

Тема    **The Apache Tomcat 5.5 Servlet/JSP Container**

Часть    **JNDI Resources**

Автор    **OMEN (omendba@gmail.com)**

10.10.2007

Tomcat 5 обеспечивает JNDI **InitialContext** экземпляр для каждого веб-приложения, которое выполняется в нем. Механизм совместим с J2EE.

Стандарт J2EE предоставляет стандартный набор элементов в /WEB-INF/web.xml файле, чтобы ссылаться на ресурсы; ресурсы, на которые ссылаются в этих элементах, должны быть определены в конфигурации сервера.

Для Tomcat 5, элементы InitialContext конфигурируются в теле тега **<Context>**. Они могут быть определены в файле \$CATALINA\_HOME/conf/server.xml, или, предпочтительно в контексте XML файла приложения (любой META-INF/context.xml).

Tomcat 5 поддерживает отдельное пространство имен глобальных ресурсов для всего сервера. Они формируются в элементе **<GlobalNamingResources>** файла \$CATALINA\_HOME/conf/server.xml. Вы можете определить эти ресурсы с использованием элемента **<ResourceLink>**.

На ресурсы, определенные в этих элементах можно сослаться следующими элементами в дескрипторе внедрения приложения (/WEB-INF/web.xml):

- **<env-entry>** – вход для среды окружения, единственный параметр может быть использован, чтобы сконфигурировать то, как приложение будет работать.
- **<resource-ref>** – ссылка на ресурс, которая типично относится к объекту фабрики для ресурсов, таких как JDBC DataSource, JavaMail Session.
- **<resource-env-ref>** – ссылка окружающей среды ресурса, новый вариант resource-ref, добавленный в Servlet 2.4, который более прост для формирования ресурсов и не требующий дополнительной информации.

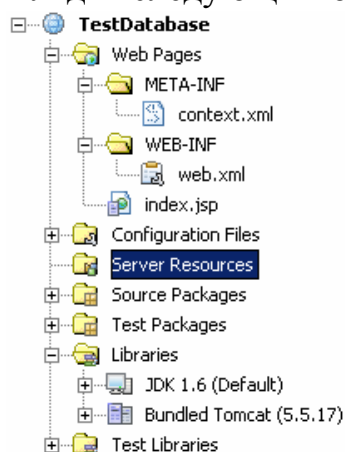
Все формируемые точки входа и ресурсы помещены в **java:comp/env** часть пространства имен JNDI.

## Первое тестовое приложение TestDatabase

Конфигурирование ресурсов будет происходить через конфигурационные файлы вручную.

Для доступа к базе данных FireBird 1.5.3 будем использовать библиотеку firebirdsql-full.jar. Данную библиотеку необходимо поместить в папку \$CATALINA\_HOME/common/lib. Тестовое приложение TestDatabase.war находится в папке \$CATALINA\_HOME/webapps.

Проект в NetBeans 5.5. выглядит следующим образом:



Как видно, приложение не содержит никаких драйверов для доступа к базе данных.

Исходный код **index.jsp** приведен ниже.

```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8" import="javax.naming.*, javax.sql.DataSource,
                                java.sql.*"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">

<%
Context initCtx=null;
Context envCtx=null;
DataSource ds=null;
Connection conn=null;
try {
    initCtx = new InitialContext();
    if (initCtx!=null) {
        envCtx = (Context) initCtx.lookup("java:comp/env");
        if (envCtx!=null) {
            ds = (DataSource) envCtx.lookup("jdbc/TestDB");
            if (ds!=null) {
                conn = ds.getConnection();
                if (conn!=null) {
                    Statement st = conn.createStatement();
                    ResultSet rset = st.executeQuery("select * from GOODS");
                    while (rset.next()) {
                        %><%=rset.getString("name")%><br><%
                    }
                    rset.close();
                    st.close();
                    conn.close();
                }
            }
            envCtx.close();
        }
        initCtx.close();
    }
} catch(NamingException e) {
    %><%= e.getMessage()%><%
} catch(SQLException e) {
    %><%= e.getMessage()%><%
}
%>
```

Исходный код **web.xml** приведен ниже.

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://java.sun.com/xml/ns/j2ee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd"
    version="2.4">
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>
            index.jsp
        </welcome-file>
    </welcome-file-list>
    <resource-ref>
        <description>DB Connection</description>
        <res-ref-name>jdbc/TestDB</res-ref-name>
        <res-type>javax.sql.DataSource</res-type>
        <res-auth>Container</res-auth>
    </resource-ref>
</web-app>
```

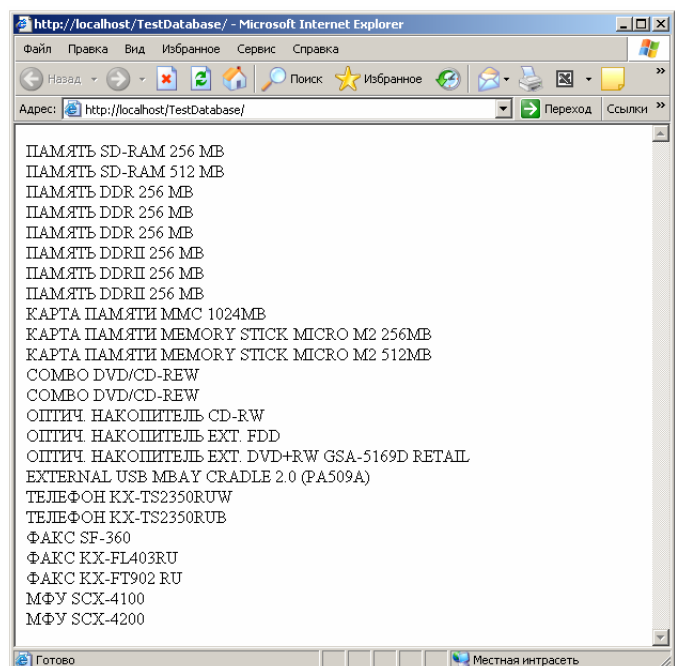
Исходный код **context.xml** приведен ниже.

```
<?xml version="1.0" encoding="UTF-8"?>
<Context path="/TestDatabase">
  <ResourceLink global="jdbc/TestDB" name="jdbc/TestDB" type="javax.sql.DataSource"/>
</Context>
```

Исходный код **server.xml** приведен ниже.

```
<?xml version="1.0" encoding="UTF-8"?>
<Server port="8025" shutdown="SHUTDOWN">
  <Listener className="org.apache.catalina.core.AprLifecycleListener"/>
  <Listener className="org.apache.catalina.mbeans.ServerLifecycleListener"/>
  <Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/>
  <Listener className="org.apache.catalina.storeconfig.StoreConfigLifecycleListener"/>
  <!-- Global JNDI resources -->
  <GlobalNamingResources>
    <Environment name="simpleValue" type="java.lang.Integer" value="30"/>
    <Resource auth="Container"
      description="FIRE"
      factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
      name="UserDatabase"
      pathname="conf/tomcat-users.xml"
      type="org.apache.catalina.UserDatabase"/>
  </GlobalNamingResources>
  <Service name="Catalina">
    <Connector URIEncoding="utf-8" acceptCount="100" connectionTimeout="20000"
      disableUploadTimeout="true" enableLookups="false"
      maxHttpHeaderSize="8192" maxSpareThreads="75" maxThreads="150"
      minSpareThreads="25" port="80" redirectPort="8443"/>
    <Connector enableLookups="false" port="8009" protocol="AJP/1.3" redirectPort="8443"/>
    <Engine defaultHost="localhost" name="Catalina">
      <Realm className="org.apache.catalina.realm.UserDatabaseRealm"
        resourceName="UserDatabase"/>
      <Host appBase="webapps" autoDeploy="false" name="localhost" unpackWARs="true"
        xmlNamespaceAware="false" xmlValidation="false">
        <Context path="/TestDatabase" docBase="TestDatabase"
          debug="5" reloadable="true" crossContext="true">
          <Resource name="jdbc/TestDB" auth="Container" type="javax.sql.DataSource"
            maxActive="100" maxIdle="30" maxWait="10000"
            username="SYSDBA" password="masterkey"
            driverClassName="org.firebirdsql.jdbc.FBDriver"
            url="jdbc:firebirdsql:localhost:shop_db"/>
        </Context>
      </Host>
    </Engine>
  </Service>
</Server>
```

Тестирование приложения:



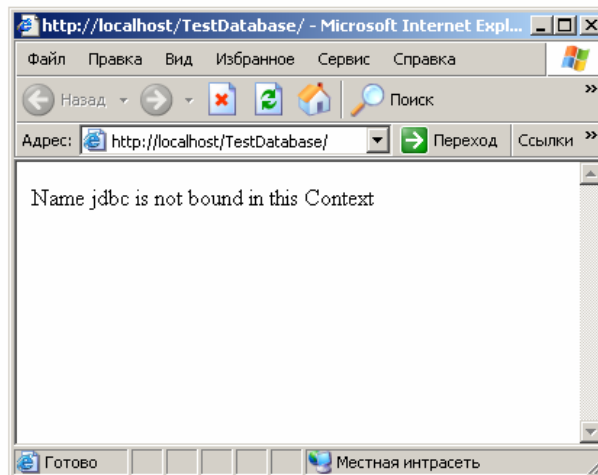
## Второе тестовое приложение TestDatabase

Попробуем сконфигурировать ресурсы через веб-шелл сервера Tomcat. Для этого нужно создать пользователя с ролью admin.

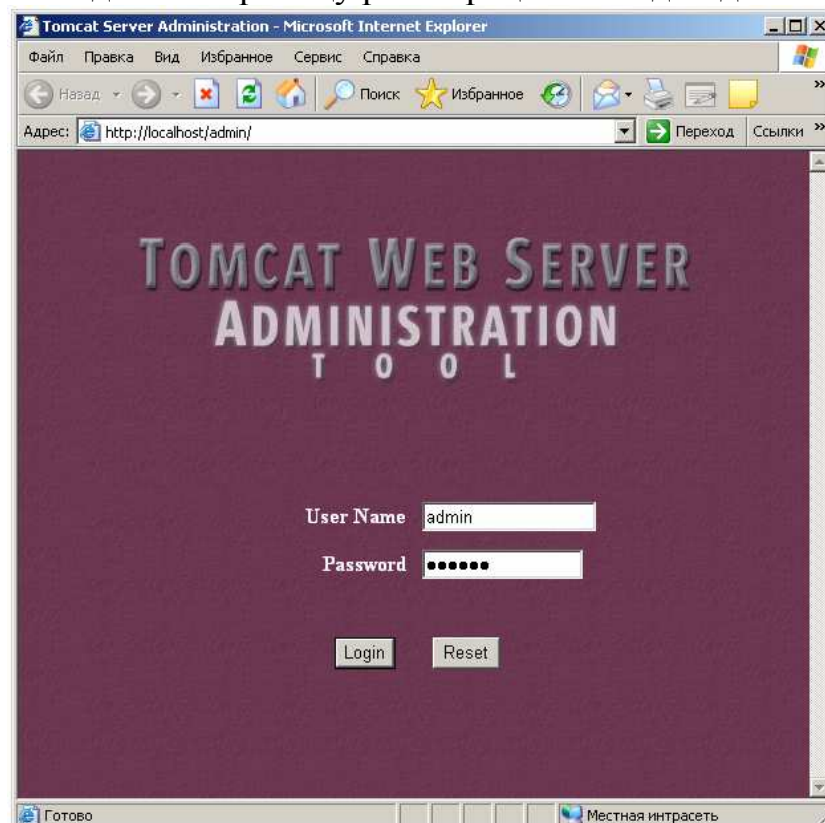
```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="manager"/>
  <role rolename="tomcat"/>
  <role rolename="admin"/>
  <role rolename="role1"/>
  <user username="both" password="tomcat" roles="tomcat,role1"/>
  <user username="tomcat" password="tomcat" roles="tomcat"/>
  <user username="admin" password="qwerty" roles="admin"/>
  <user username="role1" password="tomcat" roles="role1"/>
</tomcat-users>
```

Уберем из файла **server.xml** запись: *<Context path="/TestDatabase...*

Теперь при запуске приложения увидим следующее:



После этого зайдем на страницу регистрации и введем данные:



Далее необходимо сконфигурировать **Data Sources**:

The screenshot shows the Tomcat Web Server Administration Tool interface. On the left is a navigation tree with 'Data Sources' selected. The main area displays a configuration form for a new data source.

Property	Value
JNDI Name:	<input type="text" value="jdbc/TestDB"/>
Data Source URL:	<input type="text" value="jdbc:firebirdsql:localhost:shop_db"/>
JDBC Driver Class:	<input type="text" value="org.firebirdsql.jdbc.FBDriver"/>
User Name:	<input type="text" value="SYSDBA"/>
Password:	<input type="password" value="....."/>
Max. Active Connections:	<input type="text" value="4"/>
Max. Idle Connections:	<input type="text" value="2"/>
Max. Wait for Connection:	<input type="text" value="5000"/>

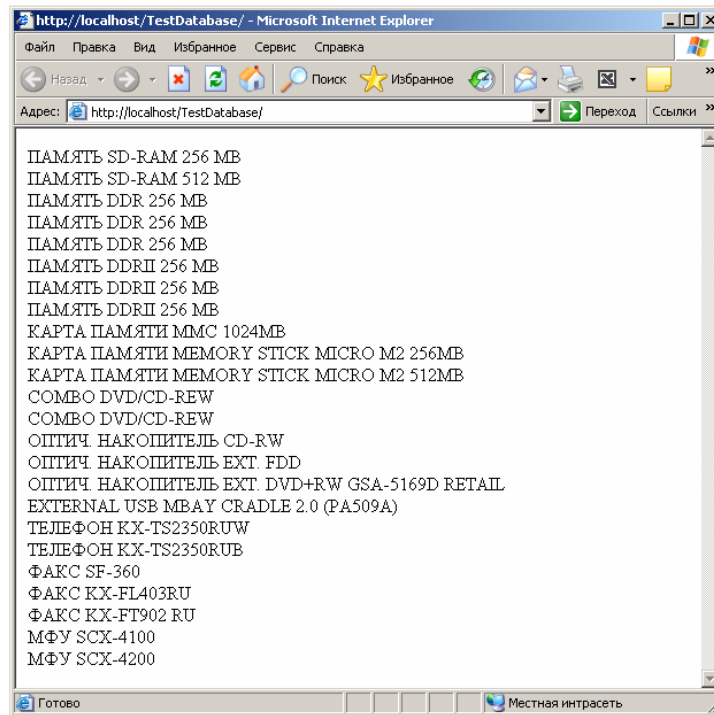
Buttons at the top right: Commit Changes, Log Out. Bottom status bar: Готово, Местная интрасеть.

The screenshot shows the Tomcat Web Server Administration Tool interface after configuration. The 'Data Sources' section now displays a table of the configured data source.

JNDI Name	JDBC Driver Class
<a href="#">jdbc/TestDB</a>	org.firebirdsql.jdbc.FBDriver

Buttons at the top right: Commit Changes, Log Out. Bottom status bar: Готово, Местная интрасеть.

А теперь:



### Третье тестовое приложение TestDatabase

Теперь используем библиотеку JSTL. Код главной страницы представлен ниже:

```
<%@page contentType="text/html"%>
<%@page pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<sql:query var="rs" dataSource="jdbc/TestDB">
  select * from GOODS
</sql:query>
<c:forEach var="row" items="${rs.rows}">
  ${row.name}<br/>
</c:forEach>
```

После этого мы увидим знакомое нам окно.