



Anwendungsmöglichkeiten der Oracle 12c Neuerungen für Entwickler

DOAG 2014 Development
Markus Fiegler
ORDIX AG

info@ordix.de
www.ordix.de

- SQL-Neuerungen in Oracle 12c
- PL/SQL-Neuerungen in Oracle 12c
- Weitere Neuerungen
- Resümee

- **DEFAULT-Constraint (mit ext. Seq.):**
 - `DEFAULT [ON NULL] <SEQ>.NEXTVAL`
 - Default-Werte werden auch ohne NOT NULL-Constraint in DD gespeichert.

- **Identity Columns:**

- `GENERATED ALWAYS AS IDENTITY`
- `GENERATED BY DEFAULT AS IDENTITY`
- `GENERATED BY DEFAULT ON NULL AS IDENTITY`

```
ALTER TABLE ...
MODIFY ...
START WITH
LIMIT VALUE ...
```

	kein Wert	Wert z.B. 5	NULL	DEFAULT
DEFAULT MA_SEQ.NEXTVAL (ext. Seq.)	Seq.-Wert	5	NULL	Seq.-Wert
DEFAULT ON NULL MA_SEQ.NEXTVAL (ext. Seq.)	Seq.-Wert	5	Seq.-Wert	Seq.-Wert
GENERATED ALWAYS AS IDENTITY	Seq.-Wert	ORA-32795	ORA-32795	Seq.-Wert
GENERATED BY DEFAULT AS IDENTITY	Seq.-Wert	5	ORA-01400	Seq.-Wert
GENERATED BY DEFAULT ON NULL AS IDENTITY	Seq.-Wert	5	Seq.-Wert	Seq.-Wert

NOT NULL Cons.

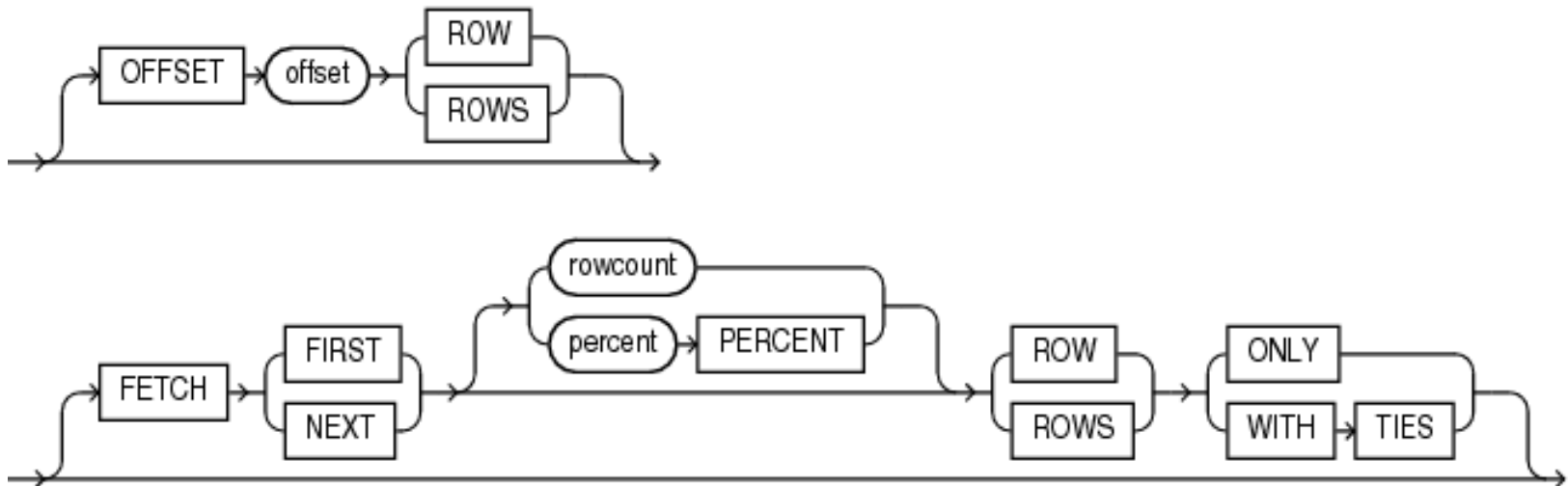
Einschränkungen:

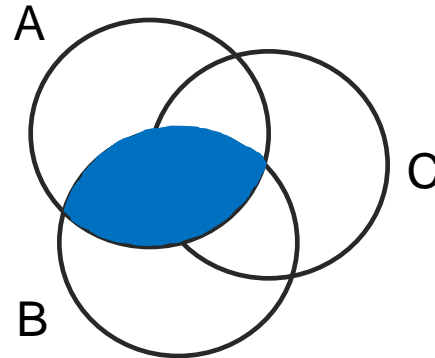
- Kein FOR UPDATE
- Kein CURRVAL und NEXTVAL
- Kein incr. refresh bei MView

nr	name
1	ma1
2	ma2
3	ma3
4	ma4
5	ma5

SELECT ... ORDER BY ...
 FETCH FIRST 2 ROWS ONLY

SELECT ... ORDER BY ...
 OFFSET 2 ROWS FETCH NEXT 2
 ROWS ONLY





- Bis Oracle 11g nur ANSI Outer-Join Syntax

```
-- ANSI Outer-Join Syntax...  
SELECT * FROM a JOIN b ON (a.nr = b.nr)  
LEFT OUTER JOIN c ON (a.nr = c.nr AND b.nr = c.nr);
```

- Ab Oracle 12c auch Native Outer-Join Syntax

```
-- Native Outer-Join Syntax...  
SELECT * FROM a, b, c  
WHERE a.nr = b.nr AND a.nr = c.nr(+) AND b.nr = c.nr(+);
```

ANSI Join-Erweiterungen: CROSS APPLY, OUTER APPLY und LATERAL

- Syntax:

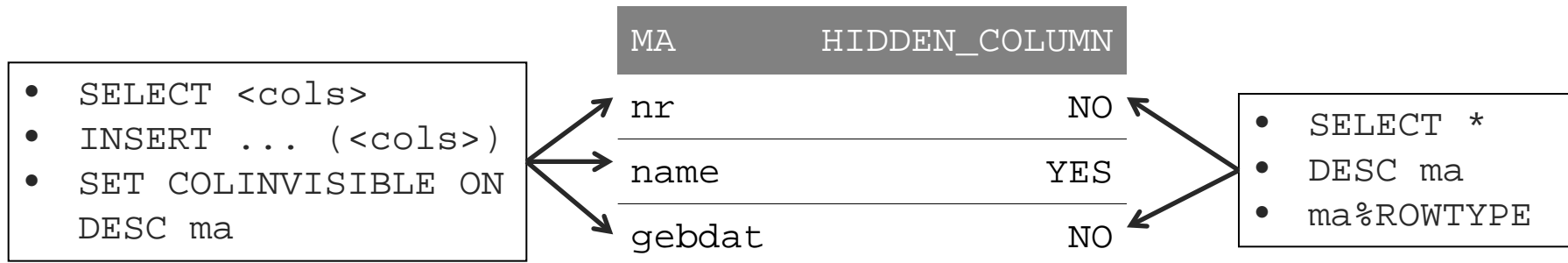
```
[cross|outer] apply [table_reference | collection_expression]  
lateral (subquery)
```

- Left Correlation Support (LCS)

```
select * from abt, LATERAL (  
  select * from ma where ma.abtnr=abt.nr  
);
```

```
select * from abt cross apply f_ma_namen(abt.nr);
```

- hidden columns != invisible columns



```
ALTER TABLE ... name VISIBLE;  
SELECT * FROM ma;
```

```
nr gebdat name  
-----  
...
```

nicht anwendbar bei:

- External tables
- Cluster tables
- Temporary tables

```
CREATE TABLE projekt(  
  nr      NUMBER,  
  name    VARCHAR(30),  
  start   DATE,  
  ende    DATE,  
  PERIOD FOR prj_zeit(  
    start, ende)  
);
```

Besonderheiten:

- fachlicher Gültigkeitszeitraum
- Statement-Ebene:
=> flashback_query_clause
- Session-Ebene:
=> DBMS_FLASHBACK_ARCHIVE.
ENABLE_AT_VALID_TIME
- Start- und Ende-Spalten optional

nr	name	start	ende
1	prj1	01.06.2014	10.06.2014
2	prj2	10.06.2014	20.06.2014
3	prj3		20.06.2014
4	prj4	10.06.2014	
5	prj5	20.06.2014	25.06.2014

```
SELECT * FROM projekt  
AS OF PERIOD FOR prj_zeit  
TO_DATE('15.06.2014',...
```

```
SELECT * FROM projekt  
VERSIONS PERIOD FOR prj_zeit  
BETWEEN  
TO_DATE('23.06.2014',...  
AND TO_DATE('30.06.2014',...
```



```
CREATE TABLE ma (  
  nr      number,  
  name    varchar2(20)  
) ROW ARCHIVAL;
```

Besonderheiten:

- Archivierung von Datensätzen

- unsichtbare Spalte:

ORA_ARCHIVE_STATE VARCHAR2(4000)

=> Wert '0' (aktiv)

=> ansonsten (inaktiv bzw. archiviert)

ora_archive _state	nr	name
-----------------------	----	------

0	10	Meier
---	----	-------

0	11	Hansen
---	----	--------

1	12	Baier
---	----	-------

```
ALTER SESSION SET
```

```
  ROW ARCHIVAL VISIBILITY = ACTIVE;
```

```
SELECT * FROM ma;
```

```
ALTER SESSION SET
```

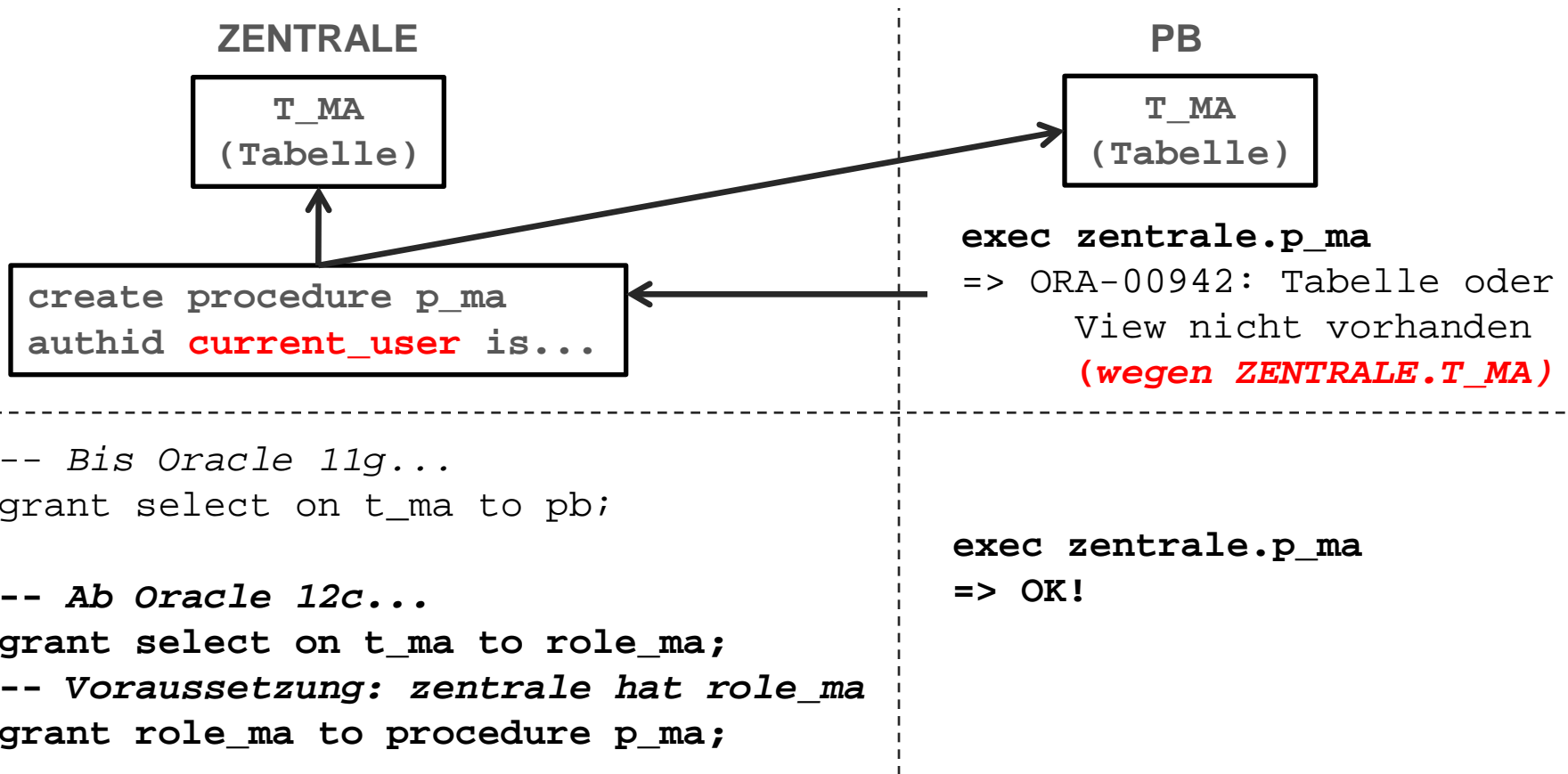
```
  ROW ARCHIVAL VISIBILITY = ALL;
```

```
SELECT * FROM ma;
```

- Zeilenübergreifende Mustererkennung
- Fragestellungen in Bereichen z.B. Aktienkurse, Sensordaten, Online-Shop

Bereiche	Beschreibung
MATCH_RECOGNIZE (... FROM < MATCH_RECOGNIZE > WHERE ...
row_pattern_partition_by	Unterteilung der Daten in Partitionen
row_pattern_order_by	Sortierung der Daten innerhalb der Partitionen
row_pattern_measures	Ausgabe-Felder und Berechnungen
row_pattern_rows_per_match	Rückgabe der Zeilen (eine Zeile oder alle Zeilen pro Mustertreffer)
row_pattern_skip_to	Wiederaufnahme der Suche nach einem Mustertreffer
PATTERN (row_pattern) [row_pattern_subset_clause]	Musterdefinition, z.B. PATTERN (GELB ROT)
DEFINE row_pattern_definition_list)	Mustervariablen, z.B. ROT AS rot.farbe = 'ROT'

- SQL-Neuerungen in Oracle 12c
- PL/SQL-Neuerungen in Oracle 12c
- Weitere Neuerungen
- Resümee



ZENTRALE

T_MA
(Tabelle)

```
create function f_ma  
return varchar2 authid  
current_user is...
```

```
create view v_ma  
as select f_ma ...
```

```
create view v_ma_beq  
bequeath current_user  
as select f_ma ...
```

PB

T_MA
(Tabelle)

```
-- Ab Oracle 12c  
select * from  
zentrale.v_ma_beq;
```

```
-- Bis Oracle 11g  
select * from  
zentrale.v_ma;
```

Zugriffsbeschränkung mit ACCESSIBLE BY-Klausel

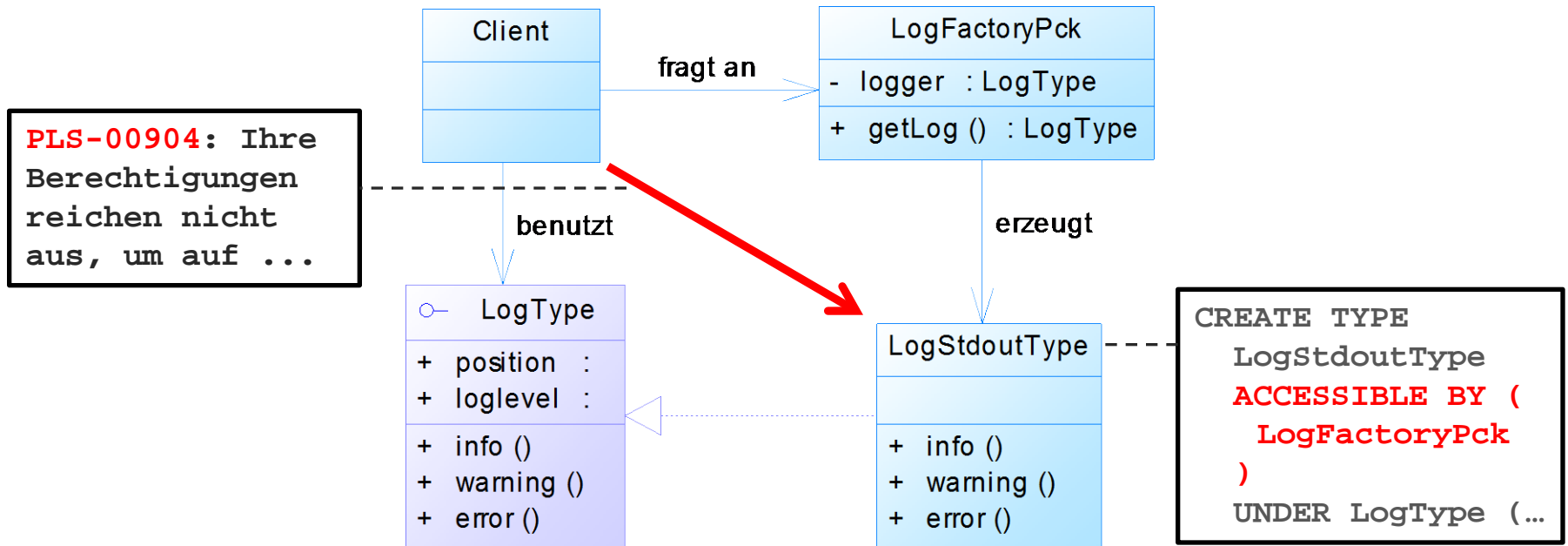
```
ACCESSIBLE BY ([<unit_kind>] [<schema>.]<unit_name>, ...)
```

Besonderheiten:

- Whitelist nur auf Schemaebene (kein INDEX und VIEW)
- SYS hat keine Sonderrechte

Anwendungsmöglichkeiten:

- Hilfs-Funktionalität
- Unbefugten Zugriff verweigern
- Singleton bei Objekttypen



- Welcher Fehler ist geworfen? (Oracle 7)

`DBMS_UTILITY.FORMAT_ERROR_STACK()`

```
ORA-01403: Keine Daten gefunden
...
```

UTL_CALL_STACK (Oracle 12c)

1403

Keine Daten gefunden

- Wo ist der Fehler aufgetreten? (Oracle 10)

`DBMS_UTILITY.FORMAT_ERROR_BACKTRACE()`

```
ORA-06512: in "MF.PKG1", Zeile 6
...
```

MF.PKG1

6

- Wie war der Programmverlauf? (Oracle 7)

`DBMS_UTILITY.FORMAT_CALL_STACK()`

```
----- PL/SQL Call Stack -----
object      line object
handle     number name
07FF83A551D8  8 procedure MF.P_LOG
07FF83C93528 10 package body MF.PKG1
```

10

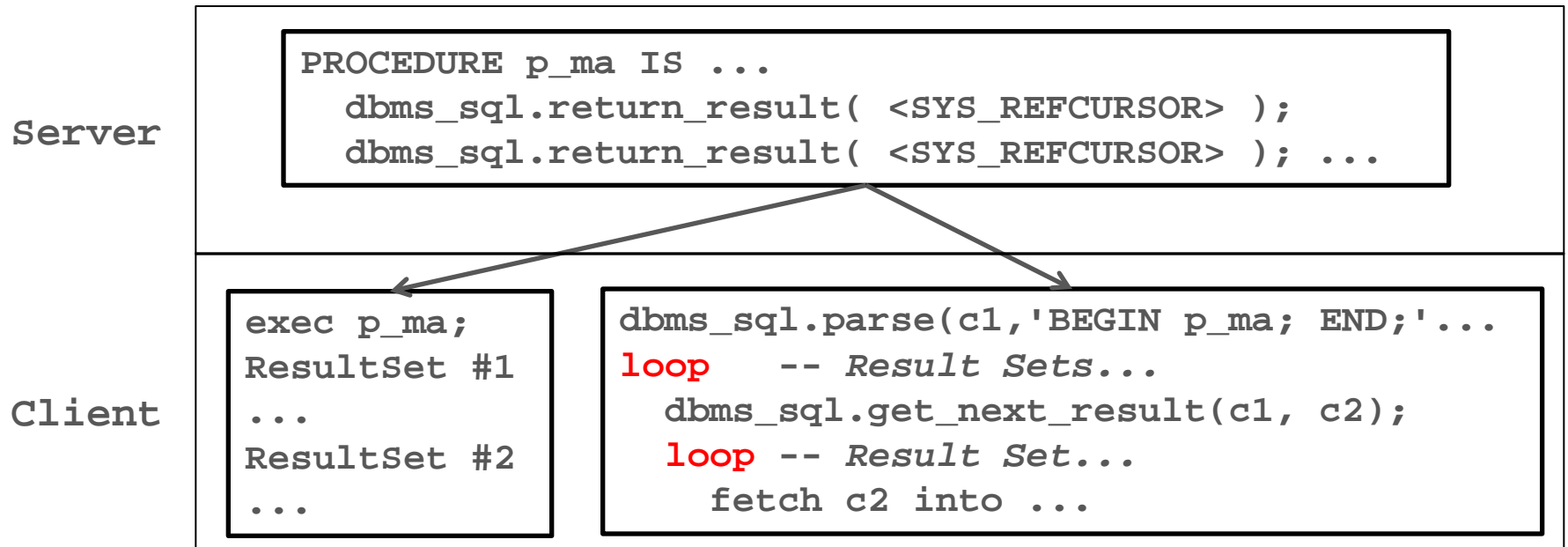
PKG1.P1.P1_SUBP1

- PL/SQL-WITH-Klausel = SQL-WITH-Klausel mit PL/SQL-Code
- Verwendung in Unterabfragen oder in DML-Statement:
 - => **ORA-32034: Nicht unterstützte Benutzung von WITH-Klausel**
 - Lösung: `/*+ WITH_PLSQL */` Hint
- PL/SQL => Kein Support im statischen SQL (nur dynamisches SQL)
- Bei SQL*Plus < 12.1 => *set sqlterminator off* oder z.B. *set sqlterminator #*

```
WITH  
  PROCEDURE p_test ...  
  FUNCTION f_date_diff ...  
    END f_date_diff;  
  SELECT f_date_diff( gebdat ) ... ;  
/
```

← Ausführung mit "/"

- Explicit Result Sets
 - PROCEDURE p_ma(p_cur OUT SYS_REFCURSOR) IS ...
- Implicit Result Sets (ab Oracle 12c)
 - PROCEDURE p_ma IS ...



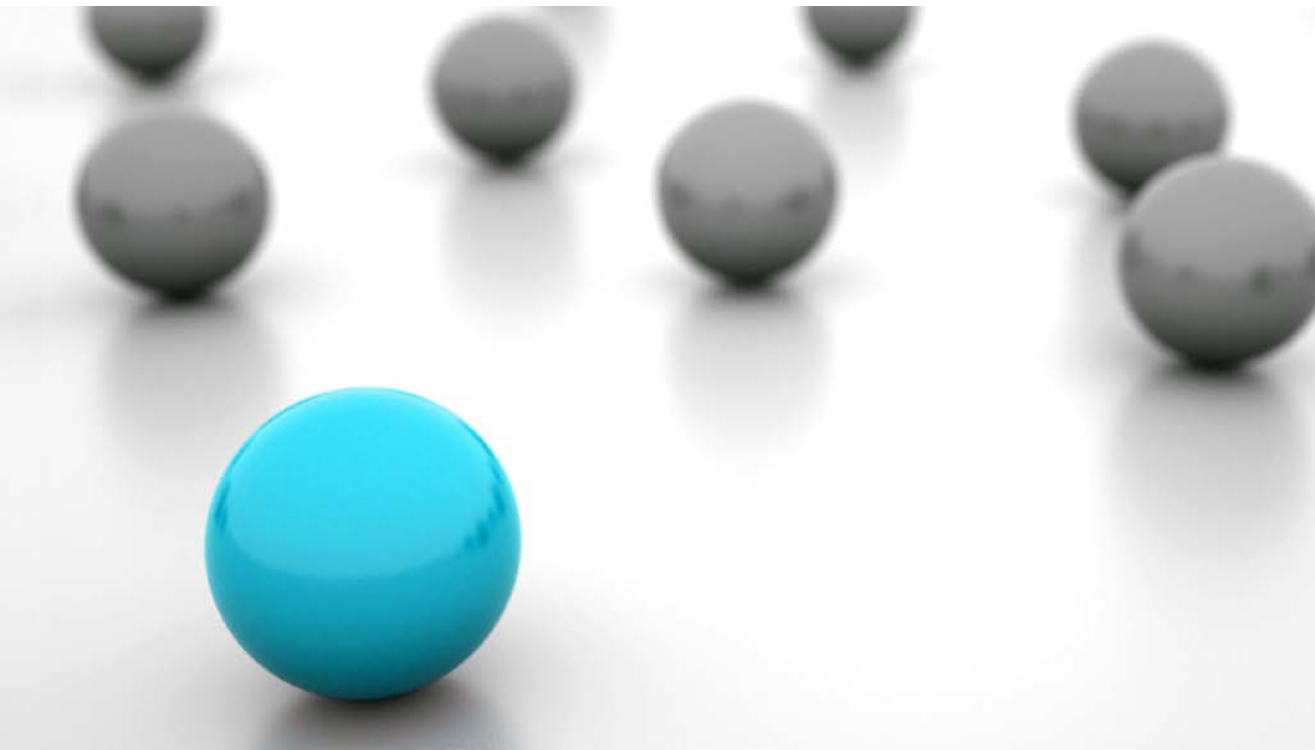
- SQL-Neuerungen in Oracle 12c
- PL/SQL-Neuerungen in Oracle 12c
- Weitere Neuerungen
- Resümee

- Extended Data Types
- Statistikerstellung bei BULK-Operationen:
 - `CREATE TABLE ... AS SELECT`
 - `INSERT INTO ... SELECT` (leere Tabelle mit direct-path insert)
- SQL Translation Framework
- Sequence-Optionen (`KEEP / NOKEEP` und `SESSION /GLOBAL`)
- SQL-Interface-Erweiterung (PL/SQL-Only Datentypen in SQL)
- Data Redaction (Sensible Daten unkenntlich machen)

- Neue Predefined Inquiry Directives:
 - `$$PLSQL_UNIT_OWNER` und `$$PLSQL_UNIT_TYPE`
- Result Cache für PL/SQL-Funktionen mit Ausführer-Rechten:
 - Zusätzliche Speicherung der Identität des Ausführers
- Java in der Datenbank (Oracle12c)
 - JavaSE6
 - Upgrade der Datenbank-JVM
 - JavaScript-Code
- View-Referenzierungen auflösen mit `DBMS_UTILITY.EXPAND_SQL_TEXT`
- PRAGMA UDF („user defined function“)

- SQL-Neuerungen in Oracle 12c
- PL/SQL-Neuerungen in Oracle 12c
- Weitere Neuerungen
- Resümee

- Verbesserungen im Bereichen von:
 - Anwendungssicherheit
 - Funktionalität
 - Kompatibilität
 - Performance
- Für die Anwendung transparente Funktionserweiterungen:
 - Weniger Code
 - Bessere Wartbarkeit
 - Weniger Fehler



Zentrale Paderborn
Westernmauer 12 - 16
33098 Paderborn
Tel.: 05251 1063-0

Seminarzentrum Wiesbaden
Kreuzberger Ring 13
65205 Wiesbaden
Tel.: 0611 77840-00

Zentrales Fax:
0180 1 67349 0
0180 1 ORDIX 0

Weitere Geschäftsstellen
in Köln, Münster und Neu-Ulm

E-Mail: info@ordix.de
Internet: <http://www.ordix.de>

Vielen Dank für Ihre Aufmerksamkeit!