How use shindig out of the box

Using a clean maven web-app.

Creating a new maven web-app:

```
mvn archetype:create -DgroupId=org.apache -DartifactId=shindig-app -DarchetypeArtifactId=maven-archetype-webapp
```

Configuring jetty:run plugin:

```
<plugin>
        <groupId>org.mortbay.jetty</groupId>
        <artifactId>maven-jetty-plugin</artifactId>
        <configuration>
                <tempDirectory>${basedir}/target/work</tempDirectory>
                <webApp>${basedir}/target/${finalName}.war</webApp>
                <contextPath>/</contextPath>
        </configuration>
</plugin>
```

Adding shindig as dependence:

```
<dependency>
        <groupId>org.apache.shindig</groupId>
        <artifactId>shindig-features</artifactId>
        <version>2.0.0</version>
</dependency>
<dependency>
        <groupId>org.apache.shindig</groupId>
        <artifactId>shindig-common</artifactId>
        <version>2.0.0</version>
</dependency>
<dependency>
        <groupId>org.apache.shindig</groupId>
        <artifactId>shindig-gadgets</artifactId>
        <version>2.0.0</version>
</dependency>
<dependency>
        <groupId>org.apache.shindig</groupId>
        <artifactId>shindig-social-api</artifactId>
        <version>2.0.0</version>
</dependency>
<dependency>
        <groupId>org.apache.shindig</groupId>
        <artifactId>shindig-extras</artifactId>
        <version>2.0.0</version>
</dependency>
```

Configuring web.xml:

```
<?xml version="1.0"?>
<web-app>

       - configuration -->
    <!-- If you have your own Guice module(s), put them here as a colon-separated -->

    <context-param>
        <param-name>guice-modules</param-name>
        <param-value>
```

```
        org.apache.shindig.common.PropertiesModule:
        org.apache.shindig.gadgets.DefaultGuiceModule:
        org.apache.shindig.gadgets.oauth.OAuthModule
    </param-value>
</context-param>

<listener>
    <listener-class>org.apache.shindig.common.servlet.GuiceServletContextListener
    </listener-class>
</listener>

<!-- Render a Gadget -->
<servlet>
    <servlet-name>xml-to-html</servlet-name>
    <servlet-class>
        org.apache.shindig.gadgets.servlet.GadgetRenderingServlet
    </servlet-class>
</servlet>

<!-- Proxy -->
<servlet>
    <servlet-name>proxy</servlet-name>
    <servlet-class>
        org.apache.shindig.gadgets.servlet.ProxyServlet
</servlet-class>
</servlet>

<servlet>
    <servlet-name>concat</servlet-name>
    <servlet-class>
        org.apache.shindig.gadgets.servlet.ConcatProxyServlet
</servlet-class>
</servlet>

<!-- OAuth callback -->
<servlet>
    <servlet-name>oauthCallback</servlet-name>
    <servlet-class>
        org.apache.shindig.gadgets.servlet.OAuthCallbackServlet
    </servlet-class>
</servlet>

<!-- Metadata RPC -->
<servlet>
    <servlet-name>metadata</servlet-name>
    <servlet-class>
        org.apache.shindig.gadgets.servlet.RpcServlet
</servlet-class>
</servlet>

<!-- javascript serving -->
<servlet>
    <servlet-name>js</servlet-name>
    <servlet-class>org.apache.shindig.gadgets.servlet.JsServlet
    </servlet-class>
</servlet>

<servlet-mapping>
    <servlet-name>js</servlet-name>
    <url-pattern>/gadgets/js/*</url-pattern>
</servlet-mapping>

<servlet-mapping>
    <servlet-name>proxy</servlet-name>
    <url-pattern>/gadgets/proxy/*</url-pattern>
</servlet-mapping>

<servlet-mapping>
    <servlet-name>concat</servlet-name>
    <url-pattern>/gadgets/concat</url-pattern>
</servlet-mapping>
```

```
    <servlet-mapping>
        <servlet-name>oauthCallback</servlet-name>
        <url-pattern>/gadgets/oauthcallback</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>xml-to-html</servlet-name>
        <url-pattern>/gadgets/ifr</url-pattern>
    </servlet-mapping>

    <servlet-mapping>
        <servlet-name>metadata</servlet-name>
        <url-pattern>/gadgets/metadata</url-pattern>
    </servlet-mapping>

</web-app>
```

Trying:

```
mvn clean package jetty:run
```

Now you can navigate to http://localhost:8080/gadgets/ifr?url=http://www.labpixies.com/campaigns/todo/todo.xml&view=home

After this point you have a minimal requirement to use gadget renderer inside your web-app, have fun.