string>
          </field>
          <field name="permissions">
            <collection type="java.util.ArrayList">
              <value>
                <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
                  <field name="identity">
                    <string>*/platform/administrators</string>
                  </field>
                  <field name="read">
                    <string>true</string>
                  </field>
                  <field name="addNode">
                    <string>true</string>
                  </field>
                  <field name="setProperty">
                    <string>true</string>
                  </field>
                  <field name="remove">
                    <string>true</string>
                  </field>
                </object>
              </value>
            </collection>
          </field>
        </object>
      </value>
    </collection>
  </field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>ksUserAvatar</string>
    </field>
    <field name="path">
      <string>/exo:applications/ksUserAvatar</string>
    </field>
    <field name="nodeType">
      <string>nt:unstructured</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

```

```

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>ForumService</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService</string>
    </field>
    <field name="nodeType">
      <string>exo:forumHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>ForumSystem</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService/ForumSystem</string>
    </field>
    <field name="nodeType">
      <string>exo:forumSystem</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>

```

```

<object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
  <field name="alias">
    <string>UserProfileHome</string>
  </field>
  <field name="path">
    <string>/exo:applications/ForumService/ForumSystem/UserProfileHome</string>
  </field>
  <field name="nodeType">
    <string>exo:userProfileHome</string>
  </field>
  <field name="permissions">
    <collection type="java.util.ArrayList">
      <value>
        <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
          <field name="identity">
            <string>*/platform/administrators</string>
          </field>
          <field name="read">
            <string>true</string>
          </field>
          <field name="addNode">
            <string>true</string>
          </field>
          <field name="setProperty">
            <string>true</string>
          </field>
          <field name="remove">
            <string>true</string>
          </field>
        </object>
      </value>
    </collection>
  </field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>StatisticHome</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService/ForumSystem/StatisticHome</string>
    </field>
    <field name="nodeType">
      <string>exo:statisticHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">

```

```

<field name="alias">
  <string>forumStatistic</string>
</field>
<field name="path">
  <string>/exo:applications/ForumService/ForumSystem/StatisticHome/forumStatistic</string>
</field>
<field name="nodeType">
  <string>exo:forumStatistic</string>
</field>
<field name="permissions">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
        <field name="identity">
          <string>*/platform/administrators</string>
        </field>
        <field name="read">
          <string>true</string>
        </field>
        <field name="addNode">
          <string>true</string>
        </field>
        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>true</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>AdministrationHome</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService/ForumSystem/AdministrationHome</string>
    </field>
    <field name="nodeType">
      <string>exo:administrationHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">

```



```

    <string>BanIPHome</string>
  </field>
  <field name="path">
    <string>/exo:applications/ForumService/ForumSystem/BanIPHome</string>
  </field>
  <field name="nodeType">
    <string>exo:banIPHome</string>
  </field>
  <field name="permissions">
    <collection type="java.util.ArrayList">
      <value>
        <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
          <field name="identity">
            <string>*:/platform/administrators</string>
          </field>
          <field name="read">
            <string>true</string>
          </field>
          <field name="addNode">
            <string>true</string>
          </field>
          <field name="setProperty">
            <string>true</string>
          </field>
          <field name="remove">
            <string>true</string>
          </field>
        </object>
      </value>
    </collection>
  </field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>forumBanIP</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService/ForumSystem/BanIPHome/forumBanIP</string>
    </field>
    <field name="nodeType">
      <string>exo:banIP</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*:/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>ForumData</string>

```

```

</field>
<field name="path">
  <string>/exo:applications/ForumService/ForumData</string>
</field>
<field name="nodeType">
  <string>exo:forumData</string>
</field>
<field name="permissions">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
        <field name="identity">
          <string>*/platform/administrators</string>
        </field>
        <field name="read">
          <string>true</string>
        </field>
        <field name="addNode">
          <string>true</string>
        </field>
        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>true</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>TopicTypeHome</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService/ForumData/TopicTypeHome</string>
    </field>
    <field name="nodeType">
      <string>exo:topicTypeHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>CategoryHome</string>
    </field>

```

```

<field name="path">
  <string>/exo:applications/ForumService/ForumData/CategoryHome</string>
</field>
<field name="nodeType">
  <string>exo:categoryHome</string>
</field>
<field name="permissions">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
        <field name="identity">
          <string>*/platform/administrators</string>
        </field>
        <field name="read">
          <string>true</string>
        </field>
        <field name="addNode">
          <string>true</string>
        </field>
        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>true</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>TagHome</string>
    </field>
    <field name="path">
      <string>/exo:applications/ForumService/ForumData/TagHome</string>
    </field>
    <field name="nodeType">
      <string>exo:tagHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>forumBBCode</string>
    </field>
    <field name="path">

```

```

    <string>/exo:applications/ForumService/ForumData/forumBBCode</string>
  </field>
  <field name="nodeType">
    <string>exo:forumBBCodeHome</string>
  </field>
  <field name="permissions">
    <collection type="java.util.ArrayList">
      <value>
        <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
          <field name="identity">
            <string>*/platform/administrators</string>
          </field>
          <field name="read">
            <string>true</string>
          </field>
          <field name="addNode">
            <string>true</string>
          </field>
          <field name="setProperty">
            <string>true</string>
          </field>
          <field name="remove">
            <string>true</string>
          </field>
        </object>
      </value>
    </collection>
  </field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>faqApp</string>
    </field>
    <field name="path">
      <string>/exo:applications/faqApp</string>
    </field>
    <field name="nodeType">
      <string>exo:faqHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>settingHome</string>
    </field>
    <field name="path">
      <string>/exo:applications/faqApp/settingHome</string>
    </field>
  </object>
</value>

```

```

</field>
<field name="nodeType">
  <string>exo:faqSettingHome</string>
</field>
<field name="permissions">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
        <field name="identity">
          <string>*/platform/administrators</string>
        </field>
        <field name="read">
          <string>true</string>
        </field>
        <field name="addNode">
          <string>true</string>
        </field>
        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>true</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>userSettingHome</string>
    </field>
    <field name="path">
      <string>/exo:applications/faqApp/settingHome/userSettingHome</string>
    </field>
    <field name="nodeType">
      <string>exo:faqUserSettingHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>categories</string>
    </field>
    <field name="path">
      <string>/exo:applications/faqApp/categories</string>
    </field>
  </object>
</value>

```

```

<field name="nodeType">
  <string>exo:faqCategory</string>
</field>
<field name="mixinTypes">
  <collection type="java.util.ArrayList">
    <value>
      <string>mix:faqSubCategory</string>
    </value>
  </collection>
</field>
<field name="permissions">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
        <field name="identity">
          <string>*/platform/administrators</string>
        </field>
        <field name="read">
          <string>true</string>
        </field>
        <field name="addNode">
          <string>true</string>
        </field>
        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>true</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>templateHome</string>
    </field>
    <field name="path">
      <string>/exo/applications/faqApp/templateHome</string>
    </field>
    <field name="nodeType">
      <string>exo:templateHome</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
      </collection>
    </field>
  </object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">

```

```

<field name="alias">
  <string>eXoPolls</string>
</field>
<field name="path">
  <string>/exo:applications/eXoPolls</string>
</field>
<field name="nodeType">
  <string>nt:unstructured</string>
</field>
<field name="permissions">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
        <field name="identity">
          <string>*/platform/administrators</string>
        </field>
        <field name="read">
          <string>true</string>
        </field>
        <field name="addNode">
          <string>true</string>
        </field>
        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>true</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</value>

<value>
  <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$JcrPath">
    <field name="alias">
      <string>groupsPath</string>
    </field>
    <field name="path">
      <string>/Groups</string>
    </field>
    <field name="permissions">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>*/platform/administrators</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>true</string>
            </field>
            <field name="setProperty">
              <string>true</string>
            </field>
            <field name="remove">
              <string>true</string>
            </field>
          </object>
        </value>
        <value>
          <object type="org.exoplatform.services.jcr.ext.hierarchy.impl.HierarchyConfig$Permission">
            <field name="identity">
              <string>any</string>
            </field>
            <field name="read">
              <string>true</string>
            </field>
            <field name="addNode">
              <string>false</string>
            </field>

```

```

        <field name="setProperty">
          <string>true</string>
        </field>
        <field name="remove">
          <string>>false</string>
        </field>
      </object>
    </value>
  </collection>
</field>
</object>
</object-param>
</init-params>
</component-plugin>

```

2.2.2. Roles Configuration

The roles plug-in component defines roles in Forum of eXo Platform 3. This convenient application defines access to a set of functions within the application. Currently, it only defines the person who has the *administrator* role. Administrators can get access to administration functions. At runtime, the application gets data from the roles plug-in to decide whether the logged user has the administrative role or not.

Configuration

The plug-in is configured in the *roles-configuration.xml* file.

When the user signs in, his or her username, group and membership will be compared with the user roles defined in the *.xml* file that is provided by the roles plug-in component.

In particular, at runtime of ForumService, the roles plug-in component is called. The role plug-in is configured in the *roles-configuration.xml* file. The plug-in component named *add.role.rules.plugin* will be referred to *org.exoplatform.ks.common.conf.RoleRulesPlugin* to create users for Forum corresponding to users who exist in the organization database. In addition, the list of users who have *administration* roles are also defined.

```

<component-plugin>
  <name>add.role.rules.plugin</name>
  <set-method>addRolePlugin</set-method>
  <type>org.exoplatform.ks.common.conf.RoleRulesPlugin</type>
  <description>add role rules</description>
  <init-params>
    <value-param>
      <name>role</name>
      <description>name of the role</description>
      <value>ADMIN</value>
    </value-param>
    <values-param>
      <name>rules</name>
      <description>rules of the role</description>
      <value>root</value>
      <!--value>admin</value -->
      <!--value>member:/demo</value-->
      <!--value>/forums/admin</value-->
      <!--value>*/forum/admin</value-->
      <!--value>/platform/administrators</value-->
      <!--value>manager:/platform/users</value-->
      <!--value>*/somegroup/somesubgroup</value-->
      <!--value>manager:/somegroup/someothergroup</value-->
    </values-param>
  </init-params>
</component-plugin>

```


In which:

Name	Set-method	Type	Description
add.role.rules.plugin	addRolePlugin	org.exoplatform.ks.common.conf.RoleRulesPlugin	Add role rules.

- Init-params:

Name	Possible value	Default value	Description
role	string	ADMIN	The name of role.
rules	string	root	The rules of role.

- When the *role-configuration.xml* file is executed, the administration role (with ADMIN value) will be checked and assigned to a matrix of *users/groups/memberships* defined inside the "value" tags as below:

```
<value>...</value>
```

For example:

```
...
<value>root</value>
<value>john</value>
<value>/platform/administrators</value>
<value>member:/VIP</value>
<value>validator:/VIP</value>
...
```

In the example above, the default administrators of Forum include *root*, *john*, users in */platform/administrators* group and users who have *member/validator* memberships in the VIP group.

When being *root*, the users who belong to the */platform/administrators* group or who have *member/validator* memberships in the VIP group and sign in the Forum, they will be identified as the default administrator of Forum.

To add or remove the default administrator of the Forum, simply edit the *roles-configuration.xml* file, add or remove the relevant "value" tags.

```
...
<values-param>
...
<value>...</value>
...
</values-param>
...
```

The default administrators of the Forum can only change their roles by editing in the *roles-configuration.xml* file.

At runtime, modifications in the *roles-configuration.xml* file will be read and database will be updated. Normal users of the Forum and default administration will be created correspondingly.

2.2.3. ProfileProvider Configuration

Forum and FAQ applications are to show some information about posters. The way to retrieve that information is pluggable through the *ContactProvider* component.

For public internet websites, users can provide personal information, such as personal email address and location. To enable, simply override the *ContactProvider* component in your configuration.

Configuration

Configure the *profile-configuration.xml* file as shown below:

```
<component>
  <key>org.exoplatform.ks.common.user.ContactProvider</key>
  <type>org.exoplatform.ks.common.user.DefaultContactProvider</type>
  <!--<type>org.exoplatform.ks.ext.common.SocialContactProvider</type> -->
</component>
```

When Knowledge is integrated in eXo Platform and if you want to use ProfileProvider from eXo Social, you need to change "**type**" *org.exoplatform.ks.common.user.DefaultContactProvider* into *org.exoplatform.ks.ext.common.SocialContactProvider*.

Use ContactProvider

You can get the *ContactProvider* as follows:

```
public CommonContact getPersonalContact(String userId) throws Exception {
  try {
    if(userId.indexOf(Utils.DELETED) > 0) return new CommonContact();
    ContactProvider provider = (ContactProvider) PortalContainer.getComponent(ContactProvider.class);
    return provider.getCommonContact(userId);
  } catch (Exception e) {
    return new CommonContact();
  }
}
```

In Knowledge, when using ContactProvider, you can use one of two following classes:

- [DefaultContactProvider](#)
- [SocialContactProvider](#)

DefaultContactProvider

When you start the Tomcat, the *DefaultContactProvider* class will be initialized and the *OrganizationService* component is set within the *DefaultContactProvider* component.

The *DefaultContactProvider* class allows you to get user information via the *OrganizationService* component.

```
public CommonContact getCommonContact(String userId) {
  CommonContact contact = new CommonContact();
  try {
    User user = orgService.getUserHandler().findUserByName(userId);
    UserProfile profile = orgService.getUserProfileHandler().findUserProfileByName(userId);
    contact.setEmailAddress(user.getEmail());
    contact.setFirstName(user.getFirstName());
    contact.setLastName(user.getLastName());
    if(profile.getUserInfoMap() != null) {
      contact.setAvatarUrl(profile.getAttribute("user.other-info.avatar.url"));
    }
  }
}
```

```

        contact.setBirthday(profile.getAttribute("user.bdate"));
        contact.setCity(profile.getAttribute("user.home-info.postal.city"));
        contact.setCountry(profile.getAttribute("user.home-info.postal.country"));
        contact.setGender(profile.getAttribute("user.gender"));
        contact.setJob(profile.getAttribute("user.jobtitle"));
        contact.setMobile(profile.getAttribute("user.home-info.telecom.mobile.number"));
        contact.setPhone(profile.getAttribute("user.business-info.telecom.telephone.number"));
        contact.setWebSite(profile.getAttribute("user.home-info.online.uri"));
    }
} catch (Exception e) {
    log.error("Could not retrieve forum user profile for " + userId + " ", e);
}
return contact;
}

```

- The information which is get by the user includes:

Name	Type	Description
email	String	Email of user.
firstName	String	First name of user.
lastName	String	Last name of user.

- The information which is get via *UserProfile* includes:

Attribute	Type	Description
user.other-info.avatar.url	String	The path containing the user's avatar.
user.bdate	String	The user's birthday.
user.home-info.postal.city	String	The home city of user.
user.home-info.postal.country	String	The home country of user.
user.gender	String	The user's gender.
user.jobtitle	String	The user's job.
user.home-info.telecom. mobile.number	String	The home phone number of user.
user.business-info.telecom. telephone.number	String	The mobile number of user.
user.home-info.online.uri	String	The individual websites of user.

SocialContactProvider The *SocailContactProvider* class gets users' profiles by *userId* via the *IdentityManager* class.

```

public CommonContact getCommonContact(String userId) {
    CommonContact contact = new CommonContact();
    try {
        IdentityManager identityM = (IdentityManager) PortalContainer.getInstance().getComponentInstanceOfType(IdentityManager.class);
        Identity userIdentity = identityM.getIdentity(OrganizationIdentityProvider.NAME, userId, true);
        Profile profile = userIdentity.getProfile();
        if (profile.contains(Profile.EMAIL)) {
            contact.setEmailAddress(profile.getProperty(Profile.EMAIL).toString());
        }
        if (profile.contains(Profile.FIRST_NAME)) {
            contact.setFirstName(profile.getProperty(Profile.FIRST_NAME).toString());
        }
        if (profile.contains(Profile.LAST_NAME)) {
            contact.setLastName(profile.getProperty(Profile.LAST_NAME).toString());
        }
        contact.setAvatarUrl(profile.getAvatarImageSource());
        if (profile.contains(Profile.GENDER)) {
            contact.setGender(profile.getProperty(Profile.GENDER).toString());
        }
    }
}

```

```

if (profile.contains(Profile.CONTACT_PHONES)) {
    contact.setPhone(profile.getProperty(Profile.CONTACT_PHONES).toString());
}
if (profile.contains(Profile.URL)) {
    contact.setWebSite(profile.getProperty(Profile.URL).toString());
}
} catch (Exception e) {
    if (LOG.isDebugEnabled()) LOG.error(String.format("can not load contact from eXo Social Profile with user [%s]", userId), e);
}
return contact;
}

```

2.2.4. Forum Configuration

2.2.4.1. BBCode Configuration

The BBCode plug-in component defines default BBCode data in the `.xml` file, including BBCode tags, for example, I, B, U, SIZE, COLOR.

When the BBCode Service runs, it will get values returned from the BBCode plug-in component to initialize default BBCode data.

Configuration of default BBCode data

The default BBCode data is configured in the `bbcodes-configuration.xml` file.

In particular, at runtime of BBCode Service, the BBCode plug-in component is called. Then, the `bbcodes-configuration.xml` file will be executed, and the component-plugin named `registerBBCodePlugin` will be referred to `org.exoplatform.ks.bbcode.spi.BBCodePlugin` to execute some objects that will generate default data.

```

<component-plugin>
  <name>forum.default.bbcodes</name>
  <set-method>registerBBCodePlugin</set-method>
  <type>org.exoplatform.ks.bbcode.spi.BBCodePlugin</type>
  <description>default supported BBCodes</description>
  <init-params>
    <object-param>
      <name>I</name>
      <description>set text in italic</description>
      <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
        <field name="tagName">
          <string>I</string>
        </field>
        <field name="replacement">
          <string>&lt;i&gt;{param}&lt;/i&gt;</string>
        </field>
        <field name="description">
          <string>Set text in italic</string>
        </field>
        <field name="example">
          <string>[I]This text is italic[/I]</string>
        </field>
        <field name="isOption">
          <string>false</string>
        </field>
        <field name="isActive">
          <string>true</string>
        </field>
      </object>
    </object-param>

    <object-param>
      <name>B</name>
      <description>
        <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
          <field name="tagName">
            <string>B</string>

```

```

    </field>
    <field name="replacement">
      <string>&lt;strong&gt;{param}&lt;/strong&gt;</string>
    </field>
    <field name="description">
      <string>Set text in bold</string>
    </field>
    <field name="example">
      <string>[B]This text is bold[/B]</string>
    </field>
    <field name="isOption">
      <string>>false</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>U</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>U</string>
    </field>
    <field name="replacement">
      <string>&lt;u&gt;{param}&lt;/u&gt;</string>
    </field>
    <field name="description">
      <string>Set text in underline</string>
    </field>
    <field name="example">
      <string>[U]This text is underline[/U]</string>
    </field>
    <field name="isOption">
      <string>>false</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>COLOR</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>COLOR</string>
    </field>
    <field name="replacement">
      <string>&lt;font color="{option}"&gt;{param}&lt;/font&gt;</string>
    </field>
    <field name="description">
      <string>The [color=option] tag allows you to change the color of your text.</string>
    </field>
    <field name="example">
      <string>[COLOR=blue]This text is blue[/COLOR]</string>
    </field>
    <field name="isOption">
      <string>true</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>SIZE</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>SIZE</string>

```

```

    </field>
    <field name="replacement">
      <string>&lt;font size="{option}"&gt;{param}&lt;/font&gt;</string>
    </field>
    <field name="description">
      <string>The [size=option] tag allows you to change the size of your text.</string>
    </field>
    <field name="example">
      <string>[size=+2]this text is two sizes larger than normal[/size]</string>
    </field>
    <field name="isOption">
      <string>true</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>FONT</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>FONT</string>
    </field>
    <field name="replacement">
      <string>&lt;font face="{option}"&gt;{param}&lt;/font&gt;</string>
    </field>
    <field name="description">
      <string>The [font=option] tag allows you to change the font of your text.</string>
    </field>
    <field name="example">
      <string>[font=courier]this text is in the courier font[/font]</string>
    </field>
    <field name="isOption">
      <string>true</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>HIGHLIGHT</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>HIGHLIGHT</string>
    </field>
    <field name="replacement">
      <string>&lt;span style="font-weight: bold; color: blue;"&gt;{param}&lt;/span&gt;</string>
    </field>
    <field name="description">
      <string>The [highlight] tag allows you to make highlight of your text.</string>
    </field>
    <field name="example">
      <string>[highlight]this text is highlighted[/highlight]</string>
    </field>
    <field name="isOption">
      <string>>false</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>LEFT</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>LEFT</string>

```

```

</field>
<field name="replacement">
  <string>&lt;div align="left"&gt;&{param}&lt;/div&gt;</string>
</field>
<field name="description">
  <string>The [left] tag allows alignment text to left. </string>
</field>
<field name="example">
  <string>[LEFT]This text is left-aligned[/LEFT]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>true</string>
</field>
</object>
</object-param>

<object-param>
  <name>RIGHT</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>RIGHT</string>
    </field>
    <field name="replacement">
      <string>&lt;div align="right"&gt;&{param}&lt;/div&gt;</string>
    </field>
    <field name="description">
      <string>The [right] tag allows alignment text to right.</string>
    </field>
    <field name="example">
      <string>[RIGHT]example[/RIGHT]</string>
    </field>
    <field name="isOption">
      <string>[RIGHT]this text is right-aligned[/RIGHT]</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>CENTER</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>CENTER</string>
    </field>
    <field name="replacement">
      <string>&lt;div align="center"&gt;&{param}&lt;/div&gt;</string>
    </field>
    <field name="description">
      <string>The [center] allows alignment text to center.</string>
    </field>
    <field name="example">
      <string>[CENTER]this text is center-aligned[/CENTER]</string>
    </field>
    <field name="isOption">
      <string>>false</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>JUSTIFY</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>JUSTIFY</string>

```

```

</field>
<field name="replacement">
  <string>&lt;div align="justify"&gt;{param}&lt;/div&gt;</string>
</field>
<field name="description">
  <string>The [justify] tag allows alignment text to justify.</string>
</field>
<field name="example">
  <string>[JUSTIFY]this text is justify-aligned[/JUSTIFY]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>true</string>
</field>
</object>
</object-param>

<object-param>
  <name>EMAIL</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>EMAIL</string>
    </field>
    <field name="replacement">
      <string>&lt;a href="mailto:{param}"&gt;{param}&lt;/a&gt;</string>
    </field>
    <field name="description">
      <string>The [email] tag allows you to link to an email address.</string>
    </field>
    <field name="example">
      <string>[email]demo@example.com[/email]</string>
    </field>
    <field name="isOption">
      <string>>false</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>EMAIL-OPT</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>EMAIL</string>
    </field>
    <field name="replacement">
      <string>&lt;a href="mailto:{option}"&gt;{param}&lt;/a&gt;</string>
    </field>
    <field name="description">
      <string>The [email=option] tag allows link to an email address and use an optional parameter to 'name' of this link.</string>
    </field>
    <field name="example">
      <string>[email=demo@example.com]Click Here to Email me[/email] </string>
    </field>
    <field name="isOption">
      <string>true</string>
    </field>
    <field name="isActive">
      <string>true</string>
    </field>
  </object>
</object-param>

<object-param>
  <name>URL</name>
  <description/>
  <object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
    <field name="tagName">
      <string>URL</string>

```



```

</field>
<field name="replacement">
  <string>&lt;a target='_blank' href="{param}"&gt;{param}&lt;/a&gt;</string>
</field>
<field name="description">
  <string>The [url] tag allows link to other websites and files.</string>
</field>
<field name="example">
  <string>[URL]http://www.exoplatfrom.com[/URL]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>>true</string>
</field>
</object>
</object-param>

<object-param>
<name>URL-OPT</name>
<description/>
<object type="org.exoplatfrom.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>URL</string>
  </field>
  <field name="replacement">
    <string>&lt;a target='_blank' href="{option}"&gt;{param}&lt;/a&gt;</string>
  </field>
  <field name="description">
    <string>The [url=option] tag allows link to other websites and files and usean optional parameter to 'name' of thislink.</string>
  </field>
  <field name="example">
    <string>[URL=http://www.exoplatfrom.com]Click goto exoplatfrom website.[/URL]</string>
  </field>
  <field name="isOption">
    <string>>true</string>
  </field>
  <field name="isActive">
    <string>>true</string>
  </field>
</object>
</object-param>

<object-param>
<name>GOTO</name>
<description/>
<object type="org.exoplatfrom.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>GOTO</string>
  </field>
  <field name="replacement">
    <string>&lt;a href="{option}"&gt;{param}&lt;/a&gt;</string>
  </field>
  <field name="description">
    <string>Allows goto directly to link instead of open a new window or a new tab. </string>
  </field>
  <field name="example">
    <string>[goto=http://www.exoplatfrom.com]Goto this link.[/goto]&gt;</string>
  </field>
  <field name="isOption">
    <string>>true</string>
  </field>
  <field name="isActive">
    <string>>true</string>
  </field>
</object>
</object-param>

<object-param>
<name>LIST</name>
<description/>
<object type="org.exoplatfrom.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>LIST</string>

```

```

</field>
<field name="replacement">
  <string>You can not define this bbcode tag. It is defined by the developer.</string>
</field>
<field name="description">
  <string>The [list] tag allows create simple, each bullet is denoted by the [*] tag.</string>
</field>
<field name="example">
  <string>[list][*]list item 1[*]list item 2[/list]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>true</string>
</field>
</object>
</object-param>

<object-param>
<name>LIST-OPT</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>LIST</string>
  </field>
  <field name="replacement">
    <string>You can not define this bbcode tag. It is defined by the developer.</string>
  </field>
  <field name="description">
    <string>The [list=option] tag allows create bulleted lists specifying an option. Within the value portion, each bullet is denoted by the [*] tag.</string>
  </field>
  <field name="example">
    <string>[list=1][*]list item 1[*]list item 2[/list]</string>
  </field>
  <field name="isOption">
    <string>true</string>
  </field>
  <field name="isActive">
    <string>true</string>
  </field>
</object>
</object-param>

<object-param>
<name>IMG</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>IMG</string>
  </field>
  <field name="replacement">
    <string>&lt;img border="0" alt="" src="{param}" class="inlineimg"/&gt;</string>
  </field>
  <field name="description">
    <string>The [img] tag allows you to shows the image indicated by {url}</string>
  </field>
  <field name="example">
    <string>[url=http://www.google.com.vn] [img]http://groups.google.com.vn/groups/img/3nb/groups_medium_vi.gif[/img] [/url]</string>
  </field>
  <field name="isOption">
    <string>>false</string>
  </field>
  <field name="isActive">
    <string>true</string>
  </field>
</object>
</object-param>

<object-param>
<name>QUOTE</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>QUOTE</string>

```

```

</field>
<field name="replacement">
  <string>&lt;div style="background:#ededf7; border:1px solid #d8d8d8; padding:6px 6px 6px 15px; margin:2px 0px;"&gt;{param}&lt;/div>&lt;/string>
</field>
<field name="description">
  <string>The [quote] tag allows attribute content of post.</string>
</field>
<field name="example">
  <string>[quote]Lorem ipsum dolor sit amet[/quote]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>true</string>
</field>
</object>
</object-param>

<object-param>
<name>QUOTE-OPT</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>QUOTE</string>
  </field>
  <field name="replacement">
    <string>&lt;div style="background:#ededf7; border:1px solid #d8d8d8; padding:6px 6px 6px 15px; margin:2px 0px;"&gt;&lt;div>&lt;div>Originally Posted by
&lt;strong>&lt;option>&lt;/strong>&lt;/div>&lt;/div>&lt;div>{param}&lt;/div>&lt;/div>&lt;/string>
  </field>
  <field name="description">
    <string>The [quote=option] tag allows attribute content and user name of poster.</string>
  </field>
  <field name="example">
    <string>[quote=John Doe]Lorem ipsum dolor sit amet[/quote]</string>
  </field>
  <field name="isOption">
    <string>true</string>
  </field>
  <field name="isActive">
    <string>true</string>
  </field>
</object>
</object-param>

<object-param>
<name>CODE</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>CODE</string>
  </field>
  <field name="replacement">
    <string>&lt;div style="background:#ededed; border:1px inset #7b7b7b; margin:5px; overflow:auto;"&gt;&lt;pre style="margin: 0px; padding: 0px; overflow: auto;
text-align: left;" dir="ltr"&gt;&lt;div>{param}&lt;/div>&lt;/pre>&lt;/div>&lt;/string>
  </field>
  <field name="description">
    <string>The [code] tag allows you to view source code html.</string>
  </field>
  <field name="example">
    <string>[code]&lt;div>&lt;div>some text or code html&lt;/div>&lt;/code></string>
  </field>
  <field name="isOption">
    <string>>false</string>
  </field>
  <field name="isActive">
    <string>true</string>
  </field>
</object>
</object-param>

<object-param>
<name>CSS</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">

```

```

<field name="tagName">
  <string>CSS</string>
</field>
<field name="replacement">
  <string>&lt;span class="{option}"&gt;{param}&lt;/span&gt;</string>
</field>
<field name="description">
  <string>The [css=option] tag allows you to add div tag and set class Name for this it.</string>
</field>
<field name="example">
  <string>[css=highlight]Text is highlight[/css]</string>
</field>
<field name="isOption">
  <string>true</string>
</field>
<field name="isActive">
  <string>true</string>
</field>
</object>
</object-param>

<object-param>
<name>SLIDESHARE</name>
<description/>
<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">
  <field name="tagName">
    <string>SLIDESHARE</string>
  </field>
  <field name="replacement">
    <string>
      &lt;div style="width:425px; height:355px;" align="center"&gt;
      &lt;object style="margin:0px" width="425" height="355"&gt;
        &lt;param name="movie" value="http://static.slidesharecdn.com/swf/ssplayer2.swf?doc={option}&amp;rel=0"/&gt;
        &lt;param name="allowFullScreen" value="true"/&gt;
        &lt;param name="allowScriptAccess" value="always"/&gt;
        &lt;embed src="http://static.slidesharecdn.com/swf/ssplayer2.swf?doc={option}&amp;rel=0" type="application/x-shockwave-flash" allowscriptaccess="always"
allowfullscreen="true" width="425" height="355"&gt;
        &lt;/embed&gt;
      &lt;/object&gt;
      &lt;strong&gt;{param}&lt;/strong&gt;&lt;/div&gt;
    </string>
  </field>
  <field name="description">
    <string>The [SLIDESHARE=option] tag allows you to run slide in slidesharecdn.com site.</string>
  </field>
  <field name="example">
    <string>[SLIDESHARE=slideId]My slide[/SLIDESHARE]</string>
  </field>
  <field name="isOption">
    <string>true</string>
  </field>
  <field name="isActive">
    <string>true</string>
  </field>
</object>
</object-param>
</init-params>
</component-plugin>

```

- In which,

Name	Set method	Type	Description
forum.default.bbcodes	registerBBCodePlugin	org.exoplatform.ks.bbcode.spi.BBCodePlugin	Define formats for data displayed on UI.

- The BBCode array is defined by the *org.exoplatform.ks.bbcode.spi.BBCodeData* object as below:

```

<object type="org.exoplatform.ks.bbcode.spi.BBCodeData">

```

```

<field name="tagName">
  <string>|</string>
</field>
<field name="replacement">
  <string>&lt;i&gt;{param}&lt;/i&gt;</string>
</field>
<field name="description">
  <string>Set text in italic</string>
</field>
<field name="example">
  <string>[i]This text is italic[/i]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>>true</string>
</field>
</object>

```

- The BBCode includes basic data which are defined in the field tag with a specific name as below:

```

<field name="tagName">
  <string>|</string>
</field>
<field name="replacement">
  <string>&lt;i&gt;{param}&lt;/i&gt;</string>
</field>
<field name="description">
  <string>Set text in italic</string>
</field>
<field name="example">
  <string>[i]This text is italic[/i]</string>
</field>
<field name="isOption">
  <string>>false</string>
</field>
<field name="isActive">
  <string>>true</string>
</field>

```

In which:

Field name	Value	Description
tagName	string	The text for the BBCode, which is put between two square brackets ([]). For example, for the bold tag, if you type [b], the BBCode tag will be b without any square brackets ([]).
replacement	string	The HTML code that replaces the BBCode entered by the user. Make sure that you include '{param}' (without quotes) to insert the text between opening and closing BBCode tags, and '{option}' for the parameter within the BBCode tag. You can only use <i>option</i> if 'Use Option' is selected.
description	string	The piece of text to describe the BBCode tag, including HTML tags if you want.

Field name	Value	Description
example	string	The sample piece of BBCode to use as an example for the particular BBCode. For example, to demonstrate the usage of [b] tag, enter [b]text[/b].
isOption	true, false	Select the [tag=option] [/tag] style tag, rather than just a [tag][/tag] style tag. This function will be created if you select this option.
isActive	true, false	Activate the BBCode tag.

2.2.4.2. Forums Initializer

The Initialization plug-in component defines the default Forum data in the **.xml** file, including categories, forums, topics and posts.

When the Forum Service runs, it will get values which are returned from the Initialization plug-in component to initialize default data of the Forum.

2.2.4.2.1. Configuration

This section covers the following topics:

- [Default forum data](#)
- [Forum array](#)
- [Forum topics](#)

2.2.4.2.1.1. Default forum data

The default forum data is configured in the *war:webapp/WEB-INF/conf/ksdemo/ks/services-configuration.xml* file.

In particular, when the *ForumService* starts, the Initialization plug-in component is called. Next, the *services-configuration.xml* file is executed. The component-plugin named *addInitialDefaultDataPlugin* will refer to *org.exoplatform.forum.service.conf.InitializeForumPlugin* to execute some objects to create default data for the Forum application.

```
<component-plugin>
  <name>default.data</name>
  <set-method>addInitialDefaultDataPlugin</set-method>
  <type>org.exoplatform.forum.service.conf.InitializeForumPlugin</type>
  <description>description</description>
  <init-params>
    <object-param>
      <name>livedemo.default.configuration</name>
      <description>initial data for live demo</description>
      <object type="org.exoplatform.forum.service.conf.ForumInitialData">
        <field name="categories">
          <collection type="java.util.ArrayList">
            <value>
              <object type="org.exoplatform.forum.service.conf.CategoryData">
                <field name="owner">
                  <string>root</string>
                </field>
                <field name="name">
                  <string>Knowledge Suite</string>
                </field>
                <field name="description">
                  <string>All about eXo KS</string>
                </field>
                <field name="forums">
```

```
<collection type="java.util.ArrayList">
  <value>
    <object type="org.exoplatform.forum.service.conf.ForumData">
      <field name="owner"><string>root</string></field>
      <field name="name"><string>Live demo</string></field>
      <field name="description"><string>Questions about this demo</string></field>
      <field name="topics">
        <collection type="java.util.ArrayList">
          <value>
            <object type="org.exoplatform.forum.service.conf.TopicData">
              <field name="name"><string>Demo data policy</string></field>
              <field name="icon"><string>Shield</string></field>
              <field name="owner"><string>root</string></field>
              <field name="content"><string>
                Welcome to eXo Knowledge Suite live demo!
                We hope you enjoy discovering eXo Forum and FAQ applications features.
                You don't need to be logged in to see the applications in action.
                But the power of KS lies in the rich set of admin/moderation features.
                We didn't want you to miss them so, when you [b][url="/portal/public/classic/register"]
                create a demo account[url]/b], you will be granted full permissions.
                Anybody can become an administrator or a moderator and play in the sandbox!
                As a consequence, the data for this forum (including the accounts) is not meant to stay.
                [b][center]WE MAY RESET FORUMS AND FAQS ANYTIME[/center][b]
                Enjoy and don't forget to send feedback at [email]ks@exoplatform.com[/email]</string></field>
            </object>
          </value>
        </collection>
      </field>
    </object>
  </value>
</collection>
</field>
</object>
</value>
</collection>
</field>
</object>
</object-param>
</init-params>
</component-plugin>
```

- In which,

Name	Set-method	Type	Description
default.data	addInitialDataPlugin	org.exoplatform.forum.service.conf. InitializeForumPlugin	The initial default data of Forum.

- Init-param

Name	Possible value	Default value	Description
livedemo.default.configuration	object	org.exoplatform.forum.service.conf. ForumInitialData	The initial data for live demo.

- Category array

After the *org.exoplatform.forum.service.conf.InitializeForumPlugin* object has been executed, the *org.exoplatform.forum.service.conf.ForumInitialData* object will be called. It returns a category array. The value of category array is defined by the *org.exoplatform.forum.service.conf.CategoryData* object as below:

```

<object-param>
  <name>livedemo.default.configuration</name>
  <description>initial data for live demo</description>
  <object type="org.exoplatform.forum.service.conf.ForumInitialData">
    <field name="categories">
      <collection type="java.util.ArrayList">
        <value>
          <object type="org.exoplatform.forum.service.conf.CategoryData">
            <field name="owner">
              <string>root</string>
            </field>
            <field name="name">
              <string>Knowledge Suite</string>
            </field>
            <field name="description">
              <string>All about eXo KS</string>
            </field>
            ...
          </object>
        </value>
      </collection>
    </field>
  </object>
</object-param>

```

- Category includes some basic data which are defined in the **field** tag with a specific name as below:

```

<field name="owner">
  <string>root</string>
</field>
<field name="name">
  <string>Knowledge Suite</string>
</field>
<field name="description">
  <string>All about eXo KS</string>
</field>

```

In which:

Field	Possible value	Default value	Description
owner	user id	root	The creator of Category.
name	string	Knowledge Suite	The title of Category.
description	string	All about eXo KS	The brief description of Category.

- Modify values of Category

Values of the default Category can be changed by editing text values in the *string* tag of each *field* by the other one. In the sample code above, the *org.exoplatform.forum.service.conf.CategoryData* object is called. It means that only one default Category is defined. If you want to define more default Categories, repeat calling the *org.exoplatform.forum.service.conf.CategoryData* object and define values for the new Category with the sample code as below:

```

<value>
  <object type="org.exoplatform.forum.service.conf.CategoryData">
    <field name=" ">
      ...
    </field>
  </object>
</value>

```



```
</object>
</value>
```

2.2.4.2.1.2. Forum array

Category may contain one or more Forums. The value of the Forum is defined in the **forums** field. It returns a forum array. The value of forum array is defined by the **org.exoplatform.forum.service.conf.ForumData** object as below:

```
<field name="forums">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.forum.service.conf.ForumData">
        <field name="owner"><string>root</string></field>
        <field name="name"><string>Live demo</string></field>
        <field name="description"><string>Questions about this demo</string></field>
        ...
      </object>
    </value>
  </collection>
</field>
```

- Basic Forum data

Forum includes some basic data which are defined in the **field** tag with the specific name as above.

In which:

Field	Possible value	Default value	Description
owner	user id	root	The creator of default Forum.
name	string	Live demo	The name or title of default Forum.
description	string	Questions about this demo	The brief description of default Forum.

The default Forum values can be changed by editing text values in the *string* tag of each *field* by the other one.

In the sample code above, the *org.exoplatform.forum.service.conf.ForumData* object is called only one time. It means that only one default Forum is defined inside the default Category named *Knowledge Suite*. If you want to define more default Forums, repeat calling the *org.exoplatform.forum.service.conf.ForumData* object and define values for the new Forum with the sample code as below:

```
<value>
  <object type="org.exoplatform.forum.service.conf.ForumData">
    <field name=" ">
      ...
    </field>
  </object>
</value>
```

2.2.4.2.1.3. Forum topics

- Forum may contain one or more topics. The value of topic is defined in the **topics** field. It returns a topic array. The value of topic array is defined by the *org.exoplatform.forum.service.conf.TopicData* object as below:

```

<field name="topics">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.forum.service.conf.TopicData">
        <field name="name"><string>Demo data policy</string></field>
        <field name="icon"><string>Tux</string></field>
        <field name="owner"><string>root</string></field>
        <field name="content"><string>Welcome to eXo Forum live demo. ... at anytime.</string></field>
        ...
      </object>
    </value>
  </collection>
</field>

```

A topic includes some basic data which are defined in the *field* tag with a specific name as above.

In which:

Field	Possible value	Default value	Description
name	string	Demo data policy	The name or title of default topic.
icon	string	Tux	The default icon of default topic.
owner	user id	root	The creator of default topic.
content	string	Welcome to eXo Forum live demo...at anytime.	The main content of default topic.

The values of default topic can be changed by editing text values in the *string* tag of each *field*.

With the sample code above, the *org.exoplatform.forum.service.conf.TopicData* object is called only one time. It means that only one default topic is defined inside the default Forum named *Live demo*. If you want to define more default topics, repeat calling the *org.exoplatform.forum.service.conf.TopicData* object and define values for the new topic with the sample code as below:

```

<value>
  <object type="org.exoplatform.forum.service.conf.TopicData">
    <field name=" ">
      ...
    </field>
  </object>
</value>

```

- Topic may contain one or more posts. The value of the post is defined in the *posts* field. It returns a post array. The value of the post array is defined by the *org.exoplatform.forum.service.conf.PostData* object as below:

```

<field name="posts">
  <collection type="java.util.ArrayList">
    <value>
      <object type="org.exoplatform.forum.service.conf.PostData">
        <field name="name"><string>Reply: Demo data policy</string></field>
        <field name="icon"><string>IconsView</string></field>
        <field name="owner"><string>root</string></field>
        <field name="content"><string>Enjoy and don't forget to send feedback at ks@exoplatform.com</string></field>
      </object>
    </value>
  </collection>
</field>

```

```

...
</object>
</value>
</collection>
</field>

```

A post includes some basic data which are defined in the *field* tag with a specific name as above.

In which:

Field	Possible value	Default value	Description
name	string	Reply: Demo data policy	The name or title of default post.
icon	string	IconsView	The default icon of default post.
owner	user id	root	The creator of default post.
content	string	Enjoy and don't forget to send feedback at ks@exoplatform.com	The main content of default post.

The default post values can be changed by editing text values in the *string* tag of each *field*.

With the sample code above, the `org.exoplatform.forum.service.conf.PostData` object is called only one time. It means that only one default post is defined inside the default topic named **Demo data policy**. If you want to define more default posts, repeat calling the `org.exoplatform.forum.service.conf.PostData` object and define values for the new post with the sample code as below:

```

<value>
  <object type="org.exoplatform.forum.service.conf.PostData">
    <field name=" ">
      ...
    </field>
  </object>
</value>

```

By default, the default Forum data can only be changed by modifying the *services-configuration.xml* file.

At runtime, the new changes in the *services-configuration.xml* file will be executed and updated. The default Forum data will be created correspondingly.

2.2.4.2.2. Initial Data Plugin

The Initial Data plugin is configured in the *services-configuration.xml* file. In details, at runtime of Forum Service, the Initialization plugin component is called, the *services-configuration.xml* file will be executed. The component-plugin named *addInitialDataPlugin* will refer to `org.exoplatform.forum.service.conf.ForumInitialDataPlugin` to import some objects to create data for the Forum service. The default data in the .zip or .xml file is initialized as follows:

```

<component-plugin>
  <name>technical.forum</name>
  <set-method>addInitialDataPlugin</set-method>
  <type>org.exoplatform.forum.service.conf.ForumInitialDataPlugin</type>
  <description>Initialize</description>
  <init-params>

```

```

<values-param>
  <name>locations</name>
  <description>location where Forum export format file is stored</description>
  <value>war:/data/forum/data-full-forum.zip</value>
  <!-- value>war:/data/forum/forumCategory.xml</value -->
</values-param>
</init-params>
</component-plugin>

```

- In which:

Name	Set-Method	Type	Description
technical.forum	<code>addInitialDataPlugin</code>	<code>org.exoplatform.forum.service.conf.InitialDataPlugin</code>	Initialize the data plugin

- Init-params

Name	Possible values	Default value	Description
locations	String	<code>war:/data/forum/data-full-forum.zip</code>	The location where the Forum export format file is stored.

2.2.4.3. Auto-prune

The Auto-prune component is to prune inactive topics which have not been viewed, edited or received for a given period. The "prune" operation does not denote to the physical removal of topics, but sets them to invisible. The function helps you not clutter busy forums from outdated information.

When the Job Scheduler runs, it will get values returned from the Auto-prune plug-in component to identify topics which have to be inactivated in the Forum application. These topics will be invisible to users.

Configuration

The properties of Auto-prune plug-in are configured in the `war:webapp/WEB-INF/ks-extension/ks/forum/prune-configuration.xml` file.

In particular, at runtime of Job Scheduler, the Auto-prune plugin component is called. Then, the `prune-configuration.xml` file will be executed. The component-plugin named `ForumDeactiveJob` will refer to `org.exoplatform.forum.service.conf.DeactivePeriodJob` to inactivate topics in Forum which meets predefined inactivation properties.

```

<component-plugin>
  <name>ForumDeactiveJob</name>
  <set-method>addPeriodJob</set-method>
  <type>org.exoplatform.forum.service.conf.DeactivePeriodJob</type>
  <description>add a Deactive job to the JobSchedulerService</description>
  <init-params>
    <properties-param>
      <name>job.info</name>
      <description>save the monitor data periodically</description>
      <property name="jobName" value="DeactiveJob"/>
      <property name="groupName" value="KnowlegedSuite"/>
      <property name="job" value="org.exoplatform.forum.service.conf.DeactiveJob"/>
      <property name="repeatCount" value="0"/>
      <property name="period" value="7200000"/> <!-- 2 hours-->
      <property name="startTime" value="+0"/>
      <property name="endTime" value=""/>
    </properties-param>
    <properties-param>
      <name>deactive.info</name>
      <description/>
    </properties-param>
  </init-params>
</component-plugin>

```

```
<property name="inactiveDays" value="15"/>
<property name="forumName" value="Live demo"/>
</properties-param>
</init-params>
</component-plugin>
```

- In which,

Name	Set-method	Type	Description
ForumDeactiveJob	addPeriodJob	org.exoplatform.forum. service.conf. DeactivePeriodJob	Add a DeactiveJob to the JobSchedulerService.

- The properties for the Auto-prune plug-in are defined in the *property* tag with the format as below:

```
...
<property name="jobName" value="DeactiveJob"/>
<property name="groupName" value="KnowlegedSuite"/>
<property name="job" value="org.exoplatform.forum.service.conf.DeactiveJob"/>
<property name="repeatCount" value="0"/>
<property name="period" value="7200000"/> <!-- 2 hours-->
<property name="startTime" value="+0"/>
<property name="endTime" value=""/>
...
<property name="inactiveDays" value="15"/>
<property name="forumName" value="Live demo"/>
...
```

In details:

Property name	Possible value	Default value	Description
jobname	String	DeactiveJob	The name of job which will be executed.
groupname	String	KnowlegedSuite	The name of application which will be executed.
job	Class path	org.exoplatform.forum. service.conf.DeactiveJob	The reference function of the job which will be executed.
repeatCount	Long	0	The repeating time for the job, meaning that how many times the job will be executed. The 0 value means that <i>DecactiveJob</i> is called at runtime only without repeating. If the value is set to 2 or 3, <i>DecactiveJob</i> will be called two or three times correspondingly.
period	Long	72000000	The interval between job executions.
starttime	Integer	+0	The start time when the function executes. The <i>starttime</i> is 0, meaning that

Property name	Possible value	Default value	Description
endtime	Integer	null	the time to start executing <i>DecactiveJob</i> is the runtime. The end time when the function stops executing. The <i>endtime</i> is blank, meaning that there is no limitation for the end time for <i>DecactiveJob</i>.

With start and end time, you can give a specific date in the format: yyyy-mm-dd HH:mm:ss.sss to define the start and end time for *DecactiveJob*. Besides, inactive information is also defined:

Property name	Possible value	Default value	Description
inactiveDays	Integer	15	The number of days the topic has not been activated. The <i>inactivateDays</i> is set to 1 , meaning that all the topics, which have one inactivated day, will be set as inactivated status. They will be invisible.
forumname	String	Live Demo	The name of Forum which will be checked for Auto-prune. In case the value of <i>forumname</i> is blank, all forums will be checked for the Auto-prune. If the <i>forumname</i> is Live demo, only the Forum named 'Live demo' is checked for the Auto-prune.

By default, the default properties can only be changed by editing its value in the *prune-configuration.xml* file.

At runtime, the new changes in the *prune-configuration.xml* file are executed and updated. After that, the Auto-prune plug-in will be executed, depending on its properties.

2.2.4.4. User Statistics

The Auto-count Active Users component is to calculate the number of active users automatically. A user is considered as the active user only when he/she adds a topic/post in the Forum and his/her last post date matches the predefined interval time.

For example, if one user does not have any new posts after 15 days, he/she is not considered as an active user.

When the Job Scheduler runs, it will get values returned from the Auto-count Active Users plug-in component to identify the number of active users. This value is updated to Active Members information when the user views Forum statistics.

Configuration

The properties of Auto-count Active Users plug-in is configured in the *war:webapp/WEB-INF/ks-extension/ks/forum/statistics-configuration.xml* file.

In details, at runtime of Job Scheduler, the Auto-count Active Users plug-in component is called. Then, the *statistics-configuration.xml* file is executed. The component-plugin named *RecountActiveUserJob* will refer to *org.exoplatform.forum.service.conf.RecountActiveUserPeriodJob* to calculate the number of active users.

```

<component-plugin>
  <name>RecountActiveUserJob</name>
  <set-method>addPeriodJob</set-method>
  <type>org.exoplatform.forum.service.conf.RecountActiveUserPeriodJob</type>
  <description>add a RecountActiveUser job to the JobSchedulerService</description>
  <init-params>
    <properties-param>
      <name>job.info</name>
      <description>save the monitor data periodically</description>
      <property name="jobName" value="RecountActiveUserJob"/>
      <property name="groupName" value="KnowlegedSuite"/>
      <property name="job" value="org.exoplatform.forum.service.conf.RecountActiveUserJob"/>
      <property name="repeatCount" value="0"/>
      <property name="period" value="7200000"/> <!-- 2 hours-->
      <property name="startTime" value="+0"/>
      <property name="endTime" value=""/>
    </properties-param>
    <properties-param>
      <name>RecountActiveUser.info</name>
      <description/>
      <property name="lastPost" value="15"/> <!-- users are active if have last posts in 15 day -->
    </properties-param>
  </init-params>
</component-plugin>

```

- In which,

Name	Method	Type	Description
RecountActiveUserJob	addPeriodJob	org.exoplatform.forum.service.conf.RecountActiveUserPeriodJob	Add a RecountActiveUser job to the JobSchedulerService.

- The properties for Auto-count Active Members plug-in are defined in the property tag as below:

```

...
<property name="jobName" value="RecountActiveUserJob"/>
<property name="groupName" value="KnowlegedSuite"/>
<property name="job" value="org.exoplatform.forum.service.conf.RecountActiveUserJob"/>
<property name="repeatCount" value="0"/>
<property name="period" value="7200000"/>
<property name="startTime" value="+0"/>
<property name="endTime" value=""/>
...
<property name="lastPost" value="15"/>
...

```

In which:

Property name	Possible value	Default value	Description
jobname	String	RecountActiveUserJob	The name of job which will be executed.
groupname	String	KnowlegedSuite	The name of application which will be executed.
job	Class path	org.exoplatform.forum.service.conf.RecountActiveUserJob	The reference function of job which will be executed.

Property name	Possible value	Default value	Description
repeatCount	Long	0	The number of times the job is repeated. If repeatCount is set to 0 , RecountActiveUserJob is called at runtime only without repeating. If the number is set to 2 or 3 , RecountActiveUserJob will be called two or three times.
period	Long	7200000 (millisecond) (equal to two hours)	The interval time to execute the job.
starttime	Integer	0	The start time when the function executes. The <i>starttime</i> is 0, meaning that the time to start executing RecountActiveUserJob is the runtime.
endtime	Integer	null	The end time when the function stops executing. The <i>endtime</i> is blank, meaning that there is no limitation for the end time for RecountActiveUserJob.

With start and end time, you can give a specific date in the format: yyyy-mm-dd HH:mm:ss.sss to define the start and end time for *RecountActiveUserJob*. The information of active time is also defined:

Property name	Possible value	Default value	Description
lastPost	Integer	15	The number of days that the user has added the last post. <i>lastPost</i> is 15, meaning that all users, who have any new posts within 15 days as from their last post date, are active members.

By default, the default properties can only be changed by editing its values in the *statistics-configuration.xml* file.

At runtime, the new changes in the *statistics-configuration.xml* file will be executed and updated. The Auto-count Active Users plug-in will be executed, depending on its properties.

2.2.4.5. Update Statistic Data

UpdateDataJob is used when there are abnormal changes in Forum data (such as migration). By default, UpdateDataJob is disabled at the server start up. When UpdateDataJob is running, it will calculate the statistic data in Forum to make sure that the statistic data are correct.

Configuration

The properties of Forum's UpdateDataJob is configured in */WEB-INF/ks-extension/ks/forum/statistics-configuration.xml* which is located in ks-extension webapp.


```
<component-plugin>
<name>UpdateDataJob</name>
<set-method>addPeriodJob</set-method>
<type>org.exoplatform.services.scheduler.PeriodJob</type>
<description>update topic count and post count to forum service</description>
<init-params>
<properties-param>
<name>job.info</name>
<description>save the monitor data periodically</description>
<property name="jobName" value="UpdateDataJob"/>
<property name="groupName" value="KnowledgeSuite-forum"/>
<property name="job" value="org.exoplatform.forum.service.conf.UpdateDataJob"/>
<property name="repeatCount" value="1"/>
<property name="period" value="30000"/>
<property name="startTime" value="+0"/>
<property name="endTime" value=""/>
</properties-param>
</init-params>
</component-plugin>
```

In which:

Name	Method	Type	Description
UpdateDataJob	addPeriodJob	org.exoplatform.services.scheduler.PeriodJob	Add a UpdateDataJob to the JobSchedulerService.

- The properties for Auto-count Active Members plug-in are defined in the property tag as below:

```
<property name="jobName" value="UpdateDataJob"/>
<property name="groupName" value="KnowledgeSuite-forum"/>
<property name="job" value="org.exoplatform.forum.service.conf.UpdateDataJob"/>
<property name="repeatCount" value="1"/>
<property name="period" value="30000"/>
<property name="startTime" value="+0"/>
<property name="endTime" value=""/>
```

Property name	Possible value	Default value	Description
jobname	String	UpdateDataJob	The name of job which will be executed.
groupname	String	KnowledgeSuite-forum	The name of application which will be executed.
job	Class path	org.exoplatform.forum.service.conf.UpdateDataJob	The fully qualified name of job which will be executed.
repeatCount	Long	1	The number of times the job is repeated. If repeatCount is set to 1 , RecountActiveUserJob is called at runtime only without repeating. If the number is set to 2 or 3 , RecountActiveUserJob will be called two or three times.
period	Long		

Property name	Possible value	Default value	Description
		30000 (millisecond) (equal to two hours)	The interval time to execute the job.
starttime	Integer	0	The start time when the function executes. The <i>starttime</i> is 0, meaning that the time to start executing <i>RecountActiveUserJob</i> is the runtime.
endtime	Integer	null	The end time when the function stops executing. The <i>endtime</i> is blank, meaning that there is no limitation for the end time for <i>UpdateDataJob</i> .

With start and end time, you can give a specific date in the format: yyyy-mm-dd HH:mm:ss.sss to define the start and end time for *UpdateDataJob*.

2.2.4.6. Default User Profile

The default Forum settings are a set of settings for a new account. It contains declarations of time zone, short date format, long date format, time format, maximum topics per page, maximum posts per page and flag for showing forum jump or not. The settings are simple, and users can change such settings to UI-based functions later.

Configuration

This configuration is declared in the file named *ks-configuration.xml*. Its path is "[tomcat source]/webapps/ks-extension/WEB-INF/ks-extension/ks/ks-configuration.xml" if you are running the tomcat and "[project source]/extension/webapp/src/main/webapp/WEB-INF/ks-extension/ks/ks-configuration.xml" if you are in the development phrase.

```

...
<external-component-plugins>
  <target-component>org.exoplatform.services.organization.OrganizationService</target-component>
  <component-plugin>
    ...
    <init-params>
      <properties-param>
        <name>user.profile.setting</name>
        <description>set default user profile</description>
        <property name="timeZone" value="GMT"/>
        <property name="shortDateFormat" value="MM/dd/yyyy"/>
        <property name="longDateFormat" value="DDD,MMM dd,yyyy"/>
        <property name="timeFormat" value="hh:mm a"/>
        <property name="maxTopic" value="10"/>
        <property name="maxPost" value="10"/>
        <property name="isShowForumJump" value="true"/>
      </properties-param>
    </init-params>
  </component-plugin>
</external-component-plugins>
...

```

In which:

Parameter	Possible value	Default value	Description
timeZone	Time zone id	GMT	

Parameter	Possible value	Default value	Description
			The time zone set by user. For example: GMT, GMT-05:00, GMT+07:00, GMT+08:30 ... Visit the website: http://java.sun.com/j2se/1.4.2/docs/api/java/util/TimeZone.html for more details.
shortDateFormat	Valid Java Date format	MM/dd/yyyy	The format to display short information of date. Visit the website: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html to ensure the exact format.)
longDateFormat	Valid Java Date format	DDD, MMM dd, yyyy	The format to display a date with more information. Visit the website http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html to ensure the exact format.
timeFormat	valid Java Date format	hh:mm a	The format to view time (for example, hour, minute,). Visit the website: http://java.sun.com/j2se/1.4.2/docs/api/java/text/SimpleDateFormat.html to ensure the exact format.
maxTopic	Integer	10	The maximum number of topics per page.
maxPost	Integer	10	The maximum number of posts per page.
isShowForumJump	true / false	true	Show the forum jump or not.

2.2.5. Answers Configuration

Answers_INITIALIZER

The Initialization plug-in component is to define the default answers data in the *.xml* or *.zip* file. It includes categories of question that should be exported from the Answers application.

When the Answers Service starts, it will get values returned from the Initialization plug-in component to initialize the default Answers data.

Configuration

- Default Answers data

The default Answers data is configured in the *services-configuration.xml* file.

In details, at runtime of Answers Service, the Initialization plug-in component is called, the *services-configuration.xml* file will be executed. The component-plugin named *addInitialDataPlugin* will refer to *org.exoplatform.faq.service.InitializeDataPlugin* to execute some objects to create default data.

The default data in the `.zip` file is initialized as follows:

```
<component-plugin>
  <name>technical-faq</name>
  <set-method>addInitialDataPlugin</set-method>
  <type>org.exoplatform.faq.service.InitialDataPlugin</type>
  <description>Initialize</description>
  <init-params>
    <value-param>
      <name>location</name>
      <description>location where Answers export format file is stored</description>
      <value>war:/data/Technical-FAQ.zip</value>
    </value-param>
  </init-params>
</component-plugin>
```

- In which,

Name	Set-Method	Type	Description
technical-faq	addInitialDataPlugin	org.exoplatform.faq.service.InitialDataPlugin	Initialize the data plugin.

- Init-param

Name	Possible value	Default value	Description
location	string	war:/data/Technical-FAQ.zip	The location where the Answers export format file is stored.

If the default data is in the *XML* format:

```
<value>war:/data/Technical-FAQ.xml</value>
```

By default, the default Answers data can only import if the importing categories do not exist in database.

To initialize default data in multiple files, it is required to declare them in multiple plugins.

```
<component-plugin>
....
</component-plugin>
```

Answers Email Templates Configuration

Answers is configured mainly in the `/webapps/faq/WEB-INF/portlet.xml` file.

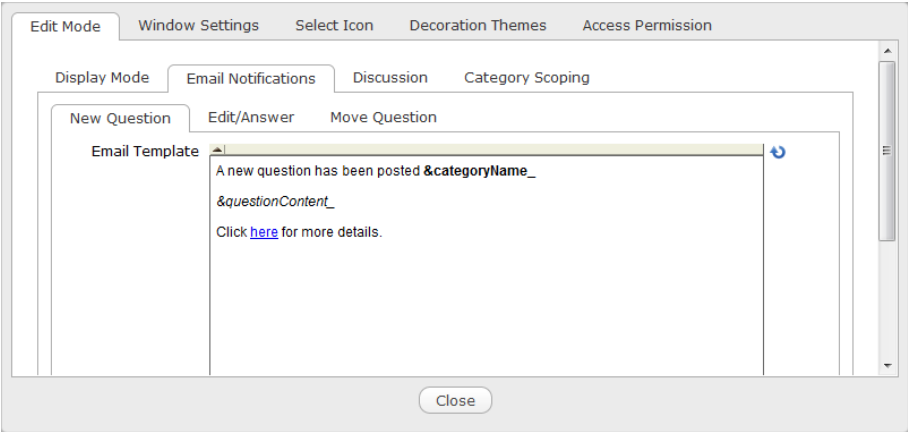


Note

For general information of the Knowledge configuration, refer to [Knowledge Configuration](#) section.

Configuration

The Mail templates use a specific syntax, enabling you to create a customized email message in the Edit mode via three templates: New question, Edit/answer, and Move question.



Parameters which are used in templates consist of:

Parameter	Description
&categoryName_	Load the name of Category.
&questionContent_	Load the question's content.
&questionResponse_	Load the question's answer.
&questionLink_	Load the link to question in the Answers portlet.

2.2.6. Poll Configuration

The Initialization plugin component defines the default Poll data in the *.xml* file, including polls. When the Poll Service runs, it will get values returned from the Initialization plugin component to initialize default Poll data.

Configuration of default Poll data

The default Poll data are configured in the *war:webapp/WEB-INF/conf/ksdemo/ks/services-configuration.xml* file.

In particular, when the Poll service starts, the Initialization plug-in component is called. Next, the *services-configuration.xml* file is executed. The component-plugin named *addInitialDefaultDataPlugin* will refer to *org.exoplatform.poll.service.InitialDeafaultDataPlugin* to execute some objects to create default data for the Poll application.

```
<component-plugin>
  <name>default.data</name>
  <set-method>addInitialDefaultDataPlugin</set-method>
  <type>org.exoplatform.poll.service.InitialDefaultDataPlugin</type>
  <description>Initialize</description>
  <init-params>
    <object-param>
      <name>livedemo.default.configuration</name>
      <description>initial data for live demo</description>
      <object type="org.exoplatform.poll.service.PollInitialData">
        <field name="pollDatas">
          <collection type="java.util.ArrayList">
            <value>
              <object type="org.exoplatform.poll.service.PollData">
                <field name="parentPath">
                  <string>ksdemo/Polls</string>
                </field>
                <field name="owner">
                  <string>root</string>
                </field>
              </object>
            </value>
          </collection>
        </field>
      </object>
    </object-param>
  </init-params>
</component-plugin>
```

```
<field name="question">
  <string>What color do you like ?</string>
</field>
<field name="timeOut">
  <string>0</string>
</field>
<field name="isMultiCheck">
  <string>>false</string>
</field>
<field name="isAgainVote">
  <string>>false</string>
</field>
<field name="isClosed">
  <string>>false</string>
</field>
<field name="options">
  <collection type="java.util.ArrayList">
    <value><string>Green</string></value>
    <value><string>Blue</string></value>
    <value><string>Red</string></value>
    <value><string>Yellow</string></value>
    <value><string>Orange</string></value>
    <value><string>Purple</string></value>
  </collection>
</field>
</object>
</value>
</collection>
</field>
</object>
</object-param>
</init-params>
</component-plugin>
```

- In which,

Name	Set-method	Type	Description
default.data	addInitialDefaultData	org.exoplatform.poll.service.InitialDefaultDataPlugin	The initial default data of Poll.

- Init-param

Name	Possible value	Default value	Description
livedemo.default.configuration	object	org.exoplatform.poll.service.PollInitialData	The initial data for live demo

- Poll array

After the `org.exoplatform.poll.service.InitialDefaultDataPlugin` object has been executed, the `org.exoplatform.poll.service.PollInitialData` object will be called. It returns a polls array. The value of poll array is defined by the `org.exoplatform.poll.service.PollData` object as below:

```
<name>livedemo.default.configuration</name>
<description>initial data for live demo</description>
<object type="org.exoplatform.poll.service.PollInitialData">
  <field name="pollDatas">
    <collection type="java.util.ArrayList">
      <value>
        <object type="org.exoplatform.poll.service.PollData">
          ....
        </object>
      </value>
    </collection>
  </field>
```

```
</object>
```

- A Poll includes some basic data which are defined in the *field* tag with a specific name as below:

```
....
<field name="parentPath">
  <string>ksdemo/Polls</string>
</field>
<field name="owner">
  <string>root</string>
</field>
<field name="question">
  <string>What color do you like ?</string>
</field>
<field name="timeOut">
  <string>0</string>
</field>
<field name="isMultiCheck">
  <string>>false</string>
</field>
<field name="isAgainVote">
  <string>>false</string>
</field>
<field name="isClosed">
  <string>>false</string>
</field>
<field name="options">
  <collection type="java.util.ArrayList">
    <value><string>Green</string></value>
    <value><string>Blue</string></value>
    <value><string>Red</string></value>
    <value><string>Yellow</string></value>
    <value><string>Orange</string></value>
    <value><string>Purple</string></value>
  </collection>
</field>
....
```

In which:

Field	Possible value	Default value	Description
parentPath	string	ksdemo/Polls	Parent path of Poll data.
owner	user id	root	The creator of Poll.
question	string	What color do you like?	The question for Poll.
timeout	number	0	The time before poll is closed. If value is set to 0, the poll will never be closed.
isMultiCheck	boolean	false	If the value is <i>true</i> , user can vote for multi-options. If the value is <i>false</i> , only one option can be voted.
isAgainVote	boolean	false	If the value is <i>true</i> , user can vote again.
isClose	boolean	false	If the value is <i>true</i> , the poll will be closed.
options	java.util.ArrayList	List of string	list of options for Poll.

- Modify values of Poll

Values of the default Poll can be changed by editing text values in the tag of each *field* by the other one.

2.3. Data Injectors

• Configuration

Instructions on how to activate the *DataInjectorService* component and the detailed configuration of plugins (including ForumDataInjector, WikiDatainjector and AnswerDataInjector) attached to it.

• How to use

Instructions on how to use plugins of the *DataInjectorService* component to inject/reject data.

Data injector is used to create data for the performance benchmark. In Knowledge, they are implemented as plugins attached to the *org.exoplatform.services.bench.DataInjectorService* service. This service is normally registered to the portal container as a general component and handled via RESTful requests.

To use this service, add the following dependency to the Classpath of the server:

```
<dependency>
<groupId>org.exoplatform.commons</groupId>
<artifactId>exo.platform.commons.component</artifactId>
<version>${org.exoplatform.commons.version}</version>
<scope>provided</scope>
</dependency>
```

When you want to inject data for a specified product, you will have to implement a class which extends *org.exoplatform.services.bench.DataInjector* and register it to *DataInjectorService* as a plugin.

In which, methods need to be installed are:

```
public abstract class DataInjector extends BaseComponentPlugin {

    /**
     * get log object.
     * @return
     */
    public abstract Log getLog();

    /**
     * This function should be implemented to execute tasks that require to response data to client.
     * <br>
     * @param params query parameters of a HTTP GET request.
     * @return object that can be serialized to JSON object.
     * @throws Exception
     */
    public abstract Object execute(HashMap<String , String> params) throws Exception;

    /**
     * This function should be implemented to inject data into the product.
     * @param params parameters for injecting. They can be query parameters of a HTTP GET request.
     * @throws Exception
     */
    public abstract void inject(HashMap<String , String> params) throws Exception;

    /**
     * This function should be implemented to clear data that is injected before by {@link #inject()}.
     * @param params parameters for rejecting. They can be query parameters of a HTTP GET request.
     * @throws Exception
     */
}
```



```
public abstract void reject(HashMap<String , String> params) throws Exception;
```

See also

- [Components](#)
- [External Component Plugins](#)

2.3.1. Configuration

To activate *DataInjectorService*, you must register this component to a portal container by the following configuration:

```
<component>
  <type>org.exoplatform.services.bench.DataInjectorService</type>
</component>
```

In Knowledge, there are three plugins attached to the *DataInjectorService* component:

- ForumDataInjector
- WikiDataInjector
- AnswerDataInjector

2.3.1.1. ForumDataInjector

The Forum Data injector is configured to register to *DataInjectorService* by the following code:

```
<external-component-plug-ins>
  <target-component>org.exoplatform.services.bench.DataInjectorService</target-component>
  <component-plug-in>
    <name>ForumDataInjector</name>
    <set-method>addInjector</set-method>
    <type>org.exoplatform.forum.bench.ForumDataInjector</type>
    <description>inject data for Forum</description>
  </component-plug-in>
</external-component-plug-ins>
```

- To inject data, the request link is in the following format:

```
http://[rest-path]/bench/inject/ForumDataInjector/?type=data&q=10,20,30,40,50&pre=cat,for,top,post,att&attSize=1
```

The query parameters description:

Param	Value	Description
q	Number	The quantity of each item in forum. For example, if the value is 10,20,30,40,50 then the injector will create 10 categories, each category will have 20 forums, each forum contains 30 topics, each topic has 40 posts which contains 50 attachments in each posts.

Param	Value	Description
pre	String	The prefixes of category, forum, topic, post and attachment. If "cat,for,top,pos,att" is inputted, the injector will create a set of data include: categories with "cat" prefix, forums with "for" prefix and so on.
attSize	Number	The size of each attachment which is created in a post.

- Every created topic can only be read/modified by user 'root' to grant permission to other members.

2.3.1.2. WikiDatainjector

The Data injector for Wiki is configured as follow:

```
<external-component-plug-ins>
<target-component>org.exoplatform.services.bench.DataInjectorService</target-component>
<component-plug-in>
  <name>WikiDataInjector</name>
  <set-method>addInjector</set-method>
  <type>org.exoplatform.wiki.bench.WikiDataInjector</type>
  <description>inject data for Wiki</description>
</component-plug-in>
</external-component-plug-ins>
```

- To inject data, the request link is in the following format:

```
http://[rest path]/bench/inject/WikiDataInjector/?type=data&q=1,2,3&pre=abc,def,mnp&wo=classic&wt=portal&maxAtt=10&mP=100
```

In which, parameters mean:

Param	Value	Description
type	[data perm]	Type of injector. It can be data or perm which injects data or permission respectively.
q	Number	The number of pages in each depth, separated by commas. For example, if the value is 1,2,3, then the injector will create 1 new child of WikiHome, 2 children per each page created in depth 1 and 3 children for each page created in depth 2.
pre	String	Prefix for page Id in each depth, separated by commas. For example, if the value is abc,def,ghk, then pages in depth 1 have title starting with "abc", title of pages in depth 2 start with "def" and in depth 3 is "ghk".
wo	String	Wiki owner.

Param	Value	Description
wt	String	Wiki type. The value can be: 'portal', 'user', 'group'.
maxAtt	Number	The size of attachment in created pages. the value is evaluated in KByte. If the value is 0 or not set, no attachment is created.
mP	Number	The maximum pages in each injection. The number of created pages must not exceed this value.

For example: To create 1000 child pages under the *root* permission, the link request is as follows:

```
http://host:8080/rest/private/bench/inject/
WikiDataInjector?type=data&q=1,1000&pre=Administrator,subAdministrator&wo=intranet&wt=portal&maxAtt=5&mP=100
```

To grant permission, the request link is in the following format:

```
http://localhost:8080/rest-ksdemo/private/bench/inject/WikiDataInjector/
?type=perm&q=1,2,3&pre=abc,def,ghk&wo=classic&wt=portal&perm=11&users=root,mary&groups=*/platform/user&rscs=true
```

For instance, in the link above, the injector will set the Read and Edit permission for pages of portal/classic which meet the constraint (q=1,2,3 and pre=abc,def,ghk).

You can use these parameters to set up permissions for pages:

Param	Value	Description
q	Number	The number of pages in each depth separated by commas. For example, if value is 1,2,3 then the injector will create 1 new child of WikiHome, 2 children per each page created in depth 1 and 3 children for each page created in depth 2.
pre	String	The prefix for page id in each depth separated by commas. For example, if the value is "abc,def,ghk" then pages in depth 1 will have title starting with "abc", title of pages in depth 2 will start with "def" and in depth 3 is "ghk"
wo	String	The wiki owner separated by commas.
wt	String	The wiki type separated by commas. This value can be: portal, user or group.
users	String,	The list of granted permissions users separated by commas.
groups	String	The list of granted permissions groups separated by commas.

Param	Value	Description
memship	String	The list of granted permissions memberships separated by commas.
perm	[number][number]	The declared permissions. The value must be a string with 2 numbers. The first number is to define Read permission of identity while the other one is for Edit permission. If the number is "zero", the privilege is denied and vise versa. For example, 11 means that both Read and Edit pages permission are granted.
rcs	Boolean	Recursive or not. If the value is true, all pages that meet the constraint will be set permission, or deepest pages (smallest descendants) will be affected.

2.3.1.3. AnswerDataInjector

The following configuration is used for the Answer Data injector:

```
<external-component-plug-ins>
<target-component>org.exoplatform.services.bench.DataInjectorService</target-component>
<component-plug-in>
<name>AnswerDataInjector</name>
<set-method>addInjector</set-method>
<type>org.exoplatform.faq.bench.AnswerDataInjector</type>
<description>inject data for Answer</description>
</component-plug-in>
</external-component-plug-ins>
```

- To inject data, the request link is in the following format:

```
http://[rest path]/bench/inject/AnswerDataInjector/?type=data&q=2,3,4,5,6&pre=cate,ques,answ,comm&att=2&attCp=100&txtCp=100
```

In which:

Param	Value	Description
type	[data perm]	Type of injector. It can be data or perm means injecting data or permission respectively.
q	number,number,etc	The number of items in each depth. For example, if the value is 2,3,4,5,6 then the injector will create 2 new categories with the depth level is '3', add 4 questions in each category, 5 answer and 6 comment in each question. Warning: Do not set the

Param	Value	Description
pre	string, string, etc	depth level to more than 5, because the number of items are calculated by number*depth. The prefix for items id in each depth. For example, if the value is "cate, ques, answ, comm", then the category has the first name/id called "cate", the question has the first name/id called "ques", the answer has the first id called "answ" and the comment has the first id called "comm".
att	number	The number of attachments in one question. If the value is '0' or not set, no attachment is created.
attCp	number	The capacity of one attachment. The value is evaluated in KByte and must be larger than KByte.
txtcp	number	The capacity text of one item (question/answer/comment). The value is evaluated in KByte. If the value is 0 or not set, texts are created randomly.



Note

All number of item injectors are calculated by:

- Categories: cats = numberCat * depth!
- Questions : ques = cats * numberQue
- Answers : anss = quest * numberAns
- Comments : coms = quest * numberComs

All = cats + ques + anss + coms

For example:

q=2,3,4,5,6: All = $2 * 3! + (2 * 3) * 4 * 5 + ((2 * 3!) * 4) * 6 = 588$ (items)

If you change the depth from '3' to '5', the number of items will be 11760.

- To grant permission, the request link is in the following format:

```
http://[rest path]/bench/inject/AnswerDataInjector/?type=perm&q=2,3,4,5,6&pre=cate,ques,answ,comm&att=2&attCp=100&&txtCp=100&view=root,demo,*/platform/user&edit=root,*/platform/manager
```

In which:

Param	Value	Description
view	string, string, etc	The list of granted permissions users/groups/memberships, if the value is

Param	Value	Description
edit	string, string, etc	"any" or not set, everyone can view all category create by prefix. The list of granted permissions users/groups/memberships, if the value is "any" or not set, only user with highest permission (Administrator) can edit all category create by prefix.



Note

In the other way, such settings have been declared in "ksdemo.war/WEB-INF/conf/ksdemo/ks/bench-configuration.xml", therefore, to save time and effort, you can import it to "ksdemo.war/WEB-INF/conf/configuration.xml" and then modify it rather than create new one.

2.3.2. How to use

You can use RESTful service to request to inject or reject data. The format of request link is:

```
http://{domain}/{rest}/bench/{inject|reject}/{plug-inName}?[params]
```

For example, after registering the WikiDataInjector plug-in as above, you can request injection as follows: <http://localhost:8080/rest-ksdemo/bench/inject/WikiDataInjector?mP=10&mA=100&mD=1&rand=false&wo=classic&wt=portal> with 10 childrens of each page created, 100 kb each attachment, 1 depth level for wiki portal classic.

To reject such created data, request this link: <http://localhost:8080/rest-ksdemo/bench/reject/WikiDataInjector>.

Developer reference

This chapter provides you with the basic knowledge of overridden components, internal APIs implemented in Knowledge. Also, you will know how to set up a default template for the FAQ portlet, and how to create a sample action extension in Wiki.

In this chapter, there are the following main topics:

- **Extension points**

Details of configuration plug-in and tutorial of `ForumEventLifeCycle`, `AnswerEventLifeCycle` and `BBCodeRenderer`.

- **Internal API**

Description of REST Services of internal APIs which the Knowledge applications (Forum, Answers and Polls) use to communicate with the server.

- **Wiki Service API**

Description of Wiki Service API used in the Knowledge function.

- **FAQ Template Configuration**

Instructions on how to configure the FQA template and to change its look and feel, information of APIs provided by the `UIComponent`.

- **Actions over a wiki page from external jars**

Instructions on how to extend your own actions packaged in external jars to the Wiki toolbar via the following particular topics:

- Create a new project for action extension
- Create new actions and their corresponding listeners
- Register new actions with `UIExtensionManager`
- Deploy new action extension

- **JCR structure**

Description of the JCR structure of the main Knowledge applications, including Forum, FAQ, Poll and Wiki.

3.1. Extension points

- **ForumEventLifeCycle**

Introduction to `ForumEventLifeCycle` that enables you to listen to the lifecycle of a forum, its configuration plug-in, and steps to use the `ForumEventLifeCycle` class.

- **AnswerEventLifeCycle**

Introduction to `AnswerEventLifeCycle` that installs event updates for the Answers data, its configuration plug-in, and steps to use the `AnswerEventLifeCycle` class.

- **BBCodeRenderer**

Introduction to `BBCodeRenderer` that is used in the core of Knowledge to render BB Codes, its configuration file, and how to register `BBCodeRenderer`.

There are some extension points in Knowledge, so that you can control how these components work by implementing or extending default implementations, then reconfigure these new components in the *configuration.xml* file.

See also

- [Internal API](#)
- [Wiki Service API](#)
- [FAQ Template Configuration](#)
- [Actions over a wiki page from external jars](#)
- [JCR structure](#)

3.1.1. ForumEventLifeCycle

ForumEventLifeCycle enables you to listen to the lifecycle of a forum. By implementing **ForumEventLifeCycle**, you can be notified of new posts and replies, categories and topics. This installation will be injected when the data flow is called to save data.

Configuration plug-in

You can find the configuration file of this component at: *ext/social-integration/src/main/resources/conf/portal/configuration.xml*.

For example, to add a Forum to a space of the Social application and keep new activities of Forum (such as new posts and topics) updated to the activities of space, do as follows:

```
<external-component-plugins>
  <target-component>org.exoplatform.forum.service.ForumService</target-component>
  <component-plugin>
    <name>ForumEventListener</name>
    <set-method>addListenerPlugin</set-method>
    <type>org.exoplatform.ks.ext.impl.ForumSpaceActivityPublisher</type>
  </component-plugin>
</external-component-plugins>
```

Tutorial

To use **ForumEventLifeCycle** class, do the following steps:

1. Create a new class that extends *ForumEventListener*.

For example: class *ABCActivityPublisher*

```
public class ABCActivityPublisher extends ForumEventListener {
  .....
}
```

2. Override functions in this created class. In each function, you can write anythings to meet your needs.

```
public class ABCActivityPublisher extends ForumEventListener {

  public void saveCategory(Category category){
    ....
  }

  public void saveForum(Forum forum){
    ....
  }
}
```



```

}

public void addTopic(Topic topic, String categoryId, String forumId){
    ....
}

public void updateTopic(Topic topic, String categoryId, String forumId){
    ....
}

public void addPost(Post post, String categoryId, String forumId, String topicId){
    ....
}

public void updatePost(Post post, String categoryId, String forumId, String topicId){
    ....
}
}
}

```

- The function *saveCategory* is called when a category is added and/or edited.
- The function *saveForum* is called when a forum is added and/or edited.
- The function *addTopic* is called when a topic is added.
- The function *updateTopic* is called when a topic is updated.
- The function *addPost* is called when a post is added.
- The function *updatePost* is called when a post is updated.

3. Add a new configuration to the *configuration.xml* file with the type that is the class created in the **Step 1**.

```

<external-component-plugins>
  <target-component>org.exoplatform.forum.service.ForumService</target-component>
  <component-plugin>
    <name>ForumEventListener</name>
    <set-method>addListenerPlugin</set-method>
    <type>{package}.{class name}</type>
    <!-- example
    <type>org.exoplatform.ks.ext.impl.ABCActivityPublisher</type>
    -->
  </component-plugin>
</external-component-plugins>

```

3.1.2. AnswerEventLifeCycle

AnswerEventLifeCycle installs event updates for the Answers data that is injected while saving answers, saving questions or posting comments.

Configuration plug-in

You can find the configuration file of this component at: *ext/social-integration/src/main/resources/conf/portal/configuration.xml*.

For example, to add Answers to a space of the Social application and keep new activities of Answers updated to the activities of space, do as follows:

```

<external-component-plugins>
  <target-component>org.exoplatform.faq.service.FAQService</target-component>
  <component-plugin>
    <name>AnswerEventListener</name>
    <set-method>addListenerPlugin</set-method>

```

```
<type>org.exoplatform.ks.ext.impl.AnswersSpaceActivityPublisher</type>
</component-plugin>
</external-component-plugins>
```

In which, **AnswersSpaceActivityPublisher** is the class to implement **ForumEventLifeCycle**.

Tutorial

To use the **AnswerEventLifeCycle** class, do the following steps:

1. Create a new class that extends *AnswerEventListener*.

For example: *ABCActivityPublisher*

```
public class ABCActivityPublisher extends AnswerEventListener {
    ....
}
```

2. Override functions in this created class. In each function, you can write anything to meet your needs.

```
public class ABCActivityPublisher extends AnswerEventListener {

    public void saveQuestion(Question question, boolean isNew){
        ....
    }

    public void saveAnswer(String questionId, Answer answer, boolean isNew){
        ....
    }

    public void saveAnswer(String questionId, Answer[] answers, boolean isNew){
        ....
    }

    public void saveComment(String questionId, Comment comment, boolean isNew){
        ....
    }
}
```

- The function *saveQuestion* is called when a question is added and/or edited.
- The function *saveAnswer* is called when an answer is added and/or edited.
- The function *saveAnswer* is called when answers are added and/or edited.
- The function *saveComment* is called when a comment is added and/or edited.

3. Add a new configuration to the *configuration.xml* file with the type that is the class created in the **Step 1**.

```
<external-component-plugins>
  <target-component>org.exoplatform.faq.service.FAQService</target-component>
  <component-plugin>
    <name>AnswerEventListener</name>
    <set-method>addListenerPlugin</set-method>
    <type>{package}.{class name}</type>
    <!-- example
    <type>org.exoplatform.ks.ext.impl.ABCActivityPublisher</type>
    -->
  </component-plugin>
</external-component-plugins>
```

3.1.3. BBCodeRenderer

BBCodeRenderer is used in the core of Knowledge to render BBCodes. In which, the data input is text, containing BBCode tags. The data output will be BBCode tags which have been encrypted into HTML tags.

You can find the configuration file of this component at: *extension/webapp/src/main/webapp/WEB-INF/ks-extension/ks/forum/bbcode-configuration.xml*.

For example, to register BBCodeRenderer, do as follows:

```
<external-component-plugins>
<target-component>org.exoplatform.ks.rendering.MarkupRenderingService</target-component>
<component-plugin>
  <name>BBCodeRenderer</name>
  <set-method>registerRenderer</set-method>
  <type>org.exoplatform.ks.rendering.spi.RendererPlugin</type>
  <description>BBCode renderer</description>
  <init-params>
    <object-param>
      <name>renderer</name>
      <description>Extended BBCodeRenderer</description>
      <object type="org.exoplatform.ks.bbcode.core.BBCodeRenderer">
        <field name="bbCodeProvider">
          <object type="org.exoplatform.ks.bbcode.core.ExtendedBBCodeProvider"/>
        </field>
      </object>
    </object-param>
  </init-params>
</component-plugin>
</external-component-plugins>
```

In which, **ExtendedBBCodeProvider** is the class to implement **BBCodeProvider**.

3.2. Internal API

This section describes REST Services of internal APIs which applications of Knowledge use to communicate with the server. There are the following applications:

- [Forum application \[69\]](#)
- [Answers application \[70\]](#)
- [Polls application \[70\]](#)

Forum application

The Forum application of Knowledge uses the ForumWebservice to provide all APIs to work with Forum, such as filtering IPs, viewing RSS, and more.

Service name	Service URL	Description	Location
ForumWebservice	\$portalName/ \$restcontextname/ ks/forum/	Call extended services of the Forum application.	Maven groupId: org.exoplatform.ks
			ArtifactId: exo.ks.web.webservice

- **APIs Usage**

Use the following APIs to build all the functions of the Forum application:

Name	Description	Service endpoint	URL	Parameters	Expected values
getMessage	Get all new posts with the number based on the <i>maxcount</i> value.	\$portalName/ \$restcontextname/ ks/ forum/getmessage/		String maxcount	integer
filterIps	Ban IPs with no access to Forum.	\$portalName/ \$restcontextname/ ks/ forum/filter/		String str	String in the IP format.
filterIpBanForum	Ban IPs with no access to a specific Forum.	\$portalName/ \$restcontextname/ ks/ forum/ filterIpBanforum/		String forumId String str	string in the IP format, string in the Forum format
filterTagNameForum	Do the quick search for tags that returns the list of tags.	\$portalName/ \$restcontextname/ ks/ forum/ filterTagNameForum/		String str String userAndTopicId	Tag name id of user and topic
viewrss	Process the request by users when they want to view RSS.	\$portalName/ \$restcontextname/ ks/ forum/rss/		String resourceid	category id forum id topic id
userrss	Process the request by users when they want to view their own collected RSS.	\$portalName / \$restcontextname/ ks/ forum/rss/user/		String resourceid	username id

Answers application

The Answers application of Knowledge uses the FAQWebservice to view RSS.

Service name	Service URL	Description	Location
FAQWebservice	\$portalName/ \$restcontextname/ ks/faq	Call the extended services of the Answers application.	Maven groupId: org.exoplatform.ks ArtifactId: exo.ks.eXoApplication. faq.service

• APIs Usage

Use the following APIs to build all the functions of Answers application:

Name	Description	Service endpoint	URL	Parameters	Expected values
viewrss	View RSS	\$portalName/ \$restcontextname/ ks/ faq/rss/		String resourceid	category id, question id

Polls application

The Polls application of Knowledge uses the PollWebservice to create and interact with the Polls gadget.

Service name	Service URL	Description	Location
PollWebservice	\$portalName/ \$restcontextname/ private/ ks/poll	Call extended services of the Polls application.	Maven groupId: org.exoplatform.ks ArtifactId: exo.ks.eXoApplication. poll.service

• APIs Usage

Use the following APIs to build all functions of the Answers application:

Name	Description	Service URL endpoint	Parameters	Expected values
viewPoll	Return data of the Polls system	\$portalName/ \$restcontextname/ private/ks/poll/ viewpoll/	String pollId	id of poll
votePoll	Update data for the Polls with the returned id	\$portalName/ \$restcontextname/ private/ks/poll votepoll/	String pollId / String indexVote	id of poll integer

See also

- [Extension points](#)
- [Wiki Service API](#)
- [FAQ Template Configuration](#)
- [Actions over a wiki page from external jars](#)
- [JCR structure](#)

3.3. Wiki Service API

• DiffService

This service is used to show the differences between two strings.

• LinkService

This service is used for the WYSIWYG editor to get the configuration.

• PageRenderingCacheService

This service is used to get the rendered content of a wiki page with the support of the cache.

• ResizeImageService

This service is used to resize the image to a desired size.

• RenderingService

This service provides functions to render between syntaxes.

• WikiRestService

This service provides functions to access Wiki data, such as tree data, Help content, images and more.

- [WikiService](#)

This service provides interface for processing database with wikis and pages, including adding, editing, removing and searching data.

See also

- [Extension points](#)
- [Internal API](#)
- [FAQ Template Configuration](#)
- [Actions over a wiki page from external jars](#)
- [JCR structure](#)

3.3.1. DiffService

This service is used to show the differences between two strings.

Method	Param	Return	Description
getDifferencesAsList (String text1, String text2) throws <i>DifferentiationFailedException</i>	text1 : The original content. text2 : The revised content.	List	Return a list of differences between lines of text1 and text2.
getWordDifferencesAsList (String text1, String text2) throws <i>DifferentiationFailedException</i> ,	text1 : The original content. text2 : The revised content.	List	Return a list of differences between words of text1 and text2.
getWordDifferencesAsHTML (String text1, String text2) throws <i>DifferentiationFailedException</i> ,	text1 : The original content. text2 : The revised content.	DiffResult	Return a <i>DiffResult</i> object representing differences between words of text1 and text2.
getDifferencesAsHTML (String text1, String text2, boolean allDoc) throws <i>DifferentiationFailedException</i> ,	text1 : The original content. text2 : The revised content. allDoc : Show all documents.	DiffResult	Return a <i>DiffResult</i> object representing differences between lines of text1 and text2.

3.3.2. LinkService

This service is used for the WYSIWYG editor to get the configuration.

Method	Param	Return	Description
getEntityConfig (EntityReference origin, ResourceReference destination);	origin : The origin of the link. destination : The destination of the link.	EntityConfig : The link configuration object that can be used to insert the link in the origin page.	Create an entity link configuration object (URL, link reference) for a link with the specified origin and destination. The link reference in the returned EntityConfig is relative to the link origin.

Method	Param	Return	Description
parseLinkReference (String linkReference, EntityReference baseReference);	linkReference : A link reference pointing to an entity of the specified type. baseReference : The entity reference used to resolve the linked entity reference.	ResourceReference : A reference to the linked entity.	Parse the given link reference and extract a reference to the linked entity. The returned entity reference is resolved relative to the given base entity reference.

3.3.3. PageRenderingCacheService

This service is used to get the rendered content of a wiki page with the support of the cache.

Method	Param	Return	Description
getRenderedContent (WikiPageParams param, String targetSyntax);	params : The identity parameter of a page to get content. targetSyntax : The syntax to be displayed.	String : The rendered content.	Get the rendered content of a wiki page.
getRenderingCache ();	N/A	ExoCache<MarkupKey, MarkupData> : The rendering cache.	Get the rendering cache.
getPageLinksMap ()	N/A	Map<WikiPageParams, List<WikiPageParams>> : The map of connection.	Return the collection of page connections. In details, a connection is built if the page content includes a link to another page.
addPageLink (WikiPageParams param, WikiPageParams entity)	param : The identity parameter of a source page. entity : The identity parameter of a target page.	void	Record a link between two pages.
invalidateCache (WikiPageParams param)	param : The identity parameter of a page to remove in cache.	void	Invalidate all cache entries linking to a page in case this page is removed, changed or renamed.

3.3.4. ResizeImageService

This service is used to resize the image to a desired size.

Method	Param	Return	Description
resizeImage (String imageName, InputStream is, int requestWidth, int requestHeight, boolean keepAspectRatio);	imageName : The name of the image that is resized. is : The input stream of the image.	InputStream : The resized input stream.	Resize the given image to the specified dimensions.

Method	Param	Return	Description
	requestWidth : The new image width.		
	requestHeight : The new image height.		
	keepAspectRatio : Keep the aspect ratio or not.		
resizeImageByWidth (String imageName, InputStream is, int requestWidth);	imageName : The name of the image that is resized. is : The input stream of the image. requestWidth : The new image width.	InputStream	Resize the given image to the requested width and keep the aspect ratio.
resizeImageByHeight (String imageName, InputStream is, int requestHeight);	imageName : The name of the image that is resized. is : The input stream of the image. requestHeight : The new image height.	InputStream	Resize the given image to the requested height and keep the aspect ratio.

3.3.5. RenderingService

This service provides functions to render between syntaxes.

Method	Param	Return	Description
getExecution() throws ComponentRepositoryException;	N/A	Execution	Get the execution from the XWIKI component manager.
getComponentManager() ;	N/A	ComponentManager : The XWIKI coponent manager.	Get the XWIKI component manager.
render (String markup, String sourceSyntax, String targetSyntax, boolean supportSectionEdit) throws Exception;	markup : The text base to convert. sourceSyntax : The original syntax of markup. targetSyntax : The syntax to convert markup. supportSectionEdit : Support users to edit the section or not.	String : The markup in the target syntax.	Render the markup from the source syntax to the target syntax.

Method	Param	Return	Description
getContentOfSection (String markup, String sourceSyntax, String sectionIndex) throws Exception;	markup : The markup that contains sections. sourceSyntax : The syntax of the markup. sectionIndex : The index of section that gets the content.	String : The content of section.	Get the content of a section specified by <i>sectionIndex</i> .
updateContentOfSection (String markup, String sourceSyntax, String sectionIndex, String newSectionContent) throws Exception;	markup : The markup that contain sections. sourceSyntax : The syntax of the markup. sectionIndex : The index of section that gets the content. newSectionContent : The new content to update to the section.	String : The new content of markup.	Update the content of a section specified by <i>sectionIndex</i> .
parse (String markup, String sourceSyntax) throws Exception;	markup : The markup to parse. sourceSyntax : The syntax of markup.	XDOM : The tree representation of the content as <i>org.xwiki.rendering.block.Block</i> .	Parse the markup to XDOM.
getCssURL ();	N/A	String : The URL of CSS.	Get the CSS URL.
setCssURL (String cssURL);	cssURL : The URL of CSS.	void	Set the CSS URL.

3.3.6. WikiRestService

This service provides functions to access Wiki data, such as tree data, Help content, images and more.

Method	Param	Return	Description
getWikiPageContent (String sessionKey, String wikiContextKey, boolean isMarkup, String data);	sessionKey : The key used to retrieve the editor input value from the session. wikiContextKey : The key used to get the wiki context from the client session. isMarkup : If <i>true</i> , then <i>markup content</i> is returned, otherwise <i>html content</i> is returned.	Response : <i>Status.OK</i> and the page content as <i>TEXT_HTML</i> . <i>Status.INTERNAL_SERVER_ERROR</i> if the exception occurs when getting the content of wiki page.	Get the content of a wiki page.

Method	Param	Return	Description
	data : The content of the wiki page.		
upload (String wikiType, String wikiOwner, String pageId)	wikiType : The wiki type. wikiOwner : The wiki owner. pageId : The page Id of the wiki page.	Response : <i>Status.OK</i> if the upload is successful. <i>HTTPStatus.BAD_REQUEST</i> if the upload fails.	Update a file to the wiki page specified by the parameters.
getTreeData (String type, String path, String currentPath, Boolean showExcerpt, String depth);	type : The tree type. path : The JCR path of the root node. currentPath : The JCR path of the current selected node. showExcerpt : Show the summary of page or not. depth : Start the depth to show the tree node.	Response : <i>Status.OK</i> and <i>APPLICATION_JSON</i> . <i>Status.INTERNAL_SERVER_ERROR</i> if the exception occurs when getting the tree data.	Get data to create a page the tree data as tree.
getRelated (String path);	path : The JCR path of the wiki page from which will get the related pages.	Response : <i>Status.OK</i> and <i>JsonRelatedData</i> as of a wiki page. <i>APPLICATION_JSON</i> . <i>Status.NOT_FOUND</i> if the wiki page cannot be found by the input JCR path. <i>Status.INTERNAL_SERVER_ERROR</i> if the exception occurs when getting related pages.	Get a list of the related pages of a wiki page.
searchData (String keyword, String wikiType, String wikiOwner) throws Exception;	keyword : The keyword to search. wikiType : The wiki type. wikiOwner : The wiki owner.	Response : <i>Status.OK</i> and search result as wiki pages by a keyword. <i>APPLICATION_JSON</i> . <i>Status.INTERNAL_SERVER_ERROR</i> if the exception occurs when searching for a page.	Search through the title of and search result as wiki pages by a keyword.
getImage (UriInfo uriInfo, String wikiType, String wikiOwner, String pageId, String imageId, Integer width);	uriInfo : The base URL information. wikiType : The wiki type.	Response : <i>Status.OK</i> if the exception occurs when getting the image. <i>Status.INTERNAL_SERVER_ERROR</i>	Get an image from a wiki page.

Method	Param	Return	Description
	wikiOwner : The wiki owner.		
	pageId : The wiki page Id.		
	imageId : The image Id.		
	width : The width of the image.		
getHelpSyntaxPage (String syntaxId);	syntaxId : The syntax Id.	Response : <i>Status.OK</i> and Help content as syntax Id. <i>TEXT_HTML</i> . <i>Status.INTERNAL_SERVER_ERROR</i> if the exception occurs when getting the Help content.	Get the Help content by the

3.3.7. WikiService

This service provides interface for processing database with wikis and pages, including adding, editing, removing and searching data.

Method	Param	Return	Description
createPage (String wikiType, String wikiOwner, String title, String parentId) throws Exception;	wikiType : The wiki type. wikiOwner : The owner of the page. title : The title of the new wiki page. parentId : The page Id of the parent page.	Page : The new wiki page.	Create a wiki page specified by the parameters.
createTemplatePage (String title, WikiPageParams params) throws Exception;	title : The title of the template. params : The parameter to specify the place to create a draft page.	Template : A new draft page.	Create a new template specified by the parameters.
initDefaultTemplatePage (String path);	path : The JCR path where the default template is created.	void	Initialize default templates of wiki.
createDraftNewPage (String draftNewPageId) throws Exception;	draftNewPageId : The Id of the draft page.	void	Create a draft page for a new wiki page.
deletePage (String wikiType, String wikiOwner,	wikiType : The wiki type.	boolean	Delete a wiki page specified by the parameters.

Method	Param	Return	Description
String pageId) throws Exception;	wikiOwner : The wiki owner.		
	pageId : The Id of wiki page that is deleted.		
deleteTemplatePage (String wikiType, String wikiOwner, String templateId) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	void	Delete a template specified by the parameters.
	templateId : The template Id which is deleted.		
deleteDraftNewPage (String draftNewPageId) throws Exception;	draftNewPageId : The Id of the draft page which is deleted.	void	Delete a draft page specified by the parameter.
renamePage (String wikiType, String wikiOwner, String pageName, String newName, String newTitle) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	boolean	Rename a wiki page specified by the parameters.
	pageName : The name of the wiki page which is renamed.		
	newName : The new name of the wiki page.		
	newTitle : The new title of the wiki page.		
movePage (WikiPageParams currentLocationParams, WikiPageParams newLocationParams) throws Exception;	currentLocationParams : Specify the current location of the wiki page. newLocationParams : Specify the new location to which the wiki page is moved.	boolean	Move a wiki page to another location.
getWikiPermission (String wikiType, String wikiOwner) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	List<PermissionEntry> : List of permission entry.	Get the permission of wiki.
setWikiPermission (String wikiType, String wikiOwner, List<PermissionEntry> permissionEntries) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	void	Set the permission for wiki.

Method	Param	Return	Description
	permissionEntries : The list of permission entry to set to permissions of the wiki.		
getPageById (String wikiType, String wikiOwner, String pageId) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	Page	Get a wiki page specified by parameters.
	pageId : The page Id.		
getRelatedPage (String wikiType, String wikiOwner, String pageId) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	Page : The wiki page.	Get a renamed wiki page by its old Id. After being renamed, the wiki page Id was changed. The old Id of the wiki page is stored to the link registry of wiki and used to get the wiki page.
	pageId : The old page Id of the wiki page.		
getExsitedOrNewDraftPage (String wikiType, String wikiOwner, String pageId) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	Page : The wiki page for the draft page.	Get a wiki page or create a draft for it if it does not exist.
	pageId : The page Id.		
getPageByUUID (String uuid) throws Exception	uuid : The JCR node UUID of the wiki page.	Page : The wiki page.	Get a wiki page by the node UUID.
getTemplatePage (WikiPageParams params, String templateId) throws Exception	params : The location of the template. templateId : The template Id.	Template	Get a template specified by the parameters.
searchContent (WikiSearchData data) throws Exception;	data : The wiki search data that contain the search query.	PageList<SearchResult>	Search through the content of wiki pages and return search results.
getBreadcumb (String wikiType, String wikiOwner, String pageId) throws Exception;	wikiType : The wiki type. wikiOwner : The wiki owner.	List<BreadcrumbData>	Get a list of breadcrumb data of a wiki page.
	pageId : The page Id.		
getWikiPageParams (BreadcrumbData data) throws Exception	data : The breadcrumb data.	WikiPageParams	Get the wiki page param from breadcrumb data.
search (WikiSearchData data) throws Exception	data : The Wiki Search data that contain the search query.	PageList<SearchResult>	Search for wiki pages that match the search query in the input search data.

Method	Param	Return	Description
searchTemplate (TemplateSearch data) throws Exception	data : The Wiki Search data that contain the search query.	List<TemplateSearchResult>	Search for wiki templates that match the search query in input search data.
searchRenamedPage (String wikiType, String wikiOwner, String pageId) throws Exception	wikiType : The wiki type. wikiOwner : The wiki owner. pageId : The page Id.	List<SearchResult>	Get a list of renamed wiki pages.
searchDataByTitle (WikiSearch data) throws Exception	data : The wiki search data that contain the search query.	List<TitleSearchResult>	Search for the title of wiki pages and return search results.
findByPath (String path, String objectNodeType) throws Exception	path : The JCR node path to get the object. objectNodeType : The node type of the JCR node.	Object : The object that has been got from the JCR node.	Get a wiki object (PageImpl, AttachmentImpl, Template) by the JCR path.
getDefaultWikiSyntaxId ()	N/A	String : The default wiki syntax Id.	Get the default wiki syntax Id.
getPageTitleOfAttachment (String path) throws Exception;	path : The JCR path of the attachment.	String : The page title.	Get the page title of an attachment.
getAttachmentAsStream (String path) throws Exception	path : The path of the attachment.	InputStream : The input stream.	Get the attachment as an stream.
getHelpSyntaxPage (String syntaxId) throws Exception	syntaxId : The syntax Id to get the Help page.	PageImpl : The wiki page that contains the Help content for the syntax.	Get the wiki page that contains the Help content for the syntax.
getMetaDataPage (MetaData metaPage) throws Exception	metaPage : The type of Metadata.	Page : The wiki page that contains the metadata information.	Get the wiki page that contains the metadata information.
getTemplates (WikiPageParams params) throws Exception	params : The location of templates.	Map<String, Template> : The map of templates.	Get the map of template specified by the parameters.
getTemplatesContainer (WikiPageParams params) throws Exception	params : The location of the template container.	TemplateContainer : The template container.	Get the template container specified by the parameters.
modifyTemplate (WikiPageParams params, Template template, String newName, String newDescription, String newContent, String newSyntaxId) throws Exception	params : The location of the template. template : The template to modify. newName : The new name of the template.	void	Modify the wiki template.

Method	Param	Return	Description
	newDescription : The new description of the template.		
	newContent : The new content of the template.		
	newSyntaxId : The new syntax Id of the template.		
isExisting (String wikiType, String wikiOwner, String pageId) throws Exception	wikiType : The wiki type. wikiOwner : The wiki owner. pageId : The page Id.	boolean	Check if the wiki page exists or not.
addComponentPlugin (ComponentPlugin plugin)	plugin : The component plugin.	void	Register a <i>PageWikiListener</i> .
addWikiTemplatePagePlugin (WikiTemplatePlugin plugin)	plugin : The template plugin.	void	Add a wiki template plugin.
getPageListeners ()	N/A	List<PageWikiListener> : The list of <i>PageWikiListener</i> .	Get page listeners.
addRelatedPage (WikiPageParams ordinaryPageParams, WikiPageParams relatedPageParams) throws Exception	ordinaryPageParams : The parameter of the target page to which related pages are added. relatedPageParams : The parameter of the related page.	boolean	Add related pages.
getRelatedPage (WikiPageParams pageParams) throws Exception	pageParams : The parameter which refers to the wiki page.	List<Page> : The list of related pages.	Get a list of related pages of a wiki page.
removeRelatedPage (WikiPageParams ordinaryPageParams, WikiPageParams relatedPageParams) throws Exception	ordinaryPageParams : The parameter of the target page from which related pages are removed. relatedPageParams : The parameter of the related page.	boolean	Remove related pages from the related pages list of a wiki page.

3.4. FAQ Template Configuration

- [Configuration plug-in](#)

Information about the configuration plug-in which is used to automatically set up a default template for the FAQ portlet, and details of properties of the template configuration plug-in.

- [How to change look and feel](#)

Instructions on how to change the template FAQ viewer, either by using plug-in or by using the *Edit* mode.

- [API provided by the UIComponent \(UIViewer.java\)](#)

Introduction to UIViewer, details of APIs and classes (CategoryInfo, QuestionInfo, SubCategoryInfo).

See also

- [Extension points](#)
- [Internal API](#)
- [Wiki Service API](#)
- [Actions over a wiki page from external jars](#)
- [JCR structure](#)

3.4.1. Configuration plug-in

Configuration plug-in is used to automatically set up a default template for the FAQ portlet. When the FAQ service starts, it will get values which are returned from the *TemplatePlugin* component to initialize the template for the FAQ portlet.

The template configuration plug-in is configured in the *templates-configuration.xml* file.

In details:

At runtime of the FAQ Service, *FAQService* component is called, then *templates-configuration.xml* file is executed. The component-plugin named *addTemplatePlugin* will be referred to *org.exoplatform.faq.service.TemplatePlugin* to execute some objects and create default data for the **Forum** application.

```
<external-component-plugins>
  <target-component>org.exoplatform.faq.service.FAQService</target-component>
  <component-plugin>
    <name>faq.default.template</name>
    <set-method>addTemplatePlugin</set-method>
    <type>org.exoplatform.faq.service.TemplatePlugin</type>
    <init-params>
      <value-param>
        <name>viewerTemplate</name>
        <value>war:/ks-extension/ks/faq/templates/FAQViewerPortlet.gtmpl</value>
      </value-param>
    </init-params>
  </component-plugin>
</external-component-plugins>
```

The properties of template configuration plug-in are defined in the *init-params* tag as follows:

```
<init-params>
  <value-param>
    <name>viewerTemplate</name>
    <value>war:/ks-extension/ks/faq/templates/FAQViewerPortlet.gtmpl</value>
  </value-param>
</init-params>
```


Name	Description	Value
viewerTemplate	Path of file template.	war:/ks-extension/ks/faq/templates/FAQViewerPortlet.gtmpl

3.4.2. How to change look and feel

You can change the template FAQ viewer in one of the following two ways:

- By using Plug-in [83]
- By using the **edit mode** [83]

Plug-in

1. Create a file named **FAQViewerPortlet.gtmpl**. The content of the file is the template of the FAQ viewer.
2. Copy this file and paste into *ks-extension/WEB-INF/ks-extension/ks/faq/templates/* that is in the **webapps** folder of the server (tomcat, jboss).

When the server runs, *FAQViewerPortlet.gtmpl* will initialize the template of the FAQ viewer.

Edit Mode

1. Run the server and open the FAQ Portlet.
2. Go to **edit mode** and open the **Edit Template** tab.
3. Edit the content of **text-area-input** and click **Save**.

3.4.3. API provided by the UIComponent (UIViewer.java)

- UIViewer is the child of the component *UIFAQPortlet*. It shows the main content of FAQ portlet.
- List of APIs:

Function name	Param	Return	Description
getCategoryInfo	Empty	CategoryInfo object	Get the object CategoryInfo .
arrangeList	(List< String > list): List of path	A new list is arranged	Arrange a list of path.
render	(String): The content of answers or comments	A new string is converted by function render	Render the content of answers or comments.

- The **CategoryInfo** class:

```
...
private String id;
private String path;
private String name;
private List<String> pathName;
private List<QuestionInfo> questionInfos = new ArrayList<QuestionInfo>();
private List<SubCategoryInfo> subCateInfos = new ArrayList<SubCategoryInfo>();
...
```

Param	Type	Description
id	String	The jcr node name of the category node.
path	String	The jcr node path of the category node.

Param	Type	Description
name	<code>String</code>	The name of the category.
pathName	<code>List<String></code>	The path to the category includes a list of category names.
questionInfos	<code>List<QuestionInfo></code>	The list of QuestionInfo object.
subCateInfos	<code>List<SubCategoryInfo></code>	The list of SubCategoryInfo object.

- The **QuestionInfo** class:

```
...
private String id;
private String question;
private String detail;
private List<String> answers = new ArrayList<String>();
...
```

Param	Type	Description
id	<code>String</code>	The jcr node name of the question node.
question	<code>String</code>	The content of the question.
details	<code>String</code>	Details of the question.
answers	<code>List<String></code>	The list of answers for the question.

- The **SubCategoryInfo** class: The params of this class are the same as those of the **CategoryInfo** class. See [here](#) for more information.

3.5. Actions over a wiki page from external jars

- [Create a new project for action extension](#)

Introduction to the structure of a new project and content of the `pom.xml` file.

- [Create new actions and their corresponding listeners](#)

Details of modifying the `ViewSourceActionComponent` class that allows creating new actions and their corresponding listeners.

- [Register new actions with `UIExtensionManager`](#)

Details of modifying the `configuration.xml` file that allows registering new actions with `UIExtensionManager`, and information of the types of action extensions.

- [Deploy new action extension](#)

Instructions on how to deploy a new action extension.

The toolbar in Wiki is built basing on the UI Extensions framework as described in [Extend eXo applications](#), so you can add your own actions packaged in external jars to it. After reading this section, you will know how to extend actions over a wiki page from external jars.

See also

- [Extension points](#)
- [Internal API](#)

- [Wiki Service API](#)
- [FAQ Template Configuration](#)
- [JCR structure](#)

3.5.1. Create a new project for action extension

Create a Maven project which has the following directory structure:

```

▼ example
  ▼ src/main/java
    ▼ com.acme
      ▸ ViewSourceActionComponent.java
  ▼ src/main/resources
    ▼ conf.portal
      ▸ configuration.xml
  ▸ src
    ▸ pom.xml
  
```

Navigating in the project's folder, you will see the following structure:

- *pom.xml*: the project's POM file.
- *src/main/java/.../ViewSourceActionComponent.java*: a simple action supporting user to view the wiki markup of a page.
- *src/main/resources/conf/portal/configuration.xml*: the configuration file to register your actions with the *org.exoplatform.webui.ext.UIExtensionManager* service.

Here is the content of the *pom.xml* file:

```

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.acme</groupId>
  <artifactId>example</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>eXo Wiki action - Example</name>
  <description>eXo Wiki action - Example</description>
  <dependencies>
    <dependency>
      <groupId>org.exoplatform.portal</groupId>
      <artifactId>exo.portal.webui.core</artifactId>
      <version>3.2.0-PLF-SNAPSHOT</version>
      <scope>provided</scope>
    </dependency>
    <dependency>
      <groupId>org.exoplatform.commons</groupId>
      <artifactId>exo.platform.commons.webui.ext</artifactId>
      <version>1.1.3-SNAPSHOT</version>
      <scope>provided</scope>
    </dependency>
    <dependency>
      <groupId>org.exoplatform.ks</groupId>
      <artifactId>exo.ks.eXoApplication.wiki.service</artifactId>
      <version>2.2.4-SNAPSHOT</version>
      <scope>provided</scope>
    </dependency>
    <dependency>
      <groupId>org.exoplatform.ks</groupId>
      <artifactId>exo.ks.eXoApplication.wiki.webapp</artifactId>
      <version>2.2.4-SNAPSHOT</version>
      <type>jar</type>
      <scope>provided</scope>
    </dependency>
  </dependencies>
  <build>
  
```

```

<plugins>
  <plugin>
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-compiler-plugin</artifactId>
    <configuration>
      <source>1.5</source>
      <target>1.5</target>
    </configuration>
  </plugin>
</plugins>
</build>
</project>

```

3.5.2. Create new actions and their corresponding listeners

Edit the *ViewSourceActionComponent* class as below:

```

package com.acme;

import java.util.Arrays;
import java.util.List;

import org.exoplatform.webui.config.annotation.ComponentConfig;
import org.exoplatform.webui.config.annotation.EventConfig;
import org.exoplatform.webui.event.Event;
import org.exoplatform.webui.ext.filter.UIExtensionFilter;
import org.exoplatform.webui.ext.filter.UIExtensionFilters;
import org.exoplatform.wiki.commons.Utils;
import org.exoplatform.wiki.mow.core.api.wiki.PageImpl;
import org.exoplatform.wiki.webui.UIWikiContentDisplay;
import org.exoplatform.wiki.webui.UIWikiPageContentArea;
import org.exoplatform.wiki.webui.UIWikiPortlet;
import org.exoplatform.wiki.webui.control.action.core.AbstractEventActionComponent;
import org.exoplatform.wiki.webui.control.filter.IsViewModeFilter;
import org.exoplatform.wiki.webui.control.listener.MoreContainerActionListener;

@ComponentConfig (
  template = "app:/templates/wiki/webui/control/action/AbstractActionComponent.gtmpl",
  events = {
    @EventConfig(listeners = ViewSourceActionComponent.ViewSourceActionListener.class)
  }
)
public class ViewSourceActionComponent extends AbstractEventActionComponent {

  public static final String      ACTION = "ViewSource";

  private static final List<UIExtensionFilter> FILTERS = Arrays.asList(new UIExtensionFilter[] { new IsViewModeFilter() });

  @UIExtensionFilters
  public List<UIExtensionFilter> getFilters() {
    return FILTERS;
  }

  @Override
  public String getActionName() {
    return ACTION;
  }

  @Override
  public boolean isAnchor() {
    return false;
  }

  public static class ViewSourceActionListener extends MoreContainerActionListener<ViewSourceActionComponent> {
    @Override
    protected void processEvent(Event<ViewSourceActionComponent> event) throws Exception {
      UIWikiPortlet wikiPortlet = event.getSource().getAncestorOfType(UIWikiPortlet.class);
      UIWikiContentDisplay contentDisplay = wikiPortlet.findFirstComponentOfType(UIWikiPageContentArea.class)
        .getChildById(UIWikiPageContentArea.VIEW_DISPLAY);
    }
  }
}

```

```
PageImpl wikiPage = (PageImpl) Utils.getCurrentWikiPage();
contentDisplay.setHtmlOutput(wikiPage.getContent().getText());
event.getRequestContext().addUIComponentToUpdateByAjax(contentDisplay);
}

}

}
```

3.5.3. Register new actions with UIExtensionManager

Edit the *configuration.xml* file as below:

```
<configuration xmlns="http://www.exoplatform.org/xml/ns/kernel_1_2.xsd" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.exoplatform.org/xml/ns/kernel_1_2.xsd http://www.exoplatform.org/xml/ns/kernel_1_2.xsd">

<external-component-plugins>
<target-component>org.exoplatform.webui.ext.UIExtensionManager</target-component>
<component-plugin>
<name>add.action</name>
<set-method>registerUIExtensionPlugin</set-method>
<type>org.exoplatform.webui.ext.UIExtensionPlugin</type>
<init-params>
<object-param>
<name>ViewSource</name>
<object type="org.exoplatform.webui.ext.UIExtension">
<field name="type"><string>org.exoplatform.wiki.webui.control.MoreExtensionContainer</string></field>
<field name="rank"><int>1000</int></field>
<field name="name"><string>ViewSource</string></field>
<field name="component"><string>com.acme.ViewSourceActionComponent</string></field>
</object>
</object-param>
</init-params>
</component-plugin>
</external-component-plugins>

</configuration>
```

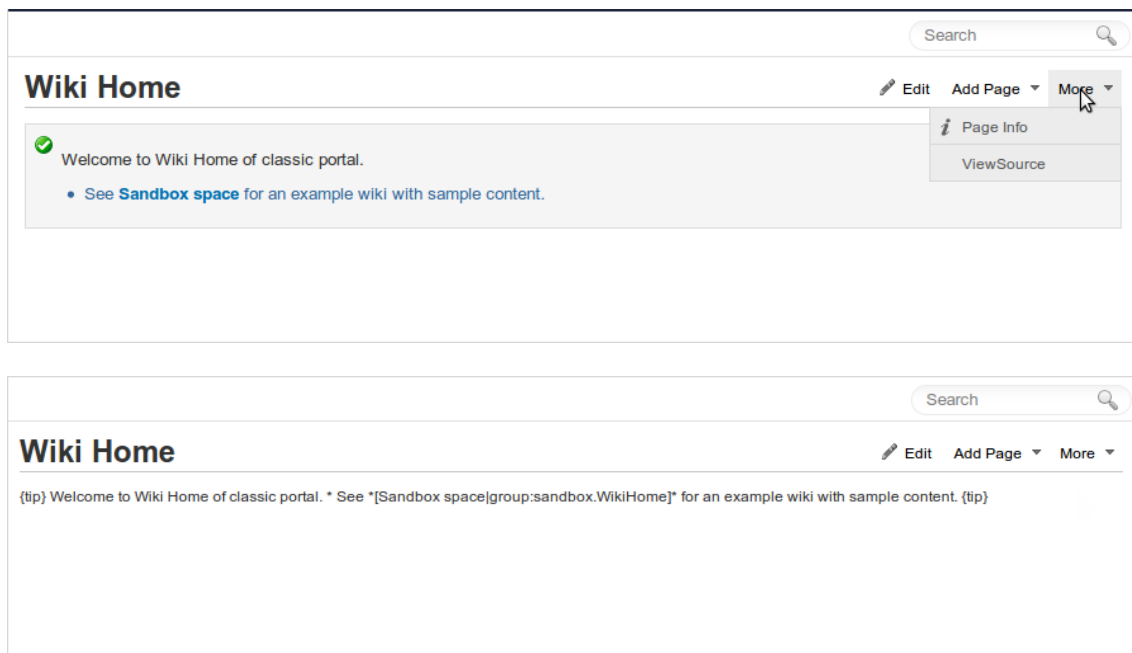
Besides the *MoreExtensionContainer* type, action extension may be one of these following types:

Type	Description
org.exoplatform.wiki.webui.control.UIPageToolBar	Actions will be placed on the Wiki page toolbar at the view mode.
org.exoplatform.wiki.webui.control.AddExtensionContainer	Actions will be placed on the Add Page drop-down list on the Wiki page toolbar at the view mode.
org.exoplatform.wiki.webui.control.MoreExtensionContainer	Actions will be placed on the More drop-down list on the Wiki page toolbar at the view mode.
org.exoplatform.wiki.webui.control.UISubmitToolBar	Actions will be placed on the Wiki page toolbar at the edit mode.
org.exoplatform.wiki.webui.control.UIEditorTabs	Actions will be placed the on Wiki editor tabs.
org.exoplatform.wiki.webui.control.BrowseExtensionContainer	Actions will be placed on the Browse drop-down list at the view mode.
org.exoplatform.wiki.webui.popup.UIWikiSettingsContainer	Actions will be placed on the Wiki setting tabs.

3.5.4. Deploy new action extension

Follow these steps to deploy your new action extension:

1. Build the project by using the **mvn clean install** command.
2. Copy the *target/example-1.0-SNAPSHOT.jar* file into the *PLATFORM_TOMCAT_HOME/webapps/wiki.war/WEB-INF/lib/* directory.
3. Run the tomcat and go to Wiki application, you should see the result as below:



3.6. JCR structure

- **Forum JCR structure**

Introduction to the whole JCR structure of Forum, and comprehensive knowledge of its main nodes: Forum System and Forum Data.

- **FAQ JCR structure**

Introduction to the whole JCR structure of FAQ, and comprehensive knowledge of its main nodes: Category, FAQ setting, Template for FAQ.

- **Poll JCR structure**

Introduction to the whole JCR structure of Poll, and properties of its node type (exo:polls).

- **Wiki JCR structure**

Introduction to the whole JCR structure of Wiki, and comprehensive knowledge of its main nodes: Wiki data and Wikimetadata.

Knowledge is a JCR-based product which uses JCR to store data. The root node of Knowledge is *exo:application* which includes these child nodes: *exo:forumHome*, *exo:faqHome*.

In this section, you will have opportunity to further understand about the JCR structure of Knowledge via the main topics above.

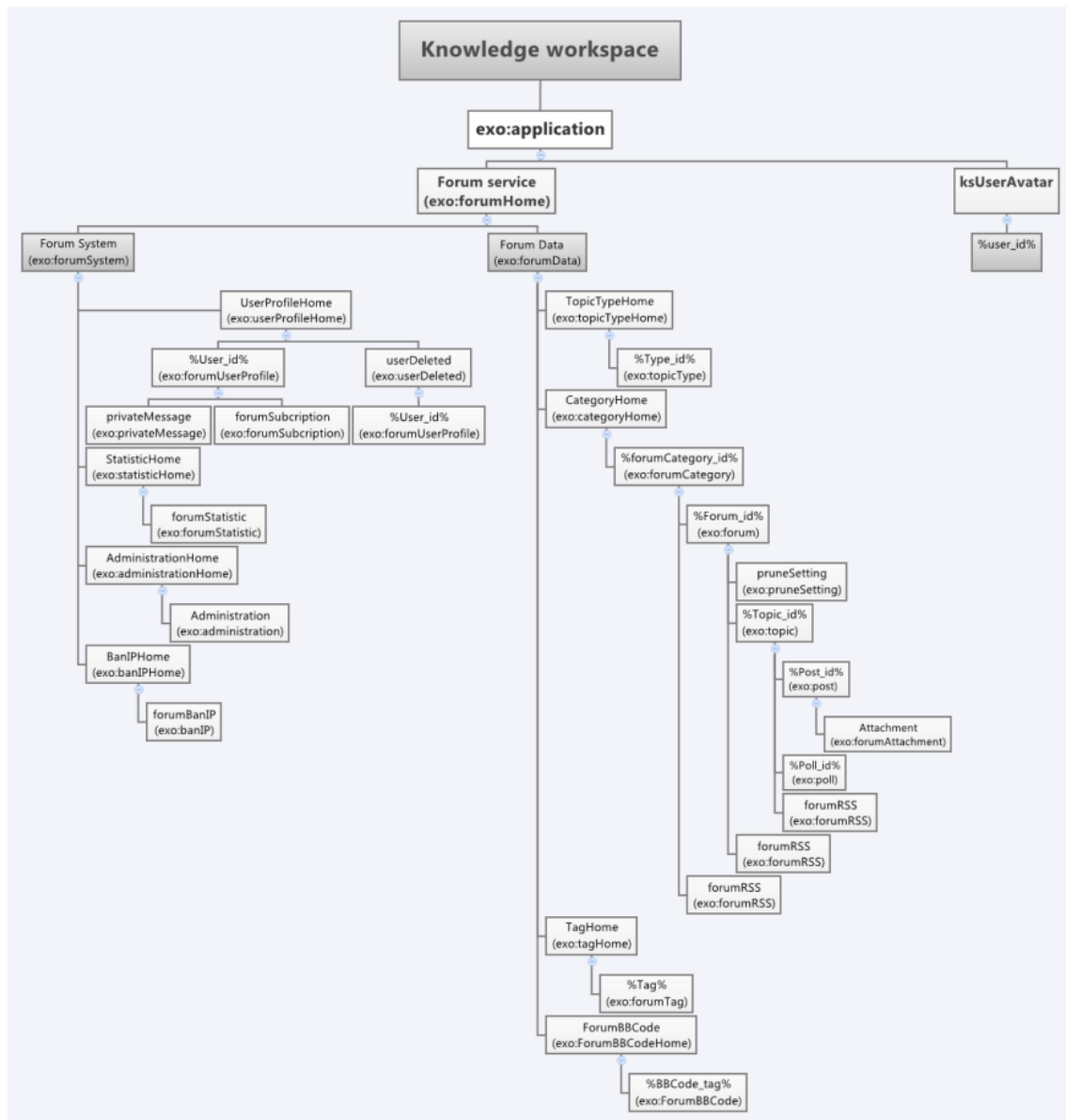
See also

- [Extension points](#)
- [Internal API](#)
- [Wiki Service API](#)

- [FAQ Template Configuration](#)
- [Actions over a wiki page from external jars](#)

3.6.1. Forum JCR structure

Forum is a JCR-based application. The Forum data are saved in eXo-JCR under the Forum Service data directory. The whole JCR structure of Forum can be visualized in the diagram below:



3.6.1.1. Forum System

The Forum System node is created from the node type `exo:forumSystem`. That is defined as a child node of Forum Service and can store nodes with these following node types: `exo:banIP`, `exo:forumUserProfile`, `exo:statistic`, `exo:administration` under the Forum System. The Forum System node is stored in `/exo:applications/ForumService/ForumSystem`.

3.6.1.1.1. User Profile and User Profile Home

The User Profile and User Profile Home node are used to store information of each user. User Profile is automatically created by a listener when a user registers to the organization service. Private message and forum subscription can be added to User Profile as a child node. These node types `exo:forumUserProfile`, `exo:userProfileHome`, `exo:privateMessage` and `exo:forumSubscription` are defined as child nodes of `exo:forumUserProfile`. The User Profile node

is stored under ForumSystem node: `/exo:applications/ForumService/ForumSystem/exo:userProfileHome/exo:forumUserProfile`.

- The node type `exo:forumUserProfile` has the following properties:

Properties name	Required type	Multiple	Description
<code>exo:userId</code>	String	false	The user id.
<code>exo:fullName</code>	String	false	The user's full name.
<code>exo:firstName</code>	String	false	The user's first name.
<code>exo:lastName</code>	String	false	The user's last name.
<code>exo:email</code>	String	false	The user's email address.
<code>exo:userTitle</code>	String	false	The user's title: Administrator, Moderator or User.
<code>exo:screenName</code>	String	false	The displayed name of user in Forum.
<code>exo:userRole</code>	Long	false	The user's role. The value can be: "0": Administrator, "1": Moderator, "2": User, "3": guest.
<code>exo:signature</code>	String	false	The signature displayed at the end of each user's post.
<code>exo:totalPost</code>	Long	false	The total posts submitted by the user.
<code>exo:totalTopic</code>	Long	false	The total topics started by the user.
<code>exo:jobWaitingForModerator</code>	Long	false	The number of jobs that are waiting to be moderated.
<code>exo:moderateForums</code>	String	true	The list of forum ids that user is the moderator.
<code>exo:moderateCategory</code>	String	true	The list of category ids that user is the moderator.
<code>exo:readTopic</code>	String	true	The list of topics that user has read.
<code>exo:readForum</code>	String	true	The list of forums that user has read.
<code>exo:lastReadPostOfTopic</code>	String	true	The list of the last read posts id in a topic that user has read.
<code>exo:lastReadPostOfForum</code>	String	true	The list of the last read posts id in a forum that user has read.
<code>exo:isAutoWatchMyTopics</code>	Boolean	false	Enable/disable the auto-watch the topics created by user. Topics created by a user will be watched

Properties name	Required type	Multiple	Description
exo:autoWatchTopicPost	Boolean	false	automatically if the value is set to "true".
exo:isAutoWatchTopicPost	Boolean	false	Enable/disable the auto-watch posts submitted by user. Topics posted by an user will be watched automatically if the value is set to "true".
exo:bookmark	String	true	The list of topics/posts bookmarked by user.
exo:lastLoginDate	Date	false	The date of the last login.
exo:joinedDate	Date	false	The date when user joined forum.
exo:lastPostDate	Date	false	The date of the last post.
exo:isDisplaySignature	Boolean	false	User's signature will be displayed at the end of their post if the value is set to "true".
exo:isDisplayAvatar	Boolean	false	User's avatar is displayed if the value is set to "true".
exo:newMessage	Long	false	The number of new messages.
exo:timeZone	Double	false	The time zone configured by user.
exo:timeFormat	String	false	The time format configured by user: 12h or 24h format.
exo:shortDateFormat	String	false	The format of short date configured by user. Example: 'dd/MM/yyyy'.
exo:longDateFormat	String	false	The format of long date configured by user. Example: 'dd mmm, yyyy'.
exo:maxPost	Long	false	The number of the maximum posts displayed per page.
exo:maxTopic	Long	false	The number of the maximum topics displayed per page
exo:isShowForumJump	Boolean	false	Display/hide the forum jump drop-down list. This jump list will be shown if the value is set to "true".
exo:collapCategories	String	true	The list of categories collapsed by user.
exo:isBanned	Boolean	false	The user's condition. User is currently banned if the value is set to "true".

Properties name	Required type	Multiple	Description
exo:banUntil	Long	false	The time when the ban period expires.
exo:banReason	String	false	The description for the reason that user was banned.
exo:banCounter	String	false	The number of bans that user has committed.
exo:banReasonSummary	String	true	The list of ban reason summaries when a user is banned for more than one time.
exo:createdDateBan	Date	false	The date when the ban period starts.

- The child node type `exo:privateMessage` has the following properties:

Properties name	Required type	Multiple	Description
exo:from	String	false	The user id of the sender.
exo:sendTo	String	false	The user id of the receiver.
exo:name	String	false	The private message subject.
exo:message	String	false	The message contents.
exo:type	String	false	The private message type: sent messages or received messages.
exo:receivedDate	Date	false	The date when the private message was received.
exo:isUnread	Boolean	false	The status of private message: read/unread.

- The child node type `exo:forumSubscription` has the following properties:

Properties name	Required type	Multiple	Description
exo:categoryIds	String	true	The ids of the subscribed categories.
exo:forumIds	String	true	The ids of the subscribed forums.
exo:topicIds	String	true	The ids of the subscribed topics.

3.6.1.1.2. Statistic and Statistic Home

Statistic and Statistic Home are used to store statistic information of forum, such as number of posts, topics, users, active users. The node type is `exo:forumStatistic`, `exo:statisticHome`.

- The Statistic node is stored under the Forum System node: `/exo:applications/ForumService/ForumSystem/exo:statisticHome/exo:forumStatistic` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:postCount	Long	false	The total number of submitted posts in Forum.
exo:topicCount	Long	false	The number of total created topics in Forum.
exo:membersCount	Long	false	The number of the registered users.
exo:newMembers	String	false	The id of the latest registered user.
exo:mostUsersOnline	String	false	The highest number of the online users.
exo:activeUsers	Long	false	The number of active users.

3.6.1.1.3. Ban IP and Ban IP Home

The Ban IP and Ban IP Home node are used to store data about banned IP addresses. The node type `exo:banIPHome` contains the child node `exo:IPHome`.

- The Ban IP node is stored under the Forum System node: `/exo:applications/ForumService/ForumSystem/exo:banIPHome/exo:banIP` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:ips	String	true	The list of ip addresses of the banned users.

3.6.1.1.4. Administration and Administration Home

Administration and Administration Home are used to store data for setting the layout, notification email format and censor jobs. The node type of the Administration Home node is `exo:administrationHome` and the its child node type is `exo:administration`. The Administration node is stored under the ForumSystem node `/exo:applications/ForumService/ForumSystem/exo:administrationHome/exo:administration` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:forumSortBy	String	false	Sort forum by criteria: post count, topic count, lock status.
exo:forumSortByType	String	false	Sort forum by ascending/descending type.
exo:topicSortBy	String	false	Sort topic by criteria.
exo:topicSortByType	String	false	Sort topic by ascending type or descending type.
exo:censoredKeyword	String	false	The list of censored words.
exo:headerSubject	String	false	The subject header.
exo:enableHeaderSubject	Boolean	false	Enable/disable the subject header. The subject header is displayed if the value is set to "true".
exo:notifyEmailContent	String	false	

Properties name	Required type	Multiple	Description
exo:notifyEmailMoved	String	false	Define if the notification email will be sent when there is a new added topic/post. Define if the notification email will be sent when there are any moved topic/post.

3.6.1.2. Forum Data

The Forum Data node is created from the node type `exo:forumData`. The data nodes like category, forum, topic, post, tag, BBcode and topic type will be stored under the Forum Data node: `/exo:applications/ForumService/ForumData`.

3.6.1.2.1. Category and Category home

The Category node is used to store all categories of forum, this node is a child node of the Forum Data node and only Category node type can be added to the Category Home. The node type of the Category Home node is `exo:categoryHome` is stored in `/exo:applications/ForumService/ForumData/CategoryHome`. The Category node has the node type `exo:forumCategory` which is a child node of the CategoryHome node. This node type is defined to allow adding child nodes as `exo:forum` and `exo:forumRSS`.

- The node type `exo:forumCategory` has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The category id.
exo:owner	String	false	The category creator.
exo:path	String	false	The node path of the category.
exo:createdDate	Date	false	The date when the category was created.
exo:modifiedBy	String	false	The id of the user who made the last modification on the category.
exo:name	String	false	The category name.
exo:modifiedDate	Date	false	The date when the modifications on category were made.
exo:description	String	false	The category description.
exo:moderators	String	true	The list of moderators of the category.
exo:tempModerators	String	true	The temporary moderator of the category.
exo:createTopicRole	String	true	The topic role.
exo:poster	String	true	The list of ids of the users and groups who can post in the category.
exo:viewer	String	true	The list of ids of the users and groups who can only view posts in the category.

Properties name	Required type	Multiple	Description
exo:categoryOrder	Long	false	The order number of category in the category list.
exo:userPrivate	String	true	The list of user ids whose access are restricted from the category.
exo:forumCount	Long	false	The total number of forums in the category.

3.6.1.2.2. Forum

The Forum node is defined as a child node of category and allowed adding child nodes as Topic and RSS type. The node type of Forum is `exo:forum`. The Forum node is stored in `/exo:applications/ForumService/ForumData/CategoryHome/%Category-id%/Forum-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The forum id.
exo:owner	String	false	The forum creator.
exo:path	String	false	The node path of the forum.
exo:name	String	false	The forum title.
exo:forumOrder	Integer	false	The order number in the list of forums. Forum with smaller number will get higher order.
exo:createdDate	Date	false	The date and time when the forum was created.
exo:modifiedBy	String	false	The id of user who modified the category.
exo:modifiedDate	Date	false	The time of modification, including date, time and time zone.
exo:lastTopicPath	String	false	The id of the last topic in the forum.
exo:description	String	false	The description of forum.
exo:postCount	Long	false	The total number of submitted posts in the forum.
exo:topicCount	Long	false	The total number of created topics in the forum.
exo:isAutoAddEmailNotify	Boolean	false	Enable/disable the notification email to moderators.
exo:notifyWhenAddTopic	String	true	Email addresses to notify when there is a new topic in the forum.
exo:notifyWhenAddPost	String	true	Email addresses to notify when there is a new post in the forum.

Properties name	Required type	Multiple	Description
exo:isModerateTopic	Boolean	false	All new topic will be moderated if the value is set to "true".
exo:isModeratePost	Boolean	false	All new posts will be moderated if the value is set to "true".
exo:isClosed	Boolean	false	The forum status: closed/open. Forum is closed if the value is set to "true".
exo:isLock	Boolean	false	The forum status: locked/unlocked. Forum is locked if the value is set to "true".
exo:createTopicRole	String	true	The list of ids of the users or groups who can create topic in the forum.
exo:poster	String	true	The list of ids of the users or groups who can submit post in the forum.
exo:viewer	String	true	The list ids of the users or groups who can view posts in the forum.
exo:moderators	String	true	The list of user ids who are the moderators of forum.
exo:tempModerators	String	true	The list of user ids who are the temporary moderators.
exo:banIPs	String	true	The list of banned IP addresses.

- The child node pruneSetting has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	False	The forum id.
exo:inActiveDay	Long	False	The number of days/weeks/months that the topics in forum have not been active.
exo:periodTime	Long	False	The number of days/weeks/months that the prune job will be executed to check for the old topics and deactivate them.
exo:isActive	Boolean	False	The current status of the prune job. If the value is set to "True", the prune job will be run.
exo:lastRunDate	Date	False	The date that prune job runs for the last time.

3.6.1.2.3. Topic

The Topic node is defined as a child node of the Forum node and allowed adding child nodes as Topic, Poll and RSS type. The node type of the Topic and Poll node is `exo:topic`, `exo:poll`.

- The Topic node is stored in `/exo:applications/ForumService/ForumData/CategoryHome/%Category-id%/Forum-id%/Topic-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
<code>exo:id</code>	String	false	The topic id.
<code>exo:owner</code>	String	false	The user id of the topic creator.
<code>exo:path</code>	String	false	The node path of the topic.
<code>exo:name</code>	String	false	The subject of the topic
<code>exo:createdDate</code>	Date	false	The time when the topic was created.
<code>exo:modifiedBy</code>	String	false	The id of the user who made the latest modification in the topic.
<code>exo:modifiedDate</code>	Date	false	The date when the modifications were made.
<code>exo:lastPostBy</code>	String	false	The user id of the last poster in topic.
<code>exo:lastPostDate</code>	Date	false	The date when the last post was submitted.
<code>exo:description</code>	String	false	The topic description.
<code>exo:topicType</code>	String	false	The id of the topic type.
<code>exo:postCount</code>	Long	false	The number of posts in the topic.
<code>exo:viewCount</code>	Long	false	The number of topic views.
<code>exo:numberAttachments</code>	Long	false	The number of attachments in the topic.
<code>exo:icon</code>	String	false	The name of the topic icon.
<code>exo:link</code>	String	false	The link to the topic. Example: <code>http://localhost:8080/ksdemo/public/classic/forum/topic/%Topic-id%</code> .
<code>exo:isModeratePost</code>	Boolean	false	All posts in the topic will have to wait for moderation if the value is set to "true".
<code>exo:isNotifyWhenAddPost</code>	Boolean	false	When there is a new post in a topic, a notification message will be sent to the topic owner if this value is set to "true".
<code>exo:isClosed</code>	Boolean	false	The state of the topic: closed/open. If the value is

Properties name	Required type	Multiple	Description
exo:isLock	Boolean	false	set to "true", the topic is closed. The lock status of the topic: lock/unlocked. If the value is set to "true", the topic is locked.
exo:isApproved	Boolean	false	The topic is approved to be published if the value is set to "true".
exo:isSticky	Boolean	false	If the value is set to "true", the topic is currently sticky.
exo:isWaiting	boolean	false	The topic status. The topic is waiting for moderation if the value is set to "true".
exo:isActive	boolean	false	The topic activity status: active/inactive. The topic is active (topic gets new posts in a period of time) if the status is set to "true".
exo:isActiveByForum	Boolean	false	The topic status based on the forum state. Example: When the topic is active and the forum that contains it is closed, this topic will be considered as inactive.
exo:canView	String	true	List of user ids who can view the topic contents.
exo:canPost	String	true	List of user ids who can post in the topic.
exo:isPoll	Boolean	false	The topic contains poll if the value is set to "true".
exo:userVoteRating	String	true	The list of user id who voted.
exo:tagId	String	true	The list of the topic tag id.
exo:voteRating	Double	false	The average vote score of the topic.

- The child node type `exo:poll` has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The poll id.
exo:owner	String	false	The user id of poll creator.
exo:createdDate	Date	false	The date and time when the poll was created.
exo:modifiedBy	String	false	The user id who modified the poll.
exo:modifiedDate	Date	false	

Properties name	Required type	Multiple	Description
			The time when the poll is modified.
exo:lastVote	Date	false	The date of the last vote.
exo:question	String	false	The contents of the question for poll.
exo:timeOut	Long	false	The time when the polled is closed.
exo:option	String	true	The list of options for poll.
exo:vote	String	true	The list of votes by users.
exo:userVote	String	true	The list of user ids who voted.
exo:isMultiCheck	Boolean	false	User can choose more than one option if the value is set to "true".
exo:isAgainVote	Boolean	false	Users can change their vote if the value is set to "true".
exo:isClosed	Boolean	false	The poll status. Poll is closed if the value is set to "true".

3.6.1.2.4. Post

The Post node is defined as child node of Topic and allowed adding only the Attachment child node type. The Post node type is `exo:post`, and the child node type is `exo:forumAttachment`.

- The Post node is stored in `/exo:applications/ForumService/ForumData/CategoryHome/%Category-id%/Forum-id%/Topic-id%/Post-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The post id.
exo:owner	String	false	The user id of the poster.
exo:path	String	false	The node path of the post.
exo:createdDate	Date	false	The date time when post is submitted, including date, time, time zone.
exo:modifiedBy	String	false	The id of the user who modified the post.
exo:editReason	String	false	The reason for editing the post.
exo:modifiedDate	Date	false	The date when the post was modified.
exo:name	String	false	The post title.
exo:message	String	false	The message of the post.
exo:remoteAddr	String	false	The remote IP address of the post.
exo:icon	String	false	The name of the icon for the post.

Properties name	Required type	Multiple	Description
exo:userPrivate	String	true	The list of user ids that are restricted from the post.
exo:link	String	false	The link to open the topic.
exo:isApproved	Boolean	false	The state of the post: approved/unapproved. The post is approved if the value is set to "true".
exo:numberAttach	Long	false	The number of attachments in the post.
exo:isActiveByTopic	Boolean	false	The post is activity status based on the topic state. If the topic is close, all post in it will be considered as inactive.
exo:isHidden	Boolean	false	The post status: shown/hidden. The post is hidden if the value is set to "true".
exo:isFirstPost	Boolean	false	The post is the first one in a topic if the value is set to "true".

3.6.1.2.5. Tag and Tag home

The Tag node is used to store data about tag name, topics with tag added, number of users using this tag, number of tags in use. The node type of the Tag node is `exo:forumTag` and its child node type is `"exo:tagHome"`. The Tag node is stored in `/exo:applications/ForumService/ForumData/TagHome/%tag-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The id of tag.
exo:name	String	false	The tag name.
exo:useCount	Long	false	The number of times that the tag was used.
exo:userTag	String	true	The number of users using the tag.

3.6.1.2.6. BBCode and BBCode home

The BBCode node is used to define what BBCode will be used in the forum. The node type of the BBCode node is `exo:forumBBCode`. The BBCode node is stored in `/exo:applications/ForumService/ForumData/forumBBCode/%BBCode_tag%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:description	String	false	The description about the tag purpose. Example: 'The [url] tag allows creating links to other websites and files'.
exo:example	String	false	

Properties name	Required type	Multiple	Description
			The example about using the tag. Example: [URL]http://www.exoplatform.com/[URL]'. The BBCode tag is active/deactive. The BBCode tag is active if the value is set to "true".
exo:isActive	Boolean	false	
exo:isOption	Boolean	false	If the value is set to "true", user can create a tag with attributes and values.
exo:replacement	String	false	The HTML code that will be replaced by the tag. Example: [url] tag replaces ' <a >{param}<="" a>'.<br="" href="{param}" target="_blank"> The BBCode tag name.
exo:tagName	String	false	

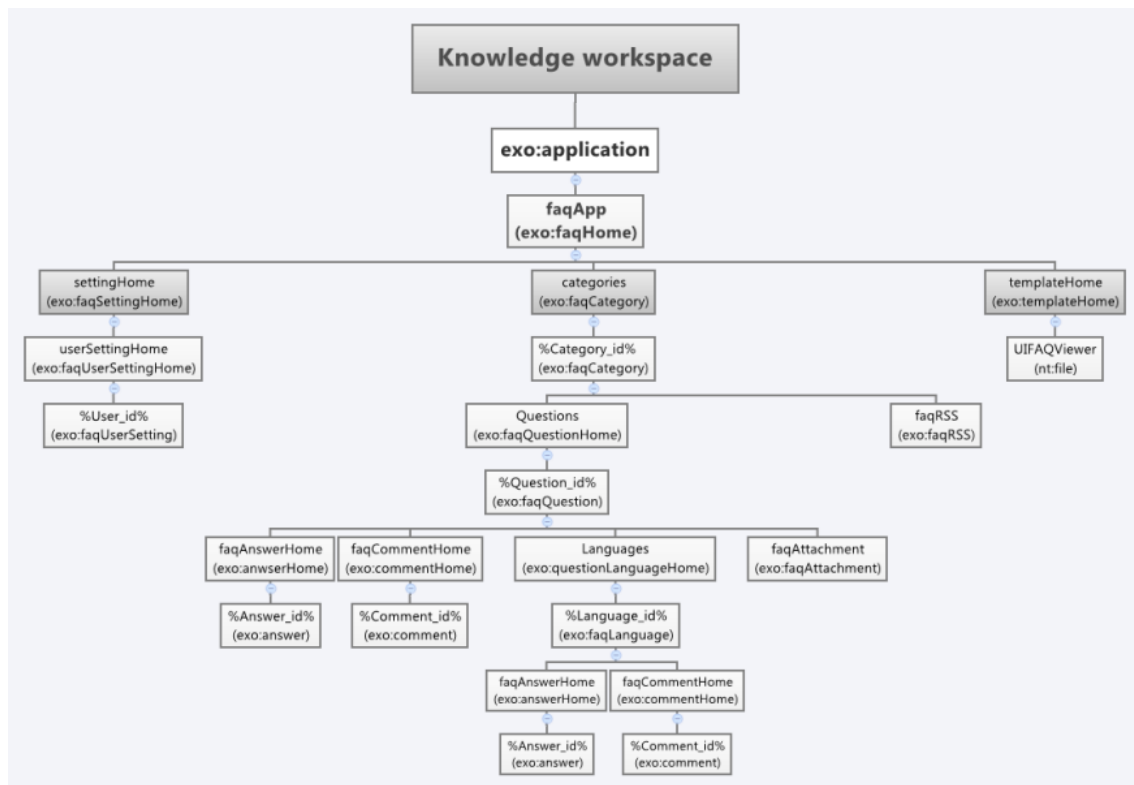
3.6.1.2.7. Topic type and Topic type home

The Topic type home contains a child node with the node type `exo:topicType`. The Topic node is stored in `/exo:applications/ForumService/ForumData/TopicTypeHome/%Topic-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The id of the topic type.
exo:name	String	false	The name of the topic type.
exo:icon	String	false	The icon of the topic type.

3.6.2. FAQ JCR structure

FAQ is a JCR-based application. The FAQ data are stored in the eXo-JCR under the `faqApp` data directory. The whole FAQ JCR structure can be visualized in the following diagram:



3.6.2.1. Category

3.6.2.1.1. Sub-category

The system will automatically create the Category Home node under the FAQ application node at the first time the user launches application. All users-created categories are the sub-categories of Category Home. The home of the Category node is automatically created in `/exo:applications/faqApp/categories`.

In fact, Sub-category is also a category. FAQ has defined a mixin node type called `mix:faqSubCategory` to allow adding a node having the same type with category to an existing category. When a category is created, this mixin node type will be mixed to that category.

The node type `exo:faqCategory` has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The sub-category id.
exo:name	String	false	The name of the sub-category.
exo:userPrivate	String	true	The list of user ids that are restricted from the category.
exo:description	String	false	The description of the sub-category.
exo:isModerateQuestions	Boolean	false	The question post moderation status. All questions posted in the sub-category will have wait for moderation if the value is set to "true".
exo:isModerateAnswers	Boolean	false	

Properties name	Required type	Multiple	Description
			The answer post moderation status. All answers posted in the sub-category will have to wait for moderation if the value is set to "true".
exo:isView	Boolean	false	The category is shown/hidden. The category will be shown if the value is set to "true".
exo:viewAuthorInfor	Boolean	false	The category enables user to view the information of questions poster if the value is set to "true".
exo:moderators	String	true	The list of user ids who are the category moderator.
exo:createdDate	Date	false	The time when the sub-category is created.
exo:index	Long	false	The index number of the category.

3.6.2.1.2. RSS

Each category has a RSS child node that stores a RSS feed representing all questions in this category as the binary data type. The RSS node is stored in `/exo:applications/faqApp/categories/ks.rss` and its node type is `exo:faqRSS`.

Properties name	Required type	Multiple	Description
exo:content	Binary	false	The content of the RSS.

3.6.2.1.3. Question and Question Home

The Question Home node is created from the `exo:faqQuestionHome` node type that is defined as a child node of category. This node contains all question nodes that created in side a category. Only the Question node type `exo:faqQuestion` can be added to the question Home. The Question Home node is created as a child node of Categories `/exo:applications/faqApp/categories/questions`.

Question node is created from `exo:faqQuestion` node type under the Question Home node. The Answers, Comments and Attachments node are defined as child nodes of the Question node. The Question node is created under the Question Home: `/exo:applications/faqApp/categories/questions/%Question-id%`.

- The `exo:faqQuestion` node type has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The question id.
exo:language	String	false	The language of the question.
exo:name	String	false	The question details.
exo:title	String	false	The question title.
exo:author	String	false	The the user id of the question poster.

Properties name	Required type	Multiple	Description
exo:email	String	false	The email of the question author
exo:isActivated	Boolean	false	The question status: activated/inactivated. The question is activated if the value is set to "true".
exo:isApproved	Boolean	false	The state of the question: approved/unapproved. The question is approved to be published if the value is set to "true".
exo:categoryId	String	false	The id of the category containing the question.
exo:createdDate	Date	false	The date and time when the question was submitted.
exo:relatives	String	true	The list of the related questions ids.
exo:usersVote	String	true	The list of user ids who voted.
exo:markVote	Double	false	The average vote scores of the question.
exo:topicIdDiscuss	String	false	The topic id in the forum where the question is discussed.
exo:nameAttachs	String	true	The file name of attachments in the question.
exo:lastActivity	String	false	The user id and time when the last activity of the question was made.
exo:numberOfPublicAnswers	Long	false	The number of all posted answers that has been published.
exo:link	String	false	The link to open the question.
exo:responses	String	true	The responses of the question.
exo:dateResponse	Date	true	The date when the question received the answer.
exo:responseBy	String	true	The user id of the answer poster.

3.6.2.1.4. Multilanguages

A question can support multilanguages, all other languages are stored as a child node of the question and can be add to the question via a mixin node type called `mix:faq18n`. After the mixin node type `mix:faq18n` is added to the question, the node type `exo:questionLanguageHome` can be added to the question node and this node type will contain all languages node with the node type `exo:faqLanguage`. All display properties of the question are defined in the node type `exo:faqLanguage`.

- The node type `exo:faqLanguage` has the following properties:

Properties name	Required type	Multiple	Description
<code>exo:language</code>	String	false	The language of the question.
<code>exo:name</code>	String	false	The name of the language.
<code>exo:title</code>	String	false	The title of the question in the selected language.
<code>exo:questionId</code>	String	false	The id of the question.
<code>exo:categoryId</code>	String	false	The id of the category.

3.6.2.1.5. Answer, Comment and Attachment

The Answer, Comment and Attachment node is defined as the child nodes of the Question node. Attachment node is defined as a `nt:file` node type and stored right under the Question node. Answers and comments node are stored under the Answer home and the Comment home node.

- The Answer node is stored in `/exo:applications/faqApp/categories/questions/%Question-id%/faqAnswerHome/%Answer-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
<code>exo:id</code>	String	false	The id of the answer.
<code>exo:answerPath</code>	String	false	The path to the answer.
<code>exo:questionId</code>	String	false	The id of the question.
<code>exo:categoryId</code>	String	false	The id of the category containing the question.
<code>exo:responses</code>	String	false	The content of the answer.
<code>exo:dateResponse</code>	Date	false	The date when the response was posted.
<code>exo:responseBy</code>	String	false	The id of the user who responded the answer.
<code>exo:responseLanguage</code>	String	false	The language of the answer response.
<code>exo:approveResponses</code>	Boolean	false	The response is pending for approval if the value is set to "false".
<code>exo:activateResponses</code>	Boolean	false	The state of the answer: activated/deactivated .
<code>exo:usersVoteAnswer</code>	String	true	The list of user ids who voted for the answer.
<code>exo:MarkVotes</code>	Long	false	The average vote scores of the answer.
<code>exo:postId</code>	String	false	The post id.
<code>exo:fullName</code>	String	false	The answer author's full name.

- The Comment node is stored in `/exo:applications/faqApp/categories/questions/%Question-id%/faqCommentHome/%Comment-id%` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The comment id.
exo:comments	String	false	The comment contents.
exo:dateComment	Date	false	The date when the comment is posted.
exo:commentBy	String	false	The user id of the comment poster.
exo:postId	String	false	The id of the post.
exo:fullName	String	false	The full name of the comment poster.
exo:categoryId	String	false	The id of the category in which the comment is posted.
exo:questionId	String	false	The id of the question in which the comment is posted.
exo:commentLanguage	String	false	The language of the comment.

- The Attachment node is stored in `/exo:applications/faqApp/categories/questions/%Question-id%/faqAttachment` and its node type has the following properties:

Properties name	Required type	Multiple	Description
exo:fileName	String	false	The name of the attachment file.

3.6.2.2. FAQ setting

This FAQ Setting node stores the user settings data, such as how answer is ordered (in alphabetical order or by created date), the order type (descending or ascending) or the user's selection to sort questions by popularity. Each user has a dedicated settings data to select the display preferences in FAQ . The default setting will be used if the users has never changed and saved their setting.

- The User Setting node of an individual user is stored in `/exo:applications/faqApp/settingHome/userSettingHome/%user-id%` and has the following properties:

Properties name	Required type	Multiple	Description
exo:ordeBy	string	false	Define how questions are ordered, by "alphabet/index" or "created date".
exo:ordeType	string	false	The value "asc" = ascending and "des" = descending.
exo:sortQuestionByVote	Boolean	false	All questions will be sorted by the popularity (based on the number of votes) if the value is set to "true".

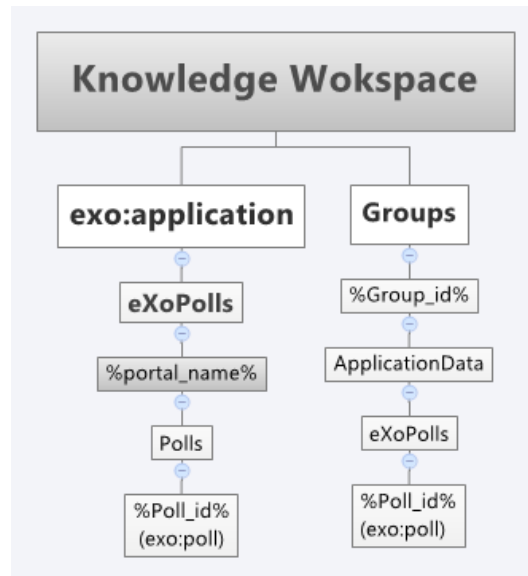
3.6.2.3. Template for FAQ

This node stores the template for FAQ portlet. The user can edit this template online in FAQ to change the layout, skins, and more.

- The template is stored in an nt:file node type under the Template Home node. `/exo:applications/faqApp/templateHome/nt:file`.

3.6.3. Poll JCR structure

The Poll data are saved in eXo-JCR under the eXoPolls data directory. The whole JCR structure of Poll can be visualized in the diagram below:



The Poll node is used to store the default data in a poll. The node type of the Poll node is `exo:poll`. The Poll node is stored under eXoPolls node `/exo:applications/eXoPolls/%PortalName%/Polls/Poll-id%` and its node type (`exo:polls`) has the following properties:

Properties name	Required type	Multiple	Description
exo:id	String	false	The poll id.
exo:owner	String	false	The user id of the poll creator.
exo:createdDate	Date	false	The date and time when the poll is created.
exo:modifiedBy	String	false	The id of the user who made the last modification on the poll.
exo:modifiedDate	Date	false	The date and time when the latest modification on poll was made.
exo:lastVote	Date	false	The date and time when the last vote was made.
exo:question	String	false	The question content of poll.
exo:timeOut	Long	false	The time when the poll will be closed.
exo:option	String	true	The list of options for poll. Each option is separated by a comma.
exo:vote	String	true	The list of votes by users.

Properties name	Required type	Multiple	Description
exo:userVote	String	true	The list of user ids who voted.
exo:isMultiCheck	Boolean	false	Enable/disable the multi check. User can vote for more than one option if the value is set to "true".
exo:isAgainVote	Boolean	false	Enable/disable the option to vote again. User can change their vote if the value is set to "true".
exo:isClosed	Boolean	false	The poll status: open/closed. The poll is closed if the value is set to "true".

3.6.4. Wiki JCR structure

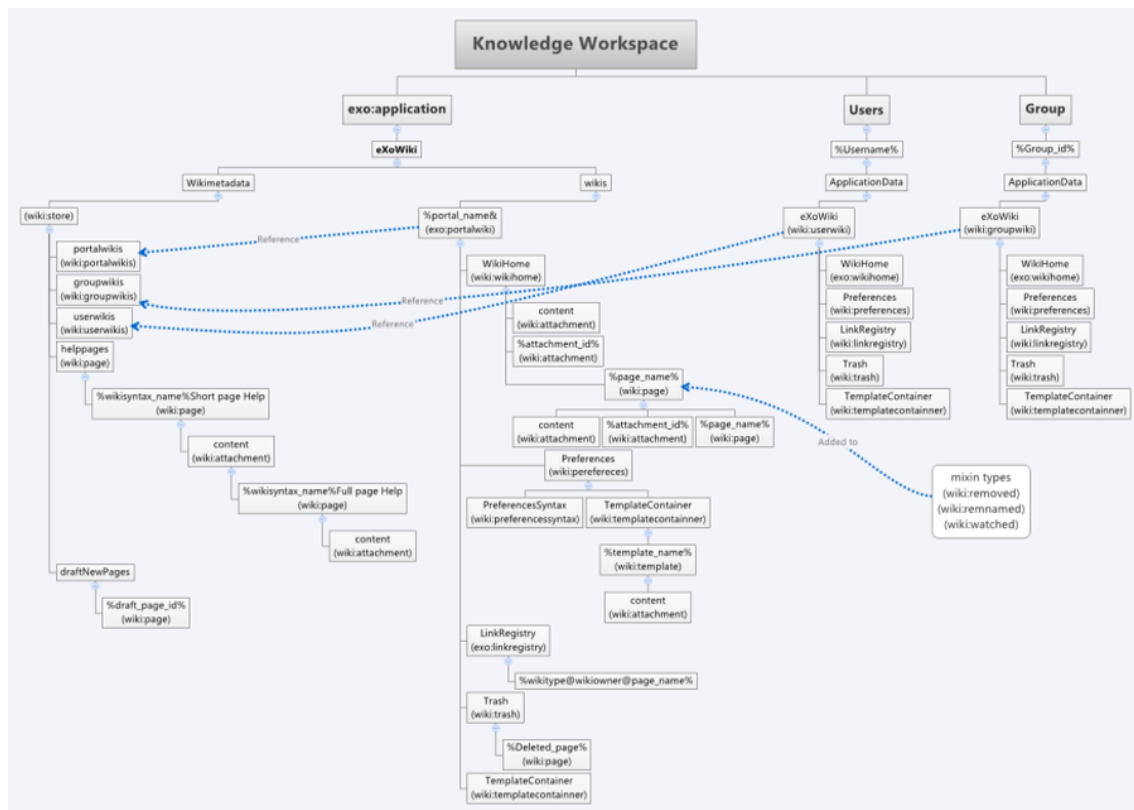
The Wiki portlet is a JCR-based application. The Wiki data are stored in the eXo JCR storage. The Wiki application of Knowledge supports several wikis, each wiki is organized as a tree of pages and hosted in different locations, depending on the type of wiki:

- Portal wikis: `/exo:applications/eXoWiki/wikis/$PORTAL/WikiHome`
- Group wikis: `/Groups/$GROUP/ApplicationData/eXoWiki/WikiHome`
- User wikis: `/Users/$USERNAME/ApplicationData/eXoWiki/WikiHome`

WikiHome a conventional name of the root page of a Wiki. The type of *WikiHome* is "exo:wikihome". Each page may have a number of sub-pages and attachments.

Other Wiki metadata are organized below `/exo:applications/eXoWiki/wikimetadata`.

The whole Wiki JCR structure can be visualized in the following diagram:



3.6.4.1. Wiki data

Depending on the Wiki type (portal, group or user); its pages, attachments and default syntax are stored under a node with either of node types: "wiki:portalwiki", "wiki:groupwiki" or "wiki:userwiki", respectively. Such a node has four child nodes: WikiHome, Preferences, LinkRegistry and Trash. These node types have the super type "wiki:wiki" that has the following properties:

Properties name	Required type	Multiple	Description
ref	Reference	false	The reference to one of three following nodes: portalwikis, groupwikis or userwikis.
owner	String	false	The name of the wiki.
wikiPermissions	String	true	The property consists of the permission information of the wiki. The permission string has the format: VIEWPAGE,EDITPAGE,ADMINPAGE,ADMINPAGE:GROUP:/platform/users; VIEWPAGE,EDITPAGE,ADMINPAGE,ADMINPAGE:GROUP:/platform/administrators.
isDefaultPermissionsInited	Boolean	true	Check whether the default permission is applied to all the wiki tree or not. Its default value is "false".

WikiHome

The WikiHome node stores the root page of a Wiki. It has the node type "exo:wikihome" that has the super type "wiki:page" inherited from "nt:folder". The super type has the following properties:

Properties name	Required type	Multiple	Description
owner	String	false	The creator of the page.
author	String	false	The last person who modifies the page.
createdDate	Date	false	The date when the page is created.
updatedAt	Date	false	The last date when the page is updated.
syntax	String	false	The Wiki syntax is used to write the page.
title	String	false	The title of the page.
comment	String	false	The comment explains what is modified in the page.
url	String	false	The URL to the page.
isOverridePermission	Boolean	false	Check whether the default permission is overridden on the page or not. Its default value is "false".
exo:relation	Reference	true	The property consists of the UUIDs of the related pages.

The nodes that have the type "wiki:page", have a child node named "content" and other child nodes including attachments with the type "wiki:attachment" inherited from "nt:file". The node type "wiki:attachment" has the following properties:

Properties name	Required type	Multiple	Description
title	String	false	The name of the attachment.
fileType	String	false	The type of the attachment.
creator	String	false	The creator of the attachment.

- Three mixin node types: "wiki:removed", "wiki:renamed" and "wiki:watched" may be added to the node type "wiki:page".

The mixin node type "wiki:removed" has the following properties:

Properties name	Required type	Multiple	Description
removedBy	String	false	The person who deleted the page.
removedDate	Date	false	The date when the page is deleted.
parentPath	String	false	The path to the parent page of the deleted page.

The mixin node type "wiki:renamed" has the following properties:

Properties name	Required type	Multiple	Description
oldPageIds	String	true	The old Ids of the renamed page.

The mixin node type "wiki:watched" has the following properties:

Properties name	Required type	Multiple	Description
watcher	String	false	The Id of the person who is watching the page.

Preferences

The "Preferences" node stores the default syntax and page templates of the Wiki. It has the node type "wiki:preferences" and two child nodes "PreferencesSyntax" and "TemplateContainer".

- The "PreferencesSyntax" node has the node type "wiki:preferencesyntax" that has the following properties:

Properties name	Required type	Multiple	Description
defaultSyntax	String	false	The default Wiki syntax of each wiki.
allowMutipleSyntaxes	Boolean	false	Specify whether multiple syntaxes are enabled or not.

- The "TemplateContainer" node stores the page templates. Its child node has the node type "wiki:template" inherited from the node type "wiki:page". The node type "wiki:template" has the following properties:

Properties name	Required type	Multiple	Description
description	String	false	The description of the template.

LinkRegistry

The "LinkRegistry" node stores the entries to keep track of renaming or moving pages. Each link entry has the node type "wiki:linkentry" that has the following properties:

Properties name	Required type	Multiple	Description
alias	String	false	The Id of a page that is moved or renamed. Its format is "wikitype@wikiowner@pagename".
newlink	Path	false	A new path to the wiki page that has been moved or renamed.

Trash

The "Trash" node stores deleted pages.

Template Container

The "Template Container" node stores the templates to create Wiki pages.

3.6.4.2. Wiki metadata

Wiki metadata are stored in the node wikimetadatas that has the node type "wiki:store". These metadata consist of information about help contents, draft contents and the way to find the wikis in the system. The nodes portalwikis, groupwikis and userwikis allow finding the wikis under the type: portal, group and user, basing on the JCR reference features. These nodes have the node types "wiki:portalwikis", "wiki:groupwikis" and "wiki:userwikis" relatively.

Help pages

Help-related information about Wiki syntaxes are stored under the "helppages" node which has the node type "wiki:page". For syntax help, there are two help pages (summary and detailed) for each syntax. The summary page suggests the simple syntax, while the detailed page provides the full syntax-related information. The detailed page is arranged as the child page of the summary page, and both of which have the same node type "wiki:page".

Draft pages

The unsaved content of a new page is stored under the "draftNewPages" node that has the node type "wiki:page".