

## Oracle GoldenGate

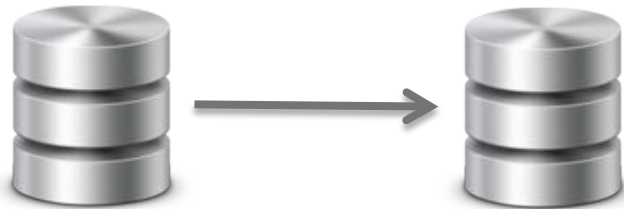
DOAG Webinar  
Thilo Fleischhauer  
13.11.2015, Wiesbaden



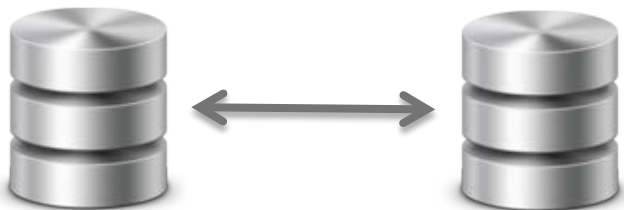
- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

## Verfahren zum Verteilen und Synchronisieren von Datenbeständen

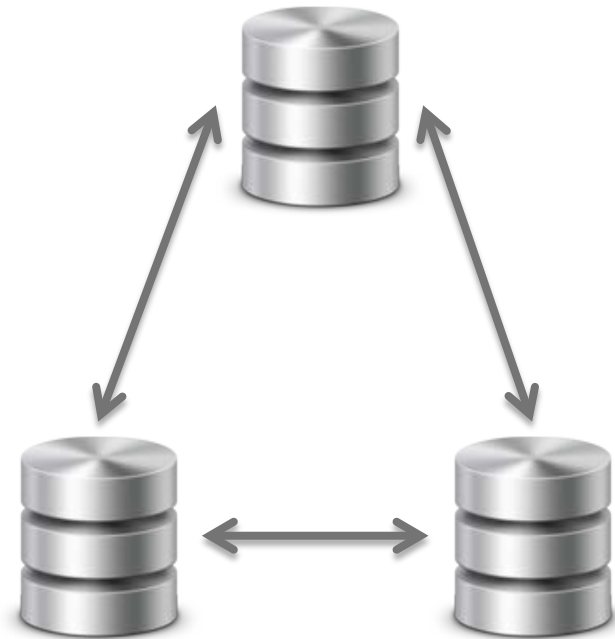
### Unidirektional/Offloading



### Bidirektional



### Multimaster



# Replikationsprodukte

- Oracle GoldenGate
- Oracle Streams
- Active Data Guard
- Logical Data Guard
- Materialized Views
  
- Nicht-Oracle Produkte
  - Dell (ehem. Quest) SharePlex for Oracle
  - IBM DataMirror Transformation Server
  - IBM CDC Infosphere
  - Dbvisit replicate
  - Tungsten Replicator

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

- Capture, Routing, Transformation und Delivery von Datenbanktransaktionen in heterogenen Umgebungen
- Erfassung aus den Transaktionslogs, z. B. Oracle Redo Logs
- Initial Load: Möglichkeit der Erfassung direkt aus der Tabelle

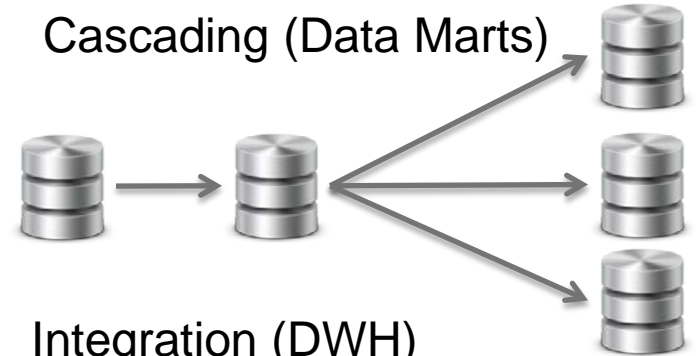


# Topologien

Unidirektional (Query Offloading)



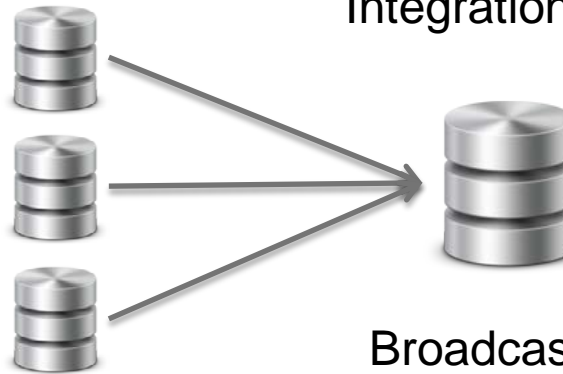
Cascading (Data Marts)



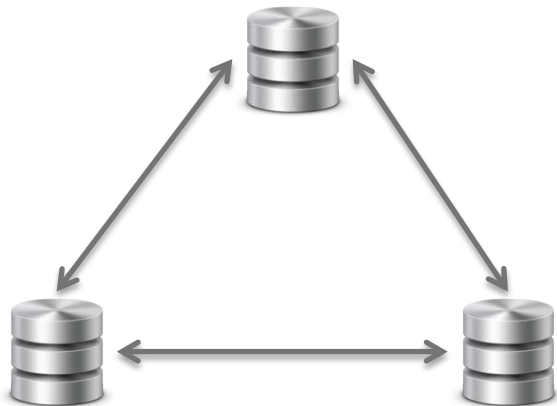
Bidirektional (Hochverfügbarkeit)



Integration (DWH)



Peer-to-Peer (Multimaster)



Broadcasting (Data Distribution)



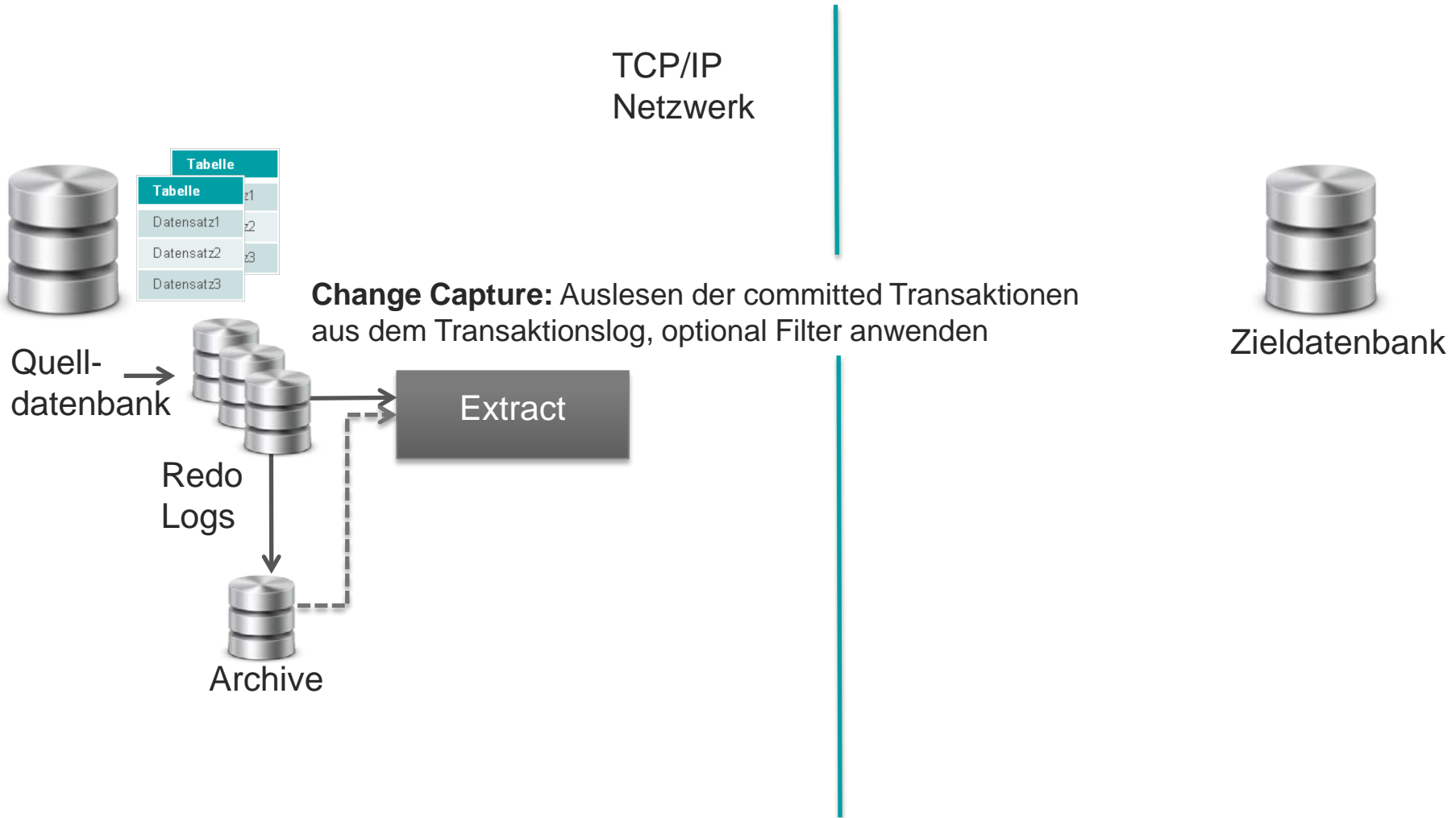
# Einsatzmöglichkeiten

- Minimierung Downtime („Zero“ Downtime)
  - Daten Migration
  - Datenbank Upgrade
- Cross Platform Migration
- Operational Reporting
- Query Offloading
- Extract, Transformation und Load (ETL) → Real Time Data Warehousing
- Disaster Recovery cross-OS oder cross-database Version
- Data Protection
- Hochverfügbarkeit Active/Active

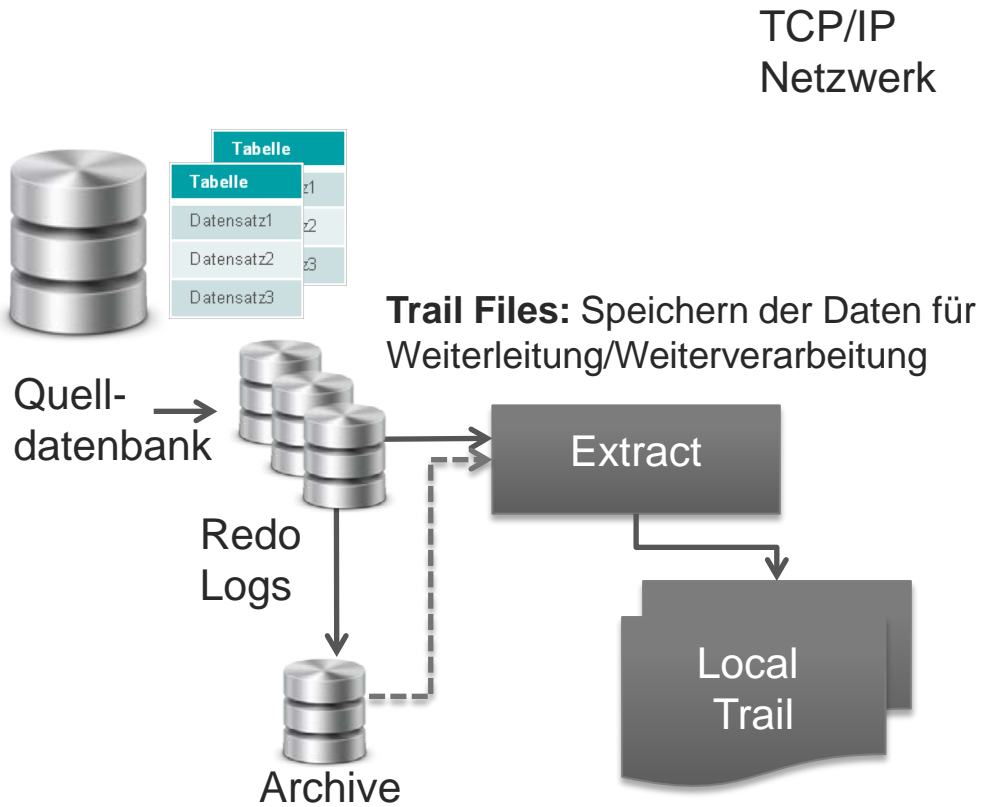


- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

# Extract

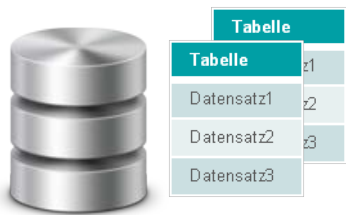


# Trail / Extract Files



# Data Pump

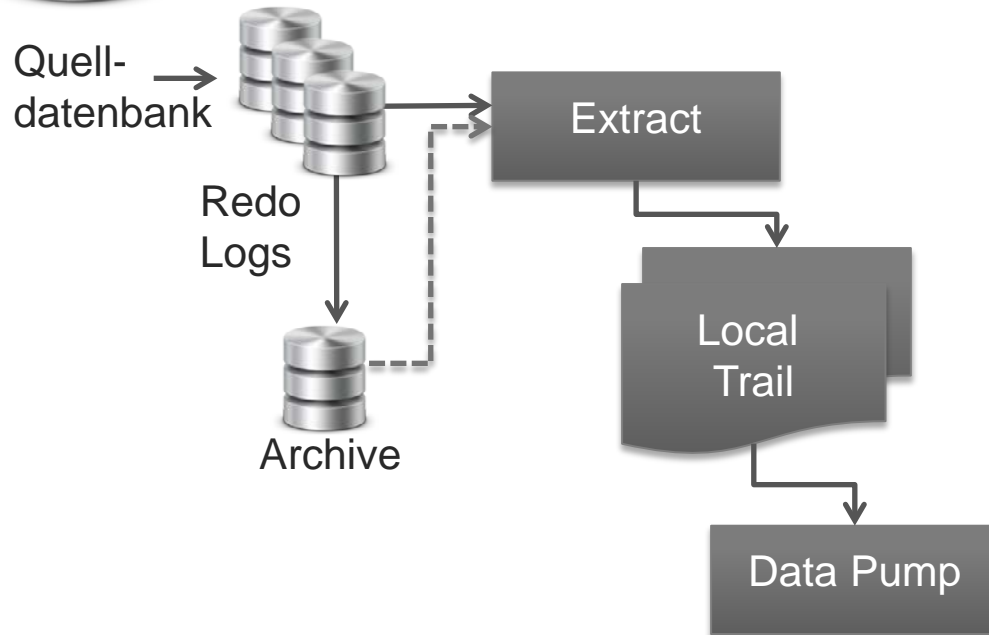
TCP/IP  
Netzwerk



**Data Pump: Datendistribution (ein oder mehrere Ziele)**

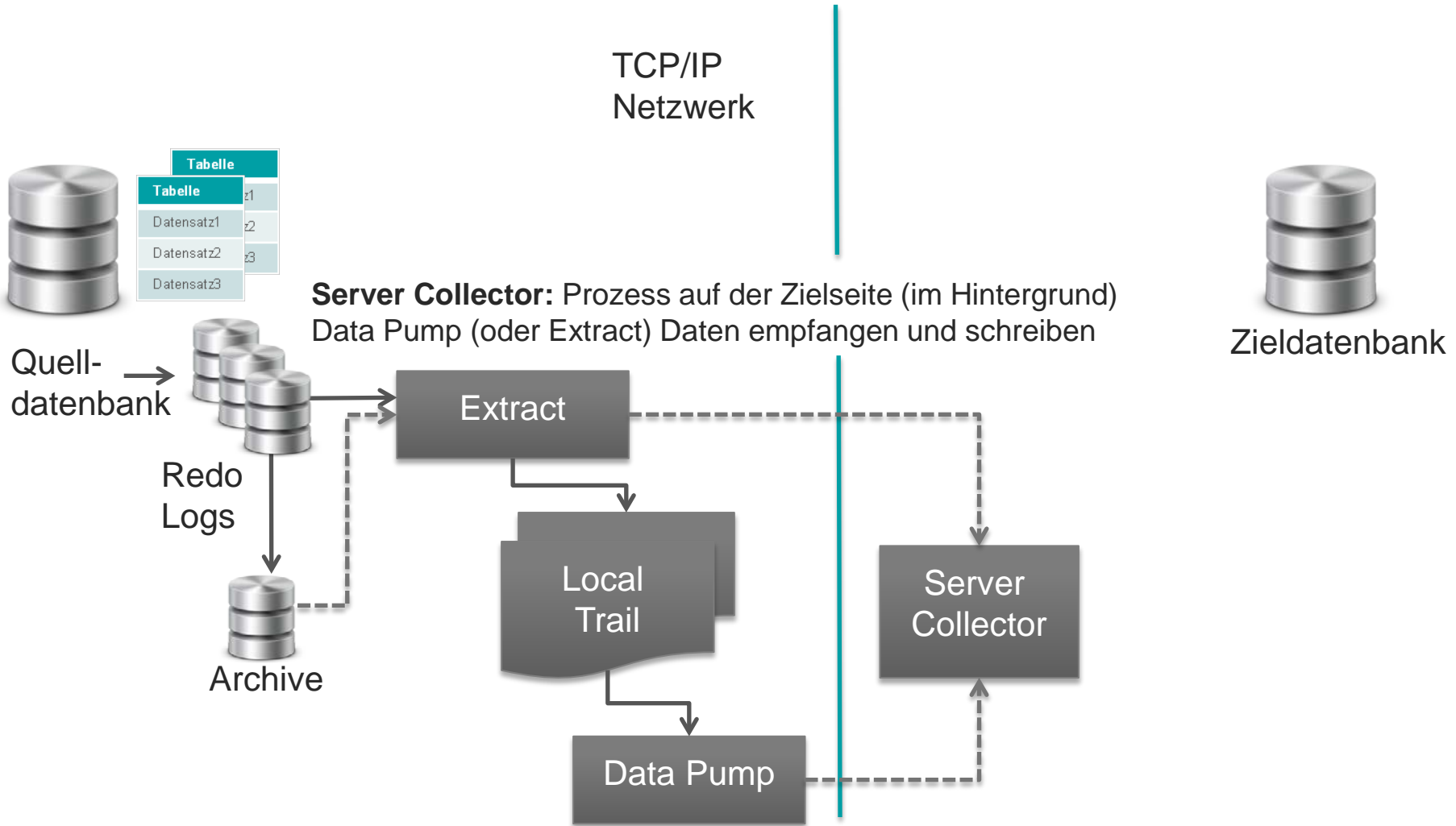


Zielfdatenbank

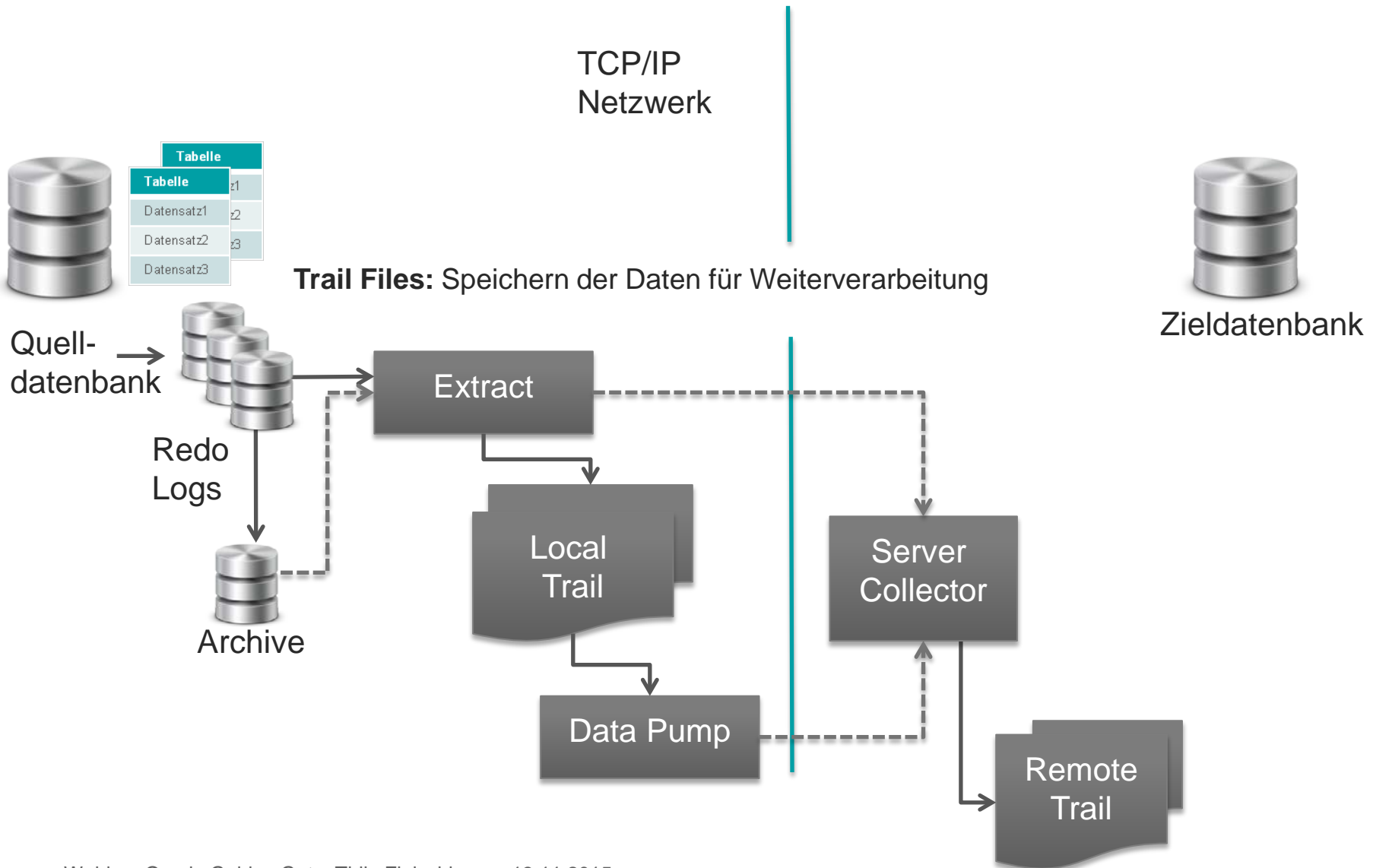


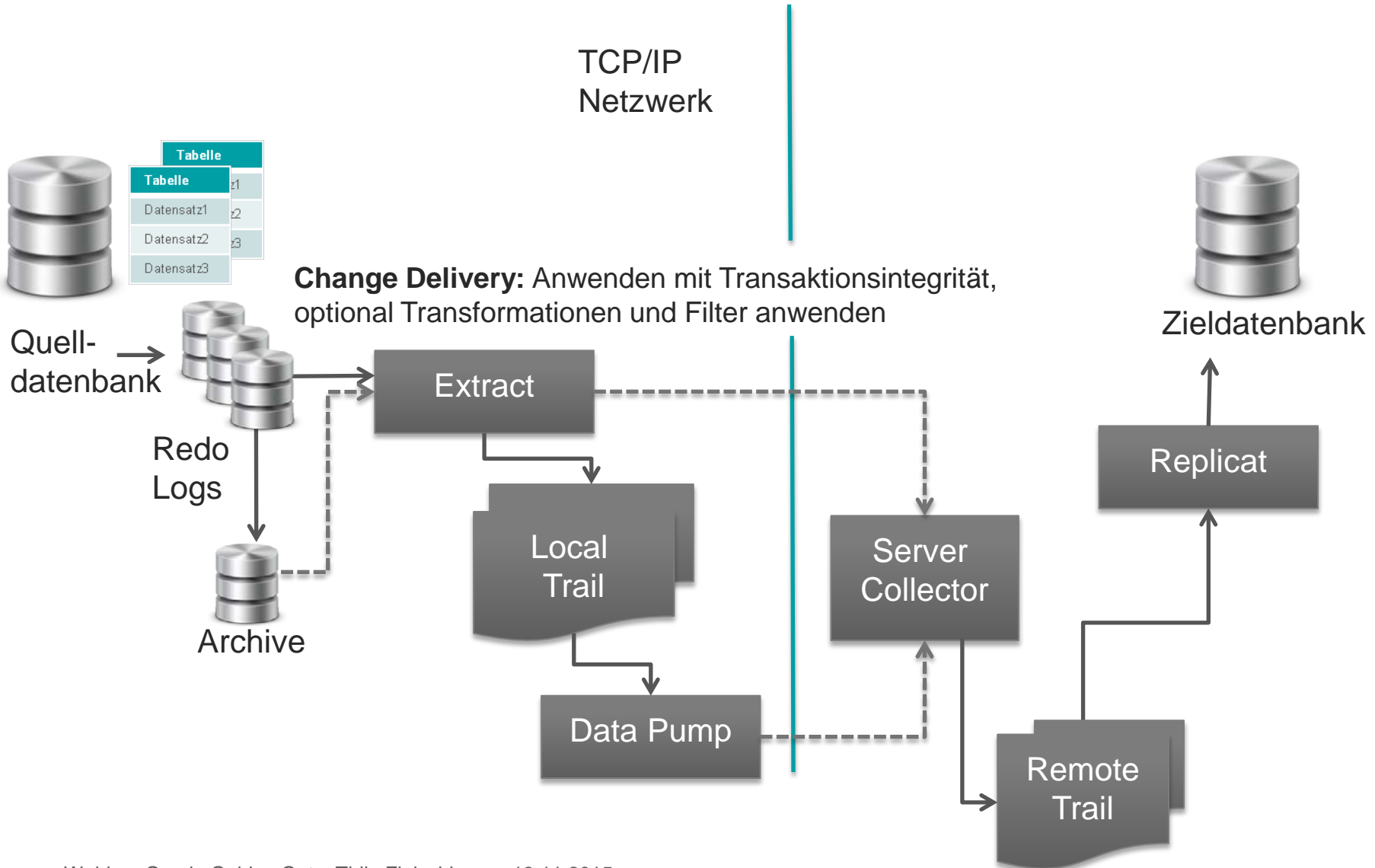
Verwendung Data Pump  
ist Best Practise:  
- Schutz vor Netzwerkfehlern  
- Schutz vor Nichtverfügbarkeit  
des Zielsystems

# Server Collector

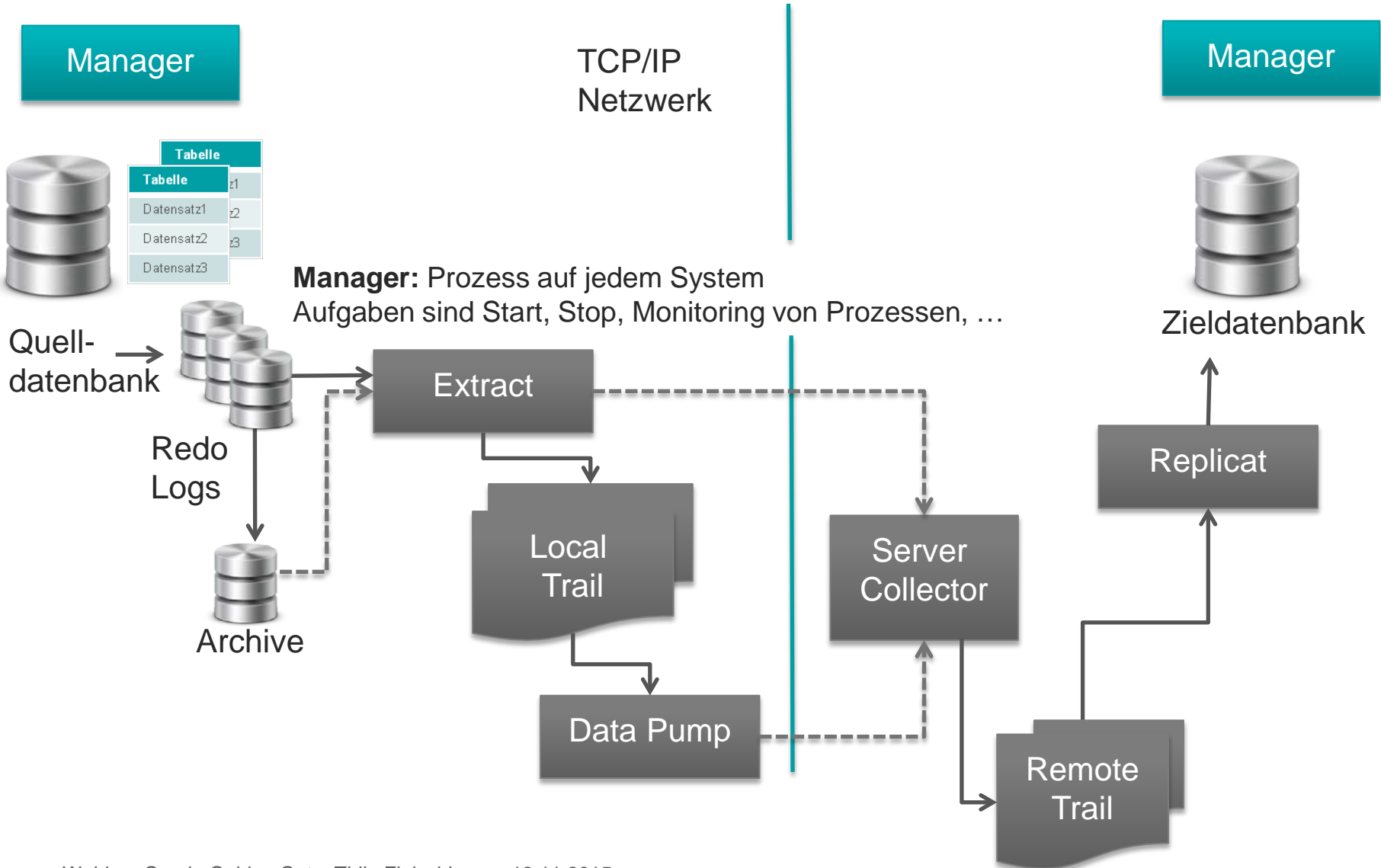


# Remote Trail





# Manager

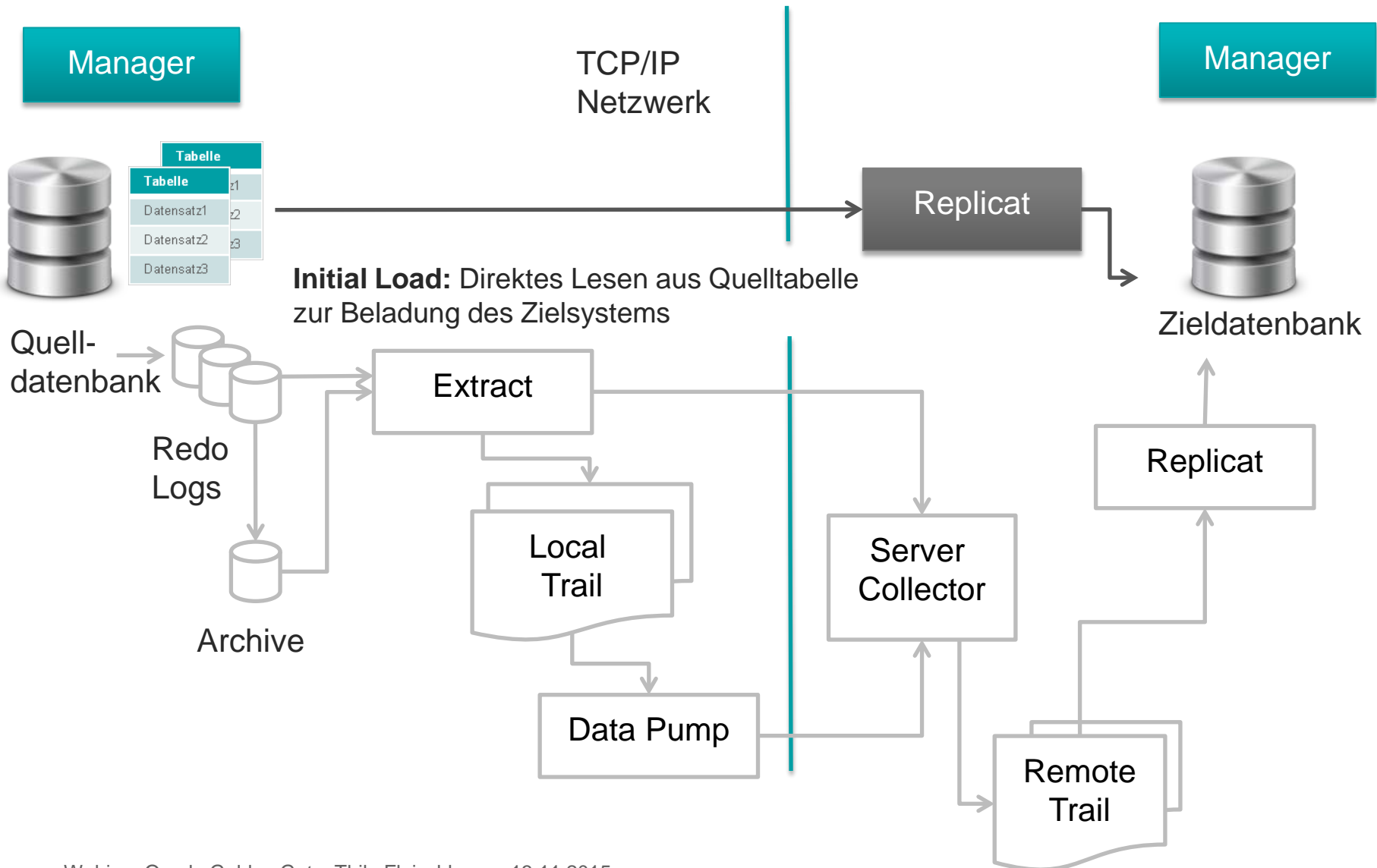




- GoldenGate-Komponenten und Prozesse
  - EXTRACT-Prozess
  - Trail Files
  - DATA-PUMP-Prozess
  - SERVER-COLLECTOR-Prozess
  - REPLICAT-Prozess
  - MANAGER-Prozess
- Bidirektionale Replikation möglich
- DDL-Replikation möglich

# Architektur (I)

Initial Load



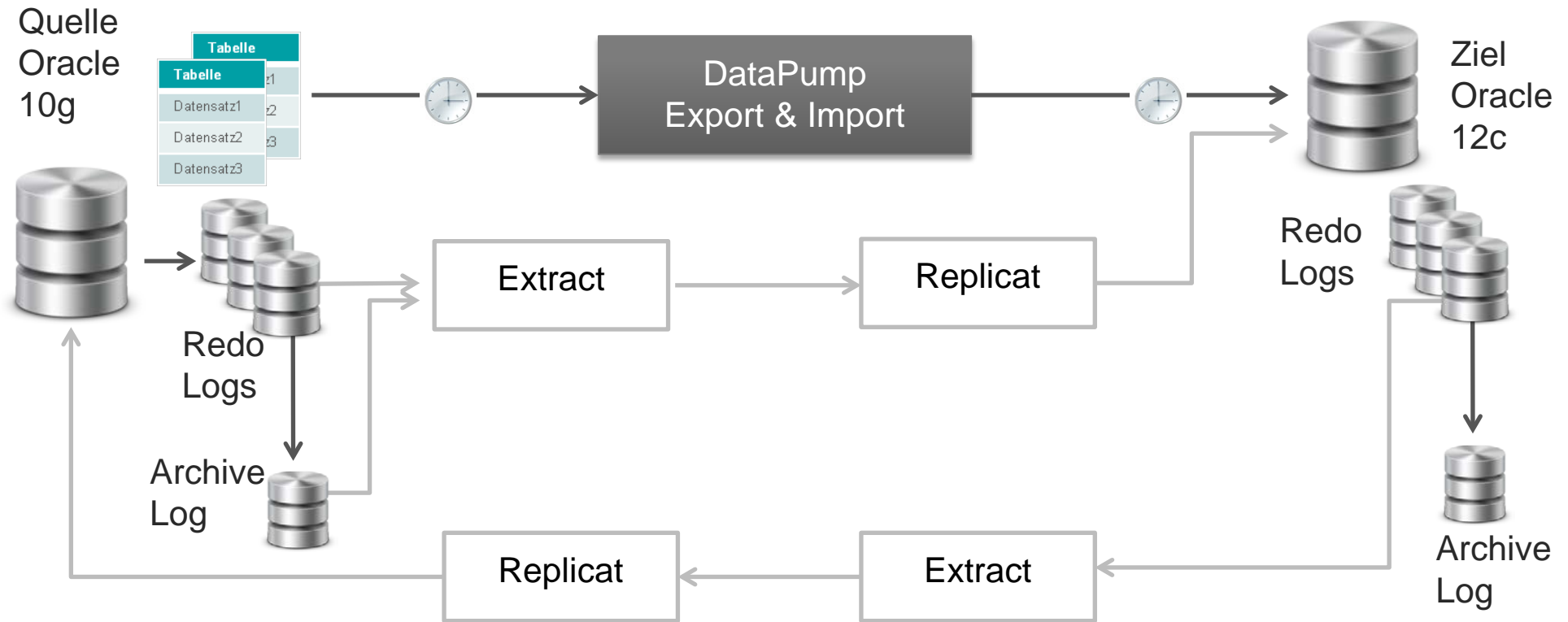
- GoldenGate-Methoden, speziell für heterogene Umgebungen (unterschiedliche Datenbanken und Betriebssystemumgebungen)
  - Direct Load
  - Direct Bulk Load
  - File (Replicat)
  - File (DB bulk load utility)
- Oracle Methoden/Alternativen für große Datenbanken
  - RMAN
  - DataPump Export & Import

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

- Minimierung Downtime Cross Platform Migration und Upgrade
- Rahmenbedingungen
- Transportable Tablespace nicht möglich
- Data Pump Export & Import mit sehr großem Wartungsfenster
- Migration und Upgrade unter Minimierung der Downtime gewünscht
- Option des Fallback muss mindestens 4 Wochen ohne Datenverlust möglich sein

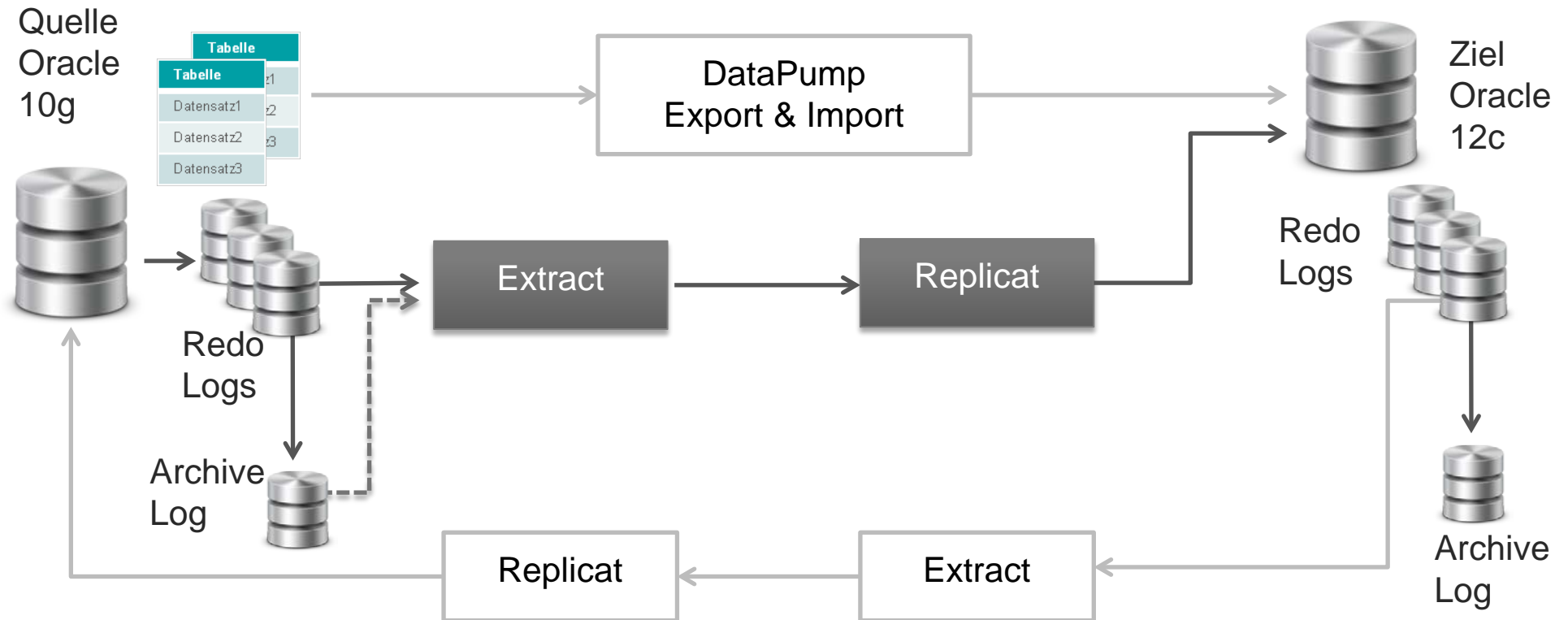
# Anwendungsszenario (II)

Initialisierung



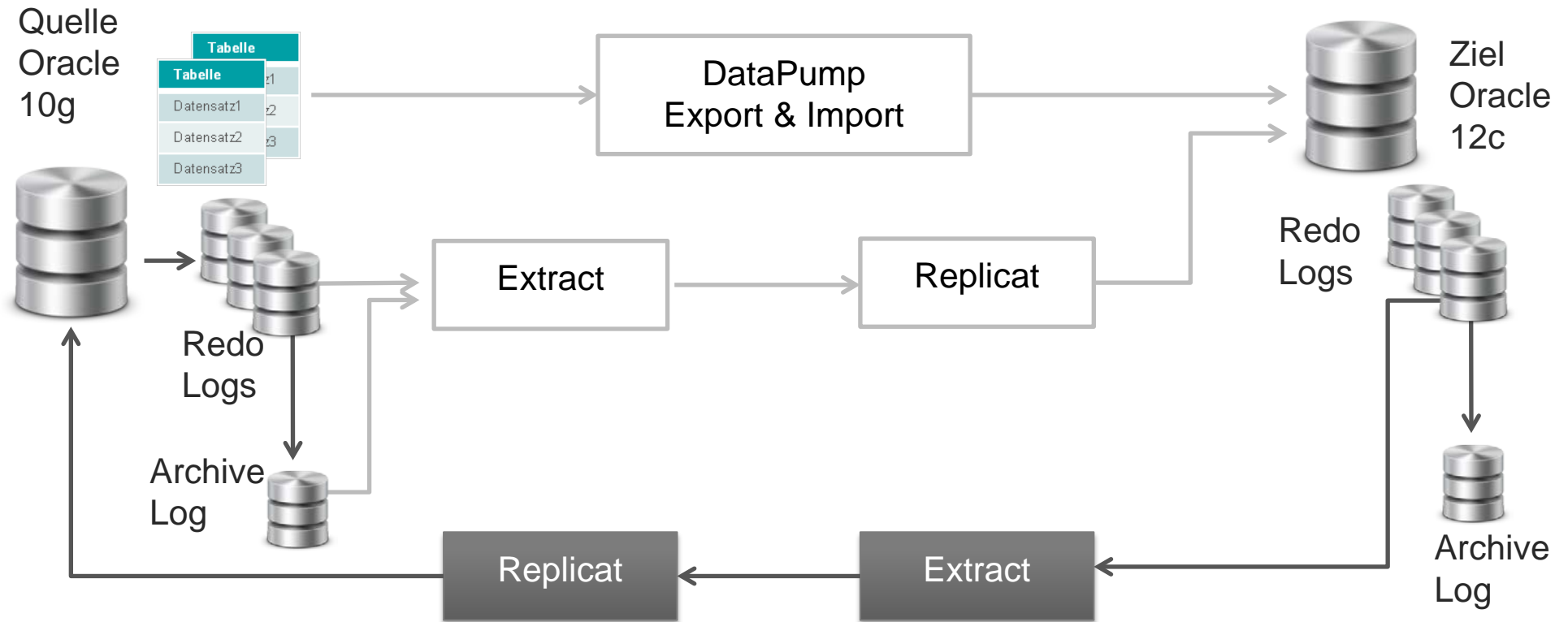
# Anwendungsszenario (III)

Testphase (Tester auf 12c)



# Anwendungsszenario (IV)

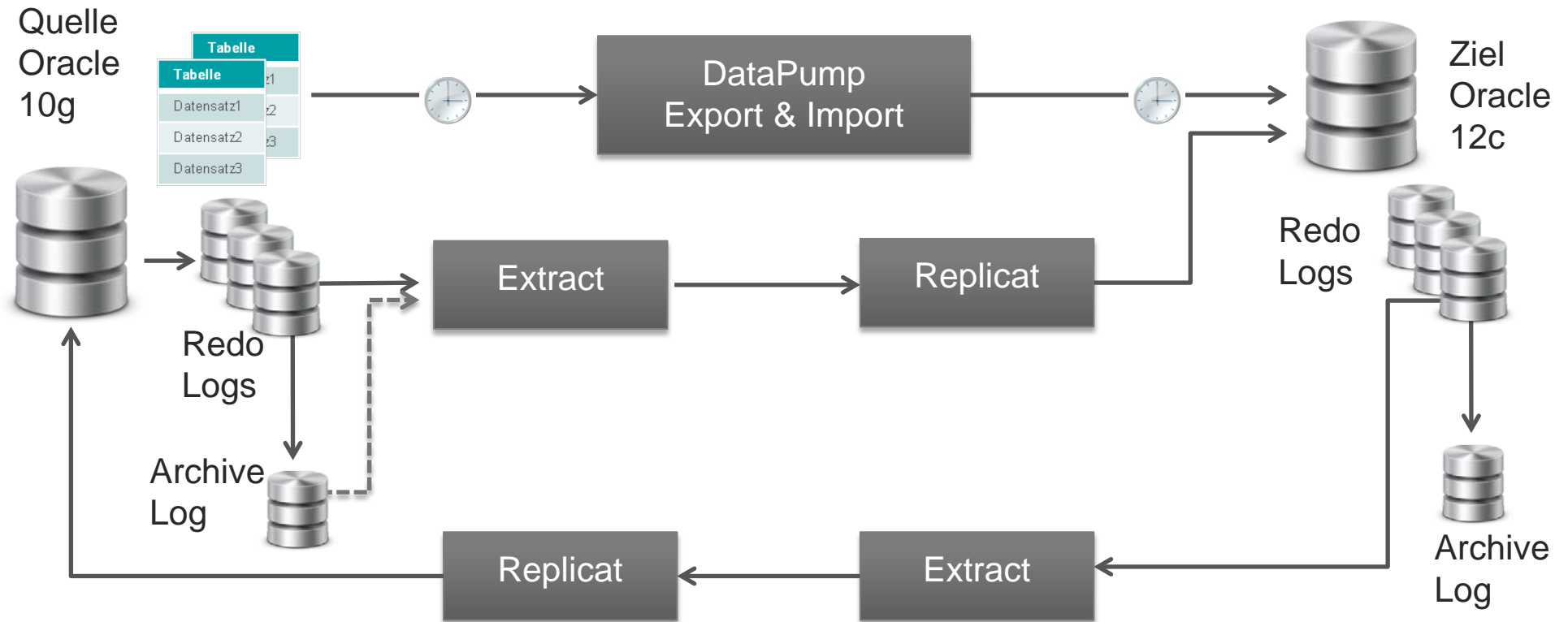
Go-Live: Replikation beenden und umgekehrt aufsetzen





# Anwendungsszenario (V)

Gesamtsicht



# Agenda

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

# Agenda

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

# Fünf Konfigurationsschritte

1. Installation durchführen
2. Umgebung vorbereiten
  - Datenbankbenutzer anlegen
  - Archive Log Modus und Supplemental Logging aktivieren
  - Manager-Prozess konfigurieren und starten
  - Tabellendefinitionen (DEFGEN) in heterogener Umgebung
3. Änderungen erfassen und weiterleiten (Change Capture)
  - Primary Extract
  - Secondary Extract / Data Pump
4. Initialisierung (Initial Load)
  - GoldenGate-Methoden
  - Oracle-Methoden
5. Änderungen anwenden (Change Delivery)
  - Replicat

# Agenda

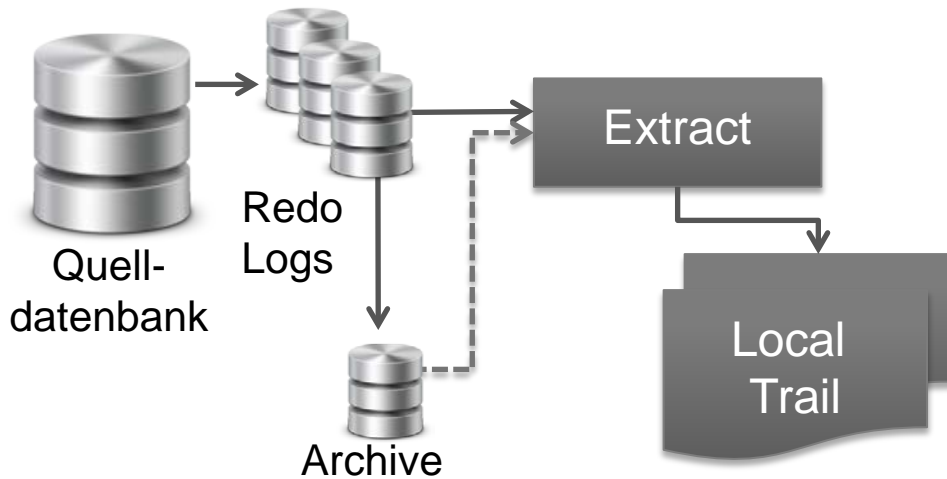
- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

# Einfache Replikation

- Anforderungen:
  - Replikation aller Tabellen im Schema ora00
  - Change Capture soll auch im Fall von Netzwerkfehler oder Nichtverfügbarkeit des Zielsystems ununterbrochen weiterlaufen  
→ Verwendung Secondary Extract (Data Pump)
  - Replikation in das Schema ora01

# Einfache Replikation

## Primary Extract



```
GGSCI (gg1) 1> edit params eora02
```

```
extract eora02  
setenv (oracle_sid=ordix)  
userid ggl_admin, password ordix  
exttrail ./dirdat/lt  
table ora00.*;
```

```
GGSCI (gg1) 2> add extract eora02,  
tranlog, begin now, threads 1
```

```
EXTRACT added.
```

```
GGSCI (gg1) 3> add exttrail ./dirdat/lt,  
extract eora02, megabytes 1
```

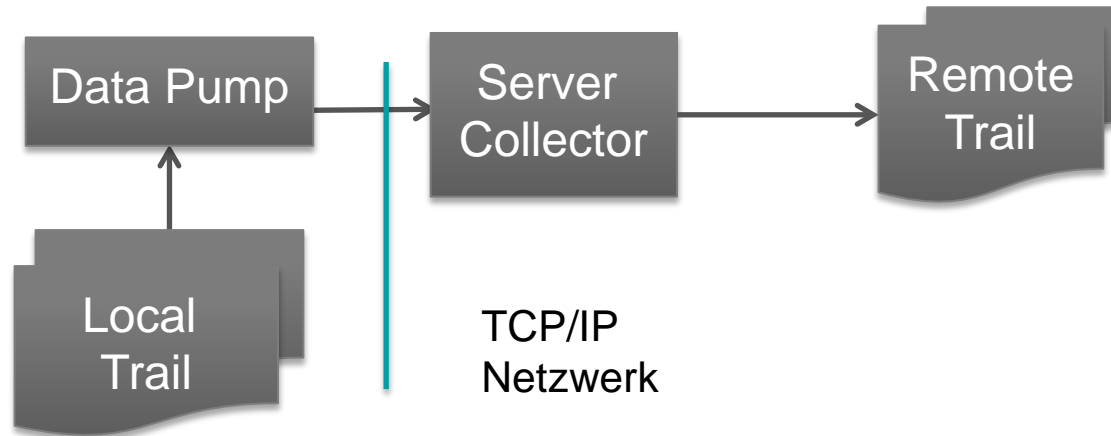
```
EXTTRAIL added.
```

```
GGSCI (gg1) 4> start extract eora02
```

```
Sending START request to MANAGER ...  
EXTRACT EORA02 starting
```

# Einfache Replikation

## Secondary Extract



```
GGSCI (gg1) 1> edit params pump
```

```
extract pump
passthru
rmthost gg2, mgrport 7809
rmttrail ./dirdat/rt
table ora00.*;
```

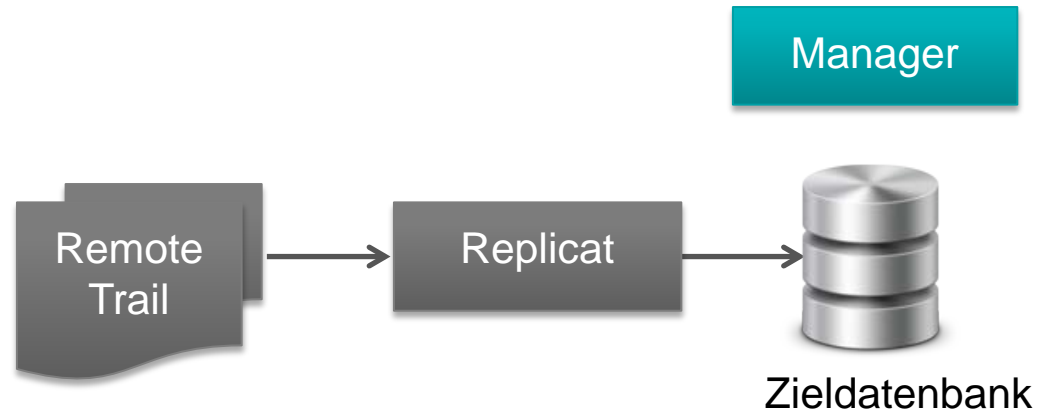
```
GGSCI (gg1) 2> add extract pump, extrailsorce ./dirdat/lt, begin now
GGSCI (gg1) 3> add rmttrail ./dirdat/rt, extract pump
GGSCI (gg1) 4> start extract pump
```



# Einfache Replikation

## Replicat

TCP/IP  
Netzwerk



```
GGSCI (gg2) 1> dblogin userid gg2_admin, password ordix
GGSCI (gg2) 2> add checkpointtable chkpt
GGSCI (gg2) 3> edit params apply02
```

```
REPLICAT apply02
userid gg2_admin, password ordix
assumtargetdefs
discardfile ./dirrpt/apply02.dsc, append
map ora00.*, target ora01.*;
```

```
GGSCI (gg2) 4> add replicat apply02, exttrail ./dirdat/rt, checkpointtable
gg2_admin.chkpt
GGSCI (gg2) 5> start replicat apply02
```

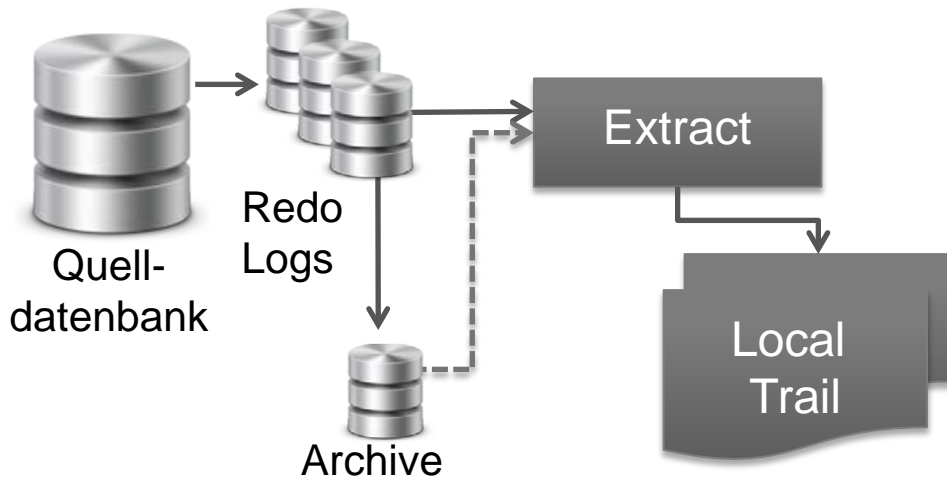
# Agenda

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

- Anforderungen:
  - Erstellen einer Tabellen-History für die Tabelle ora00.projekt
    - Jede Änderung soll neue Zeile im Zielsystem erzeugen
    - Insert, folgende Updates und Delete werden als Point-In-Time Snapshot geschrieben
  - Verwendung von Token auf der Quelle
  - Verwendung von INSERTALLRECORDS im Zielsystem
  - Change Capture soll auch im Fall von Netzwerkfehler oder Nichtverfügbarkeit des Zielsystems ununterbrochen weiterlaufen
    - Verwendung Secondary Extract (Data Pump)
  - Replikation in das Schema ora02

# Tabellen-History

## Primary Extract



```
GGSCI (gg1) 1> edit params eora03
```

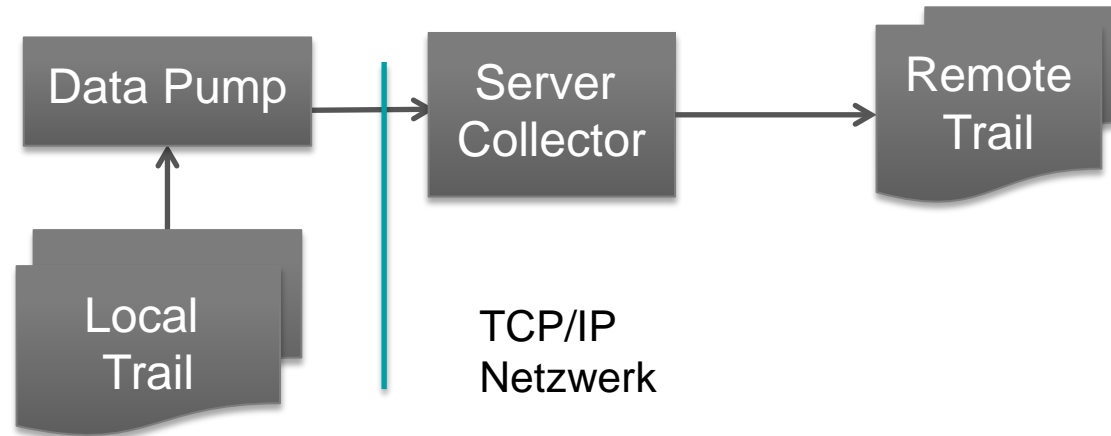
- Verwendung von Token
- Möglichkeit der Verwendung von Makros, SQL-Aufrufen, Filtern ,...

```
extract eora03
setenv(oracle_sid=ordix)
userid ggl_admin, password ordix
exttrail ./dirdat/pt
table ora00.projekt,
tokens (TKN-OSUSERNAME =
@GETENV('GGENVIRONMENT','OSUSERNAME'),
TKN-TRANSACTION-CSN = @GETENV('TRANSACTION','CSN'),
TKN-USERNAME = @GETENV('TRANSACTION','USERNAME'),
TKN-DBNAME = @GETENV('DBENVIRONMENT','DBNAME'),
TKN-HOSTNAME = @GETENV('GGENVIRONMENT','HOSTNAME'),
TKN-COMMITTIME = @GETENV('GGHEADER','COMMITTIMESTAMP'),
TKN-REC-FILESEQNO = @GETENV('RECORD','FILESEQNO'),
TKN-REC-FILERBA = @GETENV('RECORD','FILERBA'));
```

```
GGSCI (gg1) 2> add extract eora03, tranlog, begin now, threads 1
GGSCI (gg1) 3> add exttrail ./dirdat/pt, extract eora03, megabytes 1
GGSCI (gg1) 4> start extract eora03
```

# Tabellen-History

## Secondary Extract



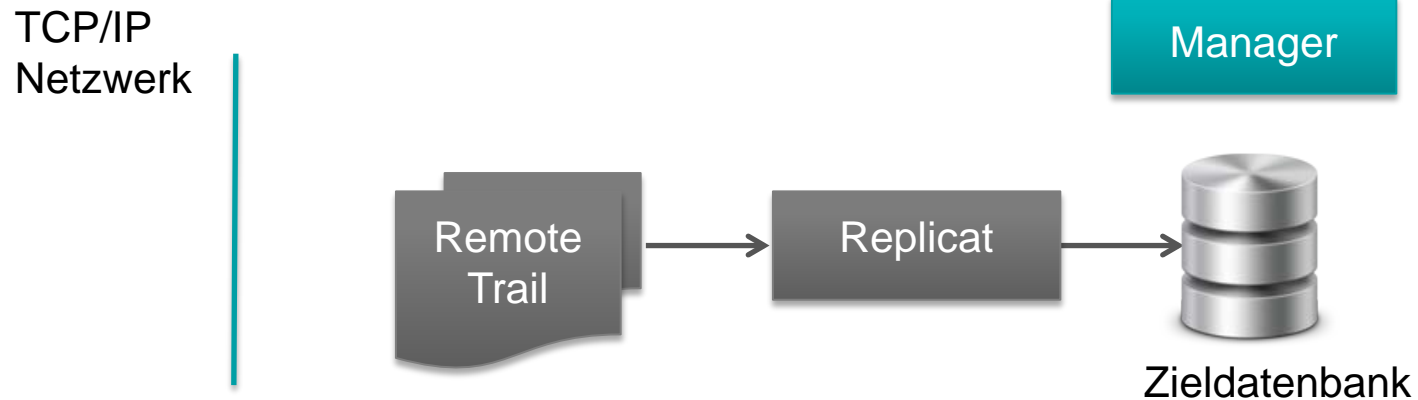
```
GGSCI (gg1) 1> edit params ppump
```

```
extract ppump  
passthru  
rmthost gg2, mgrport 7809  
rmttrail ./dirdat/pt  
table ora00.projekt;
```

```
GGSCI (gg1) 2> add extract ppump, exttrailsource ./dirdat/pt, begin now  
GGSCI (gg1) 3> add rmttrail ./dirdat/pt, extract ppump  
GGSCI (gg1) 4> start extract ppump
```

# Tabellen-History

Replicat



```
GGSCI (gg2) 1> dblogin userid gg2_admin, password ordix
GGSCI (gg2) 2> add checkpointtable chkpt
GGSCI (gg2) 3> edit params apply03
```

- Verarbeitung der zusätzlichen Information(en) (Token) in den Trail Files
- Möglichkeit der Verwendung von Filtern, ...

# Tabellen-History

Replicat (ff.)

```
replicat apply03
userid gg2_admin, password ordix
assumetargetdefs
handlecollisions ora02.*
discardfile ./dirrpt/apply01.dsc, append
insertallrecords
map ora00.projekt, target ora02.projekt
COLMAP (USEDEFAULTS,
TX_TIME = @GETENV('GGHEADER', 'COMMITTIMESTAMP'),
TX_CSN = @TOKEN('TKN-TRANSACTION-CSN'),
OP_TYPE = @GETENV('GGHEADER', 'OPTYPE'),
TX_DBNAME = @TOKEN('TKN-DBNAME'),
TX_HOSTNAME = @TOKEN('TKN-HOSTNAME'),
TX_OSUSERNAME = @TOKEN('TKN-OSUSERNAME'),
TX_USERNAME = @TOKEN('TKN-USERNAME'),
BEFORE_AFTER_IND = @GETENV('GGHEADER', 'BEFOREAFTERINDICATOR'),
REC_FILESEQNO = @TOKEN('TKN-REC-FILESEQNO')
REC_FILERBA = @TOKEN('TKN-REC-FILERBA'));
```

```
GGSCI (gg2) 4> add replicat apply03, exttrail ./dirdat/pt, checkpointtable
gg2_admin.chkpt
GGSCI (gg2) 5> start replicat apply03
```

# Agenda

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit



- Zwei Beispiele
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History

- Zwei Beispiele
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History

- Zwei Beispiele
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History

# Agenda

- Replikation
- Technologie
- Architektur
- Anwendungsszenario
- Replikation einrichten
  - Konfigurationsschritte
  - Einfache Replikation
  - Auditing mit GoldenGate: Erstellen einer Tabellen-History
- Demo
- Fazit

## Fazit

- Heterogene Initialisierung möglich
- Asynchrone Übertragung
  - Nur committed Transaktionen
  - DMLs und DDLs
- Universelles Trail-Format über heterogene Systeme
- Möglichkeiten zur Erweiterung: Token, SQL-Aufrufe, Filter, Funktionen, ...
- Betrachtung der Lizenzen notwendig





**Vielen Dank für  
Ihre Aufmerksamkeit!**

## ORDIX AG

Zentrale Paderborn  
Westernmuer 12 - 16  
33098 Paderborn  
Tel.: 05251 1063-0  
Fax: 0180 1 67349 0

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

Weitere Geschäftsstellen  
in Essen, Gersthofen,  
Köln und Münster

[info@ordix.de](mailto:info@ordix.de)  
[www.ordix.de](http://www.ordix.de)