# PrototypeBeanFactory

## Problem Description

July 6 2004, 1.0beta

The BeanFactory (http://jakarta.apache.org/hivemind/hivemind-lib/BeanFactoryBuilder.html) from the HiveMind library is able to produce plain java beans of a certain type. What it is missing is the possibility to use all the rich configuration options, that are available to services that use the BuilderFactory. This would close the missing gap to springs prototype beans and for example could be used for creating prototyped struts actions (sorry to all you tapestry folks) or command beans, that realize a command design pattern.

## Proposed Solution

Add a new PrototypeBeanFactory which uses a contribution schema that supports most of the options of BuilderFactory.

Example:

{{{service-point (id=TestPrototypeBeanFactory
interface=org.apache.hivemind.lib.PrototypeBeanFactory)
{
invoke-factory (service-id=hivemind.exlib.PrototypeBeanFactoryBuilder)
{
factory (vend-class=org.apache.hivemind.factory.PrototypeBean
configuration-id=hivemind.lib.test.SimpleBeanFactory)
}
}

configuration-point (id=SimpleBeanFactory
schema-id=hivemind.exlib.PrototypeBeanFactoryContribution)

contribution (configuration-id=SimpleBeanFactory)
{
bean (name=bean1 class=org.apache.hivemind.factory.PrototypeBean)
{
set(property=property value=testValue1)
set-service(property=service service-id=foo.businessService)
}
bean (name=bean2 class=org.apache.hivemind.factory.PrototypeBean
{
service { "foo.businessService" }
string { "testValue2" }
}
}
}}}

This example demonstrates the configuration of a factory which builds beans of the type `org.apache.hivemind.factory.PrototypeBean`. bean1 is configured by using setter methods, bean2 is configured by using constructor parameters.

Usage of the factory would be like this:

```
PrototypeBeanFactory factory = (PrototypeBeanFactory)
  r.getService("hivemind.lib.test.TestPrototypeBeanFactory",
  PrototypeBeanFactory.class);
PrototypeBean bean1 = (PrototypeBean) factory.get("bean1");
PrototypeBean bean2 = (PrototypeBean) factory.get("bean2");
```

## Discussion