Integration Community of Practice: Immutability: Managing environment-specific properties with Fuse/Fabric

Posted by Michael Thirion Apr 21, 2016

An IT infrastructure is always made of several environments, from "development" to "production"; the number and the names depending on the customer.
From an application perspective, those environments are defined by a set of properties that differs from one environment to the other. This could for instance be the URL address of a target system, or the listening port of the application itself.

To reduce the risk of introducing any kind of wrong implementation in production, an application should be treated as immutable from development to production.
This means that it shouldn't be possible to change the content of an application package without creating a new version of the application, starting back from the SCM.
Said in another way, nothing should change in the application code to adapt to the new environment properties when it's promoted from one environment to the next one.

This is something that can nicely be done within Fabric, using profile hierarchies.
The idea is to externalize the environment properties from the applications.
We can use a dedicated Fabric profile to store such properties.
The profile will only contain properties files (PID files), and we can use Maven "build profiles" to build it with property values corresponding to one specific environment.
This profile will then be configured as the parent profile of all the customer's applications running in the Fabric.
The applications would make use of the Property PlaceHolder to reference the PID properties but would never contain them.
As a result, the exact same application code can be deployed to any "Fabric environment" and simply inherits the properties reflecting the current environment configuration from a parent profile.

Here's an example that simulates this principle and that shows the details of the implementation:
GitHub - mthirion/fabric-env: Managing environment-specific properties with Fuse & Fabric

The parent profile must have a name that is well known and that doesn't change from one environment to the other.
This means that the name cannot mention any version.
However we still have to release this profile from time to time, and therefore be able to manage its version.
A nice way to do it is to rely on the Fabric versioning mechanism.
Each time the environment changes, a new Fabric is created (fabric:version-create) that virtually acts as a brand new environment; then the Karaf containers can be smoothly migrated to this new environment.
54 Views


Michael Thirion
May 5, 2016 1:26 PM
Indeed!
Well spotted! Thanks! Code is merged now!


Bruce McLachlan
May 4, 2016 12:06 PM
Update environment/pom.xml to change the maven phase for the maven-resources-plugin from compile to validate. If the plugin executes during compile phase the .properties file is only moved to the fabric8 folder after compilation, therefore the correct .properties file will not exist in the /target folder. If this the bundle is now deployed (fabric8:deploy) then the updated .properties file will not be present in the classpath and the fabric profile will not reflect the updated properties. Changing the phase to validate allows the .properties file to be updated before compilation, and the correct .properties file will be moved to the /target folder, and will therefore be available on the classpath when the bundle is deployed.
(I created a pull request on github for this change