

GateIn: GateIn 3.3

Posted by [Julien Viet](#) Jun 1, 2012

GateIn 3.3 was released today.

A few months ago we decided to change our way to drive the GateIn project and provide more frequent releases. At the same time we migrated our source repository to Git on GitHub that gives us the opportunity to have a different approach for our workflow on the GateIn project.

Until now, we used to integrate all current features in the GateIn SVN trunk, as a consequence, the trunk source code was containing different features being integrated or developed in the trunk. Some were finished, some were still a work in progress and some in an inception status. With this very centralized approach, it was never really the time to make a release because the unfinished features were not ready for releasing.

After GateIn 3.2 we moved to Git and decided instead to work with feature branches and have in the master branch only a few features scheduled for the next release. Thanks to this approach, it is easier for us to deliver more frequently GateIn community releases because we integrate only what will be released in a nearly finished state.

GateIn 3.3 has less features than the previous releases but it was released only 3 months after GateIn 3.2! Let's review them:

AS7 update

todo.

SAML2 integration

todo.

JQuery

JQuery is today almost a standard by itself, we are glad that now GateIn is based on JQuery for providing its rich UI. As a consequence JQuery is provided out of the box with GateIn and you can use it directly in your portlet.

GateIn uses JQuery 1.7.1, however if you are willing to use another JQuery version you can bundle your own version:

```
<gatein-resources
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.gatein.org/xml/ns/gatein_resources_1_2_1 http://www.gatein.org/xml/
ns/gatein_resources_1_2_1"
  xmlns="http://www.gatein.org/xml/ns/gatein_resources_1_2_1">
  <module>
    <name>jquery-1.6.4</name>
    <script>
      <name>jquery-1.6.4</name>
      <path>/javascript/jquery-1.6.4.js</path>
    </script>
  </module>
  <portlet>
    <name>jQueryPortlet</name>
    <module>
      <script>
        <name>myscript</name>
        <path>/javascript/myscript.js</path>
      </script>
      <depends>
        <module>jquery-1.6.4</module>
      </depends>
    </module>
  </portlet>
</gatein-resources>
```

This `gatein-resources.xml` file must be located in the `WEB-INF` directory of your web application. It instructs Gateln to use JQuery 1.6.4 when the JQueryPortlet appears on a page, along with a custom `myscript.js` for the portlet that would depend on JQuery. A full example is available among the examples named `gatein-jquery.war`.

Javascript modules

Javascript serving is an important concern for modern web applications and specially for portals. A portal is an aggregation engine and provides different applications in the same page having for consequence to potentially have to deliver many javascript files on the same page which can severely affect performances. We have worked to provide a better modularization of Javascript and more specially define a real dependency system for Javascript resources. This dependency system allows us to know exactly which scripts should be served when a page is displayed based on a declarative configuration.

Module definitions

Javascripts resources are now bundled in *modules*, each module is managed by Gateln asset server and are declared in the `gatein-resources.xml` file. This resource descriptor already had a mechanism for achieving this result, but the dependency system was not good enough so we had to make it evolve to support more precise semantics.

A module bundles a set of Javascript files, those files are aggregated into a single resource when they are served by Gateln for more efficiency, for instance:

```
<module>
  <name>jquery</name>
  <script>
    <name>jquery</name>
    <path>/javascript/jquery-1.7.1.js</path>
  </script>
  <script>
    <name>jquery-alias</name>
    <path>/javascript/jquery-alias.js</path>
  </script>
</module>
```

This module bundles two scripts and is named *jquery*.

Module dependencies

Of course modules can define dependencies upon each other, this module declaration defines the *common* module having a dependency on the *base* module.

```
<module>
  <name>common</name>
  <script>
    <name>eXo.core.DragDrop</name>
    <path>/javascript/eXo/core/DragDrop.js</path>
  </script>
  <depends>
    <module>base</module>
  </depends>
</module>
```

It means that when the common module is served, the base module will be served and executed before the common module scripts.

Module scopes

Scopes are very important for modules, they define when they are served by GateIn, there are three modules scopes

- Shared scope : the shared scope is just like a library and it is not served by default (except the special bootstrap module that provide base services for loading other scripts)
- Portal scope : the module is attached to a specific portal entity when it is served
- Portlet scope : the module is attached to a specific portlet and is served when the portlet appears on the screen

Module internationalization

Last but not least, modules are internationalizable. It means that GateIn can filter the scripts and replace $\${}$ keys with values from a resource bundle. This allows us to avoid resource bundle loading from the client and provides better performances.

```
<module>
  <name>base</name>
```

```
<supported-locale>de</supported-locale>  
<supported-locale>en</supported-locale>  
<supported-locale>fr</supported-locale>  
<supported-locale>ru</supported-locale>  
<supported-locale>vi</supported-locale>  
...  
</module>
```

Javascript serving

The javascript server has been rewritten to use the module system. It has also been integrated with the famous *navigation controller* that was introduced in GateIn 3.2 . This enables GateIn to provide better urls for javascript, let's see how a module url look like:

/portal/scripts/3.3.0-CR01-SNAPSHOT/SHARED/base-fr-min.js

We can see in the url several parts:

- The current server version **3.3.0-CR01-SNAPSHOT**
- The module scope **SHARED** and name **base**
- The current user language **fr** (because I'm french). This language is optional and only appears if the module contains at least a supported locale definition.
- Finally the name **min** suffix is the minified version of the module

Those parts provide true cacheability of the module which is a key point in web performance today. Note that the server is fully dynamic and you can change a part in the url to see a different result. For instance if we remove the *min* suffix you will obtain the non minified version which is useful for debugging.

This serving is made easy thanks to the *navigation controller* system in GateIn. The navigation controller provides you a fine grained configuration of the server urls, it means also that if you don't like how GateIn module url are designed you can change it easily!

In the *controller.xml* file you can find a section for the resources:

```
<route path="/scripts/{gtn:version}/{gtn:scope}/">
  <route-param qname="gtn:handler">
    <value>script</value>
  </route-param>
  <path-param qname="gtn:version" encoding="preserve-path">
    <pattern>[^/]*</pattern>
  </path-param>
  ...
</route>
```

We can easily see how it works! Of course this is fully dynamic, if you change this file and reboot (or force a reload via JMX) urls will updated instantaneously.

3 Views

There are no comments on this post