

robotron[®]

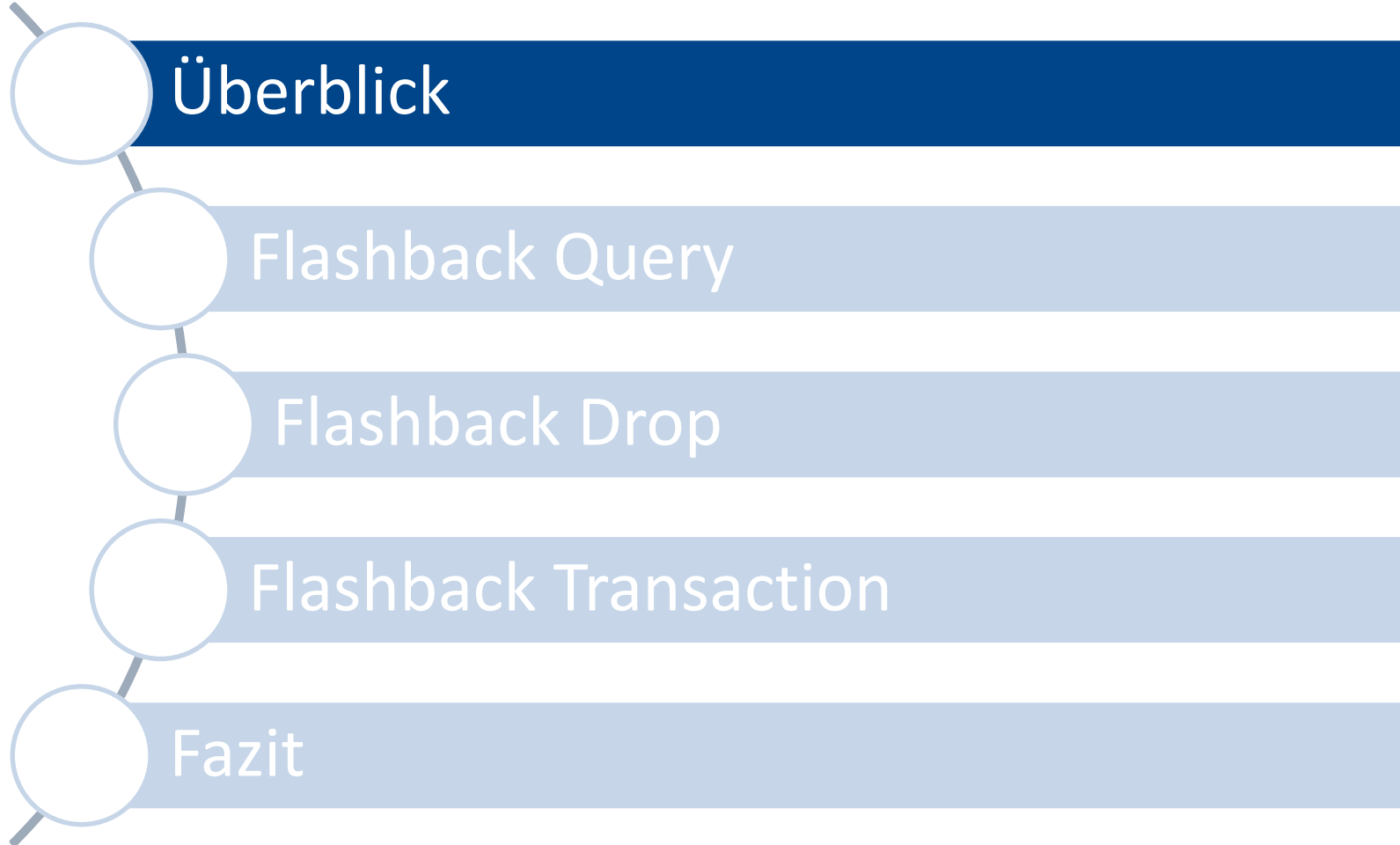
DOAG Database

Flashback – Recovery ohne Recovery

Marco Mischke – 03.06.2014



Agenda



Überblick

- ▶ **Betriebssicht des DBAs**
 - DBA verantwortet den reibungslosen Datenbank Betrieb
 - DBA plant Backups für Desaster Fälle
 - DBA implementiert Standby Systeme für schnelles Recovery
 - DBA testet und trainiert verschiedene Recovery Szenarien
- ▶ **Häufigkeit von Fehlern die ein Restore / Recover erfordern**
 - Weniger als 1x / Jahr

Überblick

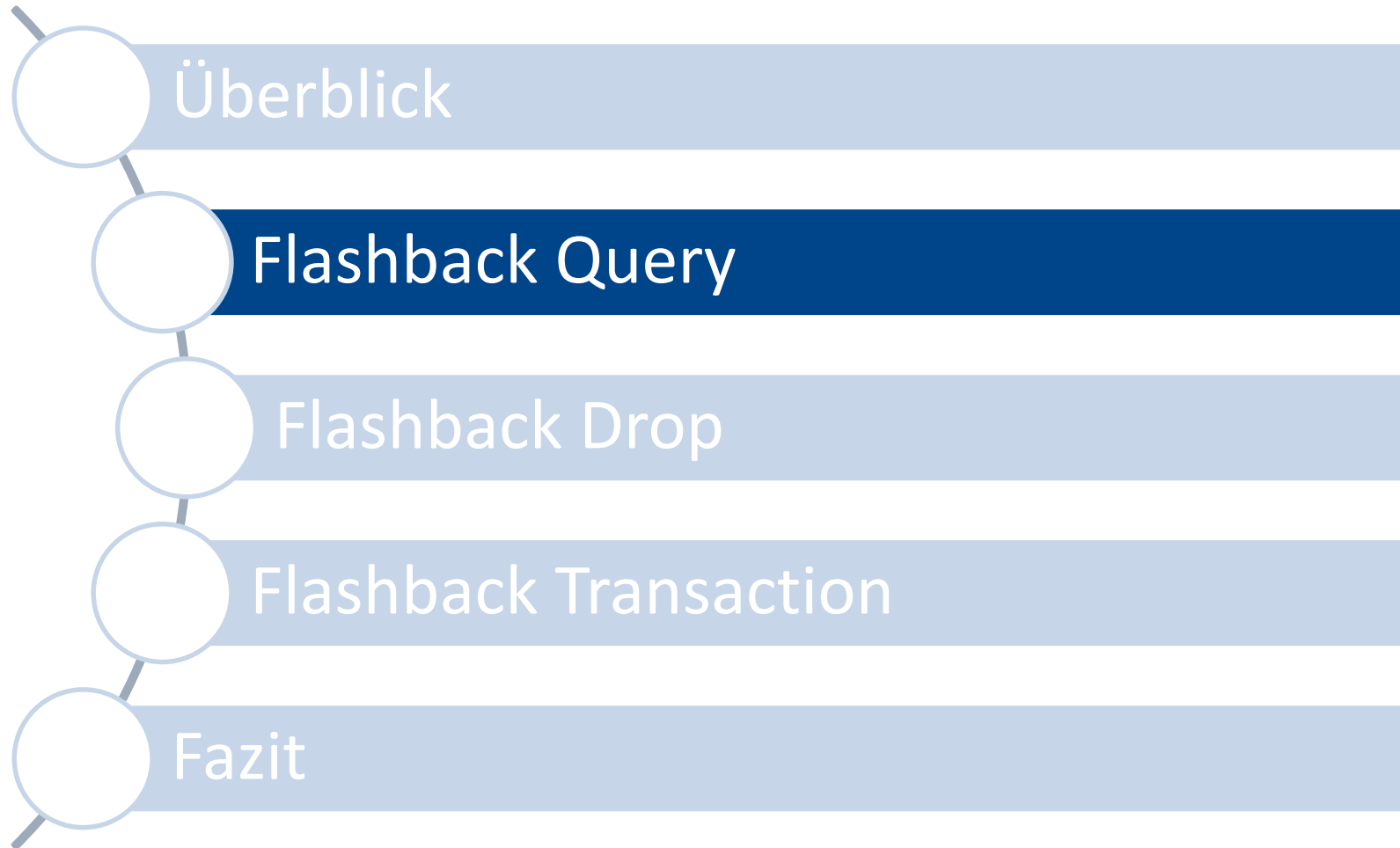
- ▶ Entwickler- / Anwendersicht
 - Wie die Datenbank intern organisiert ist, ist nicht relevant
 - Erzeugt, ändert und löscht Objekte und Dateninhalte
 - Will im Fehlerfall den Zustand zum Zeitpunkt X
- ▶ Häufigkeit von inhaltlichen Fehlern
 - Betrieb: mehrmals / Jahr
 - Entwicklung: bis hin zu mehrmals / Woche

Überblick

- ▶ DBA Ansatz um Anwenderfehler zu beheben:
 - Point in Time Recovery
 - Flashback Database
- ▶ Aufwändig
- ▶ Langwierig
- ▶ Fehleranfällig
- ▶ Beeinträchtigung der DB Verfügbarkeit (bei Flashback Database)

→ Flashback 😊

Agenda



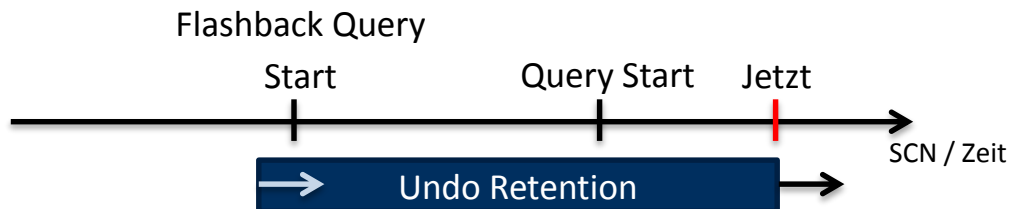
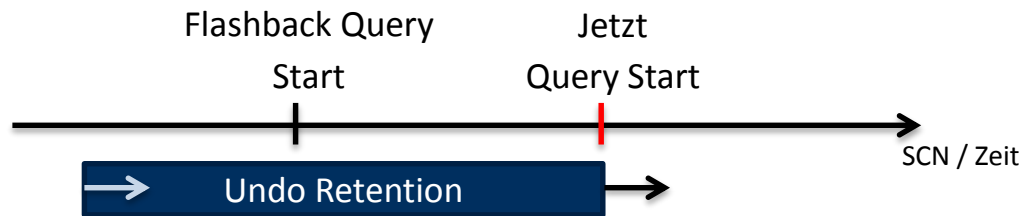
Flashback Query

- ▶ Voraussetzungen / Standards
 - UNDO Management = AUTO
 - Vorhaltezeit (undo_retention) per Standard = 15 Minuten
 - UNDO Tablespace mit AUTOEXTENT = ON
- ▶ Anpassungen
 - UNDO Tablespace mit fester Größe
 - Vorhaltezeit wird automatisch an die Größe angepasst
 - Platz im Tablespace wird voll ausgeschöpft

Flashback Query

► Funktion

- Wiederverwenden des „normalen“ konsistenten Lesens
 - Normal: SCN beim Start der Query
 - Flashback: SCN vom Zeitpunkt X



Flashback Query

▶ Anwendungsmöglichkeiten

- create table ... as select ...

```
SQL> create table emp_old
  2  as
  3  select *
  4  from emp as of timestamp
  5  to_timestamp('2014-05-01 11:00:00',
  6  'yyyy-mm-yy hh24:mi:ss');
```

- Datenstand ist erst einmal gesichert
- Anwender / Entwickler kann in Ruhe prüfen

Flashback Query

▶ Anwendungsmöglichkeiten

- Flashback Query Kombinationen

```
SQL> select * from emp
      2  minus
      3  select *
      4  from emp as of timestamp
      5  to_timestamp('2014-05-01 11:00:00',
      6  'yyyy-mm-yy hh24:mi:ss');
```

- Ermittlung von Differenzen möglich

Flashback Query

► Anwendungsmöglichkeiten

- Flashback Query – Zurückspielen von Daten

```
SQL> update emp e_live
2   set sal = (select sal
3               from emp as of timestamp
4                   to_timestamp('2014-05-01 11:00:00',
5                               'yyyy-mm-dd hh24:mi:ss') e_orig
6               where e_orig.empno = e_live.empno
7           )
8   ;
```

```
SQL> commit;
```

→ Flashback Query ist in SQL komplett transparent verwendbar

Flashback Query

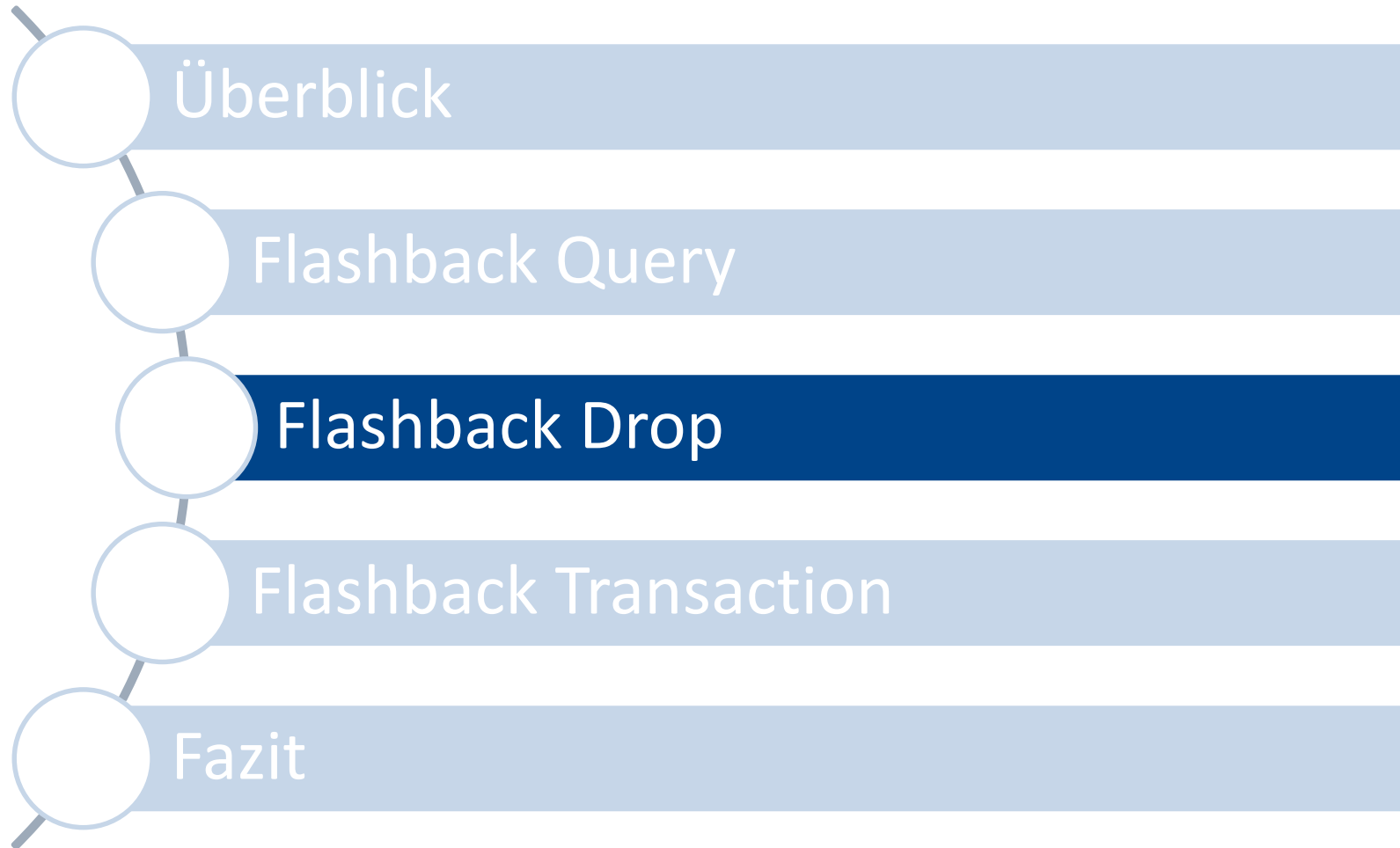
▶ Anwendungsmöglichkeiten

– Flashback Export

```
$ expdp system/Oracle-1 dumpfile=scott_flashback.dmpdp \
logfile=scott_flashback.expdp.log \
directory=data_pump_dir \
flashback_time='2014-05-01-11:00:00' schemas=scott
```

→ Sicherung von ganzen Schemas zum Zeitpunkt X

Agenda



Flashback Drop

- ▶ Retten von gelöschten Tabellen
- ▶ Nutzt nicht UNDO
- ▶ Segmente werden nicht gelöscht, nur umbenannt
- ▶ Werden erst überschrieben, wenn kein „echter“ freier Platz mehr vorhanden ist
- ▶ Regelbar über Parameter „recyclebin“, Standard = ON

Flashback Drop

► Anzeigen des Inhaltes

```
SQL> show recyclebin
```

ORIGINAL NAME	RECYCLEBIN NAME	OBJ. TYPE	DROP TIME
EMP	BIN\$yxLVQQOqC6rgQBAktBUK1A==\$0	TABLE	2014-05-02:13:29:19

– Nur Tabellen zu sehen

```
SQL> select ORIGINAL_NAME, OBJECT_NAME, OPERATION, DROPTIME  
2 from USER_RECYCLEBIN;
```

ORIGINAL_NAME	OBJECT_NAME	OPERATION	DROPTIME
EMP	BIN\$yxLVQQOqC6rgQBAktBUK1A==\$0	DROP	2014-05-02:13:29:19
PK_EMP	BIN\$yxLVQQOpC6rgQBAktBUK1A==\$0	DROP	2014-05-02:13:29:19
IX_ENAME	BIN\$yxLVQQOoC6rgQBAktBUK1A==\$0	DROP	2014-05-02:13:29:19
IX_MGR	BIN\$yxLVQQOnC6rgQBAktBUK1A==\$0	DROP	2014-05-02:13:29:19

– Alle Objekte sichtbar

Flashback Drop

▶ Zurückholen der Tabelle

```
SQL> flashback table emp to before drop;
```

Flashback abgeschlossen.

```
SQL> select ORIGINAL_NAME, OBJECT_NAME, OPERATION, DROPTIME  
       2 from USER_RECYCLEBIN;
```

Es wurden keine Zeilen ausgewählt.

▶ Ok, Indexe sind auch wieder da....

Flashback Drop

▶ Mal nachsehen...

```
SQL> select index_name from user_indexes  
2  where table_name='EMP';
```

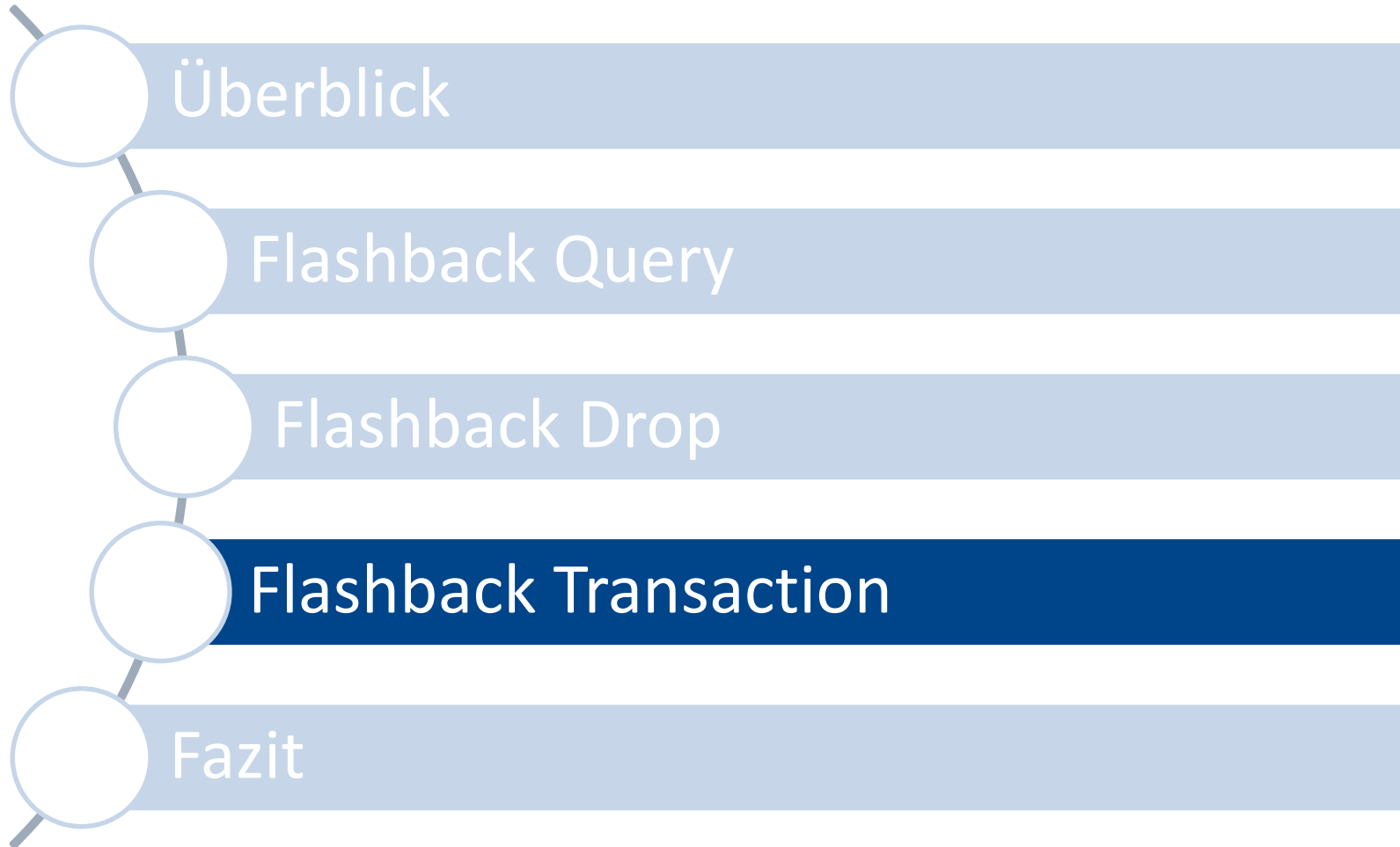
```
INDEX_NAME  
-----  
BIN$yxLVQQOoC6rgQBAKtBUK1A==$0  
BIN$yxLVQQOnC6rgQBAKtBUK1A==$0  
BIN$yxLVQQOpC6rgQBAKtBUK1A==$0
```

▶ Also doch noch umbenennen....

```
SQL> alter index "BIN$yxLVQQOoC6rgQBAKtBUK1A==$0" rename to IX_ENAME;
```

▶ Fremdschlüssel Constraints fehlen dennoch

Agenda



Flashback Transaction Query (EE)

- ▶ Zeigt den Verlauf eines Datensatzes an

```
SQL> select
  2     versions_starttime, versions_endtime,
  3     versions_operation, versions_xid,
  4     sal
  5 from
  6     emp versions between timestamp
  7         systimestamp - interval '1' hour and
  8         systimestamp
  9 where
 10     ename = 'MILLER';
```

VERSIONS_STARTTIME	VERSIONS_ENDTIME	V	VERSIONS_XID	SAL
02.05.14 14:35:41		U	07000B00F7480000	1800
02.05.14 14:29:29	02.05.14 14:35:41	U		1300
02.05.14 14:29:29				1504,91

Flashback Transaction Query (EE)

- ▶ Auch INSERTs und DELETES können ermittelt werden

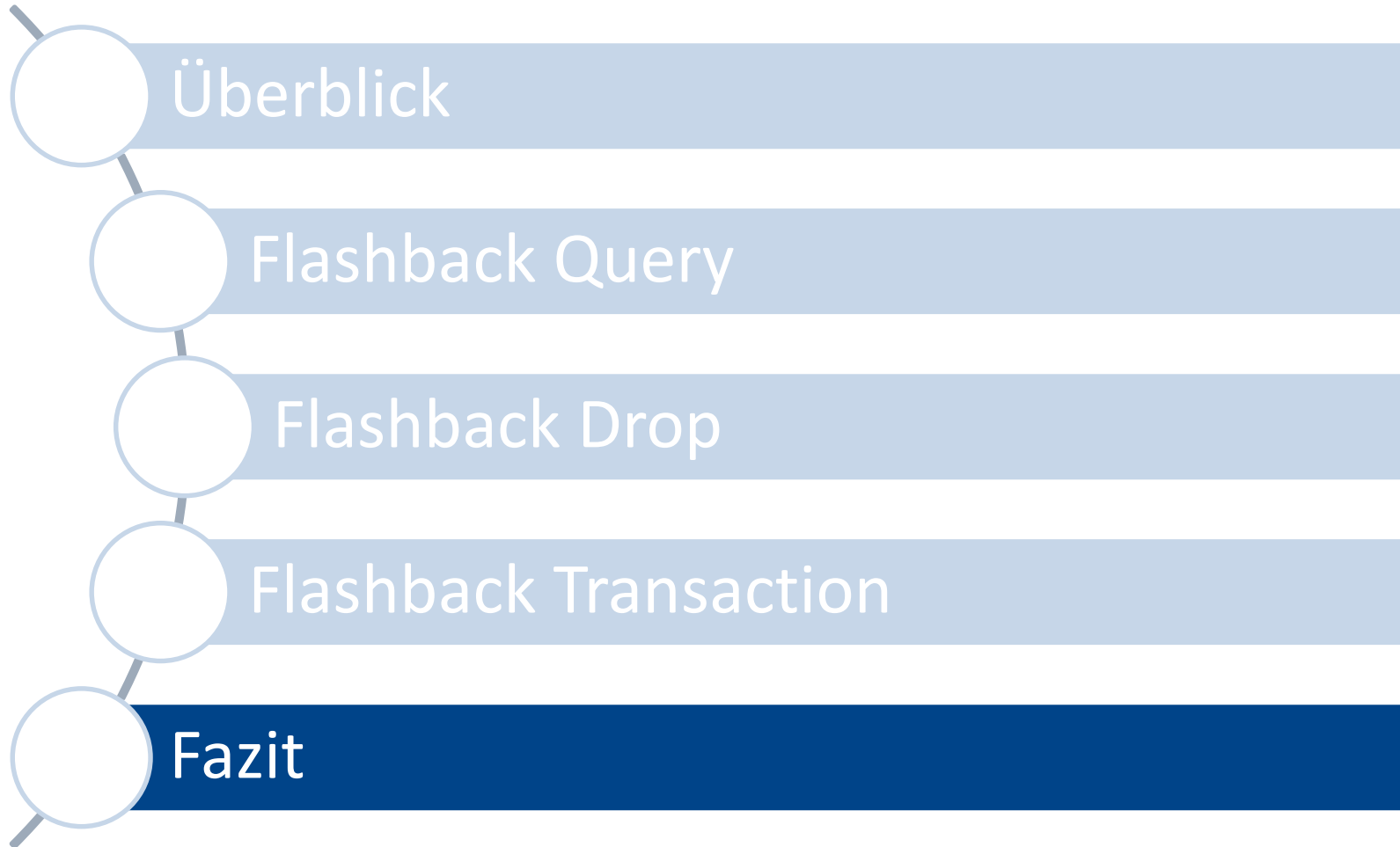
```
SQL> select
  2     versions_starttime, versions_endtime,
  3     versions_operation,
  4     sal
  5 from
  6     emp versions between timestamp
  7         systimestamp - interval '1' hour and
  8         systimestamp
  9 where
 10     ename = 'FOO';
```

VERSIONS_STARTTIME	VERSIONS_ENDTIME	V	SAL
02.10.12 15:30:05		D	5046
02.10.12 14:57:08	02.10.12 15:30:05	U	5046
02.10.12 14:45:05	02.10.12 14:57:08	I	4711

Flashback Transaction (EE)

- ▶ Macht ganze Transaktionen rückgängig
- ▶ Voraussetzungen
 - Archivelog Mode
 - Supplemental Logging
- ▶ Entfernen von Transaktionen mittels **DBMS_FLASHBACK.TRANSACTION_BACKOUT**
 - XID erforderlich
 - Wahlweise Beibehalten oder Zurückrollen der abhängigen Transaktionen

Agenda



Fazit

- ▶ Inhaltliche Fehler sind die häufigsten Fehler
- ▶ Flashback ist das erste Mittel der Wahl
 - Keine Beeinträchtigung der Verfügbarkeit
 - Einfache Handhabung
 - Kein administrativer Overhead
 - Vieles bereits in SE verfügbar



Referent

Marco Mischke

Senior Systemberater Support

T: +49 351 25859-2884

F: +49 351 25859-3699

marco.mischke@robotron.de