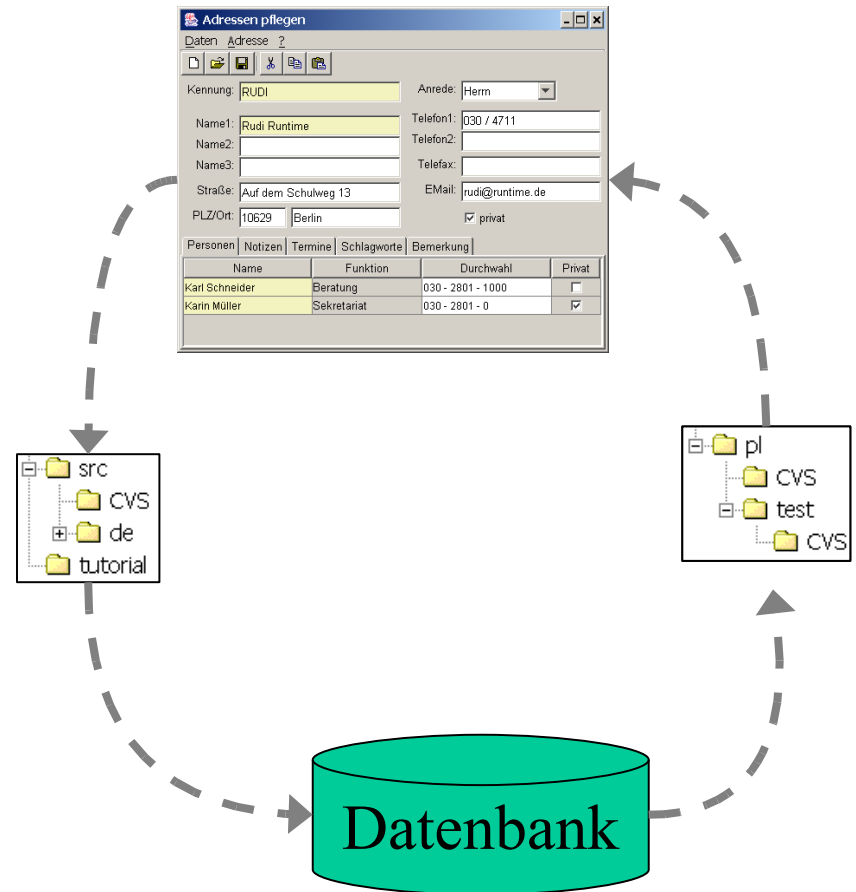


JDataSet & Persistenz Layer

Generisches Transfermodel für
Datenaustausch zwischen Datenbank
und Oberfläche

Problemstellung / Anforderungen

- Informationsobjekte sollen aus der Datenbank entnommen,
- an die Oberfläche übergeben
- und mit den dort vorgenommenen Änderungen
- zurück in die Datenbank geschrieben werden.
- Diese Steuerung soll – ähnlich wie beim GuiBuilder – über XML-Dokumente erfolgen.



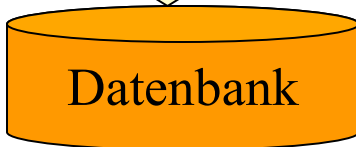
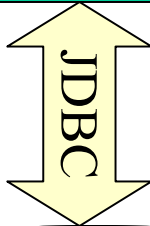
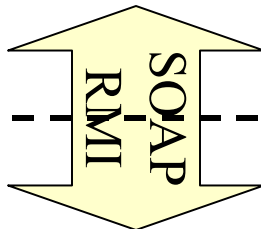
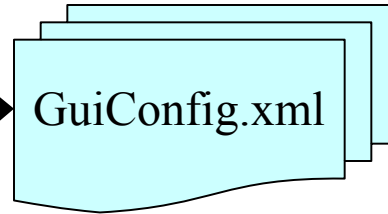
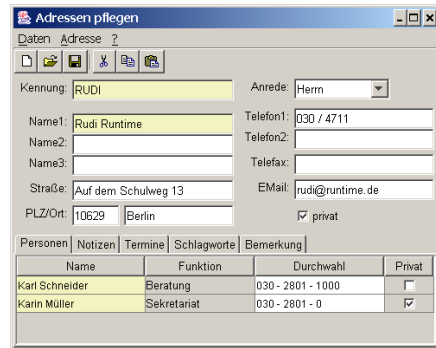
Use Case „Daten lesen / ändern / speichern“

- Im Zusammenhang mit einem **Geschäftsvorfall** fordert der Benutzer eine bestimmte Menge von Informationen aus der Datenbank an (vermittelt über einen Persistenz-Layer). Ggf. werden hierbei **Suchbegriffe** mit übergeben.
- Die Daten werden aus der Datenbank gemäß einer Vorschrift entnommen und zu einem **generischen Dataset** zusammengestellt.
- Der Dataset wird über das Netzwerk an die Oberfläche übertragen.
- Es findet ein **Binding** zwischen den Elementen des Dataset und den Widgets der Oberfläche statt.
- Der Benutzer kann über die Oberfläche den Dataset editieren (ändern, ergänzen, löschen).
- Der Dataset wird über das Netz zurück an der Persistenz-Layer übertragen.
- Gemäß den Benutzereingaben werden die Daten (im Rahmen einer **Transaktion**) in der Datenbank geändert, gelöscht, eingefügt.

Spezielle Features / Implementierungsideen

- Ein Dataset ist ein generisches Transfermodell.
- Ein Dataset hält Informationen in einer baumartigen Struktur.
- Die Regeln, nach denen die Daten zu einem Dataset zusammengestellt werden, sind nicht hart codiert, sondern werden (in XML) definiert.
- Der Persistenz-Layer erstellt anhand dieser Regeln wiederverwendbare Prepared Statements (JDBC).
- Fordert der Benutzer Daten an, wird anhand dieser Regeln ein Dataset zusammengestellt und in einem Rutsch (serialisiert) übertragen.
- Optimistisches Locking wird automatisch berücksichtigt.
- Welcher Datensatz wann erzeugt/geändert wurde, wird automatisch protokolliert.

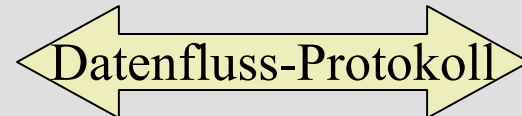
Architektur / Schichtenmodell



Client
Server

Legende

Abhängigkeit;
ist definiert durch



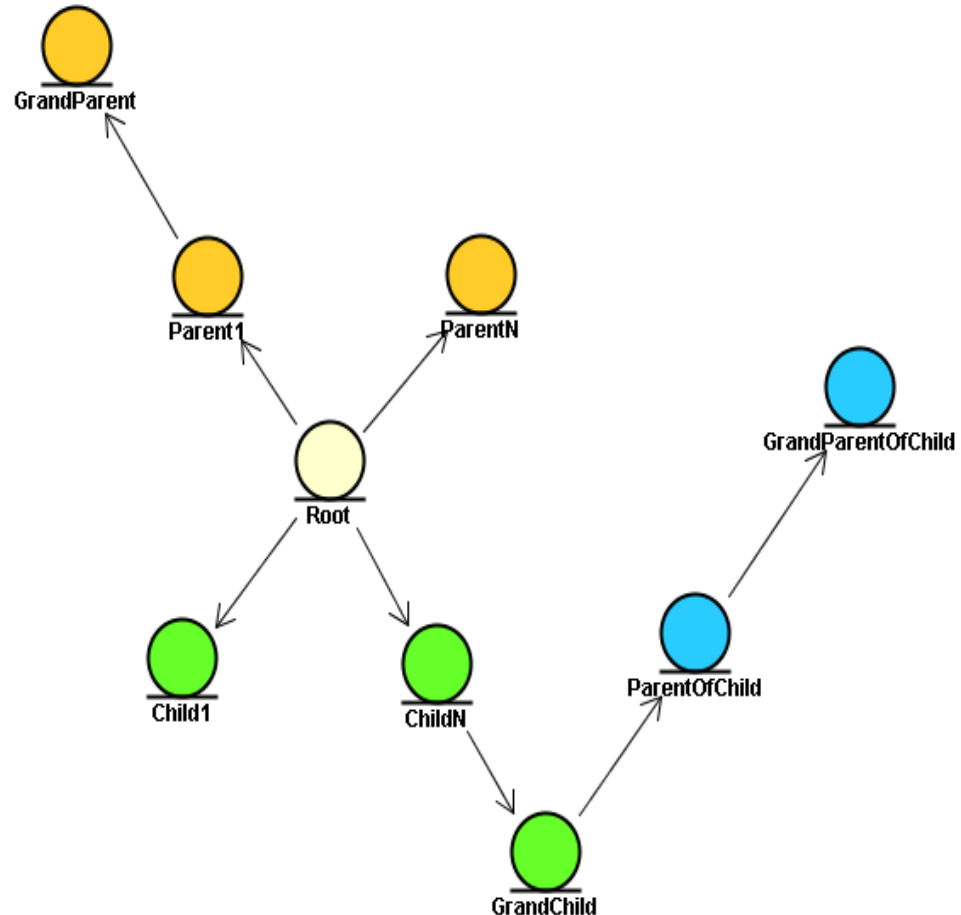
Definition Persistenz-Layer/Datenbank

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!DOCTYPE Server SYSTEM 'http://jdataset.de/PLConfig.dtd'>
<Server>
<Database name='AdrsDemo' enabled='true'>
  <JDBC-Driver>org.postgresql.Driver</JDBC-Driver>
  <URL>jdbc:postgresql://localhost/adrsdemo</URL>
  <Schema>public</Schema>
  <Username>peter</Username>
  <Password>geheim</Password>
  <!-- Definition Connection Pool (DBCP) -->
  <MaxActiveConnections value="30"/>
  <MaxIdleConnections value="15"/>
  <MinIdleConnections value="1"/>
  <!-- 300 000 = 5 Minuten -->
  <ConnectionTimeout value="300000"/>
  <!-- Ein einfaches Statement zur Überprüfung der Datenbankverbindung -->
  <ValidationQuery>"SELECT 1"</ValidationQuery>
  <!-- Default-Sequence für Primary Keys -->
  <Sequence>
    <get>select nextval('counter')</get>
  </Sequence>
  <!-- Feldname für optimistisches Locking -->
  <OptimisticLockingField value='version'/>
  <!-- Feldname für User und Datum INSERT -->
  <CreateUserField value="Created"/>
  <!-- Feldname für User und Datum letztes UPDATE -->
  <UpdateUserField value="Updated"/>
</Database>
```

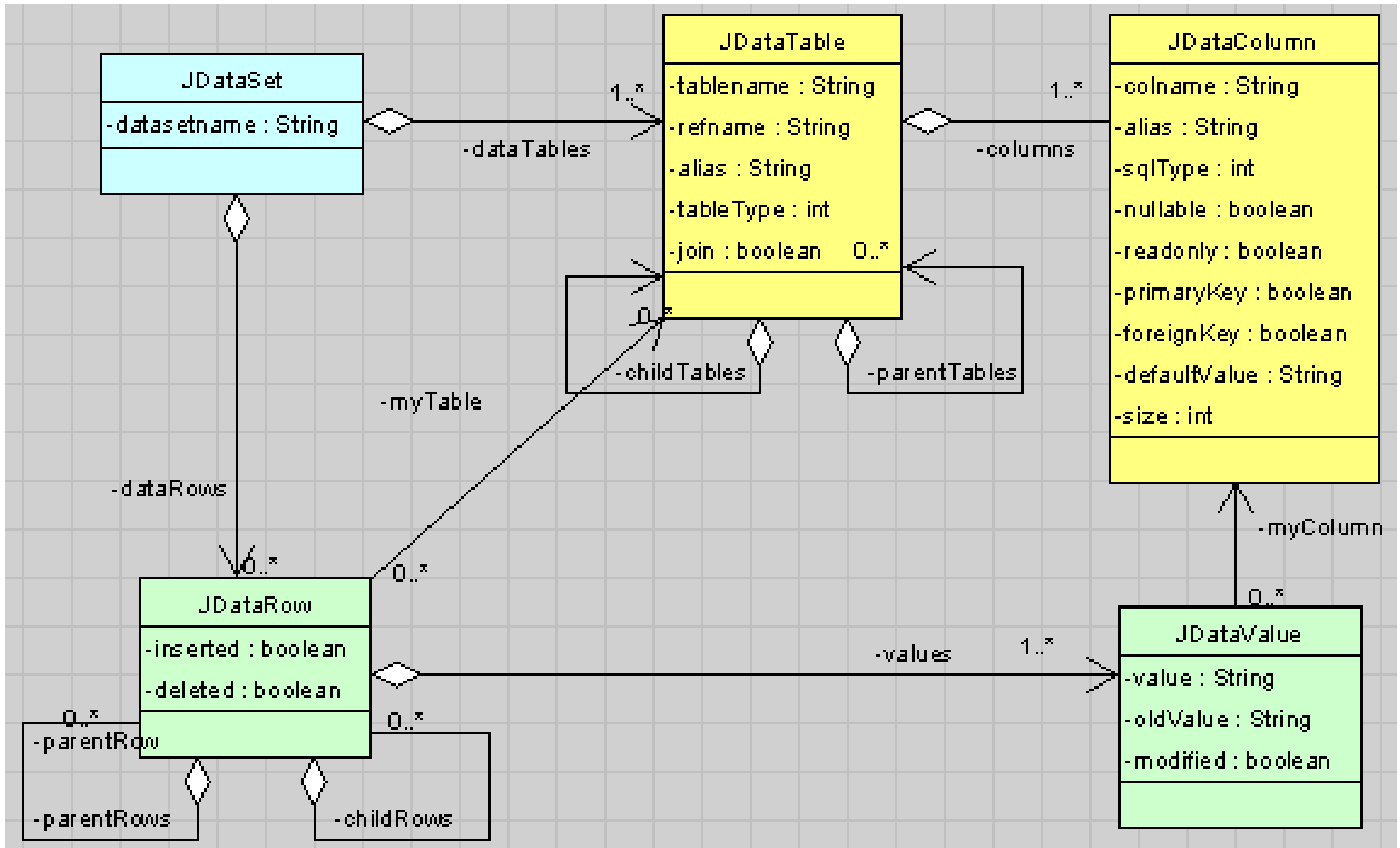
Struktur eines Dataset

Ein Dataset muss nach bestimmten Regeln aus den Tabellen einer Datenbank zusammengestellt werden:

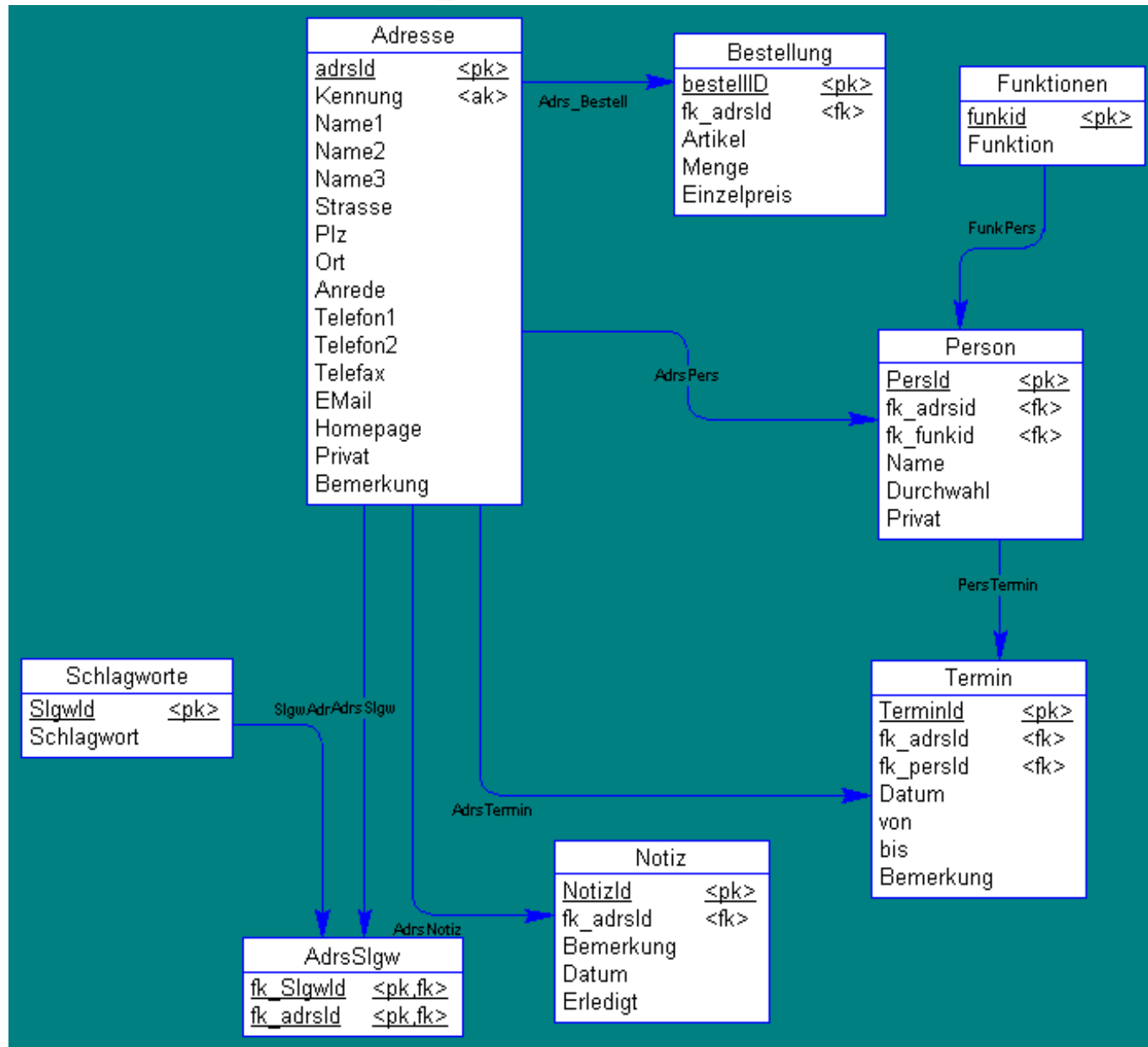
2. Es muss mindestens eine „Root-Table“ geben.
3. Vor hier aus können deren übergeordnete „Parent-Tables“ und deren Parents („Grandparent“ usw.) hinzugefügt werden.
4. Ausgehend von „Root“ können dessen „Child-Tables“ und dessen Childs („Enkel“ usw.) verwendet werden.
5. Es darf *nicht* von einem Parent zu einem Child navigiert werden!



Dataset - Objektmodell



Beispiel-Datenbank



Beispiel-Request

```
<View name='AdresseKomplett'>
  <RootTable tablename='adresse' pk='adrsid'>
    <Column name='*' />
    <Child tablename='person' pk='persid' fk='fk_adrsid' orderby='name'>
      <Column name='*' />
      <Child tablename='termin' pk='terminid' fk='fk_persid' orderby='datum,von'>
        <Column name='*' />
      </Child>
      <Parent tablename='funktionen' pk='funkid' fk='fk_funkid' join='true'>
        <Column name='funktion' alias='FunktionsBezeichnung' />
      </Parent>
    </Child>
    <Child tablename='termin' pk='terminid' fk='fk_adrsid' orderby='datum,von'>
      <Column name='*' />
    </Child>
    <Child tablename='notiz' pk='notizid' fk='fk_adrsid'>
      <Column name='*' />
    </Child>
    <Child tablename='dokument' pk='docid' fk='fk_adrsid'>
      <Column name='*' />
    </Child>
    <Child tablename='adrsslwgw' pk='fk_adrsid, fk_slgwid' fk='fk_adrsid' alias='AdressSchlagworte'>
      <Column name='*' />
      <Parent tablename='schlagworte' pk='slgwid' fk='fk_slgwid' join='true'>
        <Column name='schlagwort' />
      </Parent>
    </Child>
  </RootTable>
</View>
```

Inhalt Dataset - Schema

```
<JDataSet name='AdresseEinzeln'>
  <Schema>
    <adresse tablename='adresse' tabletype='root'>
      <Column name='adrsid' type='4' pk='true' notnull='true' />
      <Column name='kennung' type='12' notnull='true' />
      <Column name='name1' type='12' notnull='true' />
      <Column name='name2' type='12' />
      <Column name='name3' type='12' />
      <Column name='strasse' type='12' />
      <Column name='plz' type='12' />
      <Column name='ort' type='12' />
      <Column name='anrede' type='12' />
      <Column name='telefon1' type='12' />
      <Column name='telefon2' type='12' />
      <Column name='telefax' type='12' />
      <Column name='email' type='12' />
      <Column name='homepage' type='12' />
      <Column name='privat' type='-6' notnull='true' default='0' />
      <Column name='bemerkung' type='-1' />
      <Column name='version' type='4' />
    <person tablename='person' tabletype='child' element='personen' fk='fk_adrsid'>
      <Column name='persid' type='4' pk='true' notnull='true' />
      <Column name='fk_adrsid' type='4' notnull='true' fk='true' />
      <Column name='fk_funkid' type='4' />
      <Column name='name' type='12' notnull='true' />
      <Column name='durchwahl' type='12' />
      <Column name='privat' type='-6' notnull='true' default='0' />
    </person>
    <termin tablename='termin' tabletype='child' fk='fk_adrsid'>
      <Column name='terminid' type='4' pk='true' notnull='true' />
      <Column name='fk_adrsid' type='4' fk='true' />
      <Column name='fk_persid' type='4' />
      <Column name='datum' type='91' notnull='true' />
      <Column name='von' type='92' />
      <Column name='bis' type='92' />
      <Column name='bemerkung' type='-1' />
    </termin>
  </Schema>
</JDataSet>
```

Inhalt Dataset - Data

```
<Data>
  <adresse rowtype='root'>
    <adrsid>1</adrsid>
    <kennung>RUDI</kennung>
    <name1>Rudi Müller</name1>
    <name2/>
    <name3></name3>
    <strasse>Milchstraße 13</strasse>
    <plz>10969</plz>
    <ort>Berlin-Mitte</ort>
    <anrede/>
    <telefon1/>
    <telefon2>0172 / 47 11</telefon2>
    <telefax/>
    <email/>
    <homepage/>
    <privat>0</privat>
    <bemerkung>Hallo Rudi!</bemerkung>
    <version>34</version>
    <personen rowtype='node'>
      <person rowtype='child'>
        <persid>2</persid>
        <fk_adrsid>1</fk_adrsid>
        <fk_funkid>4</fk_funkid>
        <name>Johanna</name>
        <durchwahl>444</durchwahl>
        <privat>0</privat>
      </person>
    </personen>
    <personen rowtype='node'>
      <person rowtype='child'>
```

GuiBuilder – Binding des Dataset

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!DOCTYPE GDL SYSTEM 'http://guibuilder.de/gdl.dtd'>
<GDL>
  <Form label='Adressen pflegen' root-element="Adresse" size="500,450"
    name='AdressBeispiel' helpID="adresse.main" OnClose="exit">
    ...
    <Panel label='links' wy='0' wx='2' name='panelLinks' eol='false'>
      <Hidden name="adrsId" element="@adrsId" />
      <Text label='Kennung:' nn='true' element="@kennung" helpID="kennung"/>
      <Text label='Name1:' it='15' nn='true' element="@name1" helpID="name1"/>
      <Text label='Name2:' element="@name2" it='0' />
      <Text label='Name3:' element="@name3" it='0' />
      <Text label='Straße:' element="@strasse"/>
      <Text label='PLZ/Ort:' eol='false' cols='5' name='plz' element="@plz"/>
      <Text w='2' name='ort' element="@ort"/>
    </Panel>
    <Panel label='rechts' wy='0' name='panelRechts'>
      <Combo label='Anrede' element="@anrede" items='Frau|Herrn|Firma|Rechtsanwälte' w='1' wx='0' />
      <Text label='Telefon1:' it='10' element="@telefon1"/>
      <Text label='Telefon2:' it='0' element="@telefon2"/>
      <Text label='Telefax:' element="@telefax"/>
      <Text label='EMail:' element="@email"/>
      <Check label='privat' x='1' val='true' />
    </Panel>
    <Tabset layout='SCROLL'>
      <Tab label='%Personen' name='tabPersonen' layout='BORDER' OnActive="personenActive">
        <Table name='tblPersonen' element=".person">
          <Hidden name='persId' element="@persId"/>
          <Text label='Name' name='persName' nn='true' element="@name"/>
          <Combo label='Funktion' w='150' element="@fk_funkid"
            items='Beratung|Vertrieb|Geschäftsführung|Sekretariat' />
          <Text label='Durchwahl' w='120' maxLen="10" element="@durchwahl" />
          <Check label='Privat' element="@privat" tt='Kennzeichen, ob die Person privat ist' />
          <Popup label='Person'>
            <Item label='Person einfügen' file='InsertRow()' />
            <Item label='Person löschen' file='DeleteRow()' />
          </Popup>
        </Table>
      </Tab>
      <Tab label='%Notizen' name='tabNotizen' layout='BORDER' OnActive="notizenActive">
```

Attribute für
Binding:
root-element
element
dataset
displayMember
valueMember

Binding einer Tabelle
an eine „Child-Table“

Binding einer
ComboBox

Notation:
.ChildTable
#ParentTable
@Column

Zusammenspiel der Komponenten mit BeanShell #1

```
import de.pkjs.pl.*; // Persistenz-Layer
import de.jdataset.*; // DataSet
// Zugriff auf Persistenz-Layer verschaffen:
PL pl = GuiSession.getInstance().getAdapter().getDatabase();
// Daten lesen
read(event) {
    // Die Adresse mit dem Primärschlüssel „1“ wird gelesen...
    JSataSet ds = pl.getDataset("AdresseEinzel, 1);
    // ...und wir weisen dieses der Oberfläche zu.
    event.window.setDatasetValues(ds);
}
```

Zusammenspiel der Komponenten mit BeanShell #2

```
// Speichern der vom Benutzer geänderten Daten
save(event) {
    // Daten aus der Oberfläche abholen
    JDataSet ds = event.window.getDatasetValues();
    // Nur speichern wenn die Daten auch geändert wurden.
    if ( ds.hasChanges() ) {
        // Änderungen aus dem Dataset abholen...
        JDataSet dsChanges = ds.getChanges();
        // ...und die Änderungen an den Persistenzlayer übergeben
        pl.setDataset(dsChanges);
        // Alles was als geändert, ergänzt oder gelöscht
        // gekennzeichnet wurde wieder zurücksetzen.
        ds.commitChanges();
    }
}
```

Was fehlt ...

- HAVING, IN; JOIN
- Binding an TreeView