

# ***Utilisation d'une DataSource JDBC dans un serveur TOMCAT***

# Définir un pool de connexion avec TomCat

- **javax.sql.DataSource** = pool de connexions
  - *Interface dans javax.sql.DataSource*
- Objet **DataSource** pris en charge par le serveur TomCat
  - *org.apache.commons.dbcp.BasicDataSource*
  - Créé par le serveur Tomcat à partir de paramètres de configuration
    - *ressource globale à un contexte (ressource partagée par toutes les sessions d'une même application)*
      - description dans le fichier *context.xml* de l'application
    - *ressource globale partagée par tous les contextes*
      - description dans le fichier *server.xml*
  - Mis à disposition des applications par un serveur d'objets :
    - on récupère l'objet **DataSource** créé et géré par le conteneur au travers d'un identificateur (API JNDI (Java Naming Directory Interface))

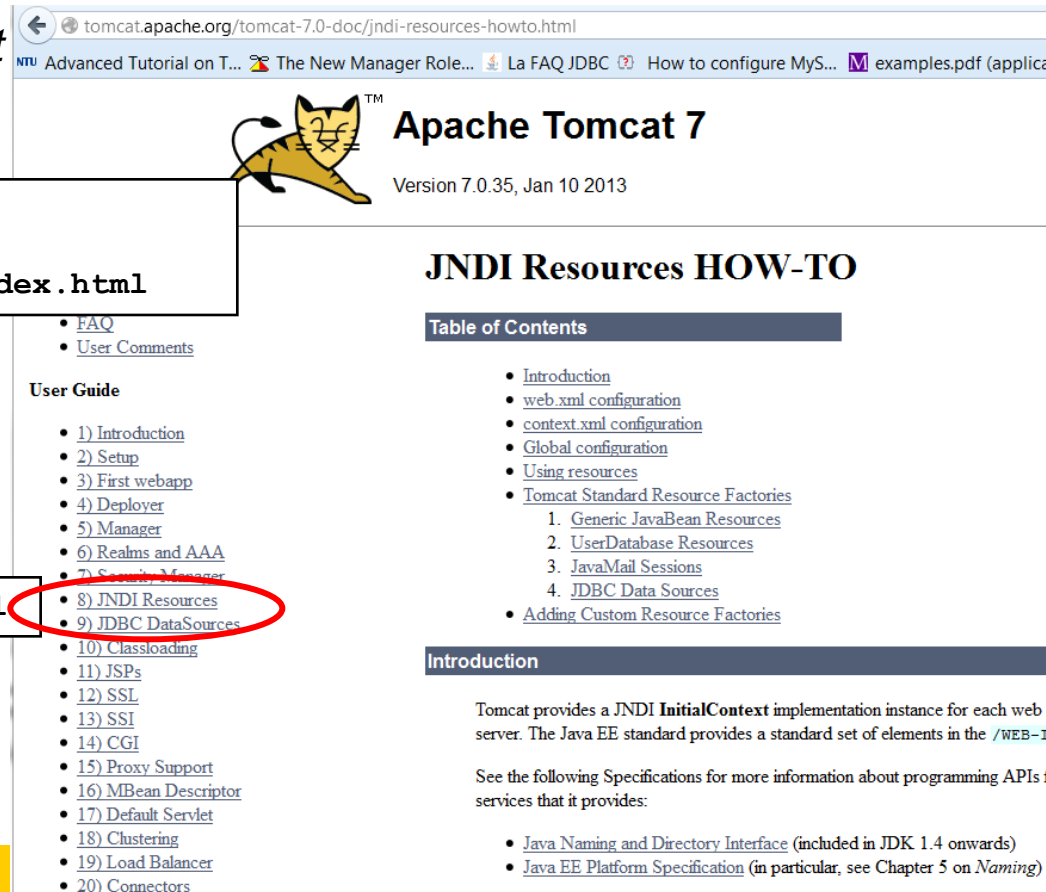
# Définir un pool de connexion avec Tomcat

- Définition d'une ressource de type DataSource pour le contexte de l'application
  - dans le fichier de configuration **context.xml** de l'application
- Désignation cette ressource dans le fichier de déploiement de l'application
  - dans le fichier **web.xml** de l'application
- Installer le pilote JDBC
  - dans le répertoire **lib** de Tomcat

Voir dans la documentation de Tomcat


<http://tomcat.apache.org/tomcat-x.x-doc/index.html>

[jndi-datasource-examples-howto.html](http://tomcat.apache.org/tomcat-x.x-doc/jndi-datasource-examples-howto.html)



tomcat.apache.org/tomcat-7.0-doc/jndi-resources-howto.html

Advanced Tutorial on T... The New Manager Role... La FAQ JDBC How to configure MyS... examples.pdf (appli...

 **Apache Tomcat 7**  
Version 7.0.35, Jan 10 2013

## JNDI Resources HOW-TO

### Table of Contents

- [Introduction](#)
- [web.xml configuration](#)
- [context.xml configuration](#)
- [Global configuration](#)
- [Using resources](#)
- [Tomcat Standard Resource Factories](#)
  1. [Generic JavaBean Resources](#)
  2. [UserDatabase Resources](#)
  3. [JavaMail Sessions](#)
  4. [JDBC Data Sources](#)
- [Adding Custom Resource Factories](#)

### Introduction

Tomcat provides a JNDI **InitialContext** implementation instance for each web server. The Java EE standard provides a standard set of elements in the `/WEB-INF`...

See the following Specifications for more information about programming APIs and services that it provides:

- [Java Naming and Directory Interface](#) (included in JDK 1.4 onwards)
- [Java EE Platform Specification](#) (in particular, see Chapter 5 on Naming)

### User Guide

- [1\) Introduction](#)
- [2\) Setup](#)
- [3\) First webapp](#)
- [4\) Deployer](#)
- [5\) Manager](#)
- [6\) Realms and AAA](#)
- [7\) Security Manager](#)
- [8\) JNDI Resources](#)
- [9\) JDBC DataSources](#)
- [10\) Classloading](#)
- [11\) JSPs](#)
- [12\) SSL](#)
- [13\) SSI](#)
- [14\) CGI](#)
- [15\) Proxy Support](#)
- [16\) MBean Descriptor](#)
- [17\) Default Servlet](#)
- [18\) Clustering](#)
- [19\) Load Balancer](#)
- [20\) Connectors](#)

# Définir un pool de connexion avec Tomcat

1

- Définition d'une ressource de type DataSource pour le contexte de l'application
  - dans le fichier de configuration **context.xml** de l'application
- Désignation cette ressource dans le fichier de déploiement de l'application
  - dans le fichier **web.xml** de l'application
- Installer le pilote JDBC
  - dans le répertoire **lib** de Tomcat

context.xml

```
<Context path="/BIMA">
```

```
  <Resource name="jdbc/UFRIMA"
```

```
    auth="Container"
```

```
    type="javax.sql.DataSource"
```

```
    maxActive="10"
```

```
    maxIdle="3"
```

```
    maxWait="10000"
```

```
    username="genoud"
```

```
    password="xxxxxxx"
```

```
    driverClassName="oracle.jdbc.OracleDriver"
```

```
    url="jdbc:oracle:thin:@im2ag-oracle.e.ujf-grenoble.fr:1521:ufrima"/>
```

```
</Context>
```

} Paramétrage du  
pool de connexion\*

} Paramétrage  
de la connexion  
JDBC

\* [http://tomcat.apache.org/tomcat-7.0-doc/jndi-resources-howto.html#UserDatabase\\_Resources](http://tomcat.apache.org/tomcat-7.0-doc/jndi-resources-howto.html#UserDatabase_Resources)

# Définir un pool de connexion avec Tomcat

- Définition d'une ressource de type DataSource pour le contexte de l'application
  - dans le fichier de configuration `context.xml` de l'application
- Désignation cette ressource dans le fichier de déploiement de l'application
  - dans le fichier `web.xml` de l'application
- Installer le pilote JDBC
  - dans le répertoire `lib` de l'application

2

`web.xml`

```
...
<servlet-mapping>
    ...
</servlet-mapping>

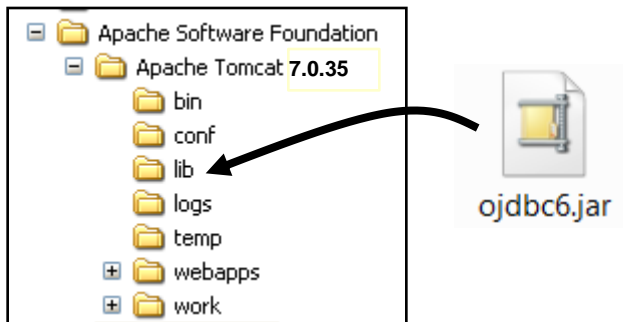
<resource-ref>
    <res-ref-name>jdbc/UFRIMA</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
</resource-ref>

...
```

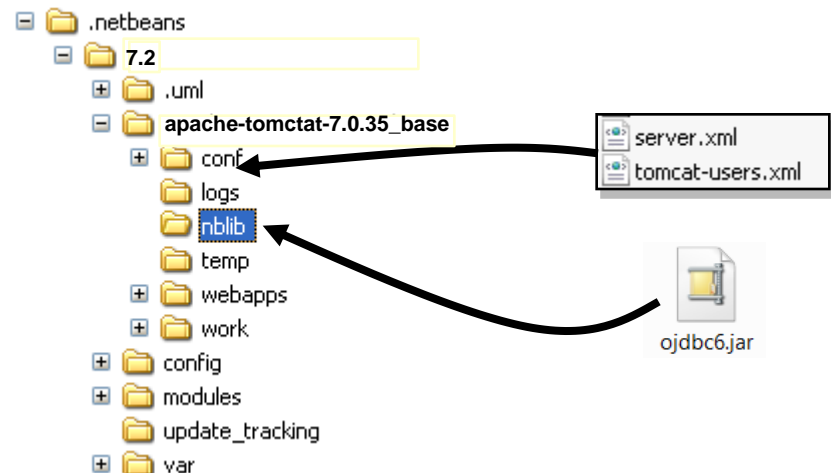
# Définir un pool de connexion avec Tomcat

- Définition d'une ressource de type DataSource pour le contexte de l'application
  - dans le fichier de configuration `context.xml` de l'application
- Désignation cette ressource dans le fichier de déploiement de l'application
  - dans le fichier `web.xml` de l'application
- Installer le pilote JDBC
  - dans le répertoire `lib` de Tomcat

3



Si vous lancez Tomcat depuis Netbeans et que vous n'avez pas le droit d'écriture dans le répertoire d'installation de Tomcat vous pouvez configurer l'instance de Tomcat que vous utilisez à l'aide des fichiers contenus dans le fichier `.netbeans` qui se trouve sur votre compte



# Définir un pool de connexion avec Tomcat

Définition d'une ressource globale partagée par tous les contextes

- Description dans le fichier de configuration du serveur (\$CATALINA\_HOME/conf/server.xml)

1

```
<GlobalNamingResources>
  <Environment
    name="simpleValue"
    type="java.lang.Integer"
    value="30"/>
  <Resource
    auth="Container"
    description="User database that can be updated and saved"
    name="UserDatabase"
    type="org.apache.catalina.UserDatabase"
    pathname="conf/tomcat-users.xml"
    factory="org.apache.catalina.users.MemoryUserDatabase"
  />
  <Resource
    name="jdbc/notes"
    type="javax.sql.DataSource"
    driverClassName="com.mysql.jdbc.Driver"
    password="yyyyyyyyy"
    maxIdle="2"
    maxWait="5000"
    username="genoud"
    url="jdbc:mysql://localhost/essai"
    maxActive="4"/>
</GlobalNamingResources>
```

La définition de la **DataSource** peut se faire par l'intermédiaire de la console d'administration Tomcat



pas disponible avec Tomcat 6 ou 7

The screenshot shows the 'Tomcat Web Server Administration Tool' interface. On the left is a tree view with 'Tomcat Server' expanded, showing 'Service (Catalina)', 'Resources', 'Data Sources', 'Mail Sessions', 'Environment Entries', 'User Databases', 'User Definition', 'Users', 'Groups', and 'Roles'. The 'Data Sources' item is selected. The main panel is titled 'Edit Data Source' and contains a table with the following properties and values:

Property	Value
JNDI Name:	jdbc/notes
Data Source URL:	<input type="text" value="jdbc:mysql://localhost/notes"/>
JDBC Driver Class:	<input type="text" value="com.mysql.jdbc.Driver"/>
User Name:	<input type="text" value="genoud"/>
Password:	<input type="password" value="yyyyyyyyy"/>
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"/>

At the top right of the main panel are buttons for 'Commit Changes' and 'Log Out'. Below the table are 'Save' and 'Reset' buttons. The 'Data Source Actions' dropdown menu is set to 'Available Actions'.

# Définir un pool de connexion avec Tomcat

Définition d'une ressource globale partagée par tous les contextes

2

- Déclaration d'un lien pour chaque contexte qui peut y accéder (fichier `context.xml` des applications concernées)

`context.xml` de l'application

```
<Context path="/HistogrammesMVC">
  <ResourceLink
    global="jdbc/notes"
    name="jdbc/notes"
    type="javax.sql.DataSource"/>
</Context>
```

3

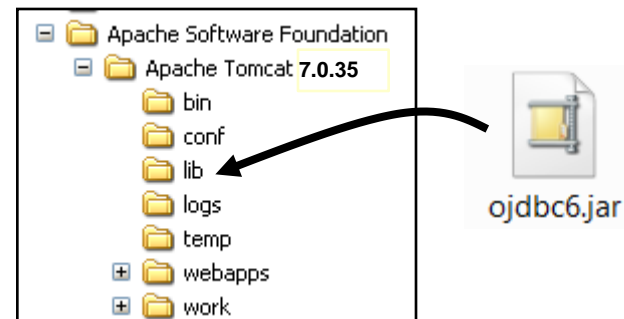
- Désignation cette ressource dans le fichier de déploiement (`web.xml`) de l'application

`web.xml` de l'application

```
<resource-ref>
  <description>Data Source pour les notes</description>
  <res-ref-name>jdbc/notes</res-ref-name>
  <res-type>javax.sql.DataSource</res-type>
  <res-auth>Container</res-auth>
</resource-ref>
</web-app>
```

4

- Ne pas oublier d'installer le pilote jdbc dans le répertoire lib de tomcat





# connexion à la BD

## pool de connexions

- Utilisation de la source de données dans le code java

```
import javax.naming.*
```

```
public class MaServlet extends HttpServlet {
```

```
/**
```

```
 * la source de données qui permettra d'accéder à la base de donnée  
 * (c'est elle qui gère un pool de connexions, et qui fournira les  
 * connexions aux différents objets en ayant l'usage).  
 */
```

```
private DataSource dataSource;
```

```
/**
```

```
 * A l'initialisation du servlet, création d'un pool de connexions  
 * qui servira à fournir des connexions à la base de données  
 * utilisée par les composants modèle.  
 */
```

```
public void init() throws ServletException {
```

```
    try {
```

```
        Context initialCtx = new InitialContext();
```

```
        dataSource = (DataSource) initialCtx.lookup("java:comp/env/jdbc/UFRIMA");
```

```
    }
```

```
    catch (NamingException ne) {
```

```
        throw new ServletException("problème lors du chargement de la data source",  
                                    ne);
```

```
    }
```

```
}
```

Récupération d'une de données  
dans une servlet

Utilisation de l'API JNDI

Le nom de la DataSource

# connexion à la BD

## pool de connexions

- Utilisation de la source de données dans le code java

```
import javax.annotation.Resource;
```

```
public class MaServlet extends HttpServlet {
```

```
/**
```

```
 * la source de données qui permettra d'accéder à la base de donnée  
 * (c'est elle qui gère un pool de connexions, et qui fournira les  
 * connexions aux différents objets en ayant l'usage).  
 */
```

```
@Resource(name = "jdbc/UFRIMA")  
private DataSource dataSource;
```

possibilité d'injecter directement la ressource  
avec annotation @Resource

Récupération d'une de données  
dans une servlet

Pour en savoir plus

<http://docs.oracle.com/javaee/6/tutorial/doc/bncjh.html>

```
}
```