

Datenvalidierung per Framework -

automatisch mit Metadaten

Stefan Winkler
- Geschäftsführer -
merlin.zwo InfoDesign GmbH & Co. KG
76228 Karlsruhe



Agenda

1. Ausgangssituation und gewünschtes Ergebnis
2. Step by Step
 - Architektur
 - Daten und Tabellen
 - Logik und Code
3. Zusätzliches und Erweiterungen



Spitzenleistung heißt, sich auf seine Stärken zu konzentrieren.

merlin.zwo

Wir machen Oracle - nur Oracle.
Aus gutem Grund.

www.merlin-zwo.de



★★★★★
WANTED

~~DEAD~~ || **ALIVE**



★ **CACHE REWARD** ★

#F4240\$

★★★★★

Oracle DBA

(m/w/d/...)

und

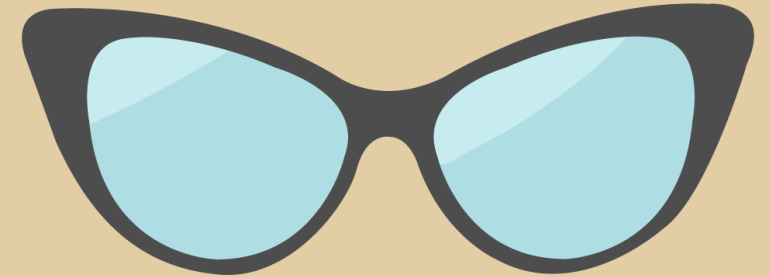
Oracle Developer

(m/w/d/...)

SQL, PL/SQL, Apex

★★★★★
WANTED

~~DEAD~~ || **ALIVE**



★ **CACHE REWARD** ★

#F4240\$

★★★★★

MyShop => MyDWH

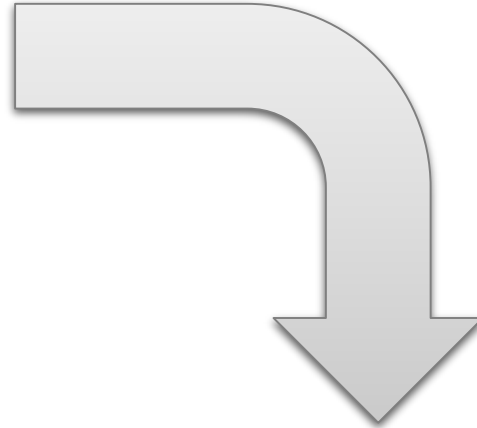
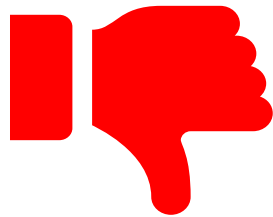


MyShop => MyDWH

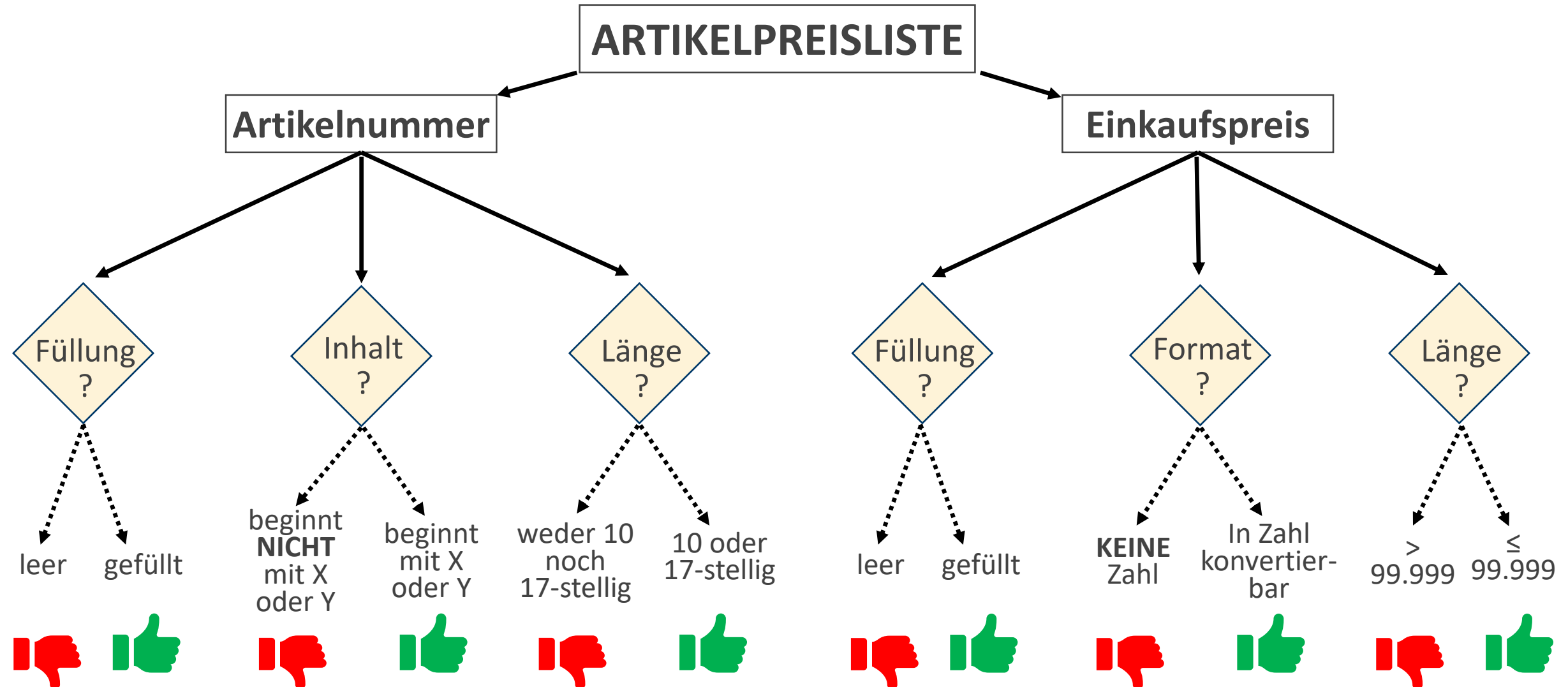


Teil 1: Ausgangssituation und gewünschtes Ergebnis

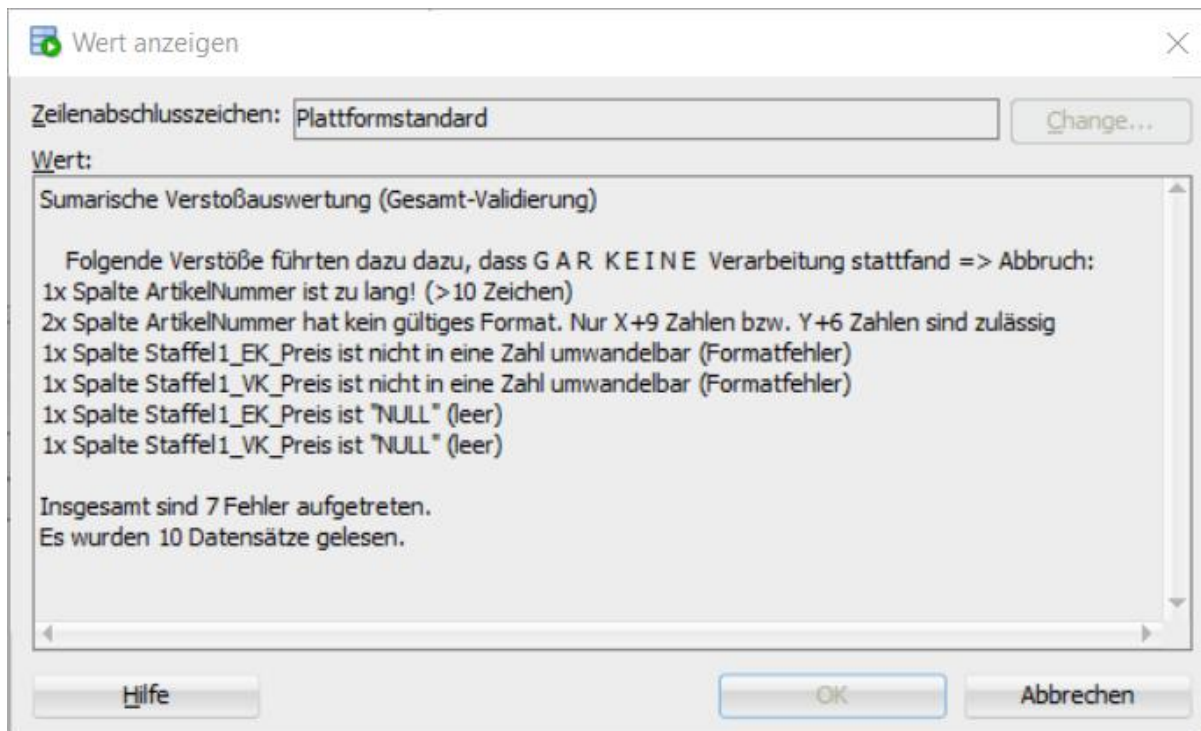
Sind die angelieferten
Daten *akzeptabel* ???



UFD_JR	M_NR	P_JR	UP_NR	AP_KEY	ZEIT_VON	ZEIT_BIS	FAHRTZEIT	...	ARBEITSKURZ	UTILITÄTSKURZ	AUFTRAG_JR	TÄTIGKEIT	MODIFIED_BY	MODIFIED_AT	TERMINART	...	M...	L...	...	KOM...	F...	FAKTOR_ID	TÄTIGKEIT_ID	M...		
1	50038900	50103	50127	(null)	(null)	17.08.04	14:00:00	17.08.04	14:30:00	(null)	J	Consulting	(null)	174-01-...	Absprech...	STEFAN	12.09.04	14:25:20	(null)	N	N	(.....)	(null)	(null)	N	J
2	50039202	50103	50074	(null)	(null)	23.08.04	13:15:00	23.08.04	18:30:00	(null)	J	Program...	(null)	290-02-...	Vorbereit...	STEFAN	12.09.04	14:25:24	(null)	N	N	(.....)	(null)	(null)	N	J
3	50038290	50103	50127	(null)	(null)	05.08.04	07:30:00	05.08.04	12:15:00	1,5	J	Meeting	(null)	174-01-...	Besprech...	STEFAN	12.09.04	14:25:08	(null)	N	N	(.....)	(null)	(null)	N	J
4	50039149	50103	50054	(null)	(null)	23.08.04	13:00:00	23.08.04	13:15:00	(null)	J	Support	(null)	260-01-...	Routing...	STEFAN	12.09.04	14:25:24	(null)	N	N	(.....)	(null)	(null)	N	J
5	50038763	50103	50053	1	(null)	13.08.04	16:00:00	13.08.04	16:45:00	(null)	J	Support	(null)	130-04-...	Problem...	STEFAN	12.09.04	14:25:17	(null)	N	N	(.....)	(null)	(null)	N	J
6	50038760	50103	50054	(null)	(null)	13.08.04	15:00:00	13.08.04	16:00:00	(null)	J	Support	(null)	260-01-...	Problem...	STEFAN	12.09.04	14:25:17	(null)	N	N	(.....)	(null)	(null)	N	J
7	50038299	50103	50127	(null)	(null)	05.08.04	13:15:00	05.08.04	15:00:00	(null)	J	Meeting	(null)	174-01-...	Besprech...	STEFAN	12.09.04	14:25:11	(null)	N	N	(.....)	(null)	(null)	N	J
8	50038329	50103	50053	1	(null)	07.08.04	14:15:00	07.08.04	17:20:00	(null)	J	Support	(null)	130-04-...	Restor...	STEFAN	12.09.04	14:25:14	(null)	N	N	(.....)	(null)	(null)	N	J
9	50039701	50103	50054	(null)	(null)	07.09.04	14:00:00	07.09.04	17:15:00	(null)	J	Support	(null)	260-01-...	Telefo...	STEFAN	09.10.04	11:05:00	(null)	N	N	(.....)	(null)	(null)	N	J
10	50046271	50102	50119	4	(null)	23.12.04	13:10:00	23.12.04	17:40:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
11	50046567	50102	50119	4	(null)	03.01.05	14:00:00	03.01.05	15:15:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
12	50046401	50102	50119	4	(null)	24.12.04	08:40:00	24.12.04	12:20:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
13	50046447	50102	50119	4	(null)	29.12.04	13:00:00	29.12.04	18:00:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
14	50043068	50041	50109	(null)	(null)	05.11.04	09:45:00	05.11.04	13:45:00	(null)	J	Support	(null)	172-01-...	Dive...	STEFAN	06.11.04	18:38:48	(null)	N	N	(.....)	(null)	(null)	N	J
15	50043069	50041	50109	(null)	(null)	04.11.04	12:00:00	04.11.04	12:45:00	(null)	J	Support	(null)	172-01-...	Dive...	STEFAN	06.11.04	18:38:47	(null)	N	N	(.....)	(null)	(null)	N	J
16	50043070	50041	50109	(null)	(null)	04.11.04	18:30:00	04.11.04	19:00:00	(null)	J	Support	(null)	172-01-...	Dive...	STEFAN	06.11.04	18:38:47	(null)	N	N	(.....)	(null)	(null)	N	J
17	50046728	50102	50119	4	(null)	03.01.05	09:30:00	03.01.05	11:30:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
18	50050628	51064	50084	4	(null)	23.02.05	13:30:00	23.02.05	14:20:00	(null)	J	Programm...	(null)	120116-...	WAV Web...	STEFAN	11.04.05	13:13:13	(null)	N	N	(.....)	(null)	(null)	N	J
19	50047656	50102	50131	1	(null)	21.01.05	12:00:00	21.01.05	17:10:00	(null)	J	Consulting	(null)	SDV1-FA...	Ort: mer...	STEFAN	11.03.05	09:38:05	(null)	N	N	(.....)	(null)	(null)	N	J
20	50047655	50102	50131	1	(null)	22.01.05	11:35:00	22.01.05	13:10:00	(null)	J	Consulting	(null)	SDV1-FA...	Ort: mer...	STEFAN	11.03.05	09:38:05	(null)	N	N	(.....)	(null)	(null)	N	J
21	50046397	50102	50119	4	(null)	27.12.04	08:30:00	27.12.04	12:30:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
22	50046399	50102	50119	4	(null)	27.12.04	14:15:00	27.12.04	18:25:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
23	50046445	50102	50119	4	(null)	29.12.04	09:30:00	29.12.04	12:35:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
24	50046422	50102	50119	4	(null)	29.12.04	13:10:00	29.12.04	17:30:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
25	50046424	50102	50119	4	(null)	29.12.04	13:10:00	29.12.04	17:30:00	(null)	J	Design	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
26	50049031	50102	50119	4	(null)	04.02.05	08:40:00	04.02.05	12:50:00	(null)	J	Programm...	SQLHavig	120130-...	Ort: mer...	STEFAN	27.02.05	20:33:44	(null)	N	N	(.....)	(null)	(null)	N	J
27	50049032	50102	50119	4	(null)	04.02.05	13:40:00	04.02.05	18:35:00	(null)	J	Programm...	SQLHavig	120130-...	Ort: mer...	STEFAN	27.02.05	20:33:44	(null)	N	N	(.....)	(null)	(null)	N	J
28	50049035	50102	50119	4	(null)	02.02.05	12:40:00	02.02.05	16:30:00	(null)	J	Programm...	SQLHavig	120130-...	Ort: mer...	STEFAN	27.02.05	20:33:44	(null)	N	N	(.....)	(null)	(null)	N	J
29	50048499	50102	50119	4	(null)	19.01.05	06:30:00	19.01.05	19:15:00	2,5	J	Meeting	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
30	50048500	50102	50119	4	(null)	20.01.05	08:30:00	20.01.05	20:30:00	2,5	J	Meeting	(null)	120130-...	Ort: mer...	STEFAN	19.02.05	16:06:42	(null)	N	N	(.....)	(null)	(null)	N	J
31	50048505	50102	50007	5	299	17.01.05	13:30:00	17.01.05	19:00:00	1,5	N	Meeting	(null)	m2-HP V3	Ort: HoWe...	STEFAN	12.02.07	12:30:22	(null)	N	N	(.....)	(null)	(null)	N	J
32	50050627	51064	50084	4	(null)	23.02.05	09:20:00	23.02.05	12:40:00	(null)	J	Programm...	(null)	120116-...	WAV Web...	STEFAN	11.04.05	13:13:13	(null)	N	N	(.....)	(null)	(null)	N	J
33	50051255	51064	50084	4	(null)	03.03.05	12:40:00	03.03.05	13:05:00	(null)	J	Programm...	(null)	120116-...	Kosten L...	STEFAN	11.04.05	13:13:13	(null)	N	N	(.....)	(null)	(null)	N	J
34	50049041	50102	50119	4	(null)	03.02.05	14:10:00	03.02.05	19:30:00	(null)	J	Programm...	SQLHavig	120130-...	Ort: mer...	STEFAN	27.02.05	20:33:44	(null)	N	N	(.....)	(null)	(null)	N	J
35	50053068	50102	50119	4	(null)	30.03.05	08:30:00	30.03.05	20:00:00	2,5	J	Meeting	(null)	120130-...	Ort: mer...	STEFAN	11.04.05	13:16:37	(null)	N	N	(.....)	(null)	(null)	N	J
36	50052800	50102	50119	4	(null)	29.03.05	06:45:00	29.03.05	17:45:00	2,25	J	Meeting	(null)	120130-...	Ort: mer...	STEFAN	11.04.05	13:16:37	(null)	N	N	(.....)	(null)	(null)	N	J
37	50053069	50102	50119	4	(null)	31.03.05	09:30:00	31.03.05	12:45:00	2,5	J	Meeting	(null)	120130-...	Ort: mer...	STEFAN	11.04.05	13:16:37	(null)	N	N	(.....)	(null)	(null)	N	J
38	50053070	50102	50119	4	(null)	31.03.05	14:30:00	31.03.05	18:45:00	2,5	J	Programm...	(null)	120130-...	Ort: mer...	STEFAN	11.04.05	13:16:37	(null)	N	N	(.....)	(null)	(null)	N	J
39	50063151	50041	50136	1	(null)	30.09.05	08:30:00	30.09.05	17:45:00	1	J	Meeting	(null)	FACHBER...	l. Pro...	STEFAN	02.12.05	17:35:55	(null)	N	N	(.....)	(null)	(null)	N	J
40	50065896	51091	50045	3	(null)	02.12.05	15:30:00	02.12.05	17:20:00	(null)	J	Support	(null)	EWG-ALL...	Fehler...	STEFAN	06.01.06	17:27:33	(null)	N	N	(.....)	(null)	(null)	N	J
41	50066308	51091	50045	1	(null)	12.12.05	09:30:00	12.12.05	18:00:00	(null)	N	Weiterbild	(null)	EWG-ALL...	Einarz...	STEFAN	18.12.05	11:33:20	(null)	J	N	(.....)	Ein...	(null)	N	J
42	50066312	51091	50045	1	(null)	13.12.05	09:00:00	13.12.05	15:40:00	(null)	N	Weiterbild	(null)	EWG-ALL...	Weiterb...	STEFAN	18.12.05	16:24:10	(null)	J	N	(.....)	Ein...	(null)	N	J
43	50066310	51091	50045	8	(null)	13.12.05	15:40:00	13.12.05	19:00:00	(null)	N	Meeting	(null)	EWG-ALL...	Treffen mit...	STEFAN	18.12.05	11:33:20	(null)	N	N	(.....)	Ein...	(null)	N	J
44	50065758	51091	50045	3	(null)	02.12.05	09:30:00	02.12.05	10:50:00	(null)	J	Support	(null)	EWG-ALL...	Fehler bei...	STEFAN	06.01.06	17:27:33	(null)	N	N	(.....)	(null)	(null)	N	J




```
SELECT INFRA.Validierung_pkg.validiereTabelleninhalt  
  ( iLaufNr      => 4711,  
    iSchemaname  => 'RawStage',  
    iTabellenname => 'ARTIKELPREISLISTE'  
  ) AS Ergebnis  
FROM dual;
```

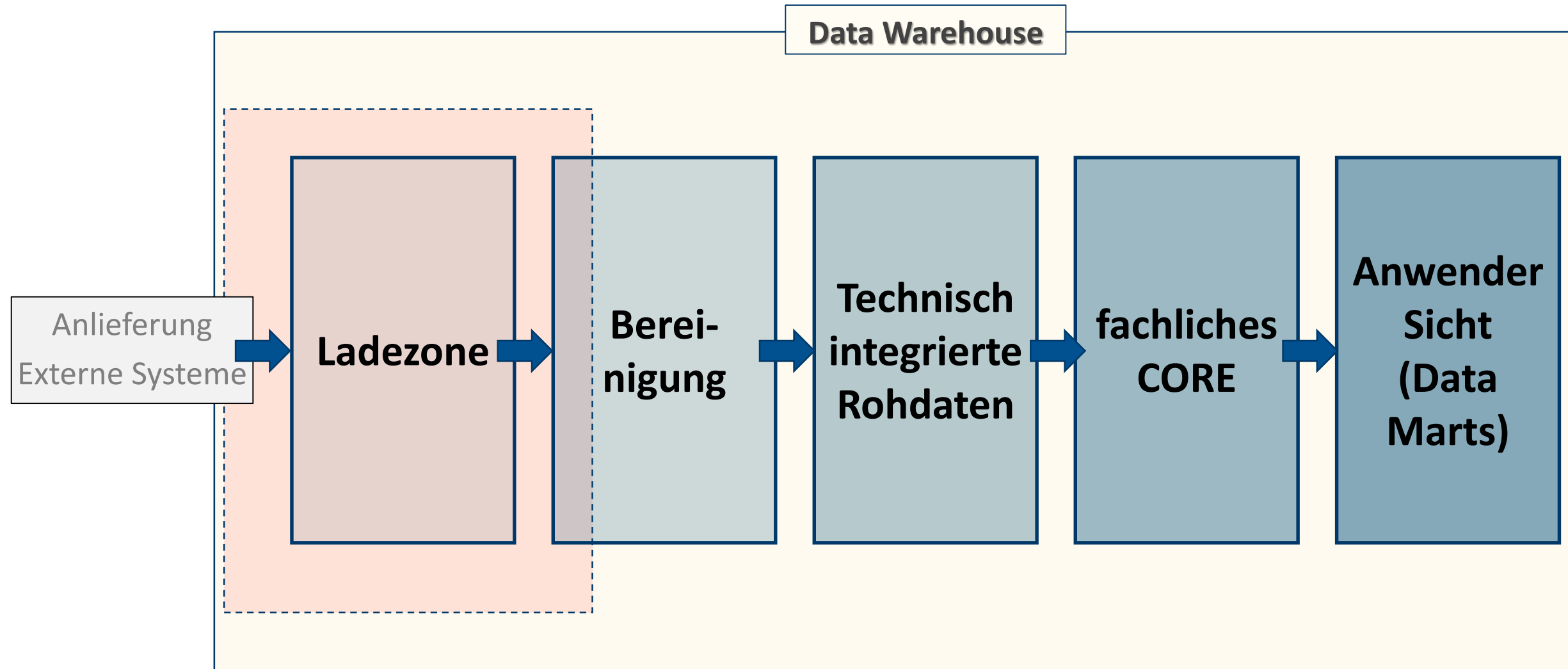


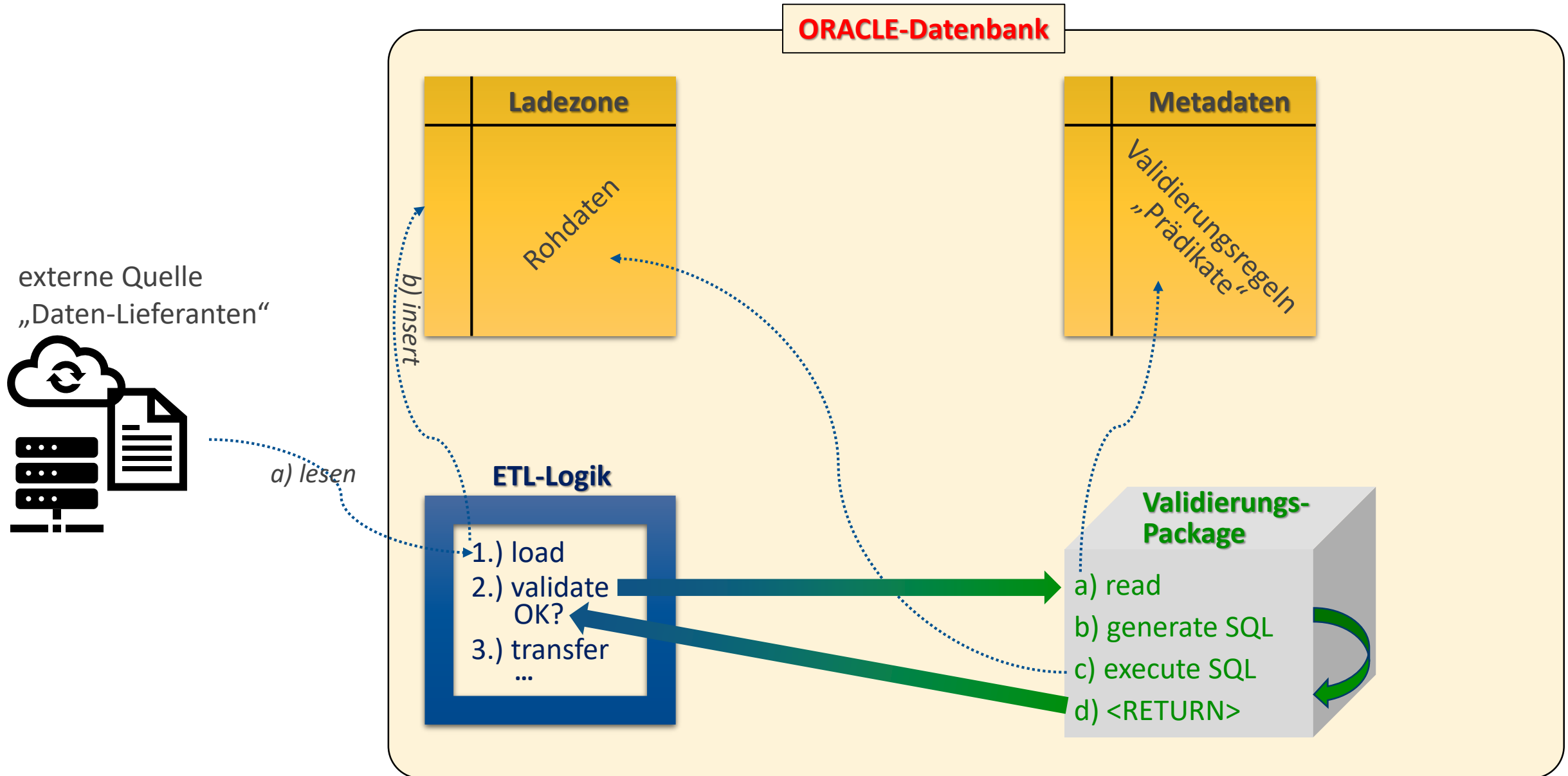
Features:

1. Filterbarkeit (bei Quellarchivierung)
2. 1x Full Table Scan (Quelltabelle)
3. RETURN als VC2 (Gesamtergebnis)
4. beliebig viele Validierungsregeln
5. Bezugnahme auf andere Tabellen (Foreign Key-Logik)
6. Bezugnahme auf andere Spalten/Inhalte (Check Constraint-Logik)
7. Zählung Verstöße (pro Regel)
8. netter Prosatext (bei Verstößen)
9. Gesamtfehleranzahl
10. Zählen aller Quelldatensätze

Teil 2: Step by Step (Architektur, Daten und Tabellen, Logik und Code)







Datenmodell des DWH-Beispiels „MyShop“

MyShop (DWH) - Datenmodell
von Stefan.Winkler@merlin-zwo.de
Stand: 08.10.2019

RAWSTAGE

RAWSTAGE.Artikel	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
ArtikelNummer	VARCHAR2 (4000 CHAR)
ArtikelKurzbezeichnung	VARCHAR2 (4000 CHAR)
ArtikelLangbezeichnung	VARCHAR2 (4000 CHAR)
ArtikelBeschreibung	VARCHAR2 (4000 CHAR)
ArtikelZusatztext	VARCHAR2 (4000 CHAR)
GrundPreis	VARCHAR2 (4000 CHAR)
GrundPreis_Waehrung	VARCHAR2 (4000 CHAR)
OnlineShop_JN	VARCHAR2 (4000 CHAR)
MindestMenge	VARCHAR2 (4000 CHAR)
MindestMenge_Einheit	VARCHAR2 (4000 CHAR)
MaximalMenge	VARCHAR2 (4000 CHAR)
MaximalMenge_Einheit	VARCHAR2 (4000 CHAR)
Gefahrenklasse	VARCHAR2 (4000 CHAR)
AltersBeschraenkung	VARCHAR2 (4000 CHAR)

RAWSTAGE.ArtikelPreisliste	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
ArtikelNummer	VARCHAR2 (4000 CHAR)
Staffel1_EK_Preis	VARCHAR2 (4000 CHAR)
Staffel1_VK_Preis	VARCHAR2 (4000 CHAR)
Staffel1_vonMenge	VARCHAR2 (4000 CHAR)
Staffel1_bisMenge	VARCHAR2 (4000 CHAR)
Staffel1_Hinweis	VARCHAR2 (4000 CHAR)
Staffel2_EK_Preis	VARCHAR2 (4000 CHAR)
Staffel2_VK_Preis	VARCHAR2 (4000 CHAR)
Staffel2_vonMenge	VARCHAR2 (4000 CHAR)
Staffel2_bisMenge	VARCHAR2 (4000 CHAR)
Staffel2_Hinweis	VARCHAR2 (4000 CHAR)
Staffel3_EK_Preis	VARCHAR2 (4000 CHAR)
Staffel3_VK_Preis	VARCHAR2 (4000 CHAR)
Staffel3_vonMenge	VARCHAR2 (4000 CHAR)
Staffel3_bisMenge	VARCHAR2 (4000 CHAR)
Staffel3_Hinweis	VARCHAR2 (4000 CHAR)

RAWSTAGE.LieferroutenTour	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE

CLEANSTAGE

CLEANSTAGE.Artikel	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
* CleanZeitpunkt	DATE
* ArtikelNummer	VARCHAR2 (10 CHAR)
* ArtikelKurzbezeichnung	VARCHAR2 (40 CHAR)
* ArtikelLangbezeichnung	VARCHAR2 (100 CHAR)
* ArtikelBeschreibung	VARCHAR2 (4000 CHAR)
ArtikelZusatztext	VARCHAR2 (4000 CHAR)
GrundPreis	NUMBER (8,2)
* GrundPreis_Waehrg_ISO4217Code	VARCHAR2 (3 CHAR)
OnlineShop_JN	VARCHAR2 (1 CHAR)
MindestMenge	NUMBER (8,2)
MindestMenge_Einheit	VARCHAR2 (10 CHAR)
MaximalMenge	NUMBER (8,2)
MaximalMenge_Einheit	VARCHAR2 (10 CHAR)
* Gefahrenklasse	VARCHAR2 (5 CHAR)
* AltersBeschraenkung	VARCHAR2 (10 CHAR)

CLEANSTAGE.ArtikelPreisliste	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
* CleanZeitpunkt	DATE
* ArtikelNummer	VARCHAR2 (10 CHAR)
* Staffel1_EK_Preis	NUMBER (8,2)
* Staffel1_VK_Preis	NUMBER (8,2)
* Staffel1_vonMenge	NUMBER (8,2)
* Staffel1_bisMenge	NUMBER (8,2)
Staffel1_Hinweis	VARCHAR2 (4000 CHAR)
Staffel2_EK_Preis	NUMBER (8,2)
Staffel2_VK_Preis	NUMBER (8,2)
Staffel2_vonMenge	NUMBER (8,2)
Staffel2_bisMenge	NUMBER (8,2)
Staffel2_Hinweis	VARCHAR2 (4000 CHAR)
Staffel3_EK_Preis	NUMBER (8,2)
Staffel3_VK_Preis	NUMBER (8,2)
Staffel3_vonMenge	NUMBER (8,2)
Staffel3_bisMenge	NUMBER (8,2)
Staffel3_Hinweis	VARCHAR2 (4000 CHAR)

CLEANSTAGE.LieferroutenTour	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE

INFRA

INFRA.VALIDIERUNGSREGEL	
P * NR	NUMBER (10)
U * VALID_FID	VARCHAR2 (100 CHAR)
U * SCHEMA	VARCHAR2 (30 CHAR)
U * TABELLE	VARCHAR2 (30 CHAR)
* PRUEFROUTINE	VARCHAR2 (100 BYTE)
U * PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
* ANGELEGT_AM	DATE
* ANGELEGT_VON	VARCHAR2 (30 CHAR)
GEAENDERT_AM	DATE
GEAENDERT_VON	VARCHAR2 (30 CHAR)
KOMMENTAR	VARCHAR2 (4000 CHAR)

VALI_PK (NR)
VALI_UK1 (VALID_FID)
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)

VALI_PK (NR)
VALI_UK1 (VALID_FID)
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)

INFRA.LOGGING	
P * LOG_ID	NUMBER (38)
* Log_Zeitpunkt	TIMESTAMP
Befehl	CLOB
* MELDUNG_VC2	VARCHAR2 (4000 CHAR)
Meldung_CLOB	CLOB
Kommentar	VARCHAR2 (4000 CHAR)
LOGGING_PK (LOG_ID)	

INFRA.WAEHRUNG_ISO4217	
P * ISO_Code	VARCHAR2 (3 CHAR)
U * ISO_Num	VARCHAR2 (3 CHAR)
* Waehrung	VARCHAR2 (30 CHAR)
Untereinheit	VARCHAR2 (30 CHAR)
* Laenderliste	VARCHAR2 (4000 CHAR)
Gueltigkeit	VARCHAR2 (50 CHAR)
WAEHRUNG_ISO4217_PK (ISO_Code)	
WAEHRUNG_ISO4217_UK (ISO_Num)	

Klassisches DWH-Staging

- **Gelb:** Ladezonen-Tabellen
- **Grün:** Cleansing-Tabellen
- **Rot:** Metadaten, Tabellen der DWH-Infrastruktur

Hier nicht mehr dargestellt sind die nachfolgenden Stages:

- ODS (operational Data Store)
- CORE (konsolidierter Kern)
- DataMart (fachlich aufbereitete Sicht)

Datenmodell des DWH-Beispiels „MyShop“

RAWSTAGE

RAWSTAGE.Artikel	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
* ArtikelNummer	VARCHAR2 (4000 CHAR)
* ArtikelKurzbezeichnung	VARCHAR2 (4000 CHAR)
* ArtikelLangbezeichnung	VARCHAR2 (4000 CHAR)
* ArtikelBeschreibung	VARCHAR2 (4000 CHAR)
* ArtikelZusatztext	VARCHAR2 (4000 CHAR)
* GrundPreis	VARCHAR2 (4000 CHAR)
* GrundPreis_Waehrung	VARCHAR2 (4000 CHAR)
* OnlineShop_JN	VARCHAR2 (4000 CHAR)
* MindestMenge	VARCHAR2 (4000 CHAR)
* MindestMenge_Einheit	VARCHAR2 (4000 CHAR)
* MaximalMenge	VARCHAR2 (4000 CHAR)
* MaximalMenge_Einheit	VARCHAR2 (4000 CHAR)
* Gefahrenklasse	VARCHAR2 (4000 CHAR)
* AltersBeschraenkung	VARCHAR2 (4000 CHAR)

CLEANSTAGE

CLEANSTAGE.Artikel	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
* CleanZeitpunkt	DATE
* ArtikelNummer	VARCHAR2 (10 CHAR)
* ArtikelKurzbezeichnung	VARCHAR2 (40 CHAR)
* ArtikelLangbezeichnung	VARCHAR2 (100 CHAR)
* ArtikelBeschreibung	VARCHAR2 (4000 CHAR)
* ArtikelZusatztext	VARCHAR2 (4000 CHAR)
* GrundPreis	NUMBER (8,2)
* GrundPreis_Waehrg_ISO4217Code	VARCHAR2 (3 CHAR)
* OnlineShop_JN	VARCHAR2 (1 CHAR)
* MindestMenge	NUMBER (8,2)
* MindestMenge_Einheit	VARCHAR2 (10 CHAR)
* MaximalMenge	NUMBER (8,2)
* MaximalMenge_Einheit	VARCHAR2 (10 CHAR)
* Gefahrenklasse	VARCHAR2 (5 CHAR)
* AltersBeschraenkung	VARCHAR2 (10 CHAR)

INFRA

INFRA.VALIDIERUNGSREGEL	
P * NR	NUMBER (10)
U * VALID_FID	VARCHAR2 (100 CHAR)
U * SCHEMA	VARCHAR2 (30 CHAR)
U * TABELLE	VARCHAR2 (30 CHAR)
* PRUEFROUTINE	VARCHAR2 (4000 BYTE)
U * PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
* ANGELEGT_AM	DATE
* ANGELEGT_VON	VARCHAR2 (30 CHAR)
GEAENDERT_AM	DATE
GEAENDERT_VON	VARCHAR2 (30 CHAR)
KOMMENTAR	VARCHAR2 (4000 CHAR)
VALI_PK (NR) VALI_UK1 (VALID_FID) VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)	
VALI_PK (NR) VALI_UK1 (VALID_FID) VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)	

RAWSTAGE.ArtikelPreisliste	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
* ArtikelNummer	VARCHAR2 (4000 CHAR)
* Staffel1_EK_Preis	VARCHAR2 (4000 CHAR)
* Staffel1_VK_Preis	VARCHAR2 (4000 CHAR)
* Staffel1_vonMenge	VARCHAR2 (4000 CHAR)
* Staffel1_bisMenge	VARCHAR2 (4000 CHAR)
* Staffel1_Hinweis	VARCHAR2 (4000 CHAR)
* Staffel2_EK_Preis	VARCHAR2 (4000 CHAR)
* Staffel2_VK_Preis	VARCHAR2 (4000 CHAR)
* Staffel2_vonMenge	VARCHAR2 (4000 CHAR)
* Staffel2_bisMenge	VARCHAR2 (4000 CHAR)
* Staffel2_Hinweis	VARCHAR2 (4000 CHAR)
* Staffel3_EK_Preis	VARCHAR2 (4000 CHAR)
* Staffel3_VK_Preis	VARCHAR2 (4000 CHAR)
* Staffel3_vonMenge	VARCHAR2 (4000 CHAR)
* Staffel3_bisMenge	VARCHAR2 (4000 CHAR)

CLEANSTAGE.ArtikelPreisliste	
* LaufNr	NUMBER (38)
* LadeZeitpunkt	DATE
* CleanZeitpunkt	DATE
* ArtikelNummer	VARCHAR2 (10 CHAR)
* Staffel1_EK_Preis	NUMBER (8,2)
* Staffel1_VK_Preis	NUMBER (8,2)
* Staffel1_vonMenge	NUMBER (8,2)
* Staffel1_bisMenge	NUMBER (8,2)
* Staffel1_Hinweis	VARCHAR2 (4000 CHAR)
* Staffel2_EK_Preis	NUMBER (8,2)
* Staffel2_VK_Preis	NUMBER (8,2)
* Staffel2_vonMenge	NUMBER (8,2)
* Staffel2_bisMenge	NUMBER (8,2)
* Staffel2_Hinweis	VARCHAR2 (4000 CHAR)
* Staffel3_EK_Preis	NUMBER (8,2)
* Staffel3_VK_Preis	NUMBER (8,2)
* Staffel3_vonMenge	NUMBER (8,2)
* Staffel3_bisMenge	NUMBER (8,2)

INFRA.LOGGING

P * LOG_ID	NUMBER (38)
* Log_Zeitpunkt	TIMESTAMP
Befehl	CLOB
* MELDUNG_VC2	VARCHAR2 (4000 CHAR)
Meldung_CLOB	CLOB
Kommentar	VARCHAR2 (4000 CHAR)
LOGGING_PK (LOG_ID)	

INFRA.WAEHRUNG_ISO4217

P * ISO_Code	VARCHAR2 (3 CHAR)
U * ISO_Num	VARCHAR2 (3 CHAR)
* Waehrung	VARCHAR2 (30 CHAR)
* Untereinheit	VARCHAR2 (30 CHAR)

Beispiel Prüfbedingung (1/5)

Single Record View

Navigation: << < > >>

NR	150
VALID_FID	Artikel_ArtikelNummer_NN
SCHEMA	RAWSTAGE
TABELLE	ARTIKEL
PRUEFROUTINE	INFRA.Validierung_pkg.validiereINFRAPruefregel
PRUEFBEDINGUNG	ArtikelNummer IS NOT NULL
VERLETZUNGSMELDUNG	Spalte ArtikelNummer ist "NULL" (leer)
ANGELEGT_AM	30.09.19 11:01:58
ANGELEGT_VON	INFRA
GEAENDERT_AM	
GEAENDERT_VON	
KOMMENTAR	

Hilfe Anwenden Abbrechen

INFRA.VALIDIERUNGSREGEL		
P	* NR	NUMBER (10)
U	* VALID_FID	VARCHAR2 (100 CHAR)
U	* SCHEMA	VARCHAR2 (30 CHAR)
U	* TABELLE	VARCHAR2 (30 CHAR)
	* PRUEFROUTINE	VARCHAR2 (100 BYTE)
U	* PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
	* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
	* ANGELEGT_AM	DATE
	* ANGELEGT_VON	VARCHAR2 (30 CHAR)
	* GEAENDERT_AM	DATE
	* GEAENDERT_VON	VARCHAR2 (30 CHAR)
	* KOMMENTAR	VARCHAR2 (4000 CHAR)
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		

Füllung der Artikelnummer ...

- a) gefüllt => 😊
- b) leer => ☹️

Beispiel Prüfbedingung (2/5)

Single Record View

Navigation: << < > >>

NR	151
VALID_FID	Artikel_ArtikelNummer_größer_10Zeichen
SCHEMA	RAWSTAGE
TABELLE	ARTIKEL
PRUEFROUTINE	INFRA.Validierung_pkg.validiereINFRAPruefregel
PRUEFBEDINGUNG	length(ArtikelNummer) <= 10
VERLETZUNGSMELDUNG	Spalte ArtikelNummer ist zu lang! (>10 Zeichen)
ANGELEGT_AM	30.09.19 11:01:58
ANGELEGT_VON	INFRA
GEAENDERT_AM	
GEAENDERT_VON	
KOMMENTAR	

Hilfe Anwenden Abbrechen

INFRA.VALIDIERUNGSREGEL		
P	* NR	NUMBER (10)
U	* VALID_FID	VARCHAR2 (100 CHAR)
U	* SCHEMA	VARCHAR2 (30 CHAR)
U	* TABELLE	VARCHAR2 (30 CHAR)
	* PRUEFROUTINE	VARCHAR2 (100 BYTE)
U	* PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
	* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
	* ANGELEGT_AM	DATE
	* ANGELEGT_VON	VARCHAR2 (30 CHAR)
	* GEAENDERT_AM	DATE
	* GEAENDERT_VON	VARCHAR2 (30 CHAR)
	* KOMMENTAR	VARCHAR2 (4000 CHAR)
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		

Länge der Artikelnummer ...

- a) höchstens 10-stellig => 😊
- b) mehr als 10 Zeichen => 😞

Beispiel Prüfbedingung (3/5)

Single Record View

Navigation: << < > >>

NR	174
VALID_FID	LieferroutenTour_FAHRTAG_ungültig
SCHEMA	RAWSTAGE
TABELLE	LIEFERROUTENTOUR
PRUEFROUTINE	INFRA.Validierung_pkg.validiereINFRAPruefregel
PRUEFBEDINGUNG	(IS_VALID_DATE(fahrttag)=1)
VERLETZUNGSMELDUNG	Spalte FAHRTAG hat kein gültiges Format (DD.MM.YYYY).
ANGELEGT_AM	30.09.19 11:01:58
ANGELEGT_VON	INFRA
GEAENDERT_AM	
GEAENDERT_VON	
KOMMENTAR	

Hilfe Anwenden Abbrechen

INFRA.VALIDIERUNGSREGEL		
P	* NR	NUMBER (10)
U	* VALID_FID	VARCHAR2 (100 CHAR)
U	* SCHEMA	VARCHAR2 (30 CHAR)
U	* TABELLE	VARCHAR2 (30 CHAR)
	* PRUEFROUTINE	VARCHAR2 (100 BYTE)
U	* PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
	* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
	* ANGELEGT_AM	DATE
	* ANGELEGT_VON	VARCHAR2 (30 CHAR)
	* GEAENDERT_AM	DATE
	* GEAENDERT_VON	VARCHAR2 (30 CHAR)
	* KOMMENTAR	VARCHAR2 (4000 CHAR)
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		

Inhalt des Fahrtags ist als Datum ...

a) z.B. 31.12.1999 => 😊

b) z.B. 32.12.1999 => 😞

Beispiel Prüfbedingung (4/5)

Single Record View

Navigation: << < > >>

NR	152
VALID_FID	Artikel_ArtikelNummer_Format_ungültig
SCHEMA	RAWSTAGE
TABELLE	ARTIKEL
PRUEFROUTINE	INFRA.Validierung_pkg.validiereINFRAPruefregel
PRUEFBEDINGUNG	regexp_like(ArtikelNummer, '^X\d{9}') OR regexp_like(ArtikelNummer, '^Y\d{6}')
VERLETZUNGSMELDUNG	Spalte ArtikelNummer hat kein gültiges Format. Nur X+9 Zahlen bzw. Y+6 Zahlen sind zulässig
ANGELEGT_AM	30.09.19 11:01:58
ANGELEGT_VON	INFRA
GEAENDERT_AM	
GEAENDERT_VON	
KOMMENTAR	

Hilfe Anwenden Abbrechen

```
regexp_like(ArtikelNummer, '^X\d{9}') OR  
regexp_like(ArtikelNummer, '^Y\d{6}')
```

INFRA.VALIDIERUNGSREGEL		
P	* NR	NUMBER (10)
U	* VALID_FID	VARCHAR2 (100 CHAR)
U	* SCHEMA	VARCHAR2 (30 CHAR)
U	* TABELLE	VARCHAR2 (30 CHAR)
	* PRUEFROUTINE	VARCHAR2 (100 BYTE)
U	* PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
	* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
	* ANGELEGT_AM	DATE
	* ANGELEGT_VON	VARCHAR2 (30 CHAR)
	GEAENDERT_AM	DATE
	GEAENDERT_VON	VARCHAR2 (30 CHAR)
	KOMMENTAR	VARCHAR2 (4000 CHAR)
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		

Inhalt der Artikelnummer zulässig:

*beginnt mit X gefolgt von 9 Zahlen oder
beginnt mit Y gefolgt von 6 Zahlen*

a) z.B. Y123456 => 😊

b) z.B. M0123456789 => 😞

Beispiel Prüfbedingung (5/5)

Single Record View

Navigation: << < > >>

NR	160
VALID_FID	Artikel_Grundpreis_Waehrung_Zahl
SCHEMA	RAWSTAGE
TABELLE	ARTIKEL
PRUEFROUTINE	INFRA.Validierung_pkg.validiereINFRAPruefregel
PRUEFBEDINGUNG	Grundpreis_Waehrung IS NOT NULL AND Grundpreis_Waehrung IN(SELECT ISO_CODE FROM WEAHRUNG_ISO4217)
VERLETZUNGSMELDUNG	Spalte Grundpreis_Waehrung enthält keinen gültigen WÄHRUNGS ISO Code
ANGELEGT_AM	30.09.19 11:01:58
ANGELEGT_VON	INFRA
GEAENDERT_AM	
GEAENDERT_VON	
KOMMENTAR	

Hilfe Anwenden Abbrechen

INFRA.VALIDIERUNGSREGEL		
P	* NR	NUMBER (10)
U	* VALID_FID	VARCHAR2 (100 CHAR)
U	* SCHEMA	VARCHAR2 (30 CHAR)
U	* TABELLE	VARCHAR2 (30 CHAR)
	* PRUEFROUTINE	VARCHAR2 (100 BYTE)
U	* PRUEFBEDINGUNG	VARCHAR2 (4000 BYTE)
	* VERLETZUNGSMELDUNG	VARCHAR2 (4000 BYTE)
	* ANGELEGT_AM	DATE
	* ANGELEGT_VON	VARCHAR2 (30 CHAR)
	* GEAENDERT_AM	DATE
	* GEAENDERT_VON	VARCHAR2 (30 CHAR)
	* KOMMENTAR	VARCHAR2 (4000 CHAR)
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		
VALI_PK (NR)		
VALI_UK1 (VALID_FID)		
VALI_UK2 (SCHEMA, TABELLE, PRUEFBEDINGUNG)		

Grundpreis_Waehrung is not null AND
Grundpreis_Waehrung IN
(SELECT ISO_CODE FROM WEAHRUNG_ISO4217)

Währung in Lookuptabelle definiert...

- a) z.B. USD (für USA) => 😊
- b) z.B. ABC => 😞

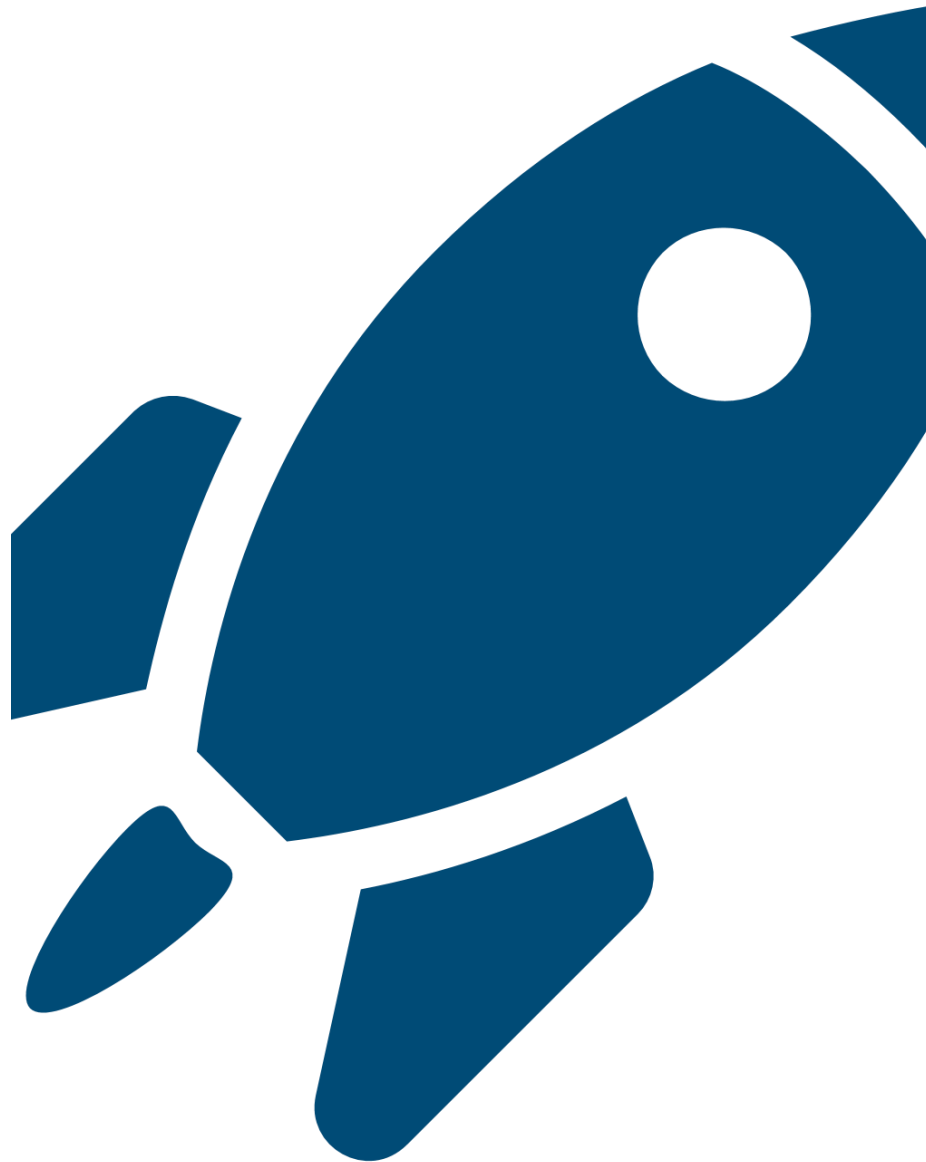


3 PL/SQL-Routinen:

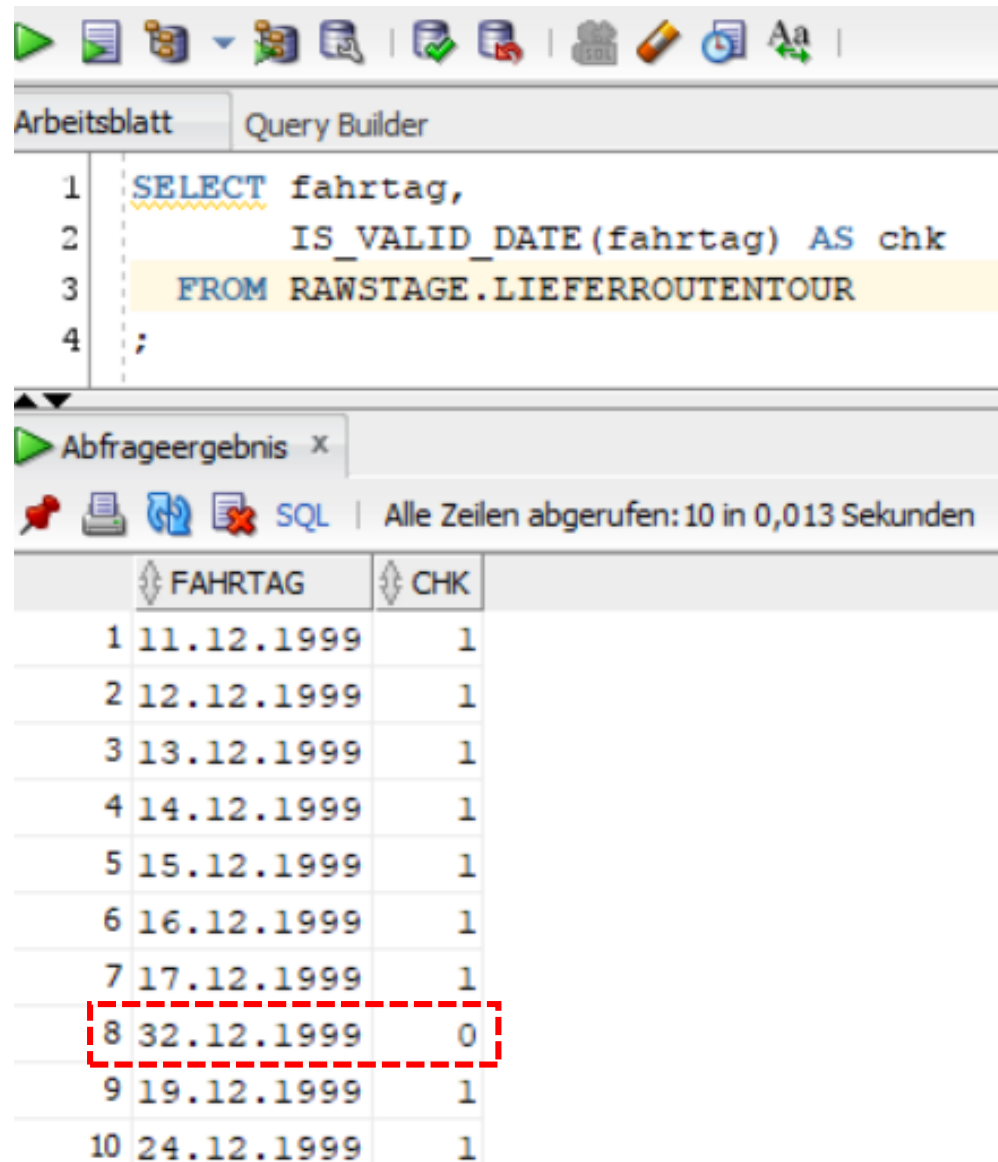
- FUNCTION
validiereTabelleninhalt
Generiert den eigentlichen
Validierungsbefehl, führt ihn aus
und gibt das erstellte Ergebnis
zurück.
- FUNCTION
validiereINFRAPruefregel
Checkt die Syntax des Prädikats
beim DML in der Metadatentabelle.
- PROCEDURE log
Führt Protokollierung durch.

Dynamisch erzeugter SELECT zur Validierung (Ergebnis von validiereTabelleninhalt)

```
SELECT CASE WHEN Gesamtanzahl_Fehler > 0
      THEN CASE WHEN substr(check1,1,1) <> '0' THEN check1 || CHR(13) END
      || CASE WHEN substr(check2,1,1) <> '0' THEN check2 || CHR(13) END
      || CASE WHEN substr(check3,1,1) <> '0' THEN check3 || CHR(13) END
      || CASE WHEN substr(check4,1,1) <> '0' THEN check4 || CHR(13) END
      || CASE WHEN substr(check5,1,1) <> '0' THEN check5 || CHR(13) END
      || CASE WHEN substr(check6,1,1) <> '0' THEN check6 || CHR(13) END
      || CASE WHEN substr(check7,1,1) <> '0' THEN check7 || CHR(13) END
      ELSE NULL
END AS Auswertung,
Gesamtanzahl_Fehler,
Gesamtanzahl_Zeilen
FROM (
  SELECT
    nv1(sum(CASE WHEN NOT (ArtikelNummer IS NOT NULL) THEN 1 ELSE 0 END),0) || 'x Spalte ArtikelNummer ist "NULL" (leer)' AS check1,
    nv1(sum(CASE WHEN NOT (length (ArtikelNummer) <= 10) THEN 1 ELSE 0 END),0) || 'x Spalte Artikelnummer ist zu lang! (>10 Zeichen)' AS check2,
    nv1(sum(CASE WHEN NOT (regexp_like (ArtikelNummer, '^X\d{9}$') OR regexp_like (ArtikelNummer, '^Y\d{6}$')) THEN 1 ELSE 0 END),0) || 'x Spalte ...' AS check3
    nv1(sum(CASE WHEN NOT (regexp_like (Staffell_EK_Preis, '^X[0-9 -]*($|[.,][0-9]*$)') THEN 1 ELSE 0 END),0) || 'x Spalte Staffell_EK_Preis ist ...' AS check4
    nv1(sum(CASE WHEN NOT (regexp_like (Staffell_VK_Preis, '^X[0-9 -]*($|[.,][0-9]*$)') THEN 1 ELSE 0 END),0) || 'x Spalte Staffell_VK_Preis ist ...' AS check5
    nv1(sum(CASE WHEN NOT (Staffell_EK_Preis IS NOT NULL) THEN 1 ELSE 0 END),0) || 'x Spalte Staffell_EK_Preis ist "NULL" (leer)' AS check6
    nv1(sum(CASE WHEN NOT (Staffell_VK_Preis IS NOT NULL) THEN 1 ELSE 0 END),0) || 'x Spalte Staffell_VK_Preis ist "NULL" (leer)' AS check7
    nv1(sum(CASE WHEN NOT (ArtikelNummer IS NOT NULL) THEN 1 ELSE 0 END),0) +
    nv1(sum(CASE WHEN NOT (length (ArtikelNummer) <=10) THEN 1 ELSE 0 END),0) +
    nv1(sum(CASE WHEN NOT (regexp_like (ArtikelNummer, '^X\d{9}$') OR regexp_like (ArtikelNummer, '^Y\d{6}$')) THEN 1 ELSE 0 END),0) +
    nv1(sum(CASE WHEN NOT (regexp_like (Staffell_EK_Preis, '^X[0-9 -]*($|[.,][0-9]*$)') THEN 1 ELSE 0 END),0) +
    nv1(sum(CASE WHEN NOT (regexp_like (Staffell_VK_Preis, '^X[0-9 -]*($|[.,][0-9]*$)') THEN 1 ELSE 0 END),0) +
    nv1(sum(CASE WHEN NOT (Staffell_EK_Preis IS NOT NULL) THEN 1 ELSE 0 END),0) +
    nv1(sum(CASE WHEN NOT (Staffell_VK_Preis IS NOT NULL) THEN 1 ELSE 0 END),0) AS Gesamtanzahl_Fehler,
    count(*) AS Gesamtanzahl_Zeilen
  FROM RAWSTAGE.ARTIKELPREISLISTE WHERE LaufNr = 4711
);
```



Überprüfung Datumformat (Beispiel für eigene, erweiterte Validierungslogik)



The screenshot shows a database query tool interface. At the top, there is a toolbar with various icons. Below it, a tab labeled 'Arbeitsblatt' and 'Query Builder' is visible. The query editor shows the following SQL query:

```
1 SELECT fahrtag,  
2     IS_VALID_DATE(fahrtag) AS chk  
3 FROM RAWSTAGE.LIEFERROUTENTOUR  
4 ;
```

Below the query editor, there is a tab labeled 'Abfrageergebnis x'. It shows the results of the query in a table format. The table has two columns: 'FAHRTAG' and 'CHK'. The results are as follows:

	FAHRTAG	CHK
1	11.12.1999	1
2	12.12.1999	1
3	13.12.1999	1
4	14.12.1999	1
5	15.12.1999	1
6	16.12.1999	1
7	17.12.1999	1
8	32.12.1999	0
9	19.12.1999	1
10	24.12.1999	1

The row with '32.12.1999' and '0' is highlighted with a red dashed border. Above the table, a status bar indicates 'Alle Zeilen abgerufen: 10 in 0,013 Sekunden'.

```
CREATE OR REPLACE
FUNCTION INFRA.is_valid_date
(
    iDATE_STRING VARCHAR2,
    iFORMATMASKE VARCHAR2 DEFAULT 'DD.MM.YYYY'
) RETURN NUMBER
IS
    v_date DATE;
BEGIN
    v_date := to_date(iDATE_STRING, iFORMATMASKE);
    RETURN 1;
EXCEPTION WHEN OTHERS THEN
    RETURN 0;
END is_valid_date;
/
```

```

PROCEDURE log (iBEFEHL      IN INFRA.Logging.Befehl%TYPE,
               iMELDUNG_VC2 IN INFRA.Logging.Meldung_VC2%TYPE,
               iMELDUNG_CLOB IN INFRA.Logging.Meldung_CLOB%TYPE,
               iKOMMENTAR    IN INFRA.Logging.Kommentar%TYPE
)
IS
    PRAGMA AUTONOMOUS_TRANSACTION;
BEGIN
    INSERT into INFRA.Logging (log_zeitpunkt, befehl, meldung_vc2, meldung_clob, kommentar)
        VALUES ( systimestamp, iBEFEHL, iMELDUNG_VC2, iMELDUNG_CLOB, iKOMMENTAR);

    COMMIT;
END log;
    
```

G_ID	LOG_ZEITPUNKT	BEFEHL	MELDUNG_VC2	MELDUNG_CLOB	KOMMENTAR
1	81 08.10.19 12:58:20,684788000	SELECT CASE WHEN Gesamtanzahl_Fehler > 0 ...	Es wurden 10 Datensätze gel...	Sumarisch...	generische Va
2	80 08.10.19 12:58:20,680613000	SELECT CASE WHEN Gesamtanzahl_Fehler > 0 ...	Gleich wird dieses SQL ausg...	(null)	generische Va
3	79 08.10.19 12:58:20,677804000	...wird gleich dynamisch als String zusammengebaut...	Für Tabelle = RawStage.ARTI...	(null)	(null)
4	78 08.10.19 12:53:36,533260000	SELECT CASE WHEN Gesamtanzahl_Fehler > 0 ...	Es wurden 10 Datensätze gelesen und dabei 7 Fehler bei der Validierung		ische Va
5	77 08.10.19 12:53:36,521372000	SELECT CASE WHEN Gesamtanzahl_Fehler > 0 ...	gefunden! Der Validierungsgesamtergebnistext ist in der Spalte		ische Va
6	76 08.10.19 12:53:36,518753000	...wird gleich dynamisch als String zusammengebaut...	iMELDUNG_CLOB hinterlegt		
7	75 08.10.19 12:53:36,517660000	SELECT CASE WHEN Gesamtanzahl_Fehler > 0 ...	Der Validierungsgesamtergeb...	(null)	generische Va

Sonstiges: Syntax-Validierung (1 von 2)

```
1 INSERT INTO INFRA.VALIDIERUNGSREGEL VALUES
2 (999,'Artikel_ArtikelNummer_größer_10Zeichen','RAWSTAGE','ARTIKEL',
3 'INFRA.Validierung_pkg.validiereINFRAPruefregel',
4 'LÄNGTH (ArtikelNummer) <=10',
5 'Spalte ArtikelNummer ist zu lang! (>10 Zeichen)',
6 to_date('30.09.19 11:01:58','DD.MM.RR HH24:MI:SS'),
7 'INFRA',null,null,null);
```

Skriptausgabe x

Aufgabe abgeschlossen in 0,035 Sekunden

Fehlerbericht -

ORA-20001: Der Inhalt der Spalte PRUEFBEDINGUNG ist N I C H T valide für die Tabelle im angegebenen Schema. Folgendes trat auf:
-904 Meldung: ORA-00904: "LÄNGTH": ungültige ID.
Folgender SQL-Testbefehl wurde ausgeführt: select CASE WHEN LÄNGTH(ArtikelNummer) <= 10 THEN 1 ELSE 0 END mein test from RAWSTAGE.ARTIKEL WHERE rownum=1
ORA-06512: in "INFRA.VALID_BIU_PRUEFBEDINGUNG_TRG", Zeile 20
ORA-04088: Fehler bei der Ausführung von Trigger 'INFRA.VALID_BIU_PRUEFBEDINGUNG_TRG'

```
FUNCTION validiereINFRAPruefregel (iSQL_BEFEHL VARCHAR2)
RETURN VARCHAR2
IS
    v_ergebnis    NUMBER;
    v_err_code     NUMBER;
    v_err_msg      VARCHAR2(1000);
BEGIN
    EXECUTE IMMEDIATE iSQL_BEFEHL INTO v_ergebnis;
    RETURN(null); -- D.h. alles gut, der INSERT/UPDATE wird durchgelassen! Fall: 1 Zeile kam zurück)
EXCEPTION
    WHEN NO_DATA_FOUND THEN -- Befehl konnte also erfolgreich, d.h. ohne Error, weil die Sytax falsch ist, ausgeführt werden.
        RETURN(null);      -- D.h. alles gut, der INSERT/UPDATE wird durchgelassen! Fall: keine Zeile kam zurück)
    WHEN OTHERS THEN -- z.B. Division durch 0, Spalte existiert nicht oder Ausdruck falsch oder ...
        v_err_code := SQLCODE;
        v_err_msg  := substr(SQLERRM, 1, 1000);
        RETURN('Der Inhalt der Spalte PRUEFBEDINGUNG ist N I C H T valide für die Tabelle im angegebenen Schema. Folgendes trat auf: '
            ||CHR(13)||v_err_code||' Meldung: '||v_err_msg||'.'||CHR(13)||'Folgender SQL-Testbefehl wurde ausgeführt: '||iSQL_BEFEHL);
END validiereINFRAPruefregel;
```



```
CREATE OR REPLACE TRIGGER INFRA.valid_BIU_Pruefbedingung_Trg
BEFORE
    INSERT OR UPDATE ON INFRA.Validierungsregel
    FOR EACH ROW
DECLARE
    v_SQL          VARCHAR2(4000);
    v_Ergebnis     VARCHAR2(4000);
    v_PruefBedingung VARCHAR2(4000);
BEGIN

    v_PruefBedingung := replace(:new.PruefBedingung,CHR(39), CHR(39)||CHR(39));

    v_SQL := 'SELECT CASE WHEN '||v_PruefBedingung||' THEN 1 ELSE 0 END mein_test from '||:new.SCHEMA||'.'||:new.TABELLE||' WHERE rownum=1';

    EXECUTE IMMEDIATE 'SELECT '||:new.PRUEFROUTINE||'( iSQL_BEFEHL => '''||v_SQL||''')
                        FROM DUAL'
        INTO v_Ergebnis
;

    IF v_Ergebnis IS NULL THEN /* D.h. der Befehl konnte fehlerfrei ausgeführt werden. */
        null;                -- => Die Sytax ist korrekt und der INSERT/UPDATE wird durchgelassen!
    ELSE -- Fehler wie z.B. Spaltenname falsch, Division durch 0, Ausdruck falsch u.s.w.
        raise_application_error(-20001, v_Ergebnis);
    END IF;
END valid_BIU_Pruefbedingung_Trg;
```

- External Tables
- Softere Plausibilisierung per Regeltypisierung (z.B. Warnung, Fehler...)
- Typische Checks statt als Regel codieren: per Spaltenattribut abbilden (z.B. NN)
- Dynamische Regelvalidierung beim Start des Frameworks (nicht nur beim Regel-INSERT)
- Erweiterung Datumsprüfung (um Zeitanteil plus Format-Strings)
- Erweiterung Zahlenprüfung (um Format-Strings)
- Kombinierbarkeit mit nachfolgender Bereinigung (Singletons, Cleansing...)
- Multitabellen-Plausibilisierung: Kontextbezogen (z.B. durch Tabellengruppierung)
- APEX GUI (Pflege der Validierungsdaten sowie Sicht auf Log-Ergebnisse)
- ...



elegant
und flexibel



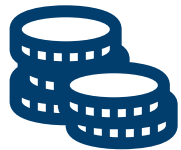
schnell
& einfach
implementiert



Vorteil DB:
integriert +
erweiterbar



transparent
für DBA
und Betreiber



keine
Lizenzen,
HW, SW



Übersicht,
Pflegeaufwand
=> GUI???



Kooperation
mit EAI /
Lieferanten



Zeit
für Tests

Fragen?



merlin.zwo

Wir kümmern uns!



merlin.zwo InfoDesign GmbH & Co. KG
Stefan Winkler

Elsa-Brändström-Straße 14
76228 Karlsruhe
Tel. 0721 – 132 096 41

stefan.winkler@merlin-zwo.de
<http://www.merlin-zwo.de>

