Transbase®

WHITE PAPER

Transbase® Replication Service

Transbase® databases can be automatically and robustly replicated from a master database. In order to activate this service the master database must be operated with „logging“, and all replicated (slave) databases must be read-only databases.

Slave databases can receive and process log records as soon as these are produced by the master database. Once they receive a commit record, the respective transaction will be committed on the slave database. The slave databases in general will be „a little“ behind the master database. A major feature of slave databases is that they can be used in read/only mode while they are processing replicated log records.

As a consequence, slave databases can be used to balance read load and at the same time can play their role as „standby“ databases. In case of a crash of the master database, any of the slave databases can almost immediately take over the role of the master database. The other databases may resynchronize with the new master and remain slave databases.

Overview

- Standby Databases
- Database Grids
- Synchronized Client Databases

The replication may happen continuously or periodically. In particular, the replication may be paused at any time and resumed at any later time. Slave databases always provide a consistent state of the master database, as they always show a committed database state.

Technical Details

1. Master Database

The master database must be configured for the default recovery method „delta logging“. Delta Logging means that each change of a database page is recorded in the log, and that a transaction is committed by writing persistently all log entries including the so-called commit-record. The changes are denoted as XOR delta between the new state and the old state of a page. XOR deltas can be used to redo transactions as well as to rollback transactions. If O, N and D denote the old state, the new state and the XOR delta, one value can be restored from the other two: In particular, O and D can be used to restore N, and N and D can restore O.

In order to allow the rollback of active transactions, the log must reach back at least to the begin of the oldest active transaction; for older log records, it can be configured if Transbase® deletes them when no longer needed or keeps them for backup reasons. In the latter case, a process must be setup to save those log records into an archive.

As a consequence of the XOR delta mechanism, the page layout must be exactly the same for master and slave databases, i.e. they must run on the same processor and under the same operating system.

2. Slave databases

First of all, it should be noted that one or more slave databases may be replicated from a single master database that is reachable via TCP/IP.

Slave databases must be read/only, i.e. the only way they can be changed is through the replicated log. Otherwise, i.e. for read/only queries they can be fully used for load balancing. In other words: Update transactions must be routed to the master database, while retrieval transactions may be processed by any of the slave databases or by the master database.

3. Database crash

A slave database may crash without affecting neither the master database nor the other slave databases. A new slave database can be established easily as described below.

When the master database crashes, anyone of the slave databases can take over. When taking over the log is replicated as far as possible as part of the take-over action; it has to be noted, however, that in this case transactions that have been active upon crash will be lost.

The slave database that takes over will become the new master database by simply „booting" the databases. As part of this, the database will be set read/write and all active transactions will be rolled back.

4. Architecture

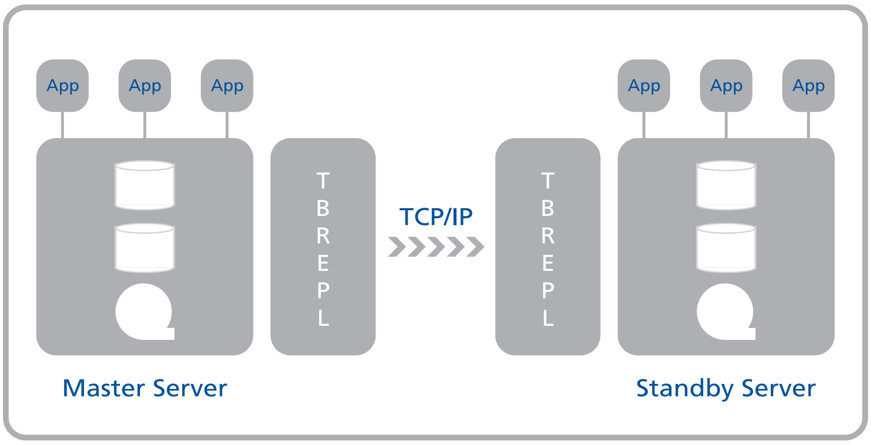
At master side, a process **tbrepl** must be established for each replicated database. This process is connected to the database shared memory and provides an internal thread for each slave to be served, in contrast to **tbkernel** which allocates one process for each client.

The command accepts the following options:

```
tbrepl master db=masterdb port=p
```

The db option must be filled with the name of the master database, the port can be specified explicitly if necessary (the default port is 2025) and should be different for each database to be replicated.

The architecture of a master database is shown by the picture on the right.



In order to start the replication into a slave database, following command is provided:

```
tbrepl init db=slavedb master=masterdb
```

This command transfers the initial state of the master database into the slave database and processes the log as far as it is available. Thus an operable snapshot of the master database is created as slave database.

After the snapshot is established the replication can be executed continuously by the command:

```
tbrepl start db=slavedb
```

or periodically by the command:

```
tbrepl db=slavedb.
```

Contact

Transaction Software GmbH
Willy-Brandt-Allee 2
81829 München, Germany
Tel.: +49 89 / 627 09 - 0
Fax: +49 89 / 627 09 - 11

info@transaction.de
www.transaction.de