How to serve webbapp A from portA and webapp B from portB

How do I serve webapp A only from port A and webapp B only from port B?

The most efficient way to do this is with two org.mortbay.jetty.Server instances. There is also a second, less efficient alternative.

Instance A will have a connector listening on port A with webapp A defined, and instance B will have webapp B and connector listening on port B defined.

Let's clarify that with an example.

Suppose we have webapp A that we want served from port 8080, and webapp B that we want served from an SSL connector on port 8443. We would set up 2 different Server instances, each in it's own jetty xml file like so:

jettyA.xml

```
<Configure id="ServerA" class="org.mortbay.jetty.Server">

    <!-- set up the port for ServerA -->
    <Set name="connectors">
      <Array type="org.mortbay.jetty.Connector">
        <Item>
          <New class="org.mortbay.jetty.nio.SelectChannelConnector">
            <Set name="port">8080</Set>
            <Set name="maxIdleTime">30000</Set>
            <Set name="Acceptors">10</Set>
          </New>
        </Item>
      </Array>
    </Set>

   <!-- set up a context deployer for Server A -->
    <Call name="addLifeCycle">
      <Arg>
        <New class="org.mortbay.jetty.deployer.ContextDeployer">
          <Set name="contexts"><Ref id="Contexts"/></Set>
          <Set name="configurationDir"><SystemProperty name="jetty.home"
default="."/>/contextsA</Set>
          <Set name="scanInterval">5</Set>
        </New>
      </Arg>
    </Call>

</Configure>
```

jettyB.xml

```
<Configure id="ServerB" class="org.mortbay.jetty.Server">

    <!-- set up the port for ServerB -->
    <Set name="connectors">
      <Array type="org.mortbay.jetty.Connector">
        <Item>
          <New class="org.mortbay.jetty.security.SslSocketConnector">
            <Set name="Port">8443</Set>
            <Set name="maxIdleTime">30000</Set>
            <Set name="Keystore"><SystemProperty name="jetty.home" default="."
/>/etc/keystore</Set>
            <Set name="Password">OBF:1vny1zlo1x8e1vnw1vn61x8g1zlu1vn4</Set>
            <Set name="KeyPassword">OBF:1u2u1wml1z7s1z7a1wnl1u2g</Set>
          </New>
        </Item>
      </Array>
    </Set>

     <!-- set up a context deployer for ServerB -->
      <Arg>
        <New class="org.mortbay.jetty.deployer.ContextDeployer">
          <Set name="contexts"><Ref id="Contexts"/></Set>
          <Set name="configurationDir"><SystemProperty name="jetty.home"
default="."/>/contextsB</Set>
          <Set name="scanInterval">5</Set>
        </New>
      </Arg>
    </Call>

</Configure>
```

Now we need to set up 2 context files, one to describe the webapp we wish to deploy on Server A and the other for the webapp to deploy on Server B. We then put these files into $JETTY_HOME/contextsA or $JETTY_HOME/contextsB respectively.

contextA.xml

```
<Configure class="org.mortbay.jetty.webapp.WebAppContext">
      <Set name="war"><SystemProperty name="jetty.home"/>/webapps/A</Set>
      <Set name="contextPath">/webappA</Set>
       ...
    </Configure>
```

contextB.xml

```
<Configure class="org.mortbay.jetty.webapp.WebAppContext">
      <Set name="war"><SystemProperty name="jetty.home"/>/webapps/B</Set>
      <Set name="contextPath">/webappB</Set>
       ...
    </Configure>
```

You run both Server instances in the same JVM by providing them both on the runline:

```
java -jar start.jar jettyA.xml jettyB.xml
```

Of course, you could also start two separate jetty instances, one with `jettyA.xml` and the other with `jettyB.xml`, if you wanted to. However, it is usually more efficient to run both Servers in the same JVM.

Alternative

There is also a alternative way to achieve the same result as above, however it is less efficient. It involves setting the list of connectors on a webapp from which it will accept requests. This is a less efficient solution than the one described above because the request is presented to each webapp, which then must decide if it will accept it or not. In the first solution, only one webapp will ever be passed the request.

Here's how to configure the alternative solution:

jetty.xml

```
<Configure class="org.mortbay.jetty.Server">

    <!-- set up both connectors -->
    <Set name="connectors">
      <Array
type="org.mortbay.jetty.Connector">
        <Item>
          <New
class="org.mortbay.jetty.nio.SelectChannelCo
nnector">
            <Set name="port">8080</Set>
```

```
            <Set
name="maxIdleTime">30000</Set>
            <Set name="Acceptors">1</Set>
            <Set name="name">connA</Set>
          </New>
        </Item>
        <Item>
          <New id="connB"
class="org.mortbay.jetty.nio.SelectChannelCo
nnector">
            <Set name="port">9090</Set>
            <Set
name="maxIdleTime">30000</Set>
            <Set name="Acceptors">1</Set>

            <Set name="name">connB</Set>
          </New>
        </Item>
      </Array>
```

```
    </Set>

</Configure>
```

"contextA.xml"

```
<Configure
class="org.mortbay.jetty.webapp.WebAppContex
t">
      <Set name="war"><SystemProperty
name="jetty.home"/>/webapps/A</Set>
      <Set name="contextPath">/webappA</Set>
      <Set name="connectorNames">
        <Array type="String">
          <Item>connA</Item>
        </Array>
       </Set>
      ...
    </Configure>
```

"contextB.xml"

```
<Configure
class="org.mortbay.jetty.webapp.WebAppContex
t">
      <Set name="war"><SystemProperty
name="jetty.home"/>/webapps/B</Set>
      <Set name="contextPath"/webappB</Set>
      <Set name="connectorNames">
        <Array type="String">
          <Item>connB</Item>
        </Array>
      </Set>
    </Configure>
```