

# SpringAspectContainer

## Overview

The **Spring Aspect Container** in AWARE will make it possible to manage your aspects using [Spring](#), it lets you treat the aspects just as any other Spring bean.

The container will read in a Spring configuration file called `aware-config.xml`, this file has to be on your classpath. In this file you can define your aspects, add parameters and data structures to them, pass in references to other components that is being used etc. This is for example documented briefly in the [Role-Based Security](#) section. For details see the [Spring documentation](#).

## Definition

To tell the [AspectWerkz](#) system that you want to deploy a specific aspect in the Spring aspect container (or any other custom aspect container) you have to specify that in the regular aspect definition file like this:

```
<aspect class="foo.bar.Baz"  
  container="org.codehaus.aware.container.SpringAspectContainer">  
  ...  
</aspect>
```

In the `aware-config.xml` file you have to configure the aspects like this:

```

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE beans PUBLIC "-//SPRING//DTD
BEAN//EN"
"http://www.springframework.org/dtd/spring-b
eans.dtd">

<beans>
    <bean id="my.own.Aspect"
        class="my.own.Aspect"
        singleton="false"
        init-method="intialize">
        <property name="someProperty">
            ...
        </property>
        ...
    </bean>
    ...
</beans>

```

Some explanations:

- `id` - specifies the name of the aspect if a custom name is define you that else use the class name of the aspect (which is the default name). Mandatory.
- `class` - specifies the class name of the aspect. Mandatory.
- `singleton` - specifies that the aspect will be instantiated using the prototype pattern, Mandatory and needed to be set to `false`
- `init-method` - the `init-method` is the method that you are using to initialize the aspect. This method will be called when all the properties have been set. Optional.
- `property` - the metadata that you want to pass to the aspect (see the [Spring](#) documentation for details on how how to define properties). Optional.

For an example on how to configure aspect using Spring see the [Role-Based Security](#) component.