

Datenbank einrichten

Die Installation erfolgt bei den verschiedenen RDBMS unterschiedlich. Nachfolgend werden die Schritte für die RDBMS MySQL, PostgreSQL, MS SQLServer und die Datenbank MS Access beschrieben. Zu jeder RDBMS existieren Frontends, mit denen die RDBMS administriert und die Daten manipuliert werden können. Wir verwenden anstelle der mit den RDBMS mitgelieferten zeilenorientierten Administrationsprogramme diese GUI-orientierten Tools.

Beachten Sie in den Kapiteln Initialisieren der Datenbank, dass die Wahl des jeweiligen Script davon abhängt, ob eine frische Datenbank aufgesetzt werden soll, oder ob die Daten von einer früheren Version von Colibri (mindestens Version 1.5) übernommen werden sollen. Die Namen der Scripte unterscheiden sich wie folgt:

Script für frisch aufgesetzte Datenbank: `colibri_inserts_xxx.sql`

Script bei Datenmigration aus Colibri: `colibri_inserts_xxx_migration.sql`

wo je nach verwendeter Datenbank xxx mit mysql, mssqlserver zu ersetzen ist.

Die Migration der Daten aus dem Colibri wird am Schluss dieses Kapitels beschrieben, da sie für alle RDBMS gleich abläuft.

MySQL

Installation MySQL-Datenbank (Windows)

(Downloadsite: <http://www.mysql.com/downloads/mysql-4.0.html>)

1. `mysql-4.0.14b-win.zip` (oder höhere Version) in ein temporäres Verzeichnis (z.B. `C:\Winnt\Temp`) entpacken.
2. In das angegebene Verzeichnis wechseln und `Setup.exe` starten. MySQL wird installiert. Bestätigen Sie die Vorschläge, die Ihnen im Laufe des Setup-Prozesses gemacht werden. MySQL wird, wenn nicht ein anderer Pfad angegeben wird, in das Verzeichnis `C:\mysql` installiert.
3. Richten Sie MySQL unter Windows 2000 wie folgt als Dienst ein. Hierzu müssen Sie als Administrator am System angemeldet sein.
 - 3.1 Öffnen Sie eine Eingabeaufforderung (vgl. 1.2), falls Sie das noch nicht getan haben.
 - 3.2 Geben Sie an der Kommandozeile `"C:\mysql\bin\mysqld-max.exe --install"` (ohne Anführungszeichen) ein und bestätigen Sie mit Return. Die Shell meldet das erfolgreiche Installieren des mysql-Daemons mit `"Service successfully installed"`.
 - 3.3 Überprüfen Sie unter `"Start > Einstellungen > Systemsteuerungen > Verwaltung > Dienste"`, ob der Dienst mit dem Namen `"MySQL"` eingerichtet ist. Als Starttyp sollte `"Automatisch"` angegeben sein, damit das RDBMS automatisch beim Systemstart hochgefahren wird.
 - 3.4 Starten Sie nun MySQL, indem Sie im Eigenschaftenfenster des MySQL-Dienstes die Schaltfläche `"Starten"` aktivieren, und schliessen Sie das Fenster, wenn der Dienst erfolgreich gestartet wurde.

Installation des MySQL-Clients

(Downloadsite: <http://www.mysql.com/downloads/mysqlcc.html>)

Das MySQL Control Center dient der Administration des Servers und weist eine GUI auf, die die Administration sehr erleichtert.

1. `mysqlcc-0.9.2-win32.zip` (oder höher) in ein temporäres Verzeichnis (z.B. `C:\Winnt\Temp`) entpacken.

2. `mysqlcc.msi` im angegebenen Verzeichnis starten. MySQLCC wird installiert. Bestätigen Sie die Vorschläge, gegebenenfalls streichen Sie in der Sprachenauswahl die Sprachpakete, die Sie nicht brauchen.
3. Starten Sie vom Desktop (oder aus Start > Programme > MySQL Control Center) "MySQL Control Center".
4. Registrieren Sie den Server im Fenster "Register Server".
- 4.1 Wenn der MySQL-Server lokal installiert ist, aktivieren Sie die Schaltfläche "Add". Falls im unteren Fenster eine Fehlermeldung erscheint, vergewissern Sie sich, ob der MySQL-Dienst tatsächlich gestartet ist (unter "Start > Einstellungen > Systemsteuerungen > Verwaltung > Dienste") und wiederholen Sie die Registrierung.
- 4.2 Falls der MySQL-Server auf einem entfernten Host installiert wurde, geben Sie im Feld "Host Name" den Hostnamen des entfernten Rechners an und bestätigen Sie mit der Schaltfläche "Add".
5. Um die Verbindung zum Server herzustellen, wählen Sie im Fenster "Console Manager" den Datenbankserver aus der Liste aus. Wählen Sie im Kontextmenu "Connect".

Initialisieren der Datenbank

1. Um die neue Datenbank zu erstellen, wählen Sie über dem Eintrag "Databases" aus dem Kontextmenu den Eintrag "New Database" aus. Geben Sie im Fenster "Create Database" den Namen "colibri" ein. Bestätigen Sie mit "OK". Die Datenbank wird erstellt.
2. Die Tabellen werden mit einem SQL-Script generiert, das in der Lieferung enthalten ist (`colibri_create_tables_mysql.sql`). Öffnen Sie den Knoten "Databases", wählen Sie die eben erstellte Datenbank und öffnen Sie die Verbindung durch Auswahl des Eintrags "Connect" im Kontextmenu.
3. Klicken Sie im Toolbar auf den Button "SQL" und wählen Sie im neuen Fenster aus dem Menu "File" den Eintrag "Open". Laden Sie die erwähnte Datei "`colibri_create_tables_mysql.sql`".
4. ergänzen Sie den Text vor dem ersten aktiven Eintrag durch

```
USE colibri;
```

(vergessen Sie das Semikolon nicht!), damit das Programm weiss, welche Datenbank es benutzen soll.
5. Klicken Sie auf die Schaltfläche "!" im Toolbar. Dadurch wird das Script ausgeführt und die Tabellen sollten jetzt angelegt werden.
6. Vergewissern Sie sich, dass die neu erstellte Datenbank jetzt Tabellen enthält, indem Sie die Verbindung aufheben (Disconnect aus dem Kontextmenu) und wieder connecten (Bug: Die Liste wird nicht aktualisiert).
8. Dieser Schritt ist abhängig davon, ob Sie eine Datenbank frisch aufsetzen oder ob Sie die Daten aus einer Vorgängerversion von Colibri übernehmen. Um die Datenbank frisch aufzusetzen, führen Sie a) aus, um die Daten von Colibri zu übernehmen, b).
- a) Laden Sie eine weitere Query, indem Sie wie in Schritt 3. und 4. vorgehen und wählen Sie diesmal das Script "`colibri_basic_inserts_mysql.sql`" aus und starten Sie es, wie in Schritt 5. beschrieben. Nun ist die Datenbank initialisiert und kann benützt werden.
- b) Laden Sie eine weitere Query, indem Sie wie in Schritt 3. und 4. vorgehen und wählen Sie diesmal das Script "`colibri_basic_inserts_mysql_migration.sql`" aus und starten Sie es, wie in Schritt 5. beschrieben. Nun müssen noch die Daten aus dem

alten Colibri übernommen werden.

Benutzer einrichten

1. Im Programm `mysqlcc.exe` wählen Sie mit dem rechten Mauszeiger aus dem Listenbaum unter dem Server den Eintrag `User Administration` und wählen aus dem Kontextmenu den Eintrag `New User`.

2. Im erscheinenden Fenster setzen Sie folgende Eigenschaften:

```
Username:   colibri
Host: <servername>      (ersetzen Sie <servername> durch den Namen des Servers)
Password:   colibri
```

Markieren Sie unter `Allow access to:` die Datenbanken, auf die der User Zugriff haben soll (`colibri`, `temp` und/oder `tutorial` zum Beispiel) mit einem Häkchen und bestätigen Sie die Änderungen mit dem Klicken auf den Button `Add`.

3. Wiederholen Sie die Schritte 1. und 2. Für jeden Client (Host), so dass Sie am Ende folgende Benutzer eingerichtet haben:

```
colibri@localhost
```

Bei Netzwerkbetrieb zusätzlich für jeden Host:

```
colibri@<client1>
colibri@<client2>
```

```
...
```

(ersetzen Sie `<clientx>` jeweils durch den Computernamen des Clients)

PostgreSQL

MS SQL-Server (MSDE2000A)

Installation des Servers (nur Windows)

1. Melden Sie sich als Benutzer mit Administrationsrechten am System an.
2. Führen Sie das Installationsprogramm "`GER_MSDE2000A.exe`" aus. Akzeptieren Sie die Nutzungsbedingungen und wählen Sie ein temporäres Verzeichnis, in das das Installationsprogramm die Dateien entpacken kann.
3. Wenn das Installationsprogramm beendet ist, wechseln Sie in das vorher gewählte Verzeichnis.
4. Öffnen Sie die Datei "`Setup.ini`" und tragen Sie die folgende Option unter `[options]` ein:
`SAPWD=<Passwort>`
(wählen Sie für `<Passwort>` einen sicheren Ausdruck) und speichern Sie die Änderungen ab.
5. Starten Sie nun `Setup.exe`. Die Installation wird nun vorbereitet und anschliessend ausgeführt. Anschliessend müssen Sie den Computer neu starten.

Installation des SQLServer Frontends DbamGR2k

1. Führen Sie das Setupprogramm "`Setup_DbamGR2k.exe`" aus. Nach der Installation wird wieder ein Neustart notwendig.

2. Starten Sie das Programm `DbaMGR2k.exe` und wählen Sie aus dem Menu `Activity` den Eintrag `"Edit connection properties/connect"`.
3. Im erscheinenden Fenster `"SQL Server connection properties"` wechseln Sie zum Register `Settings` und aktivieren die Option `"Trusted NT Connection"`. Anschliessend klicken Sie auf `Connect`.
4. Wählen Sie in der erscheinenden Liste den Server `(local) (Windows NT)` an und öffnen Sie mit der rechten Maustaste das Kontextmenu, aus dem Sie `Properties` anwählen.
5. Im neuen Fenster `"SQL Server properties"` wählen Sie das Register `Security` und aus der erscheinenden Liste den Eintrag `Security mode`. Doppelklicken Sie auf die entsprechende Valuespalte und wählen Sie aus der Dropdownliste den Eintrag `Mixed security` aus. Schliessen Sie das Fenster.
6. Wählen Sie im Menu `Tools & Settings` den Eintrag `SQL Server Network utility` an. Im erscheinenden Fenster wählen Sie aus der Liste `Deaktivierte Protokolle` den Eintrag `TCP/IP` an und klicken Sie auf `Aktivieren>>`. Der Eintrag sollte jetzt auf der rechten Seite unter `Aktivierte Protokolle` erscheinen. Klicken Sie `Ok` und beenden Sie das Programm.

Initialisieren der Datenbank

1. Programm `DbaMGR2k.exe` ausführen und am SQL Server anmelden und gegebenenfalls die Baumstruktur soweit öffnen, bis der Eintrag `Databases` erscheint.
2. Durch Anklicken des Eintrages `Databases` mit der rechten Maustaste erscheint ein Kontextmenu, in dem `New Database` angeklickt wird. Es erscheint ein Fenster `Database properties`.
3. Unter `DB Name` den gewünschten Datenbanknamen in Kleinbuchstaben eingeben (z.B. `colibri`, `temp`, `schulung`). Falls gewünscht und auf Grund von bereits gemachten Erfahrungen können hier weitere Einstellungen vorgenommen werden. Speichern und Fenster schliessen.
4. In der Baumstruktur die neu erstellte Datenbank auswählen und aus dem Menu `Activity` den Eintrag `Query` auswählen.
5. Symbol `"Open Query from file"` anklicken und in das Verzeichnis wechseln, wo sich die SQL-Skripte befinden (auf der CD:
`<Laufwerk:>\ColibriTS\scripts\db\mssqlserver`) und dort das Script `"create_tables_colibri_mssqlserver.sql"` auswählen.
6. Am Beginn des Scripts muss noch festgelegt werden, in welcher Datenbank die Tabellen erstellt werden sollen. Dazu fügt man gleich am Beginn des Scripts folgenden Eintrag ein:

`USE <Datenbank>;`

(Für `<Datenbank>` setzen Sie den Namen der oben erstellten Datenbank ein.

7. Das Script durch Anklicken des Symbols `"Execute Query (F5)"` starten. Nach dem Lauf des Scripts überzeugen Sie sich, dass keine Fehlermeldungen gezeigt worden sind, indem Sie auf das Register `Messages` klicken. Wenn keine Fehlermeldungen eingetragen sind, sollte das Script erfolgreich abgelaufen sein.
8. Dieser Schritt ist abhängig davon, ob Sie eine Datenbank frisch aufsetzen oder ob Sie die Daten aus einer Vorgängerversion von Colibri übernehmen. Um die Datenbank frisch aufzusetzen, führen Sie a) aus, um die Daten von Colibri zu übernehmen, b).

- a) Laden Sie eine weitere Query, indem Sie wie in Schritt 4. und 5. und 6. vorgehen

und wählen Sie diesmal das Script "colibri_basic_inserts_mssqlserver.sql" aus und starten Sie es, wie in Schritt 7. beschrieben.

b) Laden Sie eine weitere Query, indem Sie wie in Schritt 4. und 5. und 6. vorgehen und wählen Sie diesmal das Script "colibri_basic_inserts_mssqlserver_migration.sql" aus und starten Sie es, wie in Schritt 7. beschrieben. Nun müssen noch die Daten aus dem alten Colibri übernommen werden.

Benutzer einrichten

1. Im Programm DbMGR2k.exe wählen Sie mit dem rechten Mauszeiger aus dem Listenbaum den Eintrag Logins und wählen aus dem Kontextmenu den Eintrag New login.

2. Im Fenster "New login properties" setzen Sie folgende Eigenschaften:

```
Name: colibri
Password: colibri
Windows NT Authentication: No
Default Database: colibri
Language: German
```

Klicken Sie anschliessend auf Save und bestätigen Sie das Passwort mit colibri.

3. Wechseln Sie ins Register Database access und aktivieren Sie unter "Access granted to database for this login" die Datenbank colibri. Bestätigen Sie wiederum mit Save.
4. Wählen Sie wiederum im Listenbaum den Eintrag Users unter der eingerichteten Datenbank und klicken Sie mit der rechten Maustaste in das rechte Listenfeld und wählen Sie aus dem Kontextmenu den Eintrag New user.
5. Im neuen Fenster geben Sie folgendes ein:

```
Login colibri
Name: colibri
```

Bestätigen Sie mit Save und klicken Sie auf Close.

6. Über Start->Einstellungen->Systemsteuerung->Verwaltung->Dienste beenden Sie den Dienst SQL Server und starten ihn neu. Nun ist er einsatzbereit.

MS Access

Das objekt-relationale Datenmapping

Nachdem die Datenbank eingerichtet ist, muss der Anwendung mitgeteilt werden, wie sie auf die Datenbank zugreifen kann. Dies erfolgt in unserem Falle durch die OJB-Schicht, einem Open Source Projekt der Apache Group (<http://db.apache.org/>). Die Konfiguration primär über zwei Konfigurationsdateien, von denen uns vorerst nur "repository_database.xml" interessiert, da diese Datei die Verbindungsdaten zur Datenbank bereitstellt (die zweite Datei heisst "repository_user.xml" und beschreibt das objektrelationale Datenmapping zwischen dem Programm und der Datenbank. Ihr Inhalt ist im Anhang zu sehen.

repository_database.xml

Die Konfigurationsdatei befindet sich im Verzeichnis properties/ojb. Sie enthält alle für eine

Datenbankverbindung relevanten Daten, die sich das Programm zur Laufzeit holt. Die Verbindungskonfigurationen unterscheiden sich von Datenbank zu Datenbank, es können gleichzeitig mehrere Datenbanken benutzt werden. In unserem Falle benutzen wir immer drei Verbindungen:

Standard:	Die Datenbank, die unsere produktiven Daten enthält.
Temporary:	Diese Datenbank speichert nur bei Verbindungsproblemen mit der Standard-Verbindung die Kassentransaktionen und ermöglicht so einen reibungslosen Betrieb auch bei einem eventuellen Netzwerkausfall. Die Umstellung bei einem plötzlich auftretenden Netzwerkproblem erfolgt automatisch. Gegebenenfalls wird der Benutzer entsprechend informiert.
Tutorial:	Diese Datenbank ist für Schulungs- und Demonstrationszwecke gedacht, kurz für alle nicht produktiven Vorgänge.

Nachfolgend finden Sie Beispiele für die bislang unterstützten RDBMS MySQL, PostgreSQL, MS SQLServer und MS Access. Wir gehen bei allen Beispielen davon aus, dass:
<alias> einer der folgenden Bezeichnungen hat: standard, temporary, tutorial
<datenbankname> dem Datenbanknamen für die entsprechende Verbindung entspricht (bislang benutzen wir folgende Namen: colibri für die Standardverbindung, temp für die temporäre Verbindung und schulung für die Tutorialverbindung):

<alias>: Einer der vorgegebenen aliasse: standard, temporary, tutorial.
<datenbankname>: Einer der vorgegebenen Datenbanknamen: colibri, temp, schulung.

Ausserdem lauten in den Beispielen der Benutzername und das Passwort colibri (in jedem Falle müssen sie dem der Datenbank zugeordneten Benutzern entsprechen). Ein Vergleich der aufgelisteten jdbc-connection-descriptoren zeigt die Unterschiede zwischen den einzelnen Verbindungen.

MySQL

MySQL verfügt über eine direkte Datenbankverbindung über JDBC 3.0.

```
<jdbc-connection-descriptor
  jcd-alias="<alias>"
  default-connection="true"
  platform="mySQL"
  jdbc-level="3.0"
  driver="com.mysql.jdbc.Driver"
  protocol="jdbc"
  subprotocol="mysql"
  dbalias="//saturn:3306/<datenbankname>"
  username="colibri"
  password="colibri"
  batch-mode="false"
  useAutoCommit="2"
>

  <connection-pool
    maxActive="5"
    validationQuery=""
  />

  <sequence-manager
className="org.apache.obj.broker.util.sequence.SequenceManagerHighLowImpl">
  <attribute attribute-name="grabSize" attribute-value="1"/>
  </sequence-manager>
</jdbc-connection-descriptor>
```

PostgreSQL

PostgreSQL verfügt über eine direkte Datenbankverbindung über JDBC 3.0.

```

<jdbc-connection-descriptor
  jcd-alias="<alias>"
  default-connection="true"
  platform="PostgreSQL"
  jdbc-level="3.0"
  driver="org.postgresql.Driver"
  protocol="jdbc"
  subprotocol="postgresql"
  dbalias="//saturn:5432/<datenbankname>"
  username="colibri"
  password="colibri"
  batch-mode="false"
  useAutoCommit="2"
>

  <connection-pool
    maxActive="5"
    validationQuery=""
  />

  <sequence-manager
className="org.apache.obj.broker.util.sequence.SequenceManagerHighLowImpl">
  <attribute attribute-name="grabSize" attribute-value="1"/>
  </sequence-manager>
</jdbc-connection-descriptor>

```

MS SQLServer

MS SQLServer kann über eine direkte Datenbankverbindung über JDBC 3.0 oder über ODBC angesprochen werden. Schneller und sicherer ist die JDBC-Verbindung, die nachfolgend dokumentiert ist.

```

<jdbc-connection-descriptor
  jcd-alias="standard"
  default-connection="true"
  platform="MsSQLServer"
  jdbc-level="2.0"
  driver="com.microsoft.jdbc.sqlserver.SQLServerDriver"
  protocol="jdbc"
  subprotocol="microsoft:sqlserver"
  dbalias="//MERKUR:1433;DatabaseName=colibri;SelectMethod=cursor"
  username="colibri"
  password="colibri"
  batch-mode="false"
  useAutoCommit="2"
>

  <connection-pool
    maxActive="5"
    validationQuery=""
  />

  <sequence-manager
className="org.apache.obj.broker.util.sequence.SequenceManagerHighLowImpl">
  <attribute attribute-name="grabSize" attribute-value="1"/>
  </sequence-manager>
</jdbc-connection-descriptor>

```

Im Vergleich dazu die ODBC-Verbindung:

```

<jdbc-connection-descriptor
  jcd-alias="standard"
  default-connection="true"
  platform="MsSQLServer"
  jdbc-level="2.0"
  driver="sun.jdbc.odbc.JdbcOdbcDriver"

```

```

        protocol="jdbc"
        subprotocol="odbc"
        dbalias="mssqlserver_colibri"
        username="colibri"
        password="colibri"
        batch-mode="false"
        useAutoCommit="2"
    >

    <connection-pool
        maxActive="5"
        validationQuery="" />

    <sequence-manager
className="org.apache.obj.broker.util.sequence.SequenceManagerHighLowImpl">
        <attribute attribute-name="grabSize" attribute-value="1"/>
    </sequence-manager>
</jdbc-connection-descriptor>

```

MS Access

MS Access kann nur über die JDBC-ODBC-Bridge angesprochen werden. Dieses Protokoll eignet sich nicht für Verbindungen über ein Netzwerk.

```

<jdbc-connection-descriptor
    jcd-alias="standard"
    default-connection="true"
    platform="MsAccess"
    jdbc-level="2.0"
    driver="sun.jdbc.odbc.JdbcOdbcDriver"
    protocol="jdbc"
    subprotocol="odbc"
    dbalias="msaccess_colibri"
    username="colibri"
    password="colibri"
    batch-mode="false"
    useAutoCommit="2"
>

    <connection-pool
        maxActive="5"
        validationQuery="" />

    <sequence-manager
className="org.apache.obj.broker.util.sequence.SequenceManagerHighLowImpl">
        <attribute attribute-name="grabSize" attribute-value="1"/>
    </sequence-manager>
</jdbc-connection-descriptor>

```

Je nach Topologie benötigen wir zwei bis drei der Verbindungen:

Wenn ColibriTS nur auf eine lokale Datenbank zugreift, dann reicht ein RDBMS mit zwei installierten Datenbanken: Standard und Schulung (resp. standard und tutorial).

Wenn ColibriTS im Netzwerk läuft, dann benötigen wir ein lokales RDBMS (oder ODBC-Datenbank) für die Datenbank Temporary und ein RDBMS auf dem entfernten Server mit der Datenbank für die Standardverbindung. Auf welchem der beiden Systemen die dritte Datenbank angesiedelt wird kann frei gewählt werden.

Entsprechend diesen Angaben müssen wir in `repository_database.xml` zwei oder drei `jdbc-connection-decriptoren` einrichten. Im der ausgelieferten Fassung sind

Defaulteinstellungen, die geändert werden können. Zu diesem Zweck entscheiden Sie, mit welchem RDBMS (oder bei verschiedenen: welchen RDMBS's) Sie arbeiten wollen. Aufgrund Ihrer Entscheidung kommentieren Sie die gewünschten `jdbc-connection-descriptoren` ein

oder aus. Das Auskommentieren erfolgt nach XML-Manier, d.h. auskommentierte Bereiche sind durch `<!-- -->` eingerahmt. Wichtig ist, dass die Standardverbindung die Eigenschaft `default-connection="true"` erhält, während die anderen beim gleichen Attribut den Wert `"false"` erhalten.

Die drei Verbindungsarten unterscheiden in folgenden Eigenschaften:

Standard:	<code>jcd-alias="standard"</code>	<code>default-connection="true"</code>
Temporary:	<code>jcd-alias="temporary"</code>	<code>default-connection="false"</code>
Schulung:	<code>jcd-alias="tutorial"</code>	<code>default-connection="false"</code>

Anhang

repository_user.xml

"repository_user.xml" enthält die Tabellendefinitionen der Datenbanken. Diese Tabellendefinitionen sind für die drei verwendeten Datenbanken identisch (nicht aber unbedingt die Daten, die sich in den Datenbanken befinden. "repository_user.xml" mappt die Java-Objekte der Kassenprogramme mit den entsprechenden Tabellen in den Datenbanken.

```
<!-- Tax beinhaltet die verschiedenen Kombinationen von TaxType (Umsatzsteuer, Vorsteuer) und TaxRate (Vollstaendiger Steuersatz, Reduzierter Steuersatz, Steuerfrei), die in colibri.xml definiert sind. -->
```

```
<class-descriptor
  class="ch.eugster.pos.db.Salespoint"
  table="pos_salespoint"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"
    primarykey="true"
    indexed="true"
    access="readwrite"
    autoincrement="true"
  />
  <field-descriptor
    name="timestamp"
    column="timestamp"
    jdbc-type="TIMESTAMP"
  />
  <field-descriptor
    name="deleted"
    column="deleted"
    jdbc-type="BIT"
  />
  <field-descriptor
    name="name"
    column="name"
    jdbc-type="VARCHAR"
    length="100"
  />
  <field-descriptor
    name="place"
    column="place"
    jdbc-type="VARCHAR"
    length="255"
  />
  <field-descriptor
    name="stock"
    column="stock"
```

```

        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="currentReceiptId"
        column="current_receipt_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="active"
        column="active"
        jdbc-type="BIT"
    />
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.User"
    table="pos_user"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="username"
        column="username"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="password"
        column="password"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="posLogin"
        column="pos_login"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="status"
        column="status"
        jdbc-type="INTEGER"
    />

    <collection-descriptor
        name="userAccesses"
        element-class-ref="ch.eugster.pos.db.UserAccess"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    />

```

```

    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="userId"/>
    </collection-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Access"
    table="pos_access"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="100"
    />

    <collection-descriptor
        name="userAccesses"
        element-class-ref="ch.eugster.pos.db.UserAccess"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="accessId"/>
    </collection-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.UserAccess"
    table="pos_user_access"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="userId"
        column="user_id"

```

```

        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="accessId"
        column="access_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="state"
        column="state"
        jdbc-type="BIT"
    />

    <reference-descriptor
        name="user"
        class-ref="ch.eugster.pos.db.User"
    >
        <foreignkey field-ref="userId"/>
    </reference-descriptor>

    <reference-descriptor
        name="access"
        class-ref="ch.eugster.pos.db.Access"
    >
        <foreignkey field-ref="accessId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Receipt"
    table="pos_receipt"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="number"
        column="number"
        indexed="true"
        jdbc-type="VARCHAR"
        length="25"
    />
    <field-descriptor
        name="salespointId"
        column="salespoint_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="userId"
        column="user_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="date"
        column="date"
        jdbc-type="DATE"
    />
    <field-descriptor
        name="time"

```

```

        column="time"
        jdbc-type="TIME"
    />
    <field-descriptor
        name="status"
        column="status"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="settlement"
        column="settlement"
        jdbc-type="VARCHAR"
        length="20"
    />
    <field-descriptor
        name="amount"
        column="amount"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="payment"
        column="payment"
        jdbc-type="DOUBLE"
    />

    <collection-descriptor
        name="positions"
        element-class-ref="ch.eugster.pos.db.Position"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="receiptId"/>
    </collection-descriptor>

    <collection-descriptor
        name="payments"
        element-class-ref="ch.eugster.pos.db.Payment"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="receiptId"/>
    </collection-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Position"
    table="pos_position"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="receiptId"
        column="receipt_id"
        jdbc-type="BIGINT"
    />

```

```

<field-descriptor
  name="productId"
  column="product_id"
  jdbc-type="VARCHAR"
  length="50"
/>
<field-descriptor
  name="productGroupId"
  column="product_group_id"
  jdbc-type="BIGINT"
/>
<field-descriptor
  name="currentTaxId"
  column="current_tax_id"
  jdbc-type="BIGINT"
/>
<field-descriptor
  name="quantity"
  column="quantity"
  jdbc-type="INTEGER"
/>
<field-descriptor
  name="price"
  column="price"
  jdbc-type="DOUBLE"
/>
<field-descriptor
  name="discount"
  column="discount"
  jdbc-type="DOUBLE"
/>
<field-descriptor
  name="expense"
  column="expense"
  jdbc-type="BIT"
/>
<field-descriptor
  name="optCode"
  column="opt_code"
  jdbc-type="CHAR"
  length="1"
/>
<field-descriptor
  name="author"
  column="author"
  jdbc-type="VARCHAR"
  length="100"
/>
<field-descriptor
  name="title"
  column="title"
  jdbc-type="VARCHAR"
  length="100"
/>
<field-descriptor
  name="publisher"
  column="publisher"
  jdbc-type="VARCHAR"
  length="100"
/>
<field-descriptor
  name="isbn"
  column="isbn"
  jdbc-type="VARCHAR"
  length="13"
/>

```

```

<field-descriptor
  name="bznr"
  column="bznr"
  jdbc-type="VARCHAR"
  length="10"
/>
<field-descriptor
  name="productNumber"
  column="product_number"
  jdbc-type="VARCHAR"
  length="100"
/>
<field-descriptor
  name="ordered"
  column="ordered"
  jdbc-type="BIT"
/>
<field-descriptor
  name="orderId"
  column="order_id"
  jdbc-type="VARCHAR"
  length="100"
/>

<reference-descriptor
  name="receipt"
  class-ref="ch.eugster.pos.db.Receipt"
>
  <foreignkey field-ref="receiptId"/>
</reference-descriptor>

<reference-descriptor
  name="productGroup"
  class-ref="ch.eugster.pos.db.ProductGroup"
>
  <foreignkey field-ref="productGroupId"/>
</reference-descriptor>

<reference-descriptor
  name="currentTax"
  class-ref="ch.eugster.pos.db.CurrentTax"
>
  <foreignkey field-ref="currentTaxId"/>
</reference-descriptor>
</class-descriptor>

<class-descriptor
  class="ch.eugster.pos.db.Payment"
  table="pos_payment"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"
    primarykey="true"
    indexed="true"
    access="readwrite"
    autoincrement="true"
  />
  <field-descriptor
    name="receiptId"
    column="receipt_id"
    jdbc-type="BIGINT"
  />
  <field-descriptor

```

```

        name="paymentTypeId"
        column="payment_type_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="quotation"
        column="quotation"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="amount"
        column="amount"
        jdbc-type="DOUBLE"
    />

    <reference-descriptor
        name="receipt"
        class-ref="ch.eugster.pos.db.Receipt"
    >
        <foreignkey field-ref="receiptId"/>
    </reference-descriptor>

    <reference-descriptor
        name="paymentType"
        class-ref="ch.eugster.pos.db.PaymentType"
    >
        <foreignkey field-ref="paymentTypeId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.PaymentTypeGroup"
    table="pos_payment_type_group"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="code"
        column="code"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="255"
    />

```



```

    <field-descriptor
      name="defaultQuotation"
      column="default_quotation"
      jdbc-type="DOUBLE"
    />
    <field-descriptor
      name="defaultAccount"
      column="default_account"
      jdbc-type="VARCHAR"
      length="100"
    />

    <collection-descriptor
      name="paymentTypes"
      element-class-ref="ch.eugster.pos.db.PaymentType"
      auto-retrieve="true"
      auto-update="true"
      auto-delete="true"
    >
      <orderby name="id" sort="ASC"/>
      <inverse-foreignkey field-ref="paymentTypeGroupId"/>
    </collection-descriptor>
  </class-descriptor>

  <class-descriptor
    class="ch.eugster.pos.db.PaymentType"
    table="pos_payment_type"
  >
    <field-descriptor
      name="id"
      column="id"
      jdbc-type="BIGINT"
      primarykey="true"
      indexed="true"
      access="readwrite"
      autoincrement="true"
    />
    <field-descriptor
      name="timestamp"
      column="timestamp"
      jdbc-type="TIMESTAMP"
    />
    <field-descriptor
      name="deleted"
      column="deleted"
      jdbc-type="BIT"
    />
    <field-descriptor
      name="paymentTypeGroupId"
      column="payment_type_group_id"
      jdbc-type="BIGINT"
    />
    <field-descriptor
      name="removeable"
      column="removeable"
      jdbc-type="BIT"
    />
    <field-descriptor
      name="name"
      column="name"
      jdbc-type="VARCHAR"
      length="255"
    />
    <field-descriptor
      name="shortName"

```

```

        column="short_name"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="currency"
        column="currency"
        jdbc-type="CHAR"
        length="3"
    />
    <field-descriptor
        name="quotation"
        column="quotation"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="roundFactor"
        column="round_factor"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="amount"
        column="amount"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="account"
        column="account"
        jdbc-type="VARCHAR"
        length="100"
    />

    <reference-descriptor
        name="paymentTypeGroup"
        class-ref="ch.eugster.pos.db.PaymentTypeGroup"
    >
        <foreignkey field-ref="paymentTypeId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.PaymentTypeCash"
    table="pos_payment_type"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="paymentTypeId"

```

```

        column="payment_type_group_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="removeable"
        column="removeable"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="shortName"
        column="short_name"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="currency"
        column="currency"
        jdbc-type="CHAR"
        length="3"
    />
    <field-descriptor
        name="quotation"
        column="quotation"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="roundFactor"
        column="round_factor"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="account"
        column="account"
        jdbc-type="VARCHAR"
        length="100"
    />

    <reference-descriptor
        name="paymentTypeGroup"
        class-ref="ch.eugster.pos.db.PaymentTypeGroup"
    >
        <foreignkey field-ref="paymentTypeId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.PaymentTypeBack"
    table="pos_payment_type"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor

```

```

        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="paymentTypeGroupId"
        column="payment_type_group_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="removeable"
        column="removeable"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="shortName"
        column="short_name"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="currency"
        column="currency"
        jdbc-type="CHAR"
        length="3"
    />
    <field-descriptor
        name="quotation"
        column="quotation"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="roundFactor"
        column="round_factor"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="account"
        column="account"
        jdbc-type="VARCHAR"
        length="100"
    />

    <reference-descriptor
        name="paymentTypeGroup"
        class-ref="ch.eugster.pos.db.PaymentTypeGroup"
    >
        <foreignkey field-ref="paymentTypeGroupId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.TaxRate"
    table="pos_tax_rate"

```

```

>
<field-descriptor
  name="id"
  column="id"
  jdbc-type="BIGINT"
  primarykey="true"
  indexed="true"
  access="readwrite"
  autoincrement="true"
/>
<field-descriptor
  name="timestamp"
  column="timestamp"
  jdbc-type="TIMESTAMP"
/>
<field-descriptor
  name="deleted"
  column="deleted"
  jdbc-type="BIT"
/>
<field-descriptor
  name="code"
  column="code"
  jdbc-type="CHAR"
  length="1"
  indexed="true"
/>
<field-descriptor
  name="name"
  column="name"
  jdbc-type="VARCHAR"
  length="100"
/>

<collection-descriptor
  name="taxes"
  element-class-ref="ch.eugster.pos.db.Tax"
  auto-retrieve="true"
  auto-update="true"
  auto-delete="true"
>
  <orderby name="id" sort="ASC"/>
  <inverse-foreignkey field-ref="taxRateId"/>
</collection-descriptor>
</class-descriptor>

<class-descriptor
  class="ch.eugster.pos.db.TaxType"
  table="pos_tax_type"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"
    primarykey="true"
    access="readwrite"
    autoincrement="true"
  />
  <field-descriptor
    name="timestamp"
    column="timestamp"
    jdbc-type="TIMESTAMP"
  />
  <field-descriptor
    name="deleted"

```

```

        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="code"
        column="code"
        jdbc-type="CHAR"
        length="1"
        indexed="true"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="100"
    />

    <collection-descriptor
        name="taxes"
        element-class-ref="ch.eugster.pos.db.Tax"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="taxTypeId"/>
    </collection-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Tax"
    table="pos_tax"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="currentTaxId"
        column="current_tax_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="taxTypeId"
        column="tax_type_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="taxRateId"
        column="tax_rate_id"

```

```

        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="code"
        column="code"
        jdbc-type="VARCHAR"
        length="20"
    />
    <field-descriptor
        name="galileoId"
        column="galileo_id"
        jdbc-type="CHAR"
        length="3"
    />
    <field-descriptor
        name="fibuId"
        column="fibu_id"
        jdbc-type="VARCHAR"
        length="10"
    />
    <field-descriptor
        name="account"
        column="account"
        jdbc-type="VARCHAR"
        length="20"
    />

    <reference-descriptor
        name="taxType"
        class-ref="ch.eugster.pos.db.TaxType"
    >
        <foreignkey field-ref="taxTypeId"/>
    </reference-descriptor>

    <reference-descriptor
        name="taxRate"
        class-ref="ch.eugster.pos.db.TaxRate"
    >
        <foreignkey field-ref="taxRateId"/>
    </reference-descriptor>

    <collection-descriptor
        name="currentTaxes"
        element-class-ref="ch.eugster.pos.db.CurrentTax"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="taxId"/>
    </collection-descriptor>

    <collection-descriptor
        name="productGroups"
        element-class-ref="ch.eugster.pos.db.ProductGroup"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="defaultTaxId"/>
    </collection-descriptor>
</class-descriptor>

<!-- CurrentTax -->

```

```

<class-descriptor
  class="ch.eugster.pos.db.CurrentTax"
  table="pos_current_tax"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"
    primarykey="true"
    indexed="true"
    access="readwrite"
    autoincrement="true"
  />
  <field-descriptor
    name="timestamp"
    column="timestamp"
    jdbc-type="TIMESTAMP"
  />
  <field-descriptor
    name="deleted"
    column="deleted"
    jdbc-type="BIT"
  />
  <field-descriptor
    name="taxId"
    column="tax_id"
    jdbc-type="BIGINT"
  />
  <field-descriptor
    name="percentage"
    column="percentage"
    jdbc-type="DOUBLE"
  />
  <field-descriptor
    name="validationDate"
    column="validation_date"
    jdbc-type="DATE"
  />

  <reference-descriptor
    name="tax"
    class-ref="ch.eugster.pos.db.Tax"
  >
    <foreignkey field-ref="taxId"/>
  </reference-descriptor>
</class-descriptor>

<class-descriptor
  class="ch.eugster.pos.db.ProductGroup"
  table="pos_product_group"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"
    primarykey="true"
    indexed="true"
    access="readwrite"
    autoincrement="true"
  />
  <field-descriptor
    name="timestamp"
    column="timestamp"
    jdbc-type="TIMESTAMP"
  />

```



```

<field-descriptor
  name="deleted"
  column="deleted"
  jdbc-type="BIT"
/>
<field-descriptor
  name="galileoId"
  column="galileo_id"
  jdbc-type="VARCHAR"
  length="3"
/>
<field-descriptor
  name="defaultTaxId"
  column="default_tax_id"
  jdbc-type="BIGINT"
/>
<field-descriptor
  name="shortname"
  column="shortname"
  jdbc-type="VARCHAR"
  length="50"
/>
<field-descriptor
  name="name"
  column="name"
  jdbc-type="VARCHAR"
  length="255"
/>
<field-descriptor
  name="quantity"
  column="quantity"
  jdbc-type="INTEGER"
/>
<field-descriptor
  name="price"
  column="price"
  jdbc-type="DOUBLE"
/>
<field-descriptor
  name="optCode"
  column="opt_code"
  jdbc-type="VARCHAR"
  length="1"
/>
<field-descriptor
  name="account"
  column="account"
  jdbc-type="VARCHAR"
  length="100"
/>
<field-descriptor
  name="isExpense"
  column="is_expense"
  jdbc-type="BIT"
/>

<reference-descriptor
  name="defaultTax"
  class-ref="ch.eugster.pos.db.Tax"
>
  <foreignkey field-ref="defaultTaxId"/>
</reference-descriptor>
</class-descriptor>
<class-descriptor

```

```

class="ch.eugster.pos.db.Block"
table="pos_block"
>
<field-descriptor
  name="id"
  column="id"
  jdbc-type="BIGINT"
  primarykey="true"
  indexed="true"
  access="readwrite"
  autoincrement="true"
/>
<field-descriptor
  name="timestamp"
  column="timestamp"
  jdbc-type="TIMESTAMP"
/>
<field-descriptor
  name="visible"
  column="visible"
  jdbc-type="BIT"
/>
<field-descriptor
  name="name"
  column="name"
  jdbc-type="VARCHAR"
  length="100"
/>
<field-descriptor
  name="clazz"
  column="class"
  jdbc-type="VARCHAR"
  length="255"
/>
<field-descriptor
  name="fontSize"
  column="font_size"
  jdbc-type="DOUBLE"
/>
<field-descriptor
  name="fontStyle"
  column="font_style"
  jdbc-type="INTEGER"
/>

<collection-descriptor
  name="tabs"
  element-class-ref="ch.eugster.pos.db.Tab"
  auto-retrieve="true"
  auto-update="true"
  auto-delete="true"
>
  <orderby name="id" sort="ASC"/>
  <inverse-foreignkey field-ref="blockId"/>
</collection-descriptor>
</class-descriptor>

<class-descriptor
  class="ch.eugster.pos.db.Tab"
  table="pos_tab"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"

```

```

        primaryKey="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="blockId"
        column="block_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="defaultTab"
        column="default_tab"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="order"
        column="sequence"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="rows"
        column="rowcount"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="columns"
        column="columncount"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="visible"
        column="visible"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="fontSize"
        column="font_size"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="fontStyle"
        column="font_style"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="title"
        column="title"
        jdbc-type="VARCHAR"
        length="100"
    />

    <reference-descriptor
        name="block"
        class-ref="ch.eugster.pos.db.Block"
    >
        <foreignkey field-ref="blockId"/>
    </reference-descriptor>

    <collection-descriptor
        name="keys"
        element-class-ref="ch.eugster.pos.db.CustomKey"

```

```

        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="tabId"/>
    </collection-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.CustomKey"
    table="pos_key"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="tabId"
        column="tab_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="textEditable"
        column="editable"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="row"
        column="row"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="column"
        column="col"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgRed"
        column="bg_red"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgGreen"
        column="bg_green"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgBlue"
        column="bg_blue"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgRed2"
        column="bg_red2"

```

```

        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgGreen2"
        column="bg_green2"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgBlue2"
        column="bg_blue2"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fgRed"
        column="fg_red"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fgGreen"
        column="fg_green"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fgBlue"
        column="fg_blue"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fontSize"
        column="font_size"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="fontStyle"
        column="font_style"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="text"
        column="name"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="command"
        column="command"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="imagepath"
        column="image_path"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="relHorizontalTextPos"
        column="rel_horizontal_text_pos"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="relVerticalTextPos"
        column="rel_vertical_text_pos"
        jdbc-type="INTEGER"
    />
    <field-descriptor

```

```

        name="className"
        column="class_name"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="parentClassName"
        column="parent_class_name"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="parentId"
        column="parent_id"
        jdbc-type="BIGINT"
    />
    <field-descriptor
        name="actionType"
        column="action_type"
        jdbc-type="INTEGER"
    />

    <reference-descriptor
        name="tab"
        class-ref="ch.eugster.pos.db.Tab"
    >
        <foreignkey field-ref="tabId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.FixKey"
    table="pos_fixkey"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="fixKeyId"
        column="fix_key_group_id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="block"
        column="block"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="textEditable"

```

```

        column="text_editable"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="row"
        column="row"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="column"
        column="col"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgRed"
        column="bg_red"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgGreen"
        column="bg_green"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="bgBlue"
        column="bg_blue"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fgRed"
        column="fg_red"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fgGreen"
        column="fg_green"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fgBlue"
        column="fg_blue"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="fontSize"
        column="font_size"
        jdbc-type="DOUBLE"
    />
    <field-descriptor
        name="fontStyle"
        column="font_style"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="imagepath"
        column="image_path"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="relHorizontalTextPos"
        column="rel_horizontal_text_pos"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="relVerticalTextPos"

```

```

        column="rel_vertical_text_pos"
        jdbc-type="INTEGER"
    />
    <field-descriptor
        name="text"
        column="name"
        jdbc-type="VARCHAR"
        length="100"
    />
    <field-descriptor
        name="command"
        column="command"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="className"
        column="class_name"
        jdbc-type="VARCHAR"
        length="255"
    />
    <field-descriptor
        name="actionType"
        column="action_type"
        jdbc-type="INTEGER"
    />

    <reference-descriptor
        name="fixKeyGroup"
        class-ref="ch.eugster.pos.db.FixKeyGroup"
    >
        <foreignkey field-ref="fixKeyGroupId"/>
    </reference-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Option"
    table="pos_option"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="code"
        column="code"
        jdbc-type="VARCHAR"
        length="1"
    />
    <field-descriptor
        name="name"

```



```

        column="name"
        jdbc-type="VARCHAR"
        length="100"
    />
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Currency"
    table="pos_currency"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="deleted"
        column="deleted"
        jdbc-type="BIT"
    />
    <field-descriptor
        name="code"
        column="code"
        jdbc-type="VARCHAR"
        length="3"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="20"
    />
    <field-descriptor
        name="region"
        column="region"
        jdbc-type="VARCHAR"
        length="100"
    />
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Function"
    table="pos_function"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />

```

```

/>
<field-descriptor
  name="clazz"
  column="class"
  jdbc-type="VARCHAR"
  length="255"
/>
<field-descriptor
  name="actionType"
  column="action_type"
  jdbc-type="INTEGER"
/>
<field-descriptor
  name="name"
  column="name"
  jdbc-type="VARCHAR"
  length="255"
  indexed= "true"
/>
<field-descriptor
  name="shortName"
  column="short_name"
  jdbc-type="VARCHAR"
  length="255"
/>
</class-descriptor>

<class-descriptor
  class="ch.eugster.pos.db.KeyGroup"
  table="pos_key_group"
>
  <field-descriptor
    name="id"
    column="id"
    jdbc-type="BIGINT"
    primarykey="true"
    indexed="true"
    access="readwrite"
    autoincrement="true"
  />
  <field-descriptor
    name="timestamp"
    column="timestamp"
    jdbc-type="TIMESTAMP"
  />
  <field-descriptor
    name="name"
    column="name"
    jdbc-type="VARCHAR"
    length="255"
    indexed= "true"
  />
  <field-descriptor
    name="clazz"
    column="class"
    jdbc-type="VARCHAR"
    length="255"
  />
  <field-descriptor
    name="actionClass"
    column="action_class"
    jdbc-type="VARCHAR"
    length="255"
  />
  <field-descriptor
    name="actionType"

```

```

        column="action_type"
        jdbc-type="INTEGER"
    />
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.FixKeyGroup"
    table="pos_fixkey_group"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="timestamp"
        column="timestamp"
        jdbc-type="TIMESTAMP"
    />
    <field-descriptor
        name="name"
        column="name"
        jdbc-type="VARCHAR"
        length="255"
    />

    <collection-descriptor
        name="fixKeys"
        element-class-ref="ch.eugster.pos.db.FixKey"
        auto-retrieve="true"
        auto-update="true"
        auto-delete="true"
    >
        <orderby name="id" sort="ASC"/>
        <inverse-foreignkey field-ref="fixKeyGroupId"/>
    </collection-descriptor>
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Coin"
    table="pos_coin"
>
    <field-descriptor
        name="id"
        column="id"
        jdbc-type="BIGINT"
        primarykey="true"
        indexed="true"
        access="readwrite"
        autoincrement="true"
    />
    <field-descriptor
        name="value"
        column="value"
        jdbc-type="DOUBLE"
        indexed="true"
    />
</class-descriptor>

<class-descriptor
    class="ch.eugster.pos.db.Version"
    table="pos_version"

```

```
>
  <field-descriptor
    name="data"
    column="data_version"
    jdbc-type="INTEGER"
    primarykey="true"
    access="readwrite"
    autoincrement="false"
  />
  <field-descriptor
    name="data"
    column="data_version"
    jdbc-type="INTEGER"
  />
</class-descriptor>
```