

HOUG Konferencia 2015

**Oracle TimesTen In-Memory Database
and
TimesTen Application-Tier Database Cache**

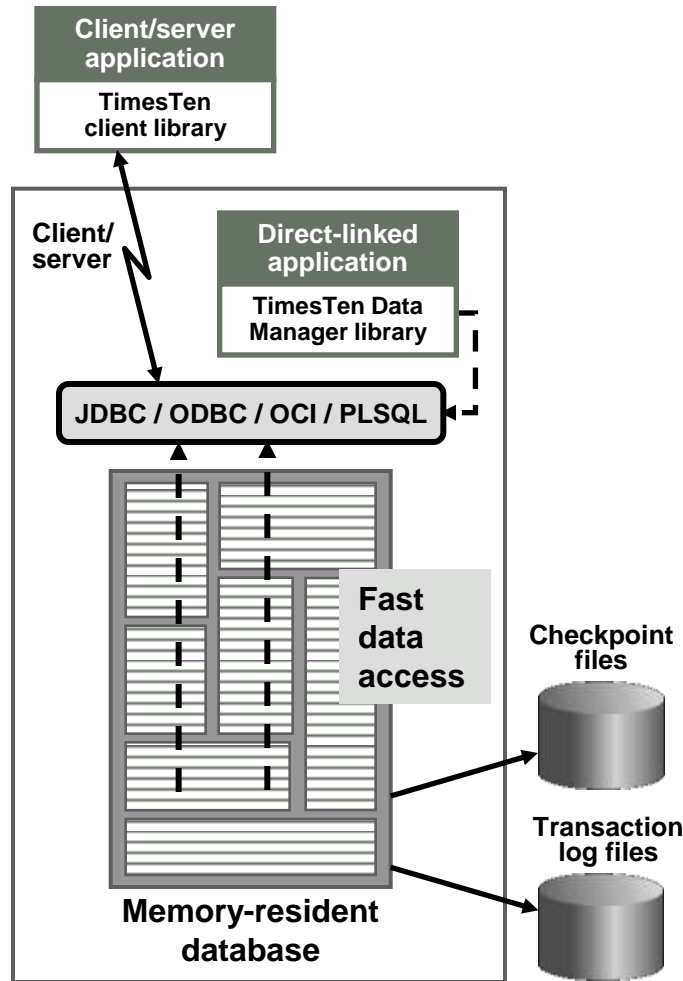
A few facts in 10 minutes

Tamas.Kerepes@webvalto.hu

What is TimesTen

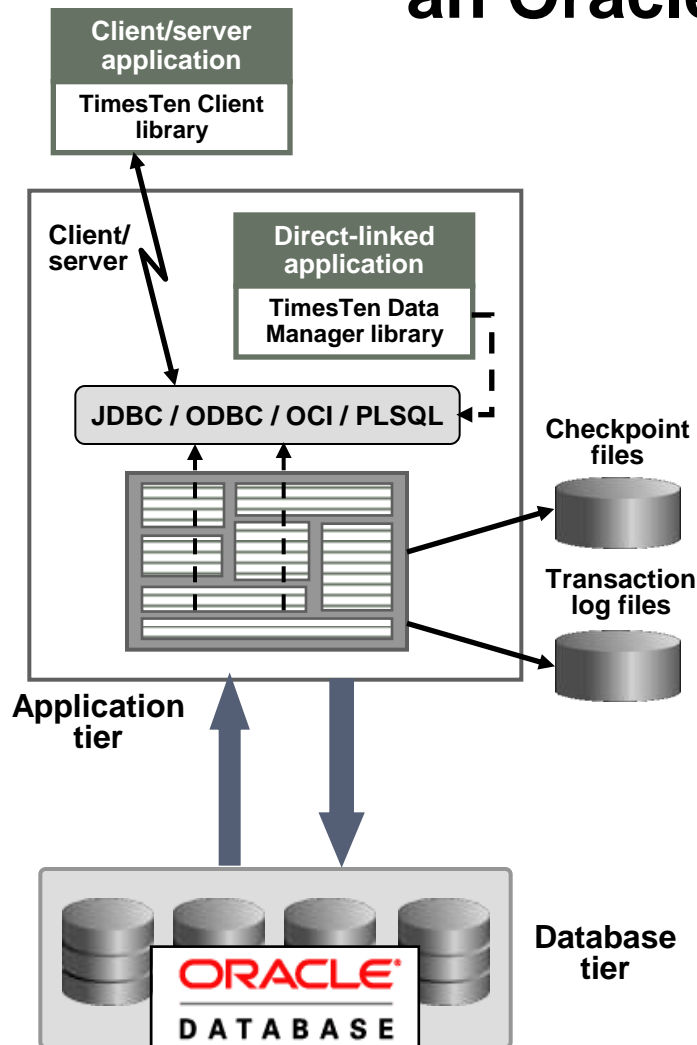
- An in-memory relational database management system originally designed at HP
- Spun out as an independent software company in 1996
- **Acquired by Oracle in 2005**
- Uses SQL and PL/SQL: a great deal of transparency
- It may be used independently from an Oracle database or as a cache layer that speeds up Oracle databases
- Mostly used for improving the performance of OLTP systems
- Different from Oracle Database In-Memory!!! 😊 😊 😊
- Developer GUI: SQL*Developer support TimesTen
- Instead of SQL*Plus: **ttlsql** line mode utility

Oracle TimesTen: an in-memory database product



- In-memory RDBMS
 - Entire database in memory
 - Standard SQL with JDBC, ODBC, OCI, Pro*C
 - PL/SQL
 - Compatible with Oracle DB
- Persistent and durable
 - Transactions with ACID
- Extreme performance
 - Fast response time
 - Very high throughput
- Embeddable

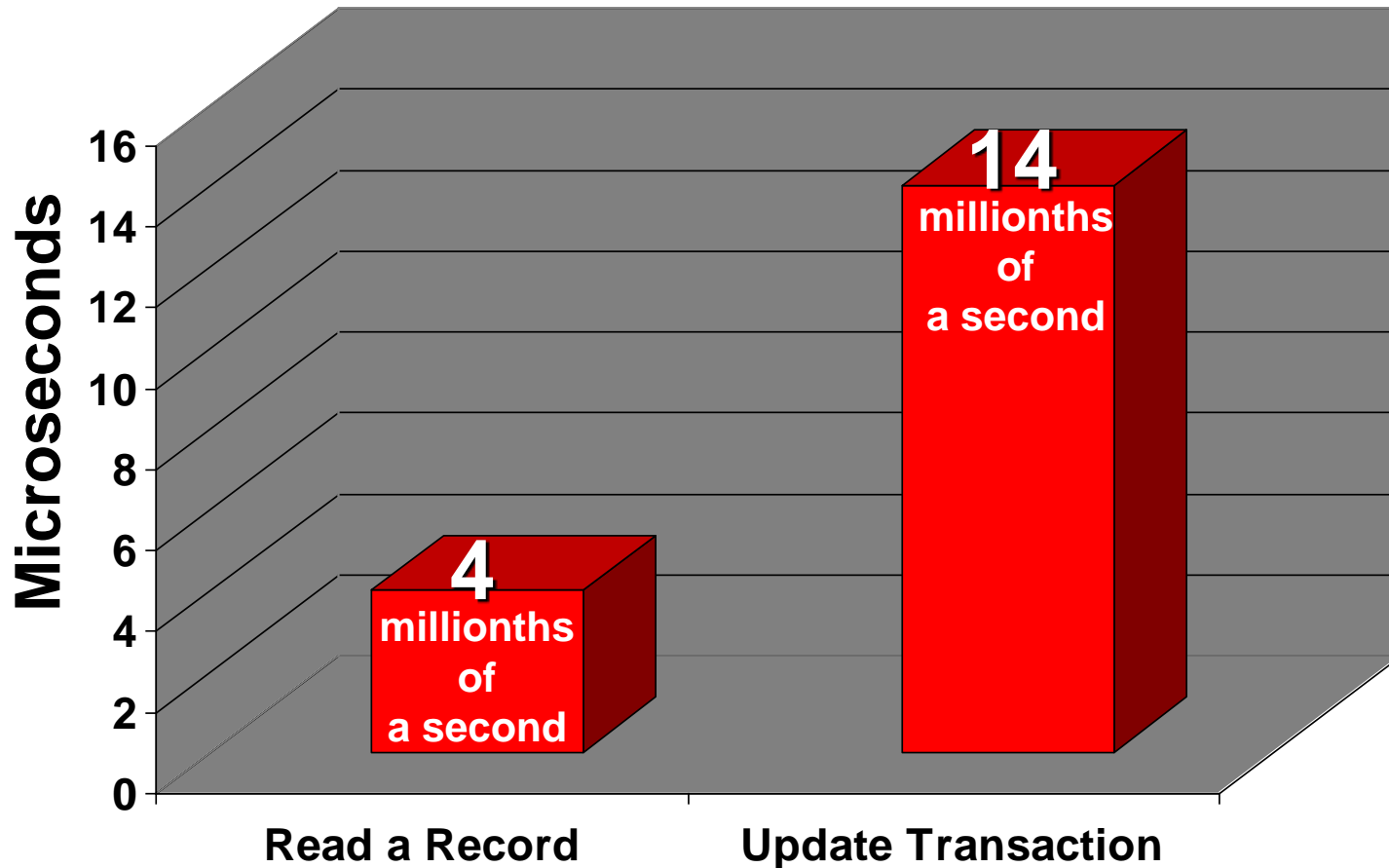
TimesTen Application-Tier Database Cache: an Oracle Database Option



- Using Oracle TimesTen as a relational cache layer
- Caches Oracle database tables in the application tier
 - Lower response time and increase throughput for Oracle Database applications
- Read-only and updatable
- Automatic data synchronization with the Oracle database

Extremely Fast Response Time

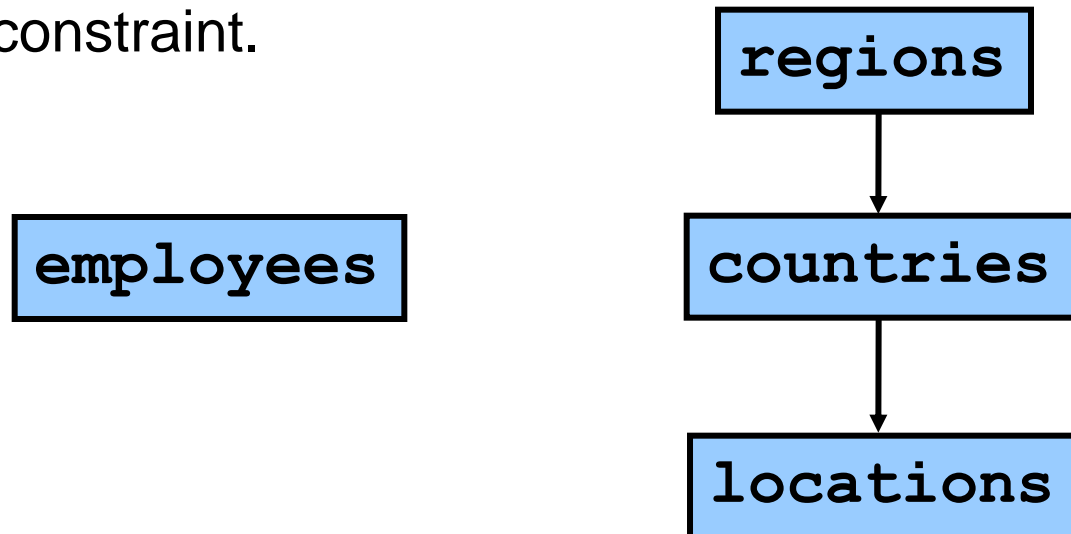
Average Response Time TimesTen In-Memory Database



TimesTen In-Memory Database 11g, Intel Xeon 3.0 GHz, 64-bit Oracle Enterprise Linux

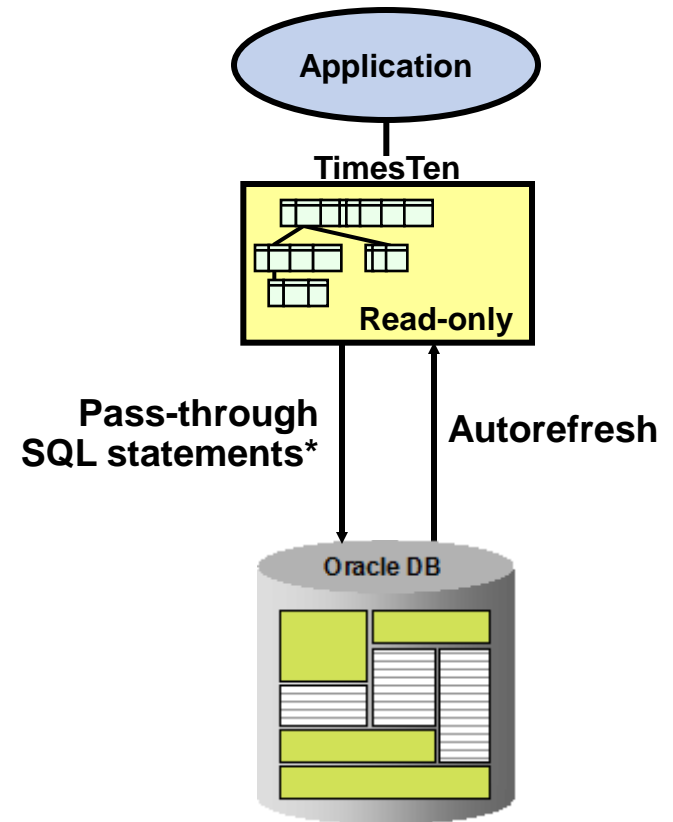
Cache Group Definition

- A cache group defines the Oracle tables and their subset of rows and columns to cache in a TimesTen database.
- A cache group can contain one or more tables.
- A cache group can have only one root table.
- In a multiple table cache group, a child cache table must reference another cache table with a primary key or foreign key constraint.



Read-Only Cache Group

- Cache tables cannot be updated directly by applications.
- Committed updates on the Oracle tables are automatically refreshed to the corresponding cache tables.



* Depending on the pass-through level

Create a Local Explicitly Loaded Read-Only Cache Group

- A local cache group does not share data in its cache tables across TimesTen databases even if the databases are members of the same cache grid.
- An explicitly loaded cache group can load data into its cache tables from Oracle in the following ways:
 - Manually by using a load or refresh operation
 - Automatically by using an automatic refresh operation

```
CREATE READONLY CACHE GROUP legacy_customers
FROM
  hr.customer (cust_num NUMBER(6) NOT NULL PRIMARY KEY,
               region VARCHAR2(10),
               name VARCHAR2(50),
               address VARCHAR2(100))
WHERE (hr.customer.cust_num < 100)
```


Read-Only Cache Group Automatic Refresh Interval

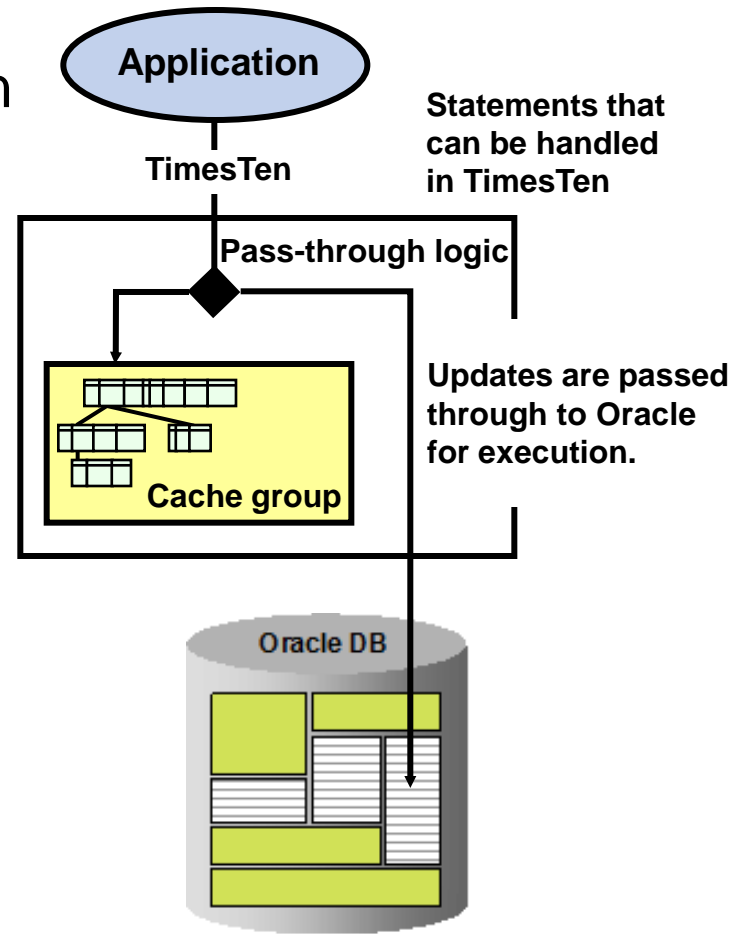
- Specifies the frequency in which committed updates on Oracle tables are automatically refreshed to the corresponding TimesTen cache tables
- Units in minutes, seconds, or milliseconds (default: 5 minutes)
- Use the SQL `ALTER CACHE GROUP` statement to modify the automatic refresh interval for an existing read-only cache group.

```
CREATE READONLY CACHE GROUP legacy_customers
  AUTOREFRESH INTERVAL 90 SECONDS
FROM
  hr.customer (cust_num NUMBER(6) NOT NULL PRIMARY KEY,
               region VARCHAR2(10),
               name VARCHAR2(50),
               address VARCHAR2(100))
WHERE (hr.customer.cust_num < 100)

ALTER CACHE GROUP legacy_customers SET AUTOREFRESH INTERVAL 10 MINUTES
```

Pass-Through of Statements from TimesTen to Oracle

- Passing a statement through to the Oracle database for execution is transparent to a TimesTen application.
- In a read-only cache group:
 - Read operations are performed in TimesTen
 - Update operations are passed through to Oracle and automatically refreshed to TimesTen



Configure Statement Pass-Through

Use one of the following mechanisms to set the statement pass-through level:

- PassThrough DSN attribute
- ttOptSetFlag built-in procedure

```
[cachealone1]
DataStore=/users/OracleCache/alone1
OracleNetServiceName=orcl
DatabaseCharacterSet=WE8ISO8859P1
PassThrough=2

% ttIsql "DSN=cachealone1;UID=cacheuser;OraclePWD=oracle"
...
Command> autocommit 0;
Command> UPDATE hr.customer SET
        > name = 'John Smith' WHERE cust_num = 50;
```

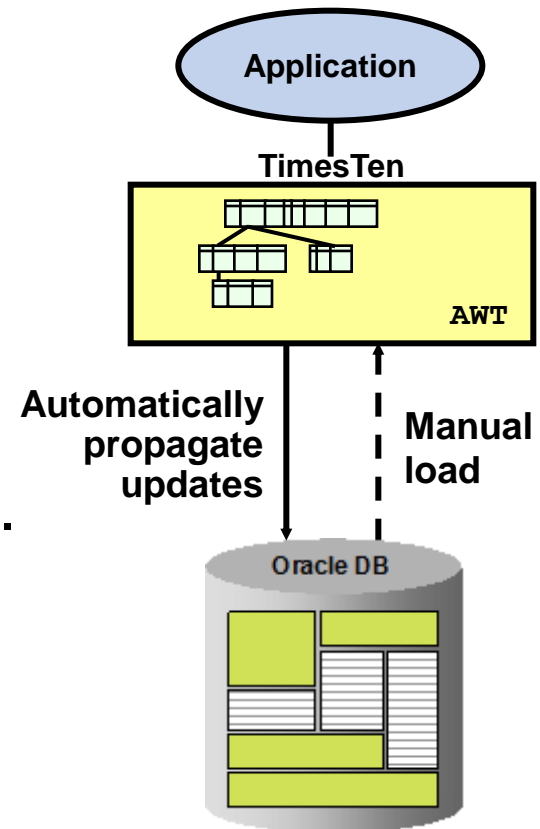
Create a Local Dynamic Read-Only Cache Group

- A dynamic cache group can load data into its cache tables from Oracle:
 - On demand by using a dynamic load operation
 - Manually by using a load operation
- Least-recently used (LRU) aging is defined by default.

```
CREATE DYNAMIC READONLY CACHE GROUP active_customers
FROM
  hr.customer (cust_num NUMBER(6) NOT NULL,
               region VARCHAR2(10),
               name VARCHAR2(50),
               address VARCHAR2(100),
               PRIMARY KEY(cust_num))
```

Asynchronous Write-Through Cache Group

- Committed updates on cache tables are first committed in the TimesTen database.
- Updates are then propagated to the Oracle database.
- New transactions can be issued on the cache tables without waiting for the Oracle transaction to complete.

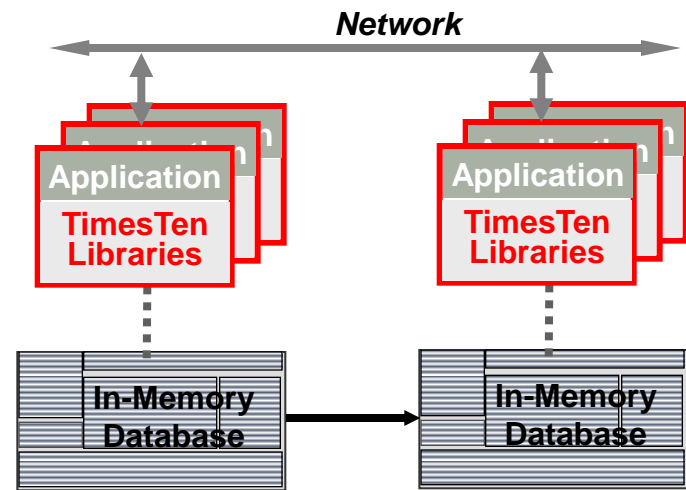


PROPAGATE and READONLY Cache Table Attributes

```
CREATE USERMANAGED CACHE GROUP western_customers
FROM
  hr.active_customer (custid NUMBER(6) NOT NULL,
                      name VARCHAR2(50),
                      addr VARCHAR2(100),
                      zip VARCHAR2(12),
                      PRIMARY KEY(custid),
                      PROPAGATE),
  hr.ordertab (orderid NUMBER(10) NOT NULL,
              custid NUMBER(6) NOT NULL,
              PRIMARY KEY(orderid),
              FOREIGN KEY(custid)
                REFERENCES hr.active_customer(custid),
              PROPAGATE),
  hr.cust_interests (custid NUMBER(6) NOT NULL,
                    interest VARCHAR2(10) NOT NULL,
                    PRIMARY KEY(custid, interest),
                    FOREIGN KEY(custid)
                      REFERENCES hr.active_customer(custid),
                    READONLY),
  hr.orderdetails (orderid NUMBER(10) NOT NULL,
                  itemid NUMBER(8) NOT NULL,
                  quantity NUMBER(4) NOT NULL,
                  PRIMARY KEY(orderid, itemid),
                  FOREIGN KEY(orderid)
                    REFERENCES hr.ordertab(orderid))
```

TimesTen to TimesTen Replication

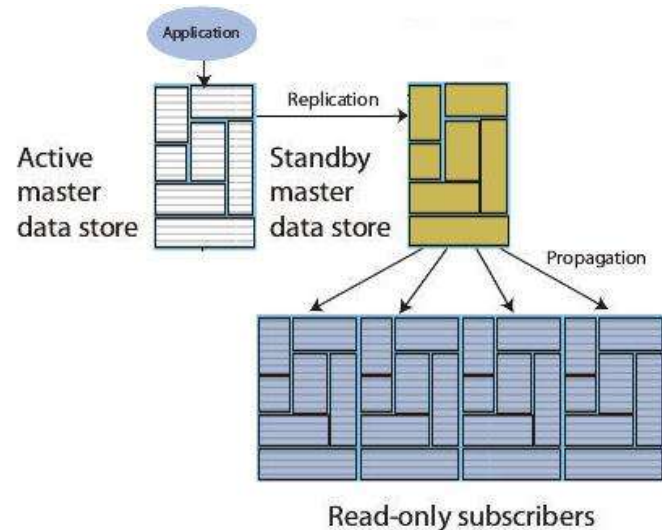
- Real-time transactional replication
- High performance
- Asynchronous or synchronous transmit
- Robust and reliable performance
- Transparent to the application (no application code changes)
- Benefits:
 - High availability
 - Online upgrades and maintenance



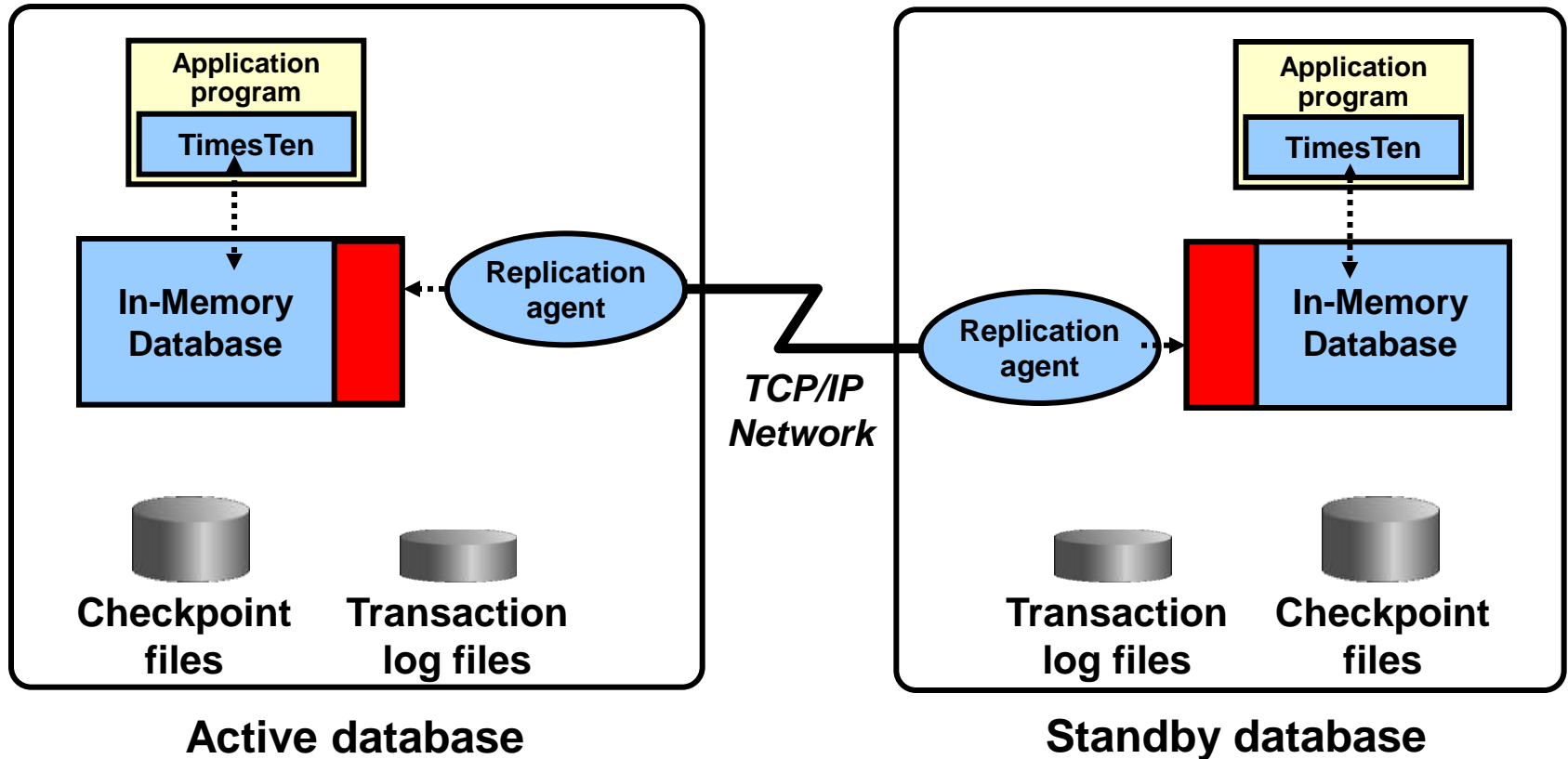
TimesTen to TimesTen Replication

Active-Standby Pair

- Applications update active database.
- Applications can read from active, standby, or subscriber databases.
- Standby database:
 - Receives replicated updates from active database
 - Propagates replicated updates to read-only subscriber databases
- If standby fails, active can be configured to replicate to subscribers directly.
- If active fails, standby can be configured to be the active database.



TimesTen Replication Architecture



Asynchronous Replication

- Committed replicated transactions are transmitted asynchronously from the active database to the standby database and then to the read-only subscriber databases.
 - Very fast response time; ideal for high-transaction rate and low-latency applications
 - Default replication mode
- Active is always ahead of the standby and subscribers.
- Replicated updates are durably committed on the active before being transmitted to the standby.

Synchronous Replication

- Committed replicated transactions are transmitted from the active database to the standby database before control is returned to the application.
 - Lower transaction throughput and longer latency than asynchronous replication.
 - No loss of replicated transactions at the cost of performance
- Synchronous replication through return two-safe service can be enabled or disabled by the application at the transaction level.

End of 10 minutes, let's summarize 😊

- TimesTen and Oracle In Memory Database Cache should be distinguished from Oracle Database In-Memory!
 - **TimesTen is for speeding up OLTP systems**
 - **It can be used independantly**
 - There is TimesTen – to – TimesTen replication for HA
 - It can be used as a cache layer for relational tables on top of an Oracle database. The name: Oracle IMDB Cache
 - Replication and caching can be combined
-
- Thanks for your attention
 - Any comments or questions?