



**zimory** going beyond...

# zimory®scale

Datenbankelastizität leicht  
gemacht

By Christian Link

# INHALT

ZIEL

MOTIVE

IDEE

VORAUSSETZUNGEN

UMSETZUNG

MESSUNGEN

ZIMORY

„Elastische, skalierbare und günstige  
Erweiterung für Oracle Datenbanken“

**ORACLE  
DATENBANKEN  
AUF  
PHYSISCHEN  
SERVERN**

**KÖNNEN NICHT  
OHNE MASSIVE  
EINGRIFFE  
ERWEITERT  
WERDEN**

Wartungsfenster für  
Erweiterungen

**LEIDEN UNTER  
ZUGRIFFS-  
KONFLIKTEN**

Applikationen  
(schreibend/lesend)

Reporting (lesend)

Data Warehouse  
(lesend)

Kundenservice  
(lesend)

**SIND  
KOSTENINTENSIV**

hohe Vorabinvestition in  
Hardware und Lizenzen

Erweiterungen  
verursachen erhöhte  
Lizenzkosten

**SIND AUF  
LASTSPITZEN  
OPTIMIERT**

keine dynamische  
Anpassung an  
Lastaufkommen

auf vordefinierte  
Leistung festgelegt

## ORACLE DATENBANKEN AUF PHYSISCHEN SERVERN

### REPLIKATION MIT STANDARD-TOOLS (STREAMS/DATA GUARD) LIMITIERT

Abhängigkeiten  
von Versionen und  
Editionen

Standby-DBs nur mit  
Aufwand oder Kosten  
(Active DG) für  
Abfragen verwendbar

### KÖNNEN NICHT BELIEBIG FÜR VERSCHIEDENE ANWENDUNGSFÄLLE OPTIMIERT WERDEN

Lese-Optimierung/  
Indizierung



Schreib-Optimierung

Replikate können nur  
mit viel Aufwand  
individuell angepasst  
werden

## EINSPARUNGEN

keine zusätzlichen  
Oracle-Lizenzen für  
Replikate

Einsparung neuer  
Lizenzen für die  
Master-DB

*Lastreduzierung durch  
Umleiten der Abfragen  
auf Replikate*

Replikate als VMs =  
reduzierte zusätzliche  
Investitionen für  
Hardware

## REPLIKATION

Oracle und non-Oracle Datenbanken

*MySQL, PostgreSQL*

*Optimierung für lesende Zugriffe  
z.B. noSQL Datenbanken*

keine Eingriffe in die Master-DB

komplett anpassbar an Kundenbedürfnisse

Replikation in heterogene Datenbank-Umgebungen

*z.B. parallele Replikation in Oracle  
und MySQL Satelliten*

## SKALIERUNG & ELASTIZITÄT

bedarfsorientierte  
Hinzu- bzw.  
Wegnahme von  
Replikaten

Replikate als  
VMs in Clouds

*Integration von Cluster-  
Lösungen wie z.B.  
Percona MySQL*

## LESEOPTIMIERUNG

umlenken der meisten lesenden Zugriffe  
auf Replikate

*Transparente Schicht zwischen  
Applikationen und Datenbanken*

Zugriffsoptimierte Indizierung

*Individuelle Datenmengen  
in Replikaten*

*Reduzierung auf benötigte Daten  
Archivierung (Blockierung von  
löschenden Aktualisierungen)*

## VORAUS- SETZUNGEN

### MASTER-DB

Stand-alone und RAC

Standard und  
Enterprise Edition

ab Oracle 11R1

### Satelliten-DBs

#### Oracle

*Standard und  
Enterprise Edition*

#### MySQL

*ab 5.1*

*empfohlen:  
Percona MySQL*

**PostgreSQL**  
(in Roadmap)

**noSQL**  
(in Roadmap)

### DATENBANK / APPLIKATIONEN

hoher Anteil lesender  
Zugriffe auf die  
Master-DB (~70:30)

Abfragen mit  
möglichst allgemeinen  
Statements

Zugriff via JDBC



## DATABASE- MIDDLEWARE

## UNABHÄNGIGKEIT VON ORACLE- VERSIONEN UND EDITIONEN

## EINFACHE INSTALLATION & ADMINISTRATION

eigener Server für  
Replikationsschicht

Konsole zur Steuerung  
der Replikation

kein Queueing auf  
Festplatte/Storage

## REPLIKATION MIT STANDARD- TOOLS/APIs

neuer DB-User für  
Replikation in  
Master-DB

Standard-APIs

LogMiner

JDBC

## DATABASE- MIDDLEWARE

BEI BEDARF  
ABSCHALTBAR  
OHNE  
PRODUKTIONS-  
UMGEBUNG ZU  
BEEINFLUSSEN

TRANSPARENTER SCHICHT ZWISCHEN DER  
DATENBANK UND DEN ZUGREIFENDEN  
APPLIKATIONEN / ANWENDERN

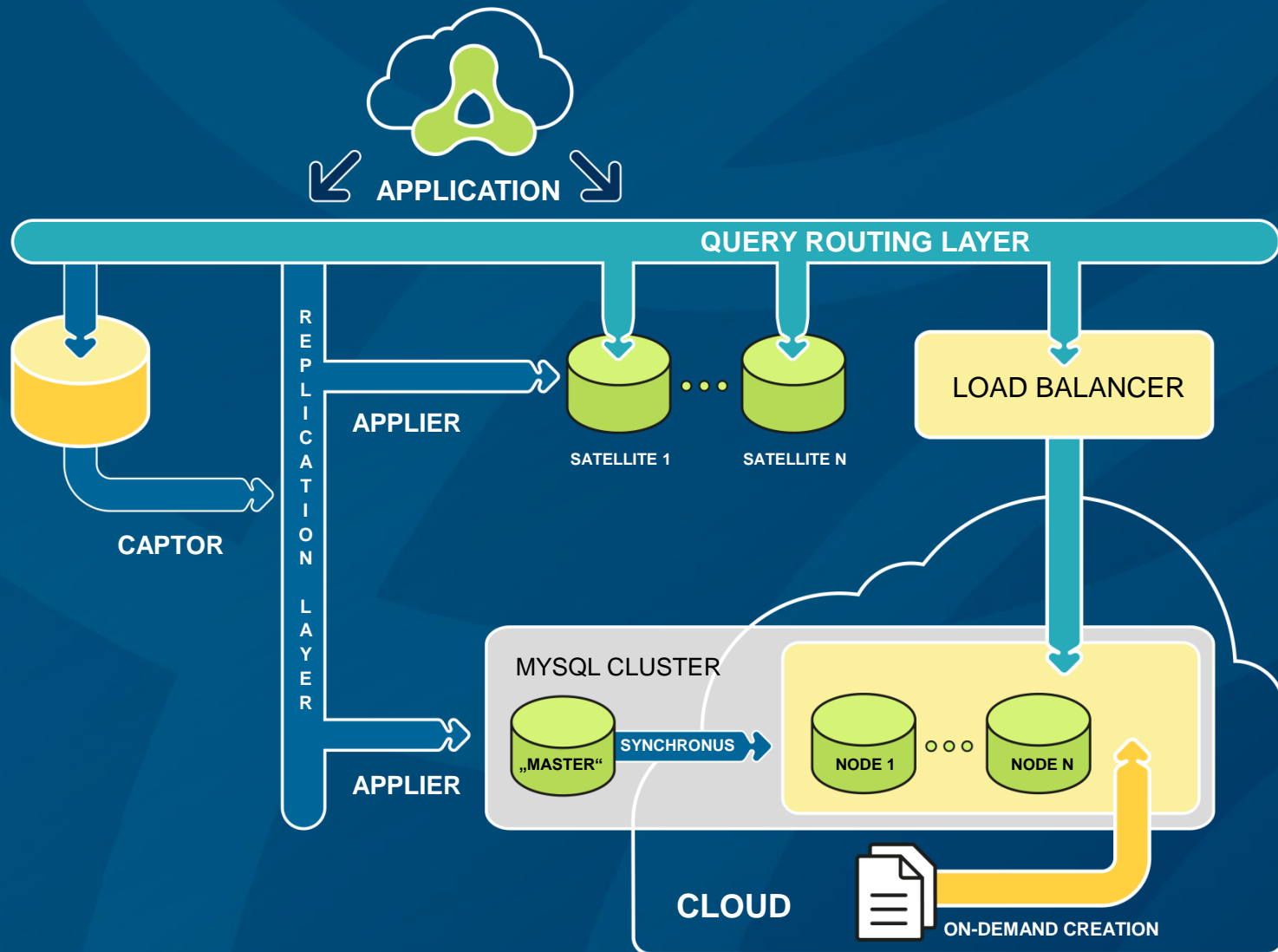
erweiterter JDBC-Treiber (Wrapper)

umleiten von lesenden Zugriffen

1:1 Durchgriff von schreibenden und  
Oracle-spezifischen Abfragen auf Master

integriertes Load Balancing

Übersetzung von SQL-Dialekten



## REPLIKATIONSSCHICHT

AUSLESEN DER REDOLOGS  
UND ARCHIVELOGS

KEIN QUEUEING AUF  
FESTPLATTE/STORAGE

IN MEMORY PROCESSING

PRO SCHEMA

AUF KUNDENBEDÜRFNISSE  
ANPASSBAR

limitierte Datenmengen

Blacklist für Tabellen

Blacklist für SQL-Anweisungen  
*z.B. nur INSERT und UPDATE*

## REPLICATION MODES

**KÖNNEN PRO  
SATELLITE ODER  
GLOBAL AKTIVIERT  
WERDEN**

**PERSISTENTE  
SPEICHERUNG  
DER KONFIGURATION**

neue Satelliten  
übernehmen  
automatisch globale  
Replication Modes

**SKIP**

gezieltes Ignorieren  
von SQL-Anweisungen

**OVERWRITE**

Überschreiben der  
Daten im Satelliten  
mit den Master-Daten

## REPLICATION MODES

### COMPAIR & REPAIR (Roadmap)

interaktive Behebung  
von Konflikten

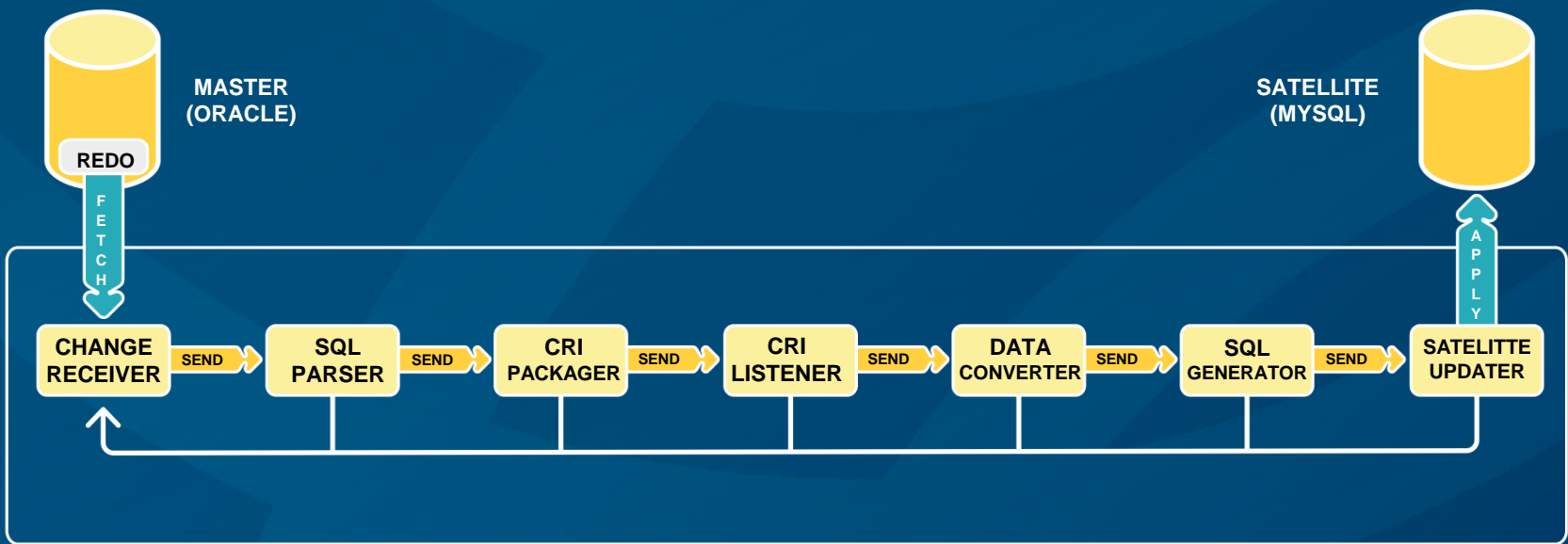
### BULK

Umschalten auf  
JDBC BATCH  
UPDATE/INSERT

Beschleunigung bei  
vielen, gleichartigen  
Statements  
(z.B. Migrationen)

### COMPRESSION AND ENCRYPTION (Roadmap)

Optimierung für WAN  
Replikation



## QUERY ROUTING LAYER

### TRANSPARENTE SCHICHT ZWISCHEN APPLIKATION UND DATENBANK

Übersetzung von SQL-  
Dialekten

Result Sets werden als  
Oracle-Result Set  
zurückgeliefert

### INTELLIGENTES QUERY ROUTING

lesende, kompatible /  
übersetzbare  
Abfragen gegen  
Satelliten

schreibende und  
Oracle-spezifische  
Abfragen gegen  
Master-DB



## QUERY ROUTING LAYER

### KONSISTENZ- CHECKS

Routing nur auf  
konsistente Satelliten

Re-Routing auf Master  
falls keine Satelliten-  
DB konsistent

### INTEGRIERTES LOAD BALANCING

Round Robin

best responding  
satellite

## SQL REWRITING

### STANDARD-SQL FUNKTIONEN

*AVG, SUM, MIN,  
MAX, etc.*

### JOINS

*INNER JOIN, OUTER  
(left/right) JOIN, FULL  
OUTER JOIN*

## DATA CONVERSION

Basic data types  
(NUMBER, CHAR,  
VARCHAR2,  
DATE, etc.)

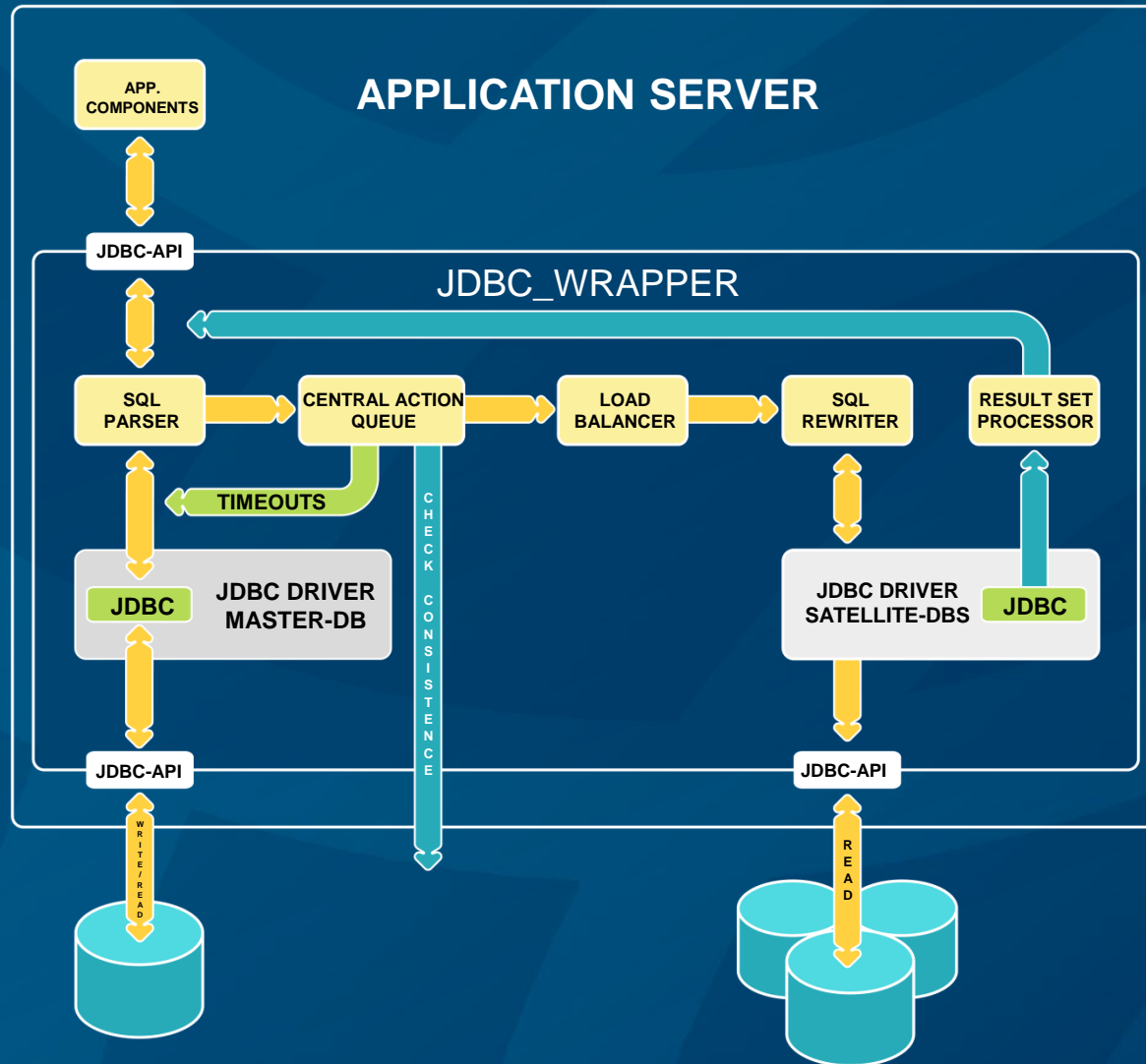
*Oracle: ja*

*MySQL: ja*

Binary data types  
(RAW, CLOB,  
LONG, etc.)

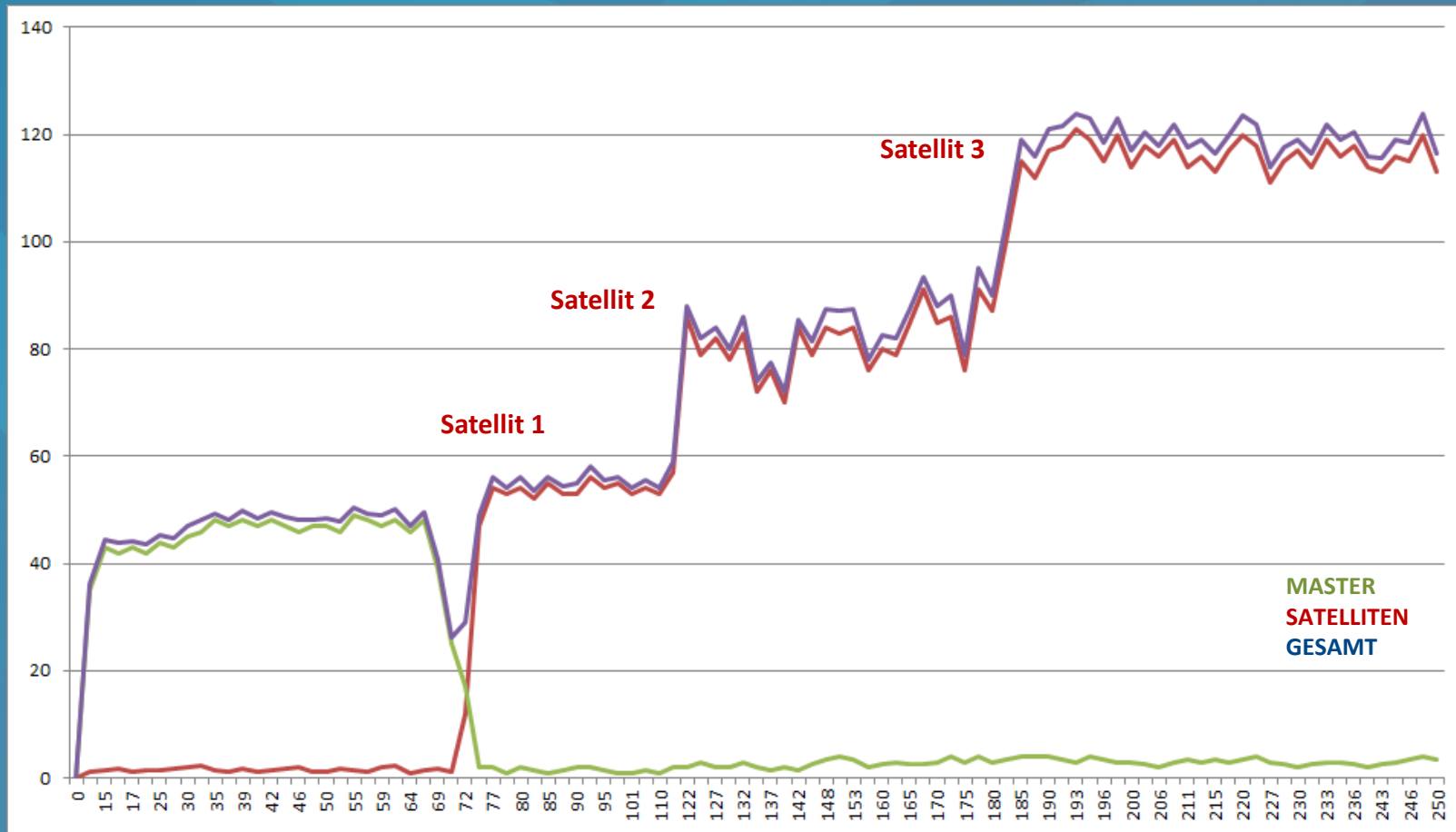
*Oracle: ja*

*MySQL: eingeschränkt,  
wird aktuell entwickelt*



Command	Description
version	Shows the Replicator version
help <command>	Shows help for a command
connect <IP/hostname>	Connects to a Replicator instance
disconnect	Disconnects from the Replicator instance
status	Shows the client status
addsat	Adds a satellite to the Replicator (interactive)
remsat <satellite_ID>	Removes a satellite from the Replicator
showsat <satellite_ID>	Shows properties about a satellite
showsat all	Shows properties about all satellites
start <satellite_ID>	Starts replication to a satellite (interactive)
start all	Starts replication to all satellites (interactive)
restart <satellite_ID>	Restarts replication to a satellite
restart all	Restarts replication to all satellites
stop <satellite_ID>	Stops replication to a satellite
stop all	Stops replication to all satellites
stop abort <satellite_ID>	Hard stops replication to a satellite
stop abort all	Hard stops replication to all satellites
stats <satellite_ID>	Shows statistics about a satellite
clearsat <satellite_ID>	Removes replicator related state from a satellite
bulk <enable/disable> <satellite_ID>	Toggles Batch Mode On or Off
createtables <satellite_ID>	Creates master db tables at the specified satellite.
createindexes <satellite_ID>	Creates indexes on specified satellite.
exit	Disconnects and exits from the console
quit	Disconnects and exits from the console
help	Shows help

# TPC-W Benchmark (Bookstore)



## PoC – Internet Plattform (Immobilien)

Test	Oracle Streams	zimory®scale
Production workload (10min live traffic)	avg: 19 sec, max: 33 sec	avg: <1 sec, max: 3 sec
Production workload with bulk delete 100k	avg: 20 sec, max: 57 sec	avg: 2 sec, max: 56 sec

# zimory

GRÜNDUNG 2007

SITZ IN BERLIN, BÜROS  
IN ERFURT UND MINSK

55 MITARBEITER

SPIN-OFF T-LABS

CLOUD MANAGEMENT  
SOFTWARE  
(Z.B. T-SYSTEMS)

zimory GmbH  
Revaler Straße 100  
10245 Berlin  
+49 (0)30 609 85 07-0  
[www.zimory.com](http://www.zimory.com)