

Table of Contents

Mit JVx können Datenbankunabhängige Applikationen entwickelt werden. Der Zugriff auf die Datenbank erfordert jedoch bestimmte Informationen wie Hostname, Datenbank Port, Datenbank Name, Benutzername, Passwort.

Diese Informationen werden üblicherweise in einer Konfigurationsdatei abgelegt. Wir verwenden dafür die Applikations Konfiguration config.xml die Serverseitig für unsere Applikation bereit liegt:

z.B.:

config.xml

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

<application>
  <securitymanager>
<class>com.sibvisions.apps.packung.security.AnonymousDBSecurityManager<
/class>
  <database>
    <url2>jdbc:oracle:thin:@localhost:1521:packung</url2>
    <url>jdbc:derby://localhost:1527/packung</url>
    <username>pack_user</username>
    <password>pack_password</password>
  </database>
</securitymanager>

  <!-- predefined life-cycle object names -->
  <lifecycle>
    <mastersession>com.sibvisions.apps.packung.Session</mastersession>
  </lifecycle>
</application>
```

Wie im Beispiel zu sehen ist, verwenden wir den url Parameter um den Zugriff auf die Datenbank zu definieren. Während der Entwicklung einer Datenbankunabhängigen Applikation muss die Datenbank geändert werden um die umgesetzte Funktionalität zu testen. Wir tauschen dazu den Inhalt von url mit url2.

Nachdem die Datenbank konfiguriert wurde, muss die Applikation nun darauf zugreifen.

Für den Zugriff auf die Datenbank ist ein Objekt vom Typ IDBAccess nötig. Es existieren unterschiedliche Implementierungen wie z.B. OracleDBAccess, DerbyDBAccess.

Der Entwickler könnte nun eine konkrete Implementierung wie z.B OracleDBAccess verwenden um ausschließlich auf Oracle Datenbanken zuzugreifen. Dadurch erhält man zwar spezielle Möglichkeiten die von OracleDBAccess angeboten werden, doch wenn die Datenbank gewechselt wird, muss auch der Source Code geändert werden.

Für diesen Fall wählen wir einen generischen Ansatz um auf die Datenbank zuzugreifen:

```
IDBAccess dba = IDBAccess.getDBAccess(cfgSession.getProperty
("/application/securitymanager/database/url"));
```

```
dba.setUsername(cfgSession.getProperty("/application/securitymanager/databas  
e/username"));  
dba.setPassword(cfgSession.getProperty("/application/securitymanager/databas  
e/password"));  
dba.open();
```

Dadurch erhalten wir immer das richtige Objekt für den Zugriff auf die konfigurierte Datenbank.

From:

<http://doc.sibvisions.com/> - **Documentation**

Permanent link:

http://doc.sibvisions.com/de/jvx/common/setup/dbindependent_config

Last update: **2018/02/02 12:46**

