

09. – 12.12.2019  
Frankfurt am Main



Vladimir Poliakov

# Datenmodellierung mit dem OSDDM

#ittage

TeamBank AG

# Datenmodellierung mit dem OSDDM IT-Tage 2019



**Autor: Vladimir Poliakov**  
Zur externen Verwendung  
Nürnberg, 03.11.2019

# Agenda

1. Über mich
2. OSDDM Einführung
3. Skripting in OSDDM
4. Beispiele
5. Eigene Skripte
6. Ausgaben von Texten und Debug
7. Typische Use Cases
8. Fragen
9. Zusammenfassung
10. Quellen, Kontakt

# Über mich

Vladimir Poliakov – Data Engineer bei der TeamBank AG

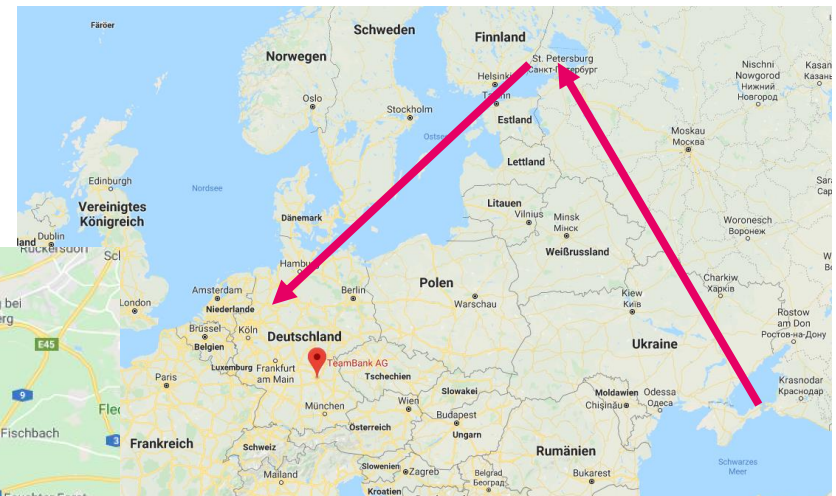


1. Geboren in der ehemaligen UdSSR, seit 1996 in Deutschland
2. Studium der Ozeanographie an der Russian State Hydrometeorological University (RSHU) in St.Petersburg
3. Entwickler → DBA → System Architekt → BI (BigData) Spezialist → Data Engineer

**ORACLE®**  
Certified Professional  
Oracle Database 12c  
Administrator

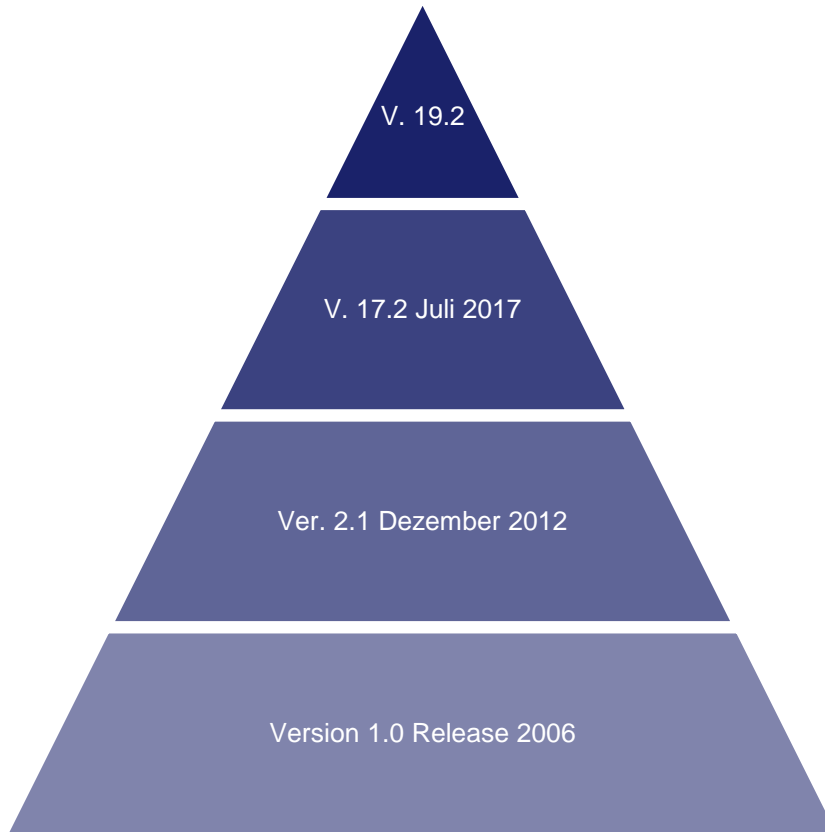
**ORACLE®**  
Certified Professional  
Oracle Database 10g  
Administrator

**ORACLE®**  
Certified Expert  
Oracle Database 10g  
Managing Oracle on Linux



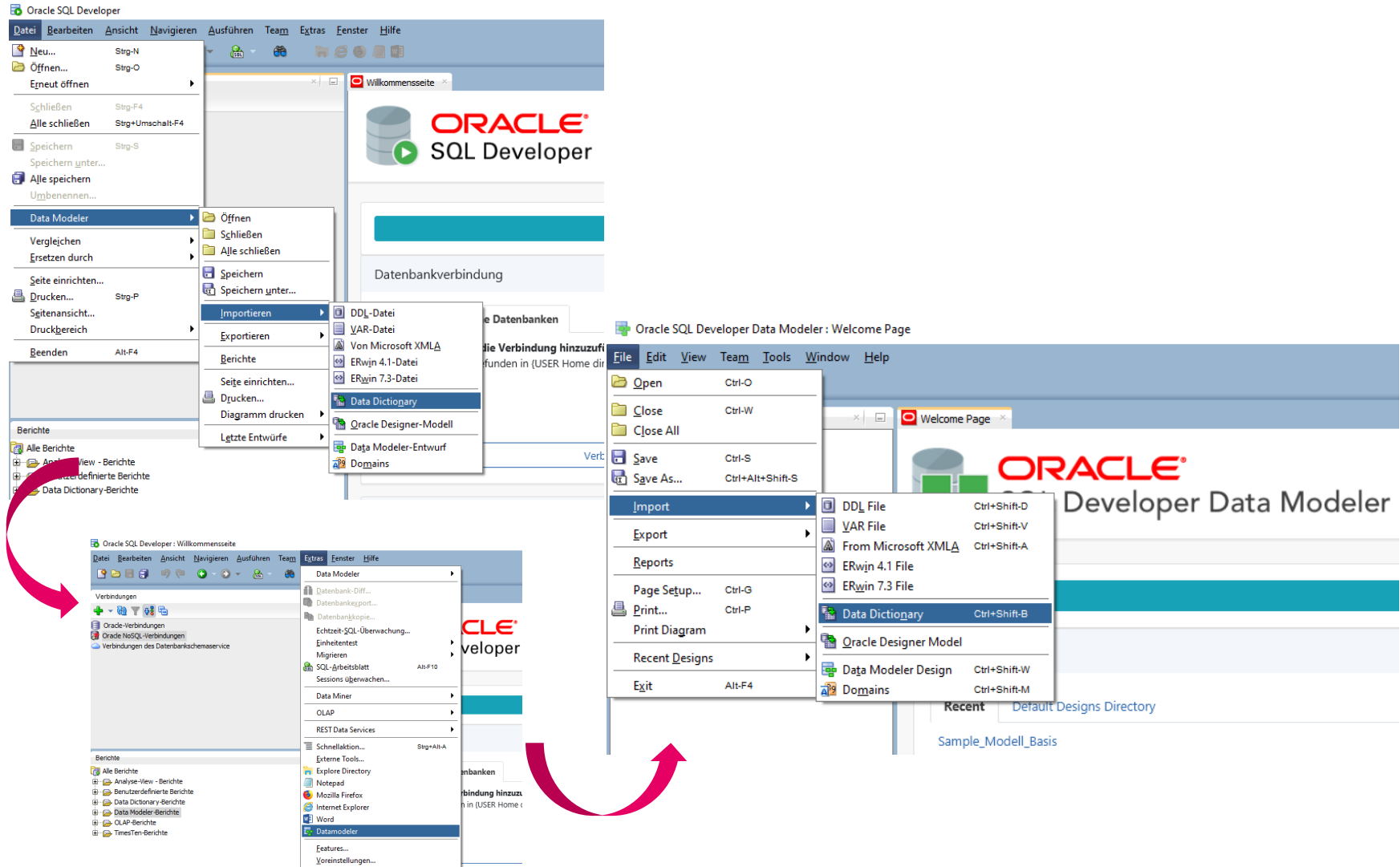
# Warum OSDDM?

Plug-In in SQL Developer oder Stand-Alone



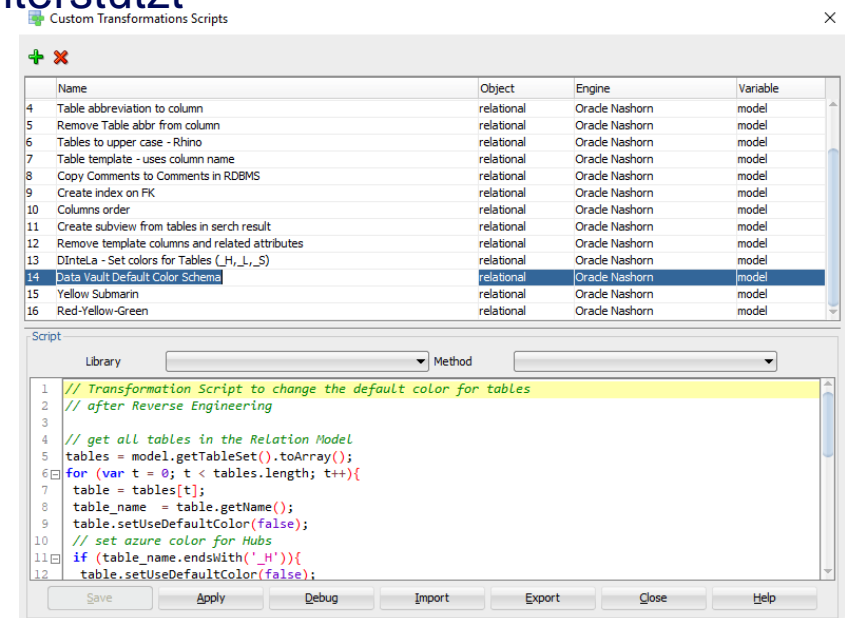
**Evolution von Oracle SQL Developer**

# OSDDM Plug-In und Stand-Alone



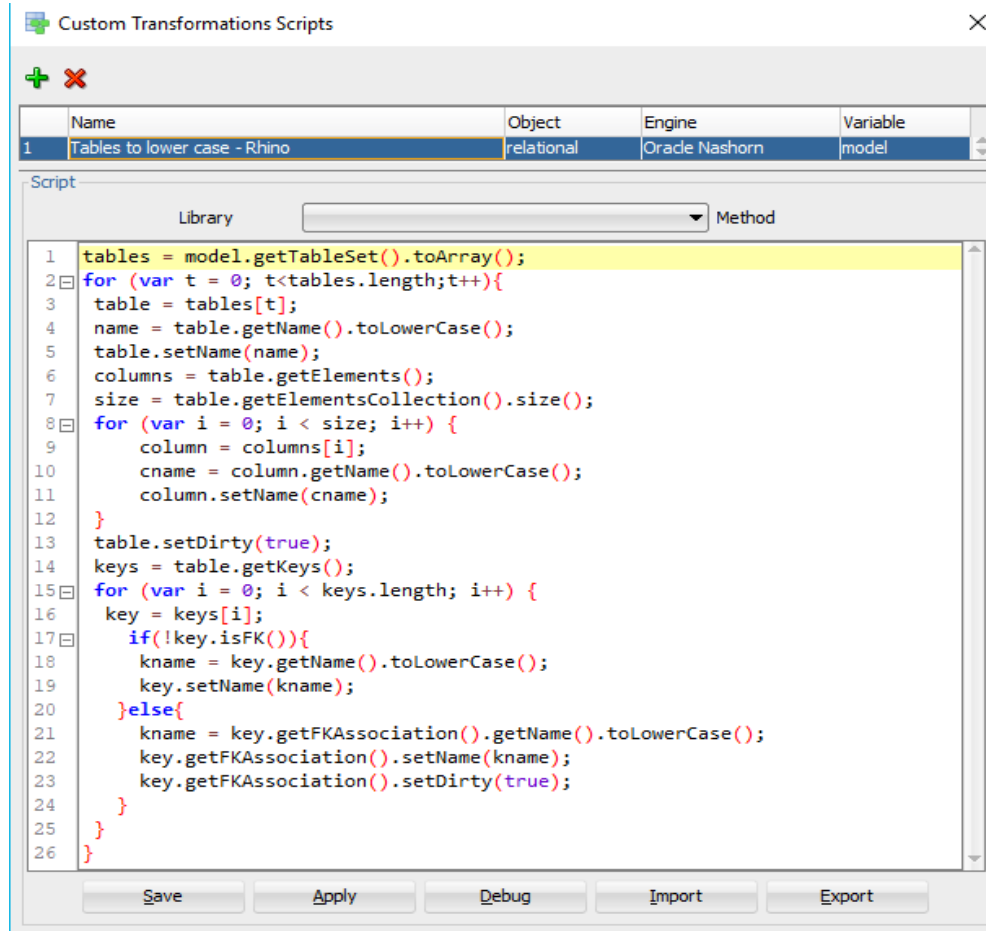
# Die Grundlagen der Skripting in OSDDM

1. Java Specification Requests (JSR) 223: Scripting for the Java™ Platform  
<https://www.jcp.org/en/jsr/detail?id=223>
2. Java Platform, Standard Edition Java Scripting Programmer's Guide von Oracle  
<https://docs.oracle.com/javase/10/scripting/toc.htm>
3. Seit JDK 6 wird bereits die JavaScript unterstützt
4. Mitgelieferte Beispiele in OSDDM
5. Eigene Skripte



# Ein Beispiel aus der Box

Tables to lower / to upper case mit Oracle Nashorn Engine



## Ein wenig Details:

- Das Datenmodell wird über die Variable `model` angesprochen
- Es wird über alle Tabellen im Modell iteriert
- Außerdem werden die Spalten und die Constraints jeder Tabelle modifiziert
- Wichtig ist die Tabelle zum Speichern markieren:  
`table.setDirty(true);`



# Meine erste Transformation

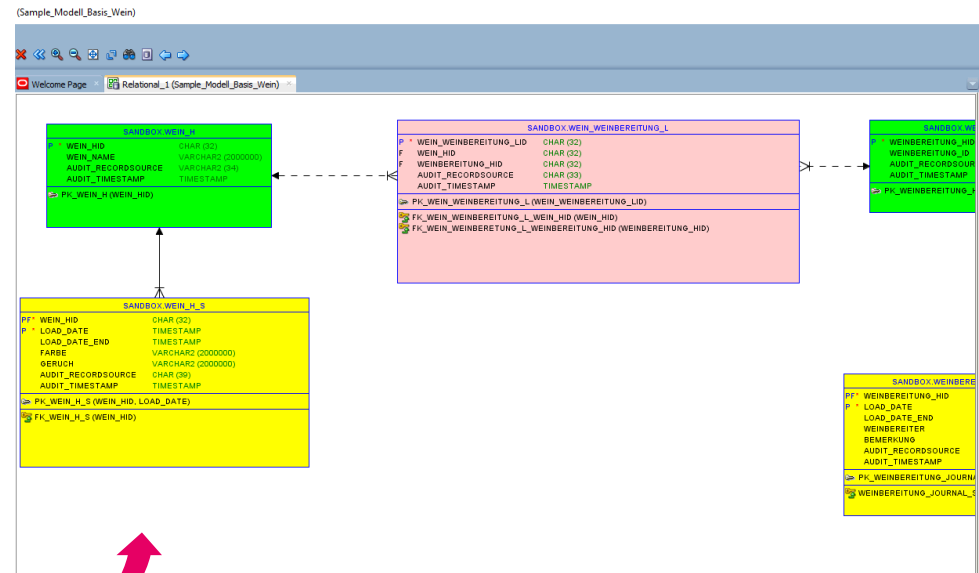
Oracle SQL Developer Data Modeler - Sample\_Modell\_Basis\_Wein

The screenshot shows the Oracle SQL Developer Data Modeler interface. The 'Tools' menu is open, displaying various options including 'Design Rules And Transformations', 'Compare/Merge Models', 'Features...', 'Preferences...', and 'Design Level Settings'. The 'Design Rules And Transformations' option is highlighted. In the background, a relational model diagram is visible, showing tables like 'SANDBOX.WEIN\_H' and 'SANDBOX.WEIN\_WEINBEREITUNG\_L' with their attributes and relationships.

The 'Custom Transformations Scripts' dialog box is open, showing a list of scripts and a script editor. The script editor contains the following code:

```

1 // Transformation Script to change the default color for tables
2 // after Reverse Engineering
3
4 // get all tables in the Relation Model
5 tables = model.getTableSet().toArray();
6 for (var t = 0; t < tables.length; t++){
7   table = tables[t];
8   table_name = table.getName();
9   table.setUseDefaultColor(false);
10  // set azure color for Hubs
11  if (table_name.endsWith('H')){
12    table.setUseDefaultColor(false);
13    table.setBackgroundColor(new java.awt.Color(204,255,255));
14  }
15  // set red color for Links
16  if (table_name.endsWith('L')){
17    table.setUseDefaultColor(false);
18    table.setBackgroundColor(new java.awt.Color(255,204,204));
19  }
20 }
    
```



**Doku von Oracle liegt im Verzeichnis `../datamodeler/xmlmetadata/doc`:**

- README.rtf
- SQLDeveloperDataModelerScripting.docx
- index.html

I. XML metadata files are part of the product

XML files in XML metadata directory ( datamodeler\datamodeler\xmlmetadat  
jects that can be created with Oracle SQL Developer Data Modeler UI or usir  
(DM) in its functionality thus any changes introduced to them could break noi

## II. Information described in XML files

XML files follow internal (class link). There is an HTML class hierarchy – direct

XML-files-in-xmlmetadata-directory  
Oracle-SQL-Developer-Data-Modeler  
functionality-thus-any-changes-intro

XML files follow internal object model following parent class link. There is which allow faster navigation in class there.¶

Two important groups can be found external-which means that property or collections stored outside the file, for example—Table has internal collection “columns”, i.e., columns are stored in table; RelationalDesign class (representing relational model) has external collection “tables”, i.e., tables are stored in RelationalDesign; LogicalModel class (representing the whole design) has external property LogicalModel and external collection “tables”.

[illegible]

```
oracle.dbtools.crest.model.design.storage.oracle.v10g.ASMDiskOraclev10g
```

## Class ASMDiskOraclev10g

Parent class : [AbstractStorageObjectOracle](#)  
Subclasses :

Properties			
Name	Type	Default Value	Getter
force	String	NO	getForce()
searchstring	String		getSearchString()
size	int	0	getSize()
sizeMeasurementUnit	String	M	getSizeMeasurementUnit()

Read Only Properties	
Name	Type
failGroup	FailGroupOraclev10g

Custom Transformations Scripts

Name	Object	Engine	Variable
14 Data Vault Default Color Schema	relational	Oracle Nashorn	model

Script

Library  Method

```
1 // Transformation Script to change the default color for tables
2 // after Reverse Engineering
3
4 // get all tables in the Relation Model
5 tables = model.getTableSet().toArray();
6 var t_changed = 0;
7 for (var t = 0; t < tables.length; t++){
8   table = tables[t];
9   table_name = table.getName();
10  // set azure color for Hubs
11  if (table_name.endsWith('_H')){
12    table.setUseDefaultColor(false);
13    table.setUseDefaultColor(false);
14    table.setBackgroundColor(new java.awt.Color(204,255,255));
15    t_changed++;
16  }
17  // set red color for Links
18  if (table_name.endsWith('_L')){
19    table.setUseDefaultColor(false);
20    table.setUseDefaultColor(false);
21    table.setBackgroundColor(new java.awt.Color(255,204,204));
22    t_changed++;
23  }
24  // set yellow color for Sattelite
25  if (table_name.endsWith('_S')){
26    table.setUseDefaultColor(false);
27    table.setUseDefaultColor(false);
28    table.setBackgroundColor(new java.awt.Color(255,255,0));
29    t_changed++;
30  }
31
32  // make sure the diagram has been changed
33  table.setDirty(true);
34 }
35
36 // Output in dialog
37 javax.swing.JOptionPane.showMessageDialog(null, "Anzahl Tabellen im Modell: " + tables.length + "\n" + "Anzahl transformierten Tabellen: " + t_changed);
```

Message

Anzahl Tabellen im Modell: 5  
Anzahl transformierten Tabellen: 5

OK

Save Apply Debug Import Export Close Help

# Ausgaben von Texten und Debug

Custom Transformations Scripts

	Name	Object	Engine	Variable
17	Demo Output Message	relational	Oracle Nashorn	model

Script

Library  Method

```
1 // Einfach eine Ausgabe im Dialog
2 javax.swing.JOptionPane.showMessageDialog(null, "Hello World!!!");
3
4 // für die längeren Ausgaben eine Notiz anlegen
5 var v_Text = "";
6 var v_Note = model.createNote();
7 v_Note.setName("Ausgabe");
8 v_Note.setComment("1. Zeile\n2. Zeile\n");
9
10 // Der Text der Notiz im Dialog ausgeben
11 v_Text = v_Note.getComment();
12 javax.swing.JOptionPane.showMessageDialog(null, v_Text);
13
14 // Interaktive Aktion mit dem Benutzer
15 var v_Answer = javax.swing.JOptionPane.showConfirmDialog (null, "Weiter?", "Question", javax.swing.JOptionPane.YES_NO_OPTION);
16 if ( v_Answer == 0 ){
17     javax.swing.JOptionPane.showMessageDialog(null, "Answer is Yes");
18 }
19 else{
20     javax.swing.JOptionPane.showMessageDialog(null, "Answer is No");
21 }
22
```

Save Apply Debug Import Export Close Help

# Typische Use Cases

1. Erstellen eines Subviews mit Objekten
2. Einfügen der Audit-Spalten
3. Arbeiten mit der Template-Table
4. Setzen des Farbschemas
5. Umwandeln der Namen in Groß- / Kleinbuchstaben
6. Umbenennen der Spalten

??? Fragen ???

1. Oracle SQL Developer oder Oracle SQL Developer Data Modeler unterstützen Skripting für Datenmodell
2. Mit der Tastenkombination `Alt+Shift+F` wird das Transformation-Fenster aufgerufen, in dem die mitgelieferten und eigenen Skripte geschrieben, aufgerufen und debuggen werden können
3. Die Dokumentation liegt im Verzeichnis  
`../datamodeler/xmlmetadata/doc`
4. Die Skripte können in Libraries zusammengefasst und verwaltet werden
5. Die Transformation des Modells kann **NICHT** rückgängig gemacht werden. Ein Backup vom Datenmodell vor der Transformation ist **ZWINGEND** notwendig

## 1. Re-engineering Your Database Using Oracle SQL Developer Data Modeler

<https://www.oracle.com/webfolder/technetwork/tutorials/obe/db/sqldevdm/r20/updatedb/UpdateDB.html>

## 2. Working with Customizable Design Rules and Transformation Scripts

<https://www.oracle.com/webfolder/technetwork/tutorials/obe/db/sqldevdm/r30/dm30desrule/dm30desrule.htm>

## 3. Transformations – die API des Oracle Data Modeler

[https://www.doag.org/formes/pubfiles/4977693/2013-K-DEV-Gudrun\\_Pabst-Transformations\\_die\\_API\\_des\\_Oracle\\_Datamodeler-Manuskript.pdf](https://www.doag.org/formes/pubfiles/4977693/2013-K-DEV-Gudrun_Pabst-Transformations_die_API_des_Oracle_Datamodeler-Manuskript.pdf)

## 4. Beispiele im GitHub <https://github.com/VladiPol/IT-Tage-2019>

## 5. Toolsvergleich [http://www.databaseanswers.org/modelling\\_tools.htm](http://www.databaseanswers.org/modelling_tools.htm) oder [https://en.wikipedia.org/wiki/Comparison\\_of\\_data\\_modeling\\_tools](https://en.wikipedia.org/wiki/Comparison_of_data_modeling_tools)



# Vielen Dank für Ihre Aufmerksamkeit!

**TeamBank**

**Vladimir Poliakov** - Data Engineer  
T +49 (0) 911 / 5390-2447  
E [vladimir.poliakov@teambank.de](mailto:vladimir.poliakov@teambank.de)

**TeamBank AG Nürnberg**  
Beuthener Straße 25  
90471 Nürnberg  
Germany

[www.teambank.de](http://www.teambank.de)

