

## **Flexible Availability Options with GoldenGate's Transactional Data Management Platform (TDM)**

February 2007

Active/active systems achieve their extraordinary availabilities by recovering from a fault in seconds. In effect, the active/active philosophy is to let a system fail, but fix it fast.

In order to recover so quickly, these systems are configured as a network of nodes, each actively processing transactions for a common application. Should a node fail, the users at that node are simply switched to a surviving node.

To ensure survivability of the application network, there must be at least two application database copies, which are separated geographically. These database copies must be synchronized so that all applications running at different nodes see the same data.

A common method to achieve database synchronism is to replicate changes that are made at each database to the other database copies in the network. An example of such a solution is the Transactional Data Management (TDM) platform from GoldenGate Software, Inc. ([www.goldengate.com](http://www.goldengate.com)).

### **The GoldenGate High Availability Offering**

GoldenGate's TDM provides real-time data movement and verification across distributed heterogeneous platforms. It is the cornerstone of the GoldenGate product suite that includes:

- TDM, the core engine which includes capture, delivery and manager components
- GoldenGate Director for monitoring and configuring the TDM components across the enterprise.
- GoldenGate Veridata for comparing two databases and reporting discrepancies.

These components are described in this product review.

There are many uses for real-time data movement and verification, such as that provided by GoldenGate's TDM:

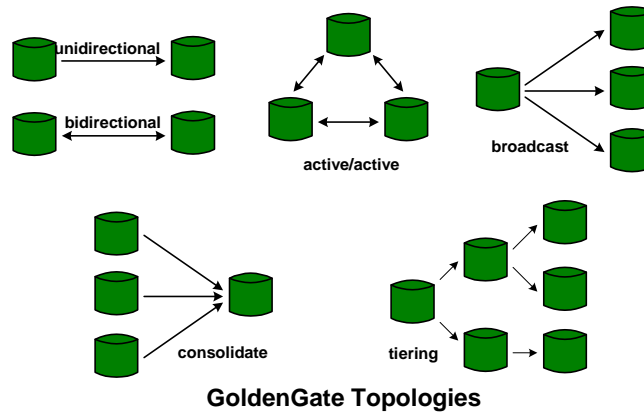
- Synchronizing the databases in an active/active application network.
- Synchronizing a hot standby system so that it is instantly ready to take over processing should the primary system fail.
- Distributing changes made to a master database in real time to other database copies for real-time warehousing or for query and reporting purposes.
- Real-time data vaulting to store transactional data at a remote site.
- Zero downtime operations to eliminate planned downtime for upgrades and migrations.

# The Transactional Data Management Platform

## What Is TDM?

TDM<sup>1</sup> captures database changes made to an operational source database and moves them asynchronously to a target database in near real time with low latency (the time from when a change is made to the source database to the time that it is applied to the target database). It can be configured to be unidirectional, replicating in one direction, or bidirectional, replicating in both directions.

Flexible network technologies are supported. TDM can be configured in a multinode peer-to-peer configuration (active/active). It can be configured to broadcast data changes from a master database to multiple slave databases or to consolidate changes made to several database copies onto a master database. It can support cascading of data down a multi-tiered database structure.



TDM configurations can be heterogeneous. The databases, the hardware platforms, and the operating systems can all be different. TDM supports data transformation between the source and target databases to support heterogeneity, whether it be simply different data structures in otherwise identical databases or database systems from different vendors.

The communication of changes between the source and target systems is via an IP network. There is no limitation on the distance between databases.

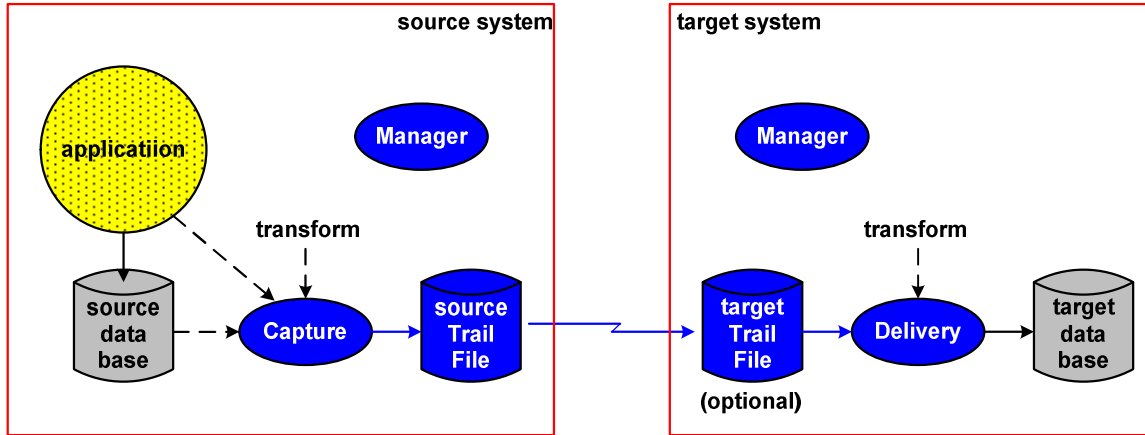
TDM supports dynamic rollback and reconstruction of a database for rapid recovery from errors that may have contaminated the database. Before-images are used to create reverse transactions to a specified point in time that is known to correspond to an accurate database. After-images are then used to create transactions to return the database to its proper current state. Dynamic rollback obviates the need for a full data reload to recover a corrupted database.

## The TDM Architecture

TDM is implemented as a set of decoupled processes that can proceed independently of each other. For instance, if the process that updates the target database is slower than the process that reads the source database, the source process does not have to pause. It can continue on at its natural speed by simply queuing database changes to disk for later processing by the target process.

The TDM processes and other components include a Capture process that receives changes to the source database, a communications facility to send source database changes to the target system, a Delivery process to apply changes to the target database, Trail Files to provide persistent storage of database changes, and a Manager process to control the processes at a node.

<sup>1</sup> GoldenGate Transactional Data Management Platform – Solution and Technology Overview, GoldenGate white paper; November, 2005.



## GoldenGate Transactional Data Management (TDM) Platform

### Capture

The role of the Capture process is to receive all changes to the source database. These changes could come from a variety of sources. They might be extracted from a transaction log (redo log or audit trail) maintained by the source database or transaction manager. If such a log is not available, changes may be fed to the Capture process directly by the application; or they may be fed by change information generated by database triggers.

The Capture process will filter out intermediate activities and rolled-back operations. It will only forward changes corresponding to completed transactions. It accomplishes this by waiting for the transaction's COMMIT operation before sending the transaction updates to the target side.

The Capture process can be instructed to only forward changes from selected tables or rows and optionally can compress change data to speed processing.

### Trail Files

Data changes captured by the Capture process are written to the source-side Trail File. Since the Capture process filters out aborted transactions, the Trail File contains only committed transactions. The Trail File is therefore a queue of transactions to be sent to the target system. There may also be a target system Trail File which will queue changes prior to applying them to the target database.

Prior to writing to the Trail File, the change data format is converted to GoldenGate's Universal Data Format (UDF). This facilitates later conversion to the format of the target database.

The Trail Files also function as consistent en-route checkpoints for the data replication process so that replication can be continued without data loss following a fault, at least so far as the data that has made it to the Trail File is concerned. The Trail Files are implemented outside of the databases to protect them from database failures.

### Communications

The transfer of data changes from the source system to the target system is via IP (the Internet Protocol). The communication channel can be any medium that supports IP, such as WANs, LANs, the Internet, or fiber. Therefore, there is no geographical limit on the separation of the source system from the target system.

Data changes are blocked and optionally compressed to improve communication efficiency. Encryption using the 128-bit Blowfish algorithm is available if needed.

### Delivery

Data changes are delivered to the Delivery process in UDF format from the target-side Trail File if there is one or otherwise directly from the communication channel.

Where possible, the Delivery process uses the data-change information to update the target database via the target database's native SQL. If the database does not have a native SQL, ODBC is used to do the update. Any database that has an ODBC driver can be a target database for GoldenGate TDM.

TDM can be configured so that the order of transaction commits at the source database is preserved when applying these commits to the target database. In this way, referential integrity at the target is guaranteed.

A single source database can feed multiple target databases.

### Manager

The Manager controls the TDM processes on its node via a command-line interface. It provides the facilities for TDM configuration, critical event and threshold monitoring, log resource management, and the management and purging of old Trail Files.

TDM resources across the enterprise can be monitored with the GoldenGate Director, described later.

### ***Transformation and Mapping***

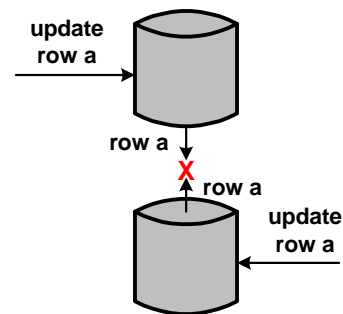
Database changes captured from the source database can be mapped and transformed into the formats required by the target database via GoldenGate-supplied functions, by user-supplied code installed as user exits, or by stored procedures in the database. Data mapping and transformation can be done either by the Capture process, by the Delivery process, or by both.

The user can specify table or row filtering and which columns are affected. GoldenGate functions available for mapping and transformation include date, math, string, and utility functions.

### ***Conflict Detection and Resolution***

A data collision occurs when two users each attempt to modify the same row at the same time at two different nodes. TDM can detect collisions and can be directed to take one of several actions, including:

- Ignore the collision.
- Retry the operation.
- Suspend replication for external action.
- Apply the increments to the field.
- Select a winning update based on the position of the source node in a nodal hierarchy.
- Resolve with user-defined rules coded as user exits.
- Log for manual resolution.



## ***Zero Downtime Operations***

TDM can support upgrades to the system as well as migration to new system components without incurring planned downtime. Migration is accomplished by starting capture on the source system, loading a snapshot of the database onto the new target system, and then applying all of the new database updates that have accumulated since the capture was begun. Once the target has been placed into service, reverse replication can be turned on so that the original system can be placed back into service if the new system should experience problems.

In a multinode system, planned downtime for maintenance and upgrades can be avoided by switching users from the node to be worked on to other nodes in the application network so that their service continues while their original node is out of service.

## ***Database Tiering***

Copies of parts or all of the database can be moved to other systems for browsing, querying, or reporting so as to reduce the load on the primary transaction-processing system. Database tiering is especially useful in applications where the “look-to-book” ratio is very high.

## ***Heterogeneity***

TDM can replicate between a wide variety of databases and operating systems, including the following:

<b>Databases</b>	<b>Operating Systems</b>
Oracle	Windows
DB2 UDB for z/OS	Linux
Microsoft SQL Server	Sun Solaris
MySQL	HP-UX
HP Enscribe	HP NonStop
HP NonStop SQL (MP and MX)	HP Tru64
Sybase	IBM OS/390
Teradata	IBM AIX
Ingres	IBM z/OS

In addition, any database that is ODBC-compatible can be a target.

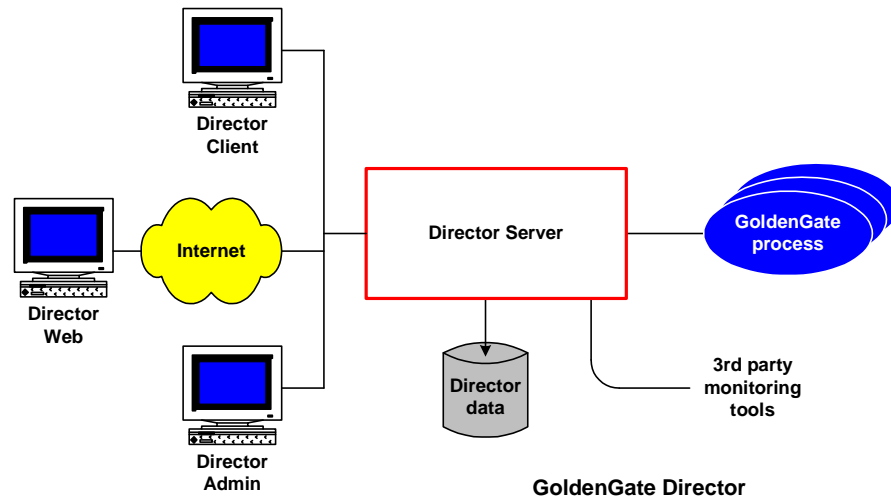
## **GoldenGate Director**

The GoldenGate Director is an optional add-on product that can be used to configure and manage multiple host platforms from a single console. It runs in its own server and provides a GUI management interface either directly or over the Web. It can integrate other 3<sup>rd</sup> party monitoring facilities via built-in or custom adapters.

The GoldenGate Director comprises a Director Server, a fat Director Client, an optional Web-based Director Client, and a Director Administrator. The Director Server sits on a central server and maintains a repository of security settings, event logs, and statistics, such as latency times. It manages the connections to the various GoldenGate processes in the network, triggering email alerts if there is a problem that needs to be reported.

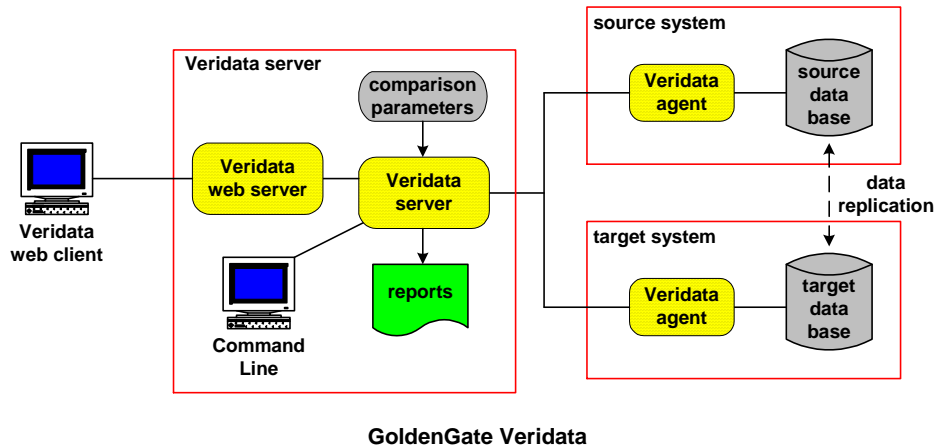
The Director Client and Director Web provide views of process status, alerts, and consolidated logs via a GUI interface.

The Director administrator manages the initial system setup and configuration. It provides intelligent parameter control by suggesting appropriate parameter values and trapping invalid parameters.



## GoldenGate Veridata

GoldenGate Veridata is a stand-alone product that identifies and reports on discrepancies between two distinct operational databases. It does this by comparing two databases online with no interruption to the applications. Verification is accomplished by comparing patent-pending row signatures. Veridata provides reports of discrepancies for manual review and correction.



The user can restrict the scope of verification by defining only the pertinent tables or data fields to be compared. Veridata identifies records in flight and users have the flexibility to determine how to handle in-flight data based on their own requirements.

As a standalone product, Veridata can be used in the absence of any other GoldenGate products.

## Summary

The GoldenGate TDM platform is used by over 300 customers in 35 countries. GoldenGate is also used in conjunction with a number of mission critical applications for high availability and real-time data integration, specifically ACI's BASE24 financial applications.

Other TDM application areas include banking, financial services, healthcare, public sector, aerospace, e-business, telecommunications, and retail.