



# TimesTen In-Memory Database for Extreme Performance

Try Oracle Cloud Platform For Free



[INTRODUCTION](#), [TIMESTEN TALK](#) | July 25, 2018

## An Introduction to Oracle TimesTen Scaleout

[Doug Hood](#)

TIMESTEN CLOUD PRODUCT MANAGER

### Introduction

Finally there is a scale out database where you don't need to trade data consistency for scalability. A database where you don't need to de-normalize to get [extreme performance](#). An OLTP SQL database that is faster than NoSQL databases. This database is called **TimesTen Scaleout**.

Oracle TimesTen In-Memory Database has been a leading relational in-memory database for mission critical applications for over 20 years. Oracle TimesTen In-Memory Database has an exciting new scalability feature called TimesTen Scaleout. TimesTen Scaleout is a [distributed database](#) which has a [shared nothing architecture](#) suitable for low latency, high throughput [OLTP](#), [IoT](#) and [analytic](#) workloads. TimesTen Scaleout supports [high availability](#), [ACID](#) transactions, SQL & PL/SQL and does not require database [de-normalization](#).



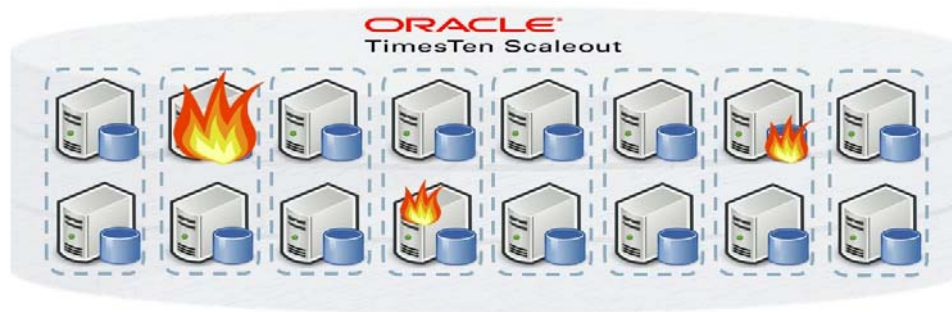
## Scale Out

To enable linear scalability, TimesTen Scaleout uses a shared nothing architecture. This means that the database can scale horizontally over a set of machines. Adding more machines to a TimesTen Scaleout database enables more connections, greater throughput, larger capacity and higher availability. Adding more machines also enables SQL operations to perform in parallel. In general, the more machines the more effective the parallel SQL processing.



## High Availability

A TimesTen Scaleout database will usually span several machines. Multiple copies of the data can be configured so that there are no single points of failure. Having multiple copies of the data means that transient hardware and software failures do not affect the availability of the database. If all copies of the data become permanently unavailable, then the database can be recovered from backup.



## ACID Transactions

Despite being an 'In-Memory Database', TimesTen databases are persistent across system or power failures. TimesTen persists data to a disk or flash filesystem. TimesTen Scaleout uses **ACID** (Atomic, Consistent, Isolated & Durable) transactions, just like Oracle databases.

TimesTen Scaleout achieves strong consistency via an optimized **Two Phase Commit Protocol [2PC]**. Unlike various 'eventual consistency' mechanisms, the 2PC protocol ensures that all changes are atomic. This strong consistency means that the transactional model is the same as if a single instance RDBMS is being used. This means that developers do not need to worry about how transactions work, they just need to commit or rollback their units of work.

Simple  
2PC ✓  
✓

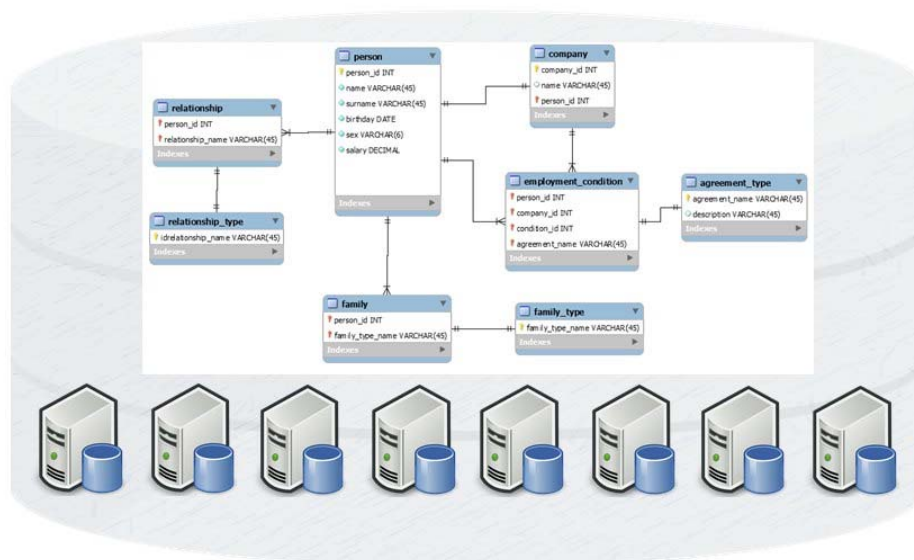
Commit  
Rollback ✓  
✓

Strong Consistency ✓  
Eventual Consistency ✗

## Single Database Image

Although the database can consist of many physical or virtual machines, it still appears to be a single logical database to developers and applications. Developers just see a single database with Schemas, Tables, Primary Keys, Foreign Keys, Indexes, Views, Materialized Views and Sequences.

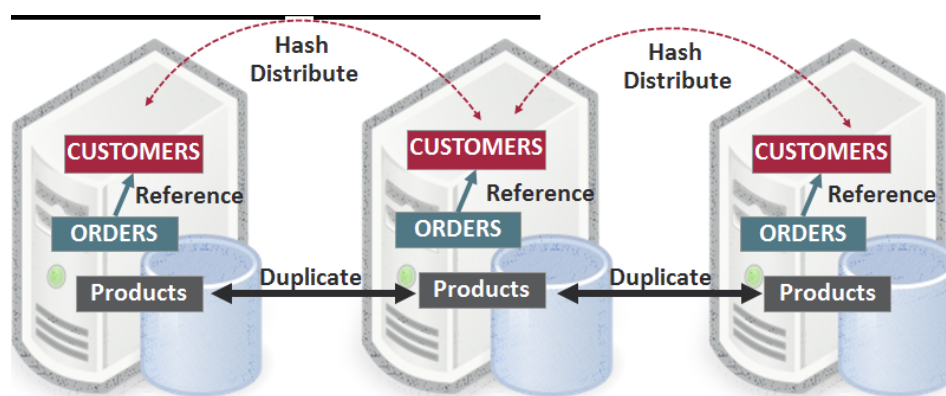
Applications can **modify any number of rows** stored on any combination of machines in a **single transaction**.



## Table Distribution Clauses

As TimesTen is an In-Memory Database, it can execute extremely fast SQL Select/Insert/Update/Delete and Join operations within a single machine.

With other scale-out databases, as you spread your database across multiple machines you will tend to have performance problems. TimesTen Scaleout solves this by allowing you to co-locate commonly accessed data on the same machine. Depending on the SQL operation, [data co-location](#) can minimize or even eliminate network messaging and has a dramatic effect on performance and scalability.



Data co-location is configured via table distribution clauses. By default, rows in a table are evenly distributed across the machines in the TimesTen Scaleout database via [consistent hash](#). For detail tables that are often joined, the foreign keys can be co-located with their parent table via the **Distribute By Reference** syntax. Reference tables which are 'read mostly' can be duplicated across all machines in a TimesTen Scaleout database via the **Duplicate** syntax.



```
CREATE USER scott IDENTIFIED BY tiger;
GRANT CONNECT, CREATE TABLE TO scott;
```

```
CREATE TABLE Customers
( CustId VARCHAR2(60) NOT NULL,
  FirstName VARCHAR2(27),
  LastName VARCHAR2(42),
  ...
  PRIMARY KEY(CustId)
)
DISTRIBUTE BY HASH (CustId);
```



#### Note

The **Distribution** clauses are optional  
If no **Distribution** clause exists, then the table will default to 'distribute by hash (the PK columns)'

```
CREATE TABLE Orders (
  OrderId NUMBER(16) NOT NULL,
  CustId VARCHAR2(60) NOT NULL,
  OrderDate TIMESTAMP,
  ...
  PRIMARY KEY (CustId, OrderId),
  CONSTRAINT fk_orders_parent
  FOREIGN KEY (CustId) REFERENCES
  Customers(CustId)
)
DISTRIBUTE BY REFERENCE (fk_orders_parent);
```

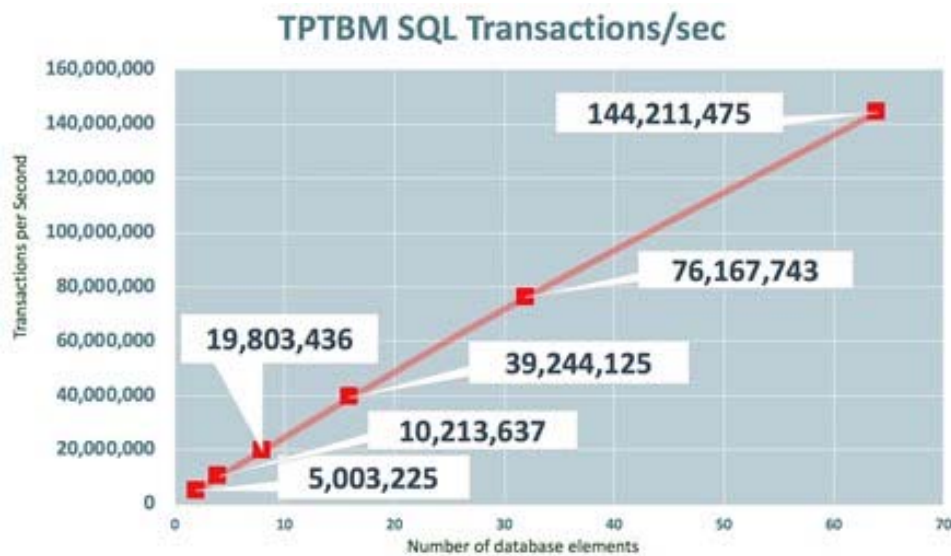
```
CREATE TABLE Products (
  ProductId INTEGER PRIMARY KEY,
  Name VARCHAR2(128),
  LastPrice NUMBER(19,4),
  ...
)
DUPLICATE;
```

## Benchmarks

As TimesTen Scaleout uses a shared nothing architecture, it is possible to achieve linear scalability for well tuned SQL workloads. At Oracle Open World 2017, a **beta** version of TimesTen Scaleout achieved **1.2 Billion SQL selects per second** for 100% reads with the **TPTBM** workload.

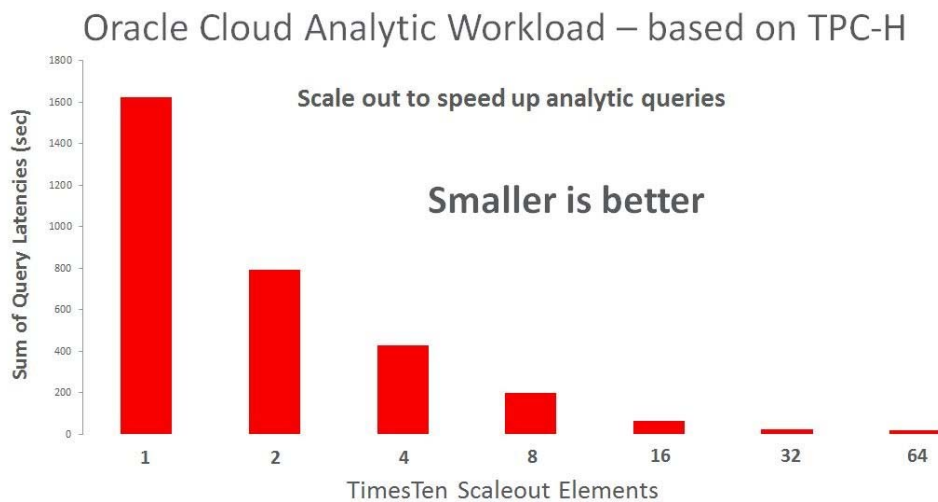


This benchmark used 64 BM.HighIO1.36 compute nodes in the Oracle Cloud. By adding more Oracle Cloud compute nodes to the TimesTen Scaleout database the throughput continued to increase and this benchmark showed linear scalability up to 64 machines.



The same **TPTBM** workload also showed linear scalability with 80% reads and 20% writes achieving **144 million transactions per second** using 64 database elements on 32 BM.HighIO1.36 compute nodes in the Oracle Cloud.

The Oracle Cloud Analytic Workload is a good example of a complex analytic workload [it uses the TCP-H schema and data but uses more complex queries] that scales as more machines are added to the TimesTen Scaleout database.



This chart shows that as more machines are added to the TimesTen Scaleout database, the time to run all of the analytic queries is reduced. Specifically as the number of machines doubles, the latency halves. This dramatic reduction in latency is constrained by [Amdahl's law](#).

## SQL APIs



TimesTen Scaleout supports SQL and PL/SQL via JDBC, ODBC, OCI, Pro\*C, ttClasses and ODP.NET.



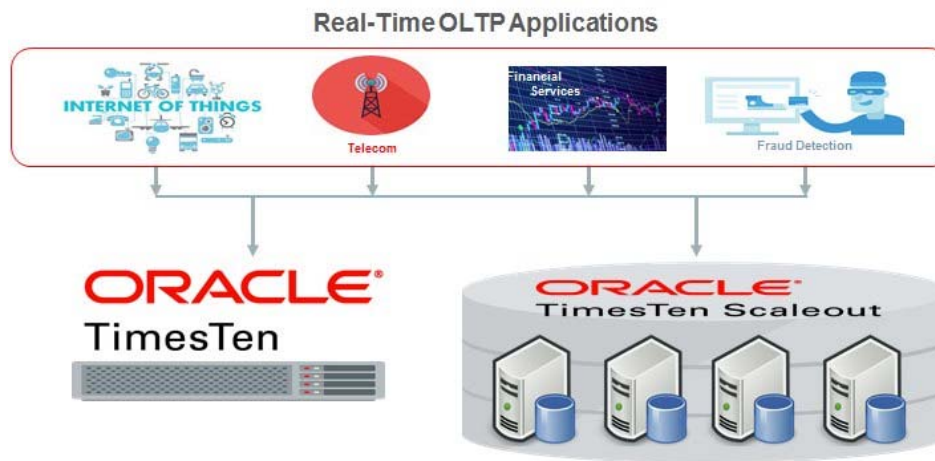
TimesTen Scaleout also works with open source languages with SQL APIs that use an Oracle OCI driver or ODPI.



## Developing Applications for TimesTen Scaleout

Existing TimesTen applications should work on TimesTen Scaleout with little or no change. Chongqing Mobile, [a subsidiary of China Mobile] recently went into production using TimesTen Scaleout by deploying an existing TimesTen application with only two changes. They changed the connect string to point to TimesTen Scaleout and they added distribution clauses to the tables to optimize the queries.

Also as TimesTen Scaleout uses a subset of Oracle Database compatible APIs, SQL, PL/SQL and data types, Oracle Database application developers can easily target their applications at TimesTen.



Details on how to develop applications for TimesTen Scaleout can be found [here](#).

## SQL Developer Support for TimesTen Scaleout

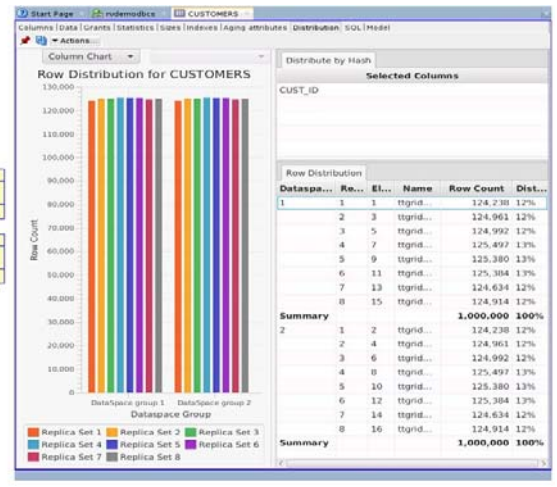
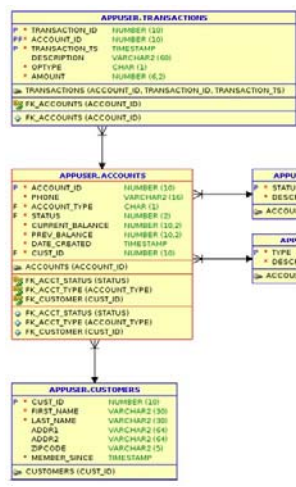
SQL Developer 18.2 supports full life cycle management for TimesTen Scaleout 18.1.

A SQL Developer window showing that the accounts table uses the **distribute by reference** table distribution clause.

```
-- Database is in Oracle type mode
create table APPUSER.ACCOUNTS (
  ACCOUNT_ID      NUMBER(10) NOT NULL,
  PHONE           VARCHAR2(16 BYTE) INLINE NOT NULL,
  ACCOUNT_TYPE    CHAR(1 BYTE) NOT NULL,
  STATUS          NUMBER(2) NOT NULL,
  CURRENT_BALANCE NUMBER(10,2) NOT NULL,
  PREV_BALANCE   NUMBER(10,2) NOT NULL,
  DATE_CREATED    DATE NOT NULL,
  CUST_ID         NUMBER(10) NOT NULL,
  primary key (ACCOUNT_ID),
  constraint FK_ACCT_STATUS foreign key (STATUS) references APPUSER.ACCOUNT_STATUS (STATUS),
  constraint FK_ACCT_TYPE foreign key (ACCOUNT_TYPE) references APPUSER.ACCOUNT_TYPE (TYPE),
  constraint FK_CUSTOMER foreign key (CUST_ID) references APPUSER.CUSTOMERS (CUST_ID))
distribute by reference (FK_CUSTOMER);
```

Some SQL Developer windows showing a **data model** and the **hash distribution** of rows for the customers table.





## Platforms

TimesTen Scaleout is officially supported on these [platforms](#) and has been tested on the following Linux distributions:

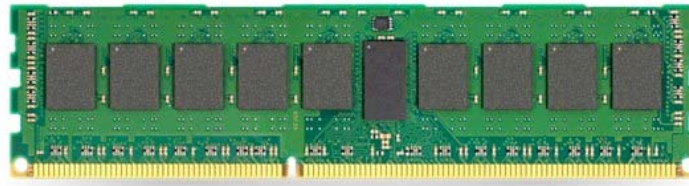


TimesTen Scaleout is officially supported on Oracle VM and has been tested on the following VMs:



## Hardware

TimesTen Scaleout is software that runs on Linux x8664 hardware. The performance of TimesTen Scaleout is dependent on the [performance of the hardware components](#) - you get what you pay for. Dedicated bare metal machines will tend to give better performance than Virtual Machines.



## Downloads

- Oracle TimesTen 18.1 Scaleout can be downloaded from [here](#)
- Oracle SQL Developer 18.2 can be downloaded from [here](#)
- A VirtualBox VM with a Hands On Lab for TimesTen 18.1 Scaleout can be downloaded from [here](#)
- Sample code for TimesTen 18.1 Scaleout can be downloaded from [here](#)

**Disclaimer:** These are my personal thoughts and do not represent Oracle's official viewpoint in any way, shape, or form.

## Be the first to comment

Comments ( 0 )

## Recent Content



## TIMESTEN TALK

[TimesTen 18c - One Download, Two Products, Three Use Cases](#)

Oracle TimesTen 18c [18.1.2.1.0] is now available for download. With this release, the Linux x8664 download ZIP file contains two products...

## TIMESTEN TALK

[Is the Python cx\\_Oracle SQL Driver fast enough for Database Benchmarks?](#)

I have been working on a customer database benchmark where latency and throughput are critical. The customer provided me with some sample...

## TIMESTEN TALK

[Using Intel Opt Memory with O Memory Datab](#)

If you missed t World 2018 liv YouTube videc Optane DC Pe Oracle TimesT

[Site Map](#)[Legal Notices](#)[Terms of Use](#)[Privacy](#)[Cookie Preferences](#)[Ad Choices](#)[Oracle Content Marketing Login](#)