HPE Shadowbase Streams for Data Integration

A Gravic, Inc. White Paper

Executive Summary

The HPE Shadowbase product suite (built by Gravic, Inc. sold by HPE) distributes real-time critical data and information to target databases and to application environments throughout the enterprise. Uses for Shadowbase products include achieving high or continuous availability for business continuity environments by synchronizing data changes between active or passive redundant systems, integrating applications, feeding data warehouses and real-time business intelligence facilities, and driving extract, transform, and load (ETL) utilities.

HPE Shadowbase Streams, a member of the HPE Shadowbase software solutions suite, uses change data capture (CDC) technology to stream data generated by one application to other applications as well as to target database environments. HPE Shadowbase Streams for data integration and HPE Shadowbase Streams for application integration provide the facilities for integrating existing applications at the data or event-driven level in order to create new and powerful functionality, supporting the Real-Time Enterprise (RTE).

Using Shadowbase Streams, changes made in any database can be quickly and easily integrated into other data environments to keep that target information synchronized. The changes can be made in real-time or can be batched for periodic snapshot or micro-batch updating.

Additionally, using Shadowbase Streams, applications that once were isolated can now interoperate in an event-driven fashion in real-time. Critical data generated by one application is distributed and acted upon immediately by other applications, enabling the implementation of Event-Driven Architectures (EDA).

Shadowbase Streams supports many models for data distribution, including maintaining a remote database of critical data; sending critical data to client applications or servers directly via queues, inter-process messages, TCP/IP sessions, or client APIs; publishing data to applications that have subscribed to the data; and responding to poll queries from client applications.

Shadowbase Streams is extensible. It allows the user to embed custom processing logic into the data-processing path. It readily filters, transforms, and adapts data from one application or database environment into the protocol or format required by another application or database environment, all without requiring any changes to the existing source application or database.

Shadowbase Streams modernizes legacy applications by integrating diverse applications across the enterprise so that new and valuable services may be generated to enhance competitiveness, to reduce costs or to increase revenue, to satisfy regulatory requirements, and to generally improve the user experience.

In this paper, several customer production examples of Shadowbase Streams for data integration follow a description of how Shadowbase Streams works.

Note that this paper focuses on using Shadowbase Streams for data integration. For application integration needs, please refer to the companion document, *HPE Shadowbase Streams for Application Integration*.

Table of Contents

Executive Summary......2
The Data Continuum 4

The Need to Integrate Data 4
The Solution......5

HPE Shadowbase Streams for Data Integration......5
Data Mapping 6
Distributing Data to Multiple Targets 7
Consolidating Data from Multiple Sources 7
Extract, Transform, and Load (ETL) 8
Data Warehouses and Data Marts 9
Real-Time Business Intelligence (RTBI) 9
Application Interoperability 10
Offloading Reporting and Querying......10

HPE Shadowbase Streams for Data Integration Architecture 11
Data Transformation......13

Case Studies 14
A Large Internet Service Provider Distributes Profile Changes in Real-Time......14
Integrating HPE NonStop Trading Applications with Online Trading Applications......15
Integrating HPE NonStop Trading Applications with Brokerage Data Warehouses in a Service Bureau Environment......16
Integrating HPE NonStop Telco Applications with a Data Warehouse via ETL Technology 17
Service Bureau Provides Customer Access to Securities Account Information (plus co-existence with RDF)......18

Summary......19

International Partner Information......20

Gravic, Inc. Contact Information 20

Table of Figures

Figure 1 – The Data Continuum 4
Figure 2 – Data Integration......5
Figure 3 – HPE Shadowbase Streams for Data Integration......6
Figure 4 – HPE Shadowbase Streams Data Mapping 6
Figure 5 – HPE Shadowbase Streams Fan Out......7
Figure 6 – HPE Shadowbase Streams Consolidation......8
Figure 7 – Updating a Data Warehouse or RTBI System in Real-time 9
Figure 8 – The HPE Shadowbase Data Replication Engine 11
Figure 9 – HPE Shadowbase Supported Platforms, Databases, and Operating Environments......13
Figure 10 – HPE Shadowbase Streams Distributes Profile Changes in Real-Time 14
Figure 11 – HPE Shadowbase Streams Integrates HPE NonStop Trading Application with Online Trading Applications......15
Figure 12 – HPE Shadowbase Streams Integrates HPE NonStop Trading Applications with Brokerage Data Warehouses in a Service Bureau Environment......16
Figure 13 – HPE Shadowbase Streams Integrates HPE NonStop Telco Applications with a Data Warehouse via ETL Technology......17
Figure 14 – HPE Shadowbase in a Multiple Replication Engine Environment 18

HPE Shadowbase Streams for Data Integration

HPE Shadowbase Streams (built by Gravic, sold by HPE) is a powerful and flexible facility that enables diverse applications to interoperate with each other at the data level (called data integration) and at the event-driven level (called application integration). This capability is provided even for legacy applications that were never intended nor designed to work together. Shadowbase Streams is a member of the HPE Shadowbase product suite and focuses on distributing information in real-time throughout the enterprise, enabling the provision of a Real-Time Enterprise (RTE).[1] It rapidly delivers information where and when it is needed without customer application modification. The application's responsibility is simply to make functional use of the data delivered by Shadowbase Streams.

With data integration, Shadowbase Streams creates a real-time copy of selected data from a source system's database in a target system's database. Target system applications may use this copy of data changes from the source application for expanded functionality. With application integration, events generated by a source application are sent in real-time to a target application for processing, and is known as an Event-Driven Architecture (EDA).[2] In this case, target application processing may or may not result in a reply back to the source application. As such, application integration using Shadowbase Streams typically involves sending a source application's database changes from one environment to an application in another.

In this white paper, the use of Shadowbase Streams for data integration is described. A description of Shadowbase Streams for application integration can be found in Gravic's companion white paper, *HPE Shadowbase Streams for Application Integration*.

The Data Continuum

From onset to action, there is value in data. The Data Continuum, shown in Figure 1, represents data's typical lifecycle:

1. **Create** – From the Edge, through IoT and other means, data is measured and tracked.
2. **Store** – Once identified, the data is written to a database or other form of storage.
3. **Analyze** – Analytics programs inspect the data, scanning for anomalies and other valuable information.
4. **Learn** – The data is translated into meaningful information stating facts, key insights, and accurate metrics.
5. **Act** – Strategic analysis of the information creates specific and advantageous actions.

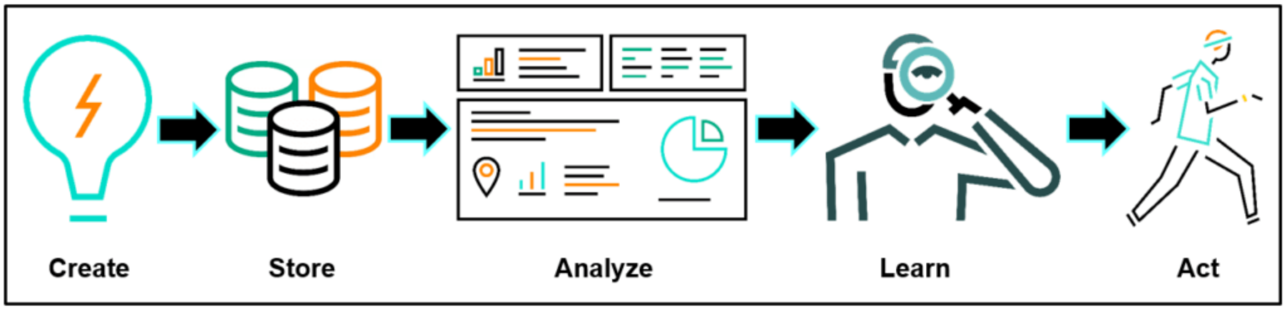

Figure 1 – The Data Continuum

Companies that record and analyze data about their business can analyze their performance, which can help determine the next action to take.

The Need to Integrate Data

Over time, the number of legacy applications developed to support an enterprise's operations grows significantly. Applications maintained databases of information, but the contents of these databases were typically not exposed for use by other applications that were independently developed.

[1] Real-Time Enterprise is a concept in business systems design focused on ensuring organizational responsiveness.
[2] Event-Driven Architecture is a software architecture pattern promoting the production, detection, consumption of, and reaction to events.

As companies grew more reliant on their IT resources, it became apparent that many new applications could benefit from use of the data stored in the databases of siloed legacy applications. In addition, as companies merged, it became necessary to somehow marry their disparate databases into a common repository of knowledge for the newly merged corporation.

Middleware such as Open Database Connectivity (ODBC) and Java Database Connectivity (JDBC) allow external applications to gain access to databases encapsulated by other applications. These databases could then be queried, or polled, by external applications so that important data could be shared among diverse applications.

But in today's 24x7 online environment, having query access to a remote database is not sufficient. Querying for data is a lengthy and complex process, and applications must react far more quickly to data changes than querying allows.[3]

What is needed is a way for one application to immediately have real-time access to the data updated by another application, just as if that data were stored locally. This access is known as a "push" paradigm, whereby the data is processed when it changes rather than at some random time later on.

The Solution

Data integration solves this challenge. As shown in Figure 2, data integration seamlessly moves selected data in real-time from a source database to a target database where it can be used by a target application. As changes are made to the source database (change data capture), they are immediately replicated to the target database to keep it synchronized with the source database (for this reason, "data integration" is often called "data synchronization"). Data replication is transparent to both the source application and to the target application. Upon delivery, the target application can then make use of this real-time data.

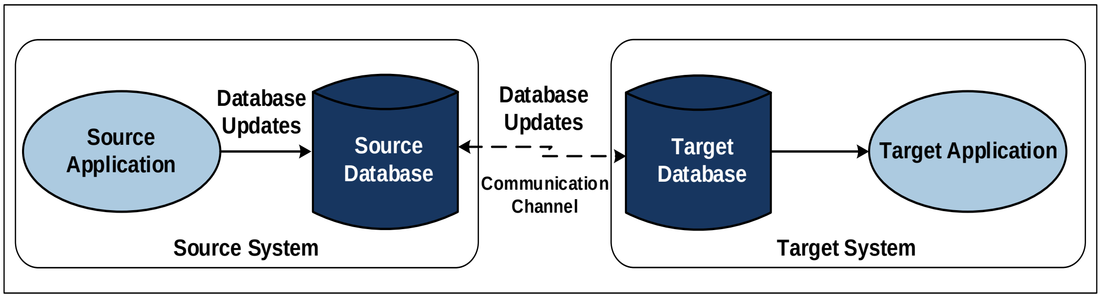

Figure 2 – Data Integration

HPE Shadowbase Streams for Data Integration

Data integration is a role fulfilled by Shadowbase Streams, which uses the transaction log of the source system to obtain database changes to replicate, as shown in Figure 3. In transaction-processing systems, a transaction manager writes every database update to a transaction log. It uses the transaction log to reconstruct the database if it becomes lost or corrupted. Examples of transaction logs are the TMF Audit Trail in HPE NonStop

[3] Querying is a "pull" paradigm. Data may have been updated for some time before it is pulled from the database by the query application.

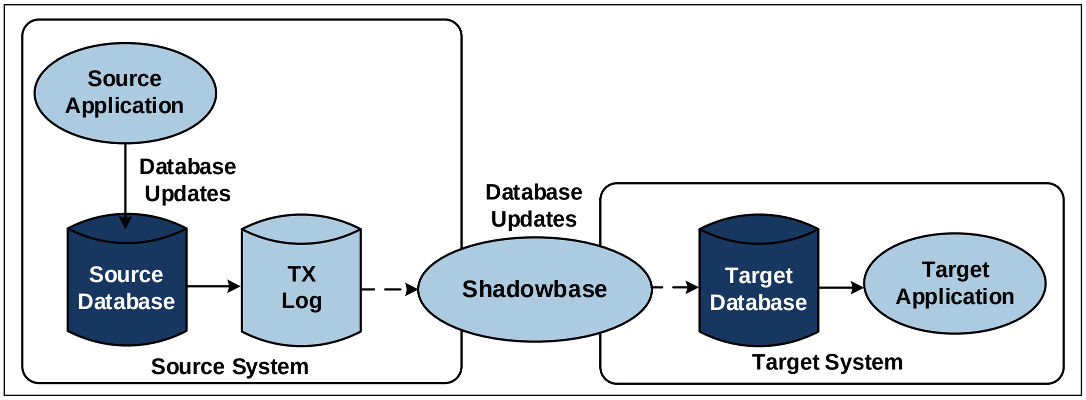

Figure 3 – HPE Shadowbase Streams for Data Integration

systems and the Redo log in Oracle databases. If a transaction log is unavailable, Shadowbase Streams can be driven by database triggers or by an application change log.

Thus, Shadowbase Streams has real-time access to database changes as they occur. As soon as Shadowbase Streams detects a database update, it replicates that update to the target database. In the process, it makes any necessary format changes to the update in order to meet the needs of the target system. In addition, it can filter and eliminate changes that are of no interest to the target system.

Data Mapping

When data is integrated between two diverse databases, the source and target databases can be different in structure and even from different vendors. For example, one can be a relational SQL database and the other a flat-file database. The source and target databases may have different schemas. Columns may be in different orders, and their names and data types may be different. Even the units of measure may be different (e.g., miles versus kilometers). Data may be represented as arrays in one schema but as separate rows or fields in another schema. Some source columns may not be resident in the target database and some target columns may not be resident in the source database.

Consequently, to deal with these differences, Shadowbase Streams provides powerful facilities to reformat, cleanse, and enrich data as it is being replicated to meet the needs of the target database and application. The mapping is driven by the source and target database schemas and is implemented at several levels, as shown in Figure 4:

- Many data types are mapped automatically, such as CHAR, numerics, and many date/time formats.

- *Data mapping scripts* can be defined for mapping more complex data types. Scripts are text files that are read by HPE Shadowbase replication at startup time and are interpreted as each database change is processed. Scripts invoke data conversion capabilities built into Shadowbase Streams.

- *User exits* allow user-generated code modules to be embedded into Shadowbase Streams to implement more complex mapping. Any mapping function may be implemented via user exits, including calls to other subsystems or processing environments. User exits provide the full power of the source and/or target environment through compiled and embedded code for processing the data as it is replicated.

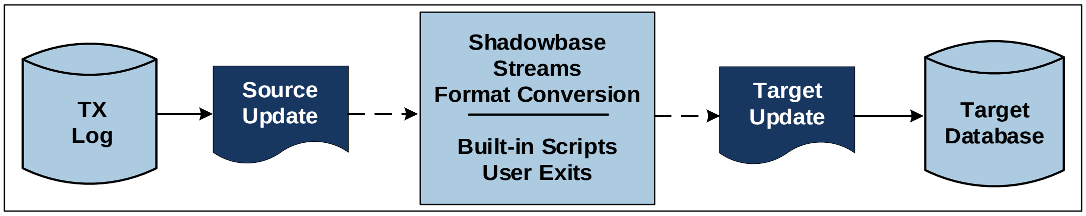

Figure 4 – HPE Shadowbase Streams Data Mapping

Many data conversions can be implemented simply via scripting. The more complex conversions are typically implemented via user exits. Since scripts are interpreted, they take more processing time than does a coded and compiled user exit. Shadowbase Streams can be tested with scripting, and scripts then can be moved to a user exit to improve processing efficiency. In general, the data mapping can be performed on the sending side, the receiving side, or on both sides as user needs dictate. In any case, these sophisticated customization capabilities allow Shadowbase Streams to perform virtually any data mapping function required.

Distributing Data to Multiple Targets

Shadowbase Streams can be configured to replicate changes to multiple databases. For each target database, a separate Shadowbase Streams thread is implemented, as shown in Figure 5. Each target database may be different, running on a different platform, and each thread manages its data replication stream to suit the needs of its target.

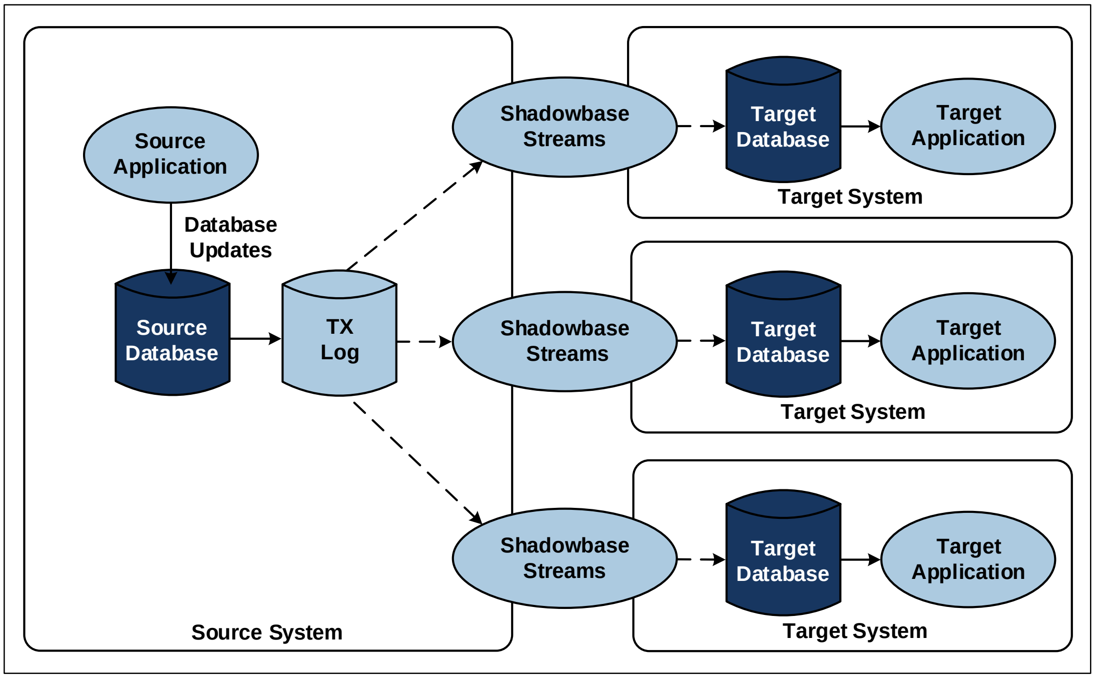

Figure 5 – HPE Shadowbase Streams Fan Out

Consolidating Data from Multiple Sources

Likewise, Shadowbase Streams can consolidate data from several source databases into one target database. As shown in Figure 6, each source database is configured with a Shadowbase Streams data replicator. All replicators feed a single target database, and each instance of Shadowbase Streams provides the conversion necessary between its source database and the target database. As a result, the target database contains changes from all of the source systems.

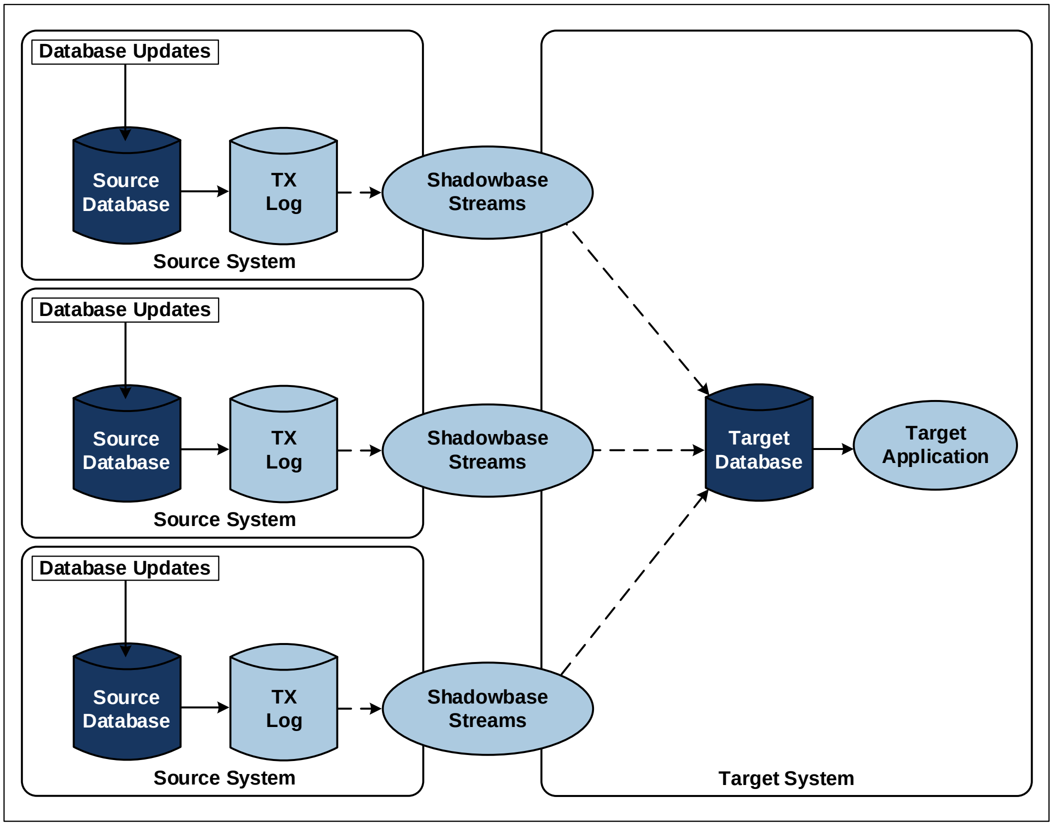

Figure 6 – HPE Shadowbase Streams Consolidation

Data Integration in Practice

Shadowbase Streams for data integration has many uses. In every case, Shadowbase Streams is used to synchronize a target database with a source database by replicating pertinent database changes from the source database to the target database with appropriate data conversion, in real-time.

Extract, Transform, and Load (ETL)

Shadowbase ETL is a member of the HPE Shadowbase product suite that makes use of HPE Shadowbase Streams data replication. It is comprised of a set of utilities and toolkits to extract, transform, and load (ETL) data from a variety of databases, transforming that data to fit operational needs, and loading the data efficiently into a target database. The Shadowbase ETL capabilities can be used in total, or separately, depending on the needs of the project. For example, other commercial ETL products can be used to do the extract to feed the data into Shadowbase ETL for subsequent delivery; alternatively, Shadowbase ETL can produce the flat files of data to be loaded into a target database by other commercial ETL utilities. Many powerful ETL utilities are being offered today.

More specifically, Shadowbase Streams can be used to feed an ETL utility with source database data that is properly formatted for the ETL utility to load into the target database. Shadowbase Streams is configured on each source system, and its powerful data mapping capabilities are tailored to convert the formats of all source databases to the format of the consolidated database that is being loaded. This configuration eliminates the complex task of implementing the “transform” capabilities of the ETL tool. The ETL tool is effectively being given the data to load already cleansed and in the proper format.

As an alternative to a traditional ETL facility, Gravic provides its HPE Shadowbase SOLV utility that can load a copy of a source database into a target database even as the source database is online and being updated. In this case, Shadowbase SOLV reads the source data to be loaded, Shadowbase Streams reads the application changes to that data as the load progresses, and the HPE Shadowbase replication engine merges the two to properly sequence the target information. Shadowbase SOLV also provides the necessary data transformation capabilities to move data between heterogeneous environments.

Of course, once the consolidated database has been loaded, it is usually necessary to keep that data up-to-date. An ETL utility often is employed to perform this function. Periodically, perhaps weekly, it is rerun to move source database changes to the consolidated database. A problem with this approach is that the data in the consolidated database rapidly becomes stale.

Shadowbase Streams can solve this problem. It can move database changes from the source databases to the consolidated database in real-time as they occur. The consolidated database is therefore always current, and the applications it drives never see stale data. Shadowbase Streams also can accept data generated by other ETL loaders and can inject that data into the replication engine for subsequent delivery to downstream applications or to target database environments.

Data Warehouses and Data Marts

A data warehouse is a database of all pertinent information relating to an enterprise. It is a repository that integrates data from disparate systems throughout the enterprise. This data is scrubbed, cleansed, and transformed into a common format to support the data warehouse's needs. The data is organized to support complex queries. A data warehouse stores current as well as historical data useful for reporting, querying, and data analysis. Its contents can be used for a variety of purposes, such as historical quarterly and annual reports, trending reports, and predictive analyses.

A data mart is a small version of a data warehouse and serves an individual business unit, or some other subject-specific purpose, rather than the entire enterprise. Being smaller, data marts are less expensive to set up and maintain and may be directly managed by a business unit rather than by the corporate IT department. The data in a data warehouse or data mart is initially loaded via an ETL facility. However, this data must be kept up-to-date. A common technique is to continue to use the ETL facility to upload new data periodically.

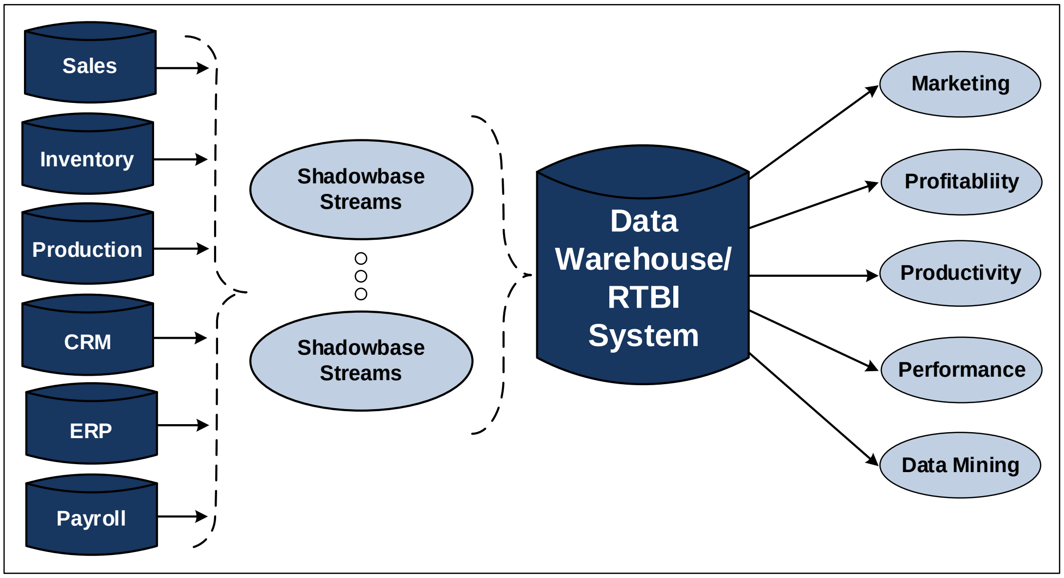

Figure 7 – Updating a Data Warehouse or RTBI System in Real-time

The upload might, for instance, be undertaken weekly. A problem with this technique, as mentioned above, is that data in the data warehouse quickly becomes stale. This limits the functionality that can be derived from the stored data, and is usually not acceptable in today's "always on" business marketplace.

Shadowbase Streams addresses this issue. As shown in Figure 7, it is configured on each source system and feeds data changes as they occur to the data warehouse or to the data mart. Its powerful data-conversion capabilities allow it to reformat the diverse formats of the source systems' databases to that required by the data warehouse or data mart. Thus, the data in the repository is kept up-to-date in real-time and can be used to respond immediately to events as they occur.

Real-Time Business Intelligence (RTBI)

Real-time business intelligence (RTBI) consolidates the database activity of an enterprise's diverse systems to deliver information about business operations in real-time. It provides the information necessary to take tactical advantage of events as they occur as well as to instantly analyze trends to strategically move the company into a direction that will improve its operations. A store, for instance, needs to know that men who purchase diapers on Saturday also tend to buy beer at the same time (strategic – put the beer near the diapers, and have beer sales on Saturday). A credit card company needs to know that a credit card being used to purchase an item in New York City was used thirty minutes ago in Amsterdam (tactical – deny the transaction, and put a credit hold on the card).

In effect, RTBI is the movement of classical data warehousing into real-time. Business transactions are fed in real-time to a business intelligence system that maintains the current state of the enterprise. This maintenance allows the enterprise to drive actions that immediately respond to events. Management decisions are based on the most current data to improve customer relations, to increase revenue, and to maximize operational efficiencies. In totality, the result is the creation of an RTE.

The data in the RTBI system is organized to meet these objectives. It is structured to provide support for online analytical processing (OLAP) so that rapid responses can be obtained to complex queries. It can generate events via database triggers to signal events of interest as they are occurring and is an example of an EDA. For instance, a customer making a purchase of the company's products at a store can be given an immediate offer for discounts on related products[4].

An RTBI system is much like a data warehouse that is fed with data changes in real-time. Shadowbase Streams can be used to feed a real-time business intelligence system just as it does a data warehouse, as shown in Figure 7.

Application Interoperability

Data integration can be used to advantage to allow applications to interoperate. In many cases, an application (the *target* application) may depend upon the data generated by another application (the *source* application). Rather than the target application having to read the source application's database remotely or to poll the source application for that data, Shadowbase Streams can replicate the data of interest from the source application's database to a database local to the target application in real-time.

In this way, a target application will have access to foreign data that has been moved to the target application's local database. Access is rapid and improves the performance of the application. Since the data is available from the target application's local database, in the correct format, little if any application code modification is necessary; it's simply a matter of reading the new data using the same mechanisms already employed. Furthermore, if the application that is the source of this data fails, then the target application will still have access to the latest copy of that data.

For instance, a sales application may need the order history for a subset of customers that have recently ordered certain products. The order history is kept in an order database by the company's order-processing system. The order history for the subset of customers of interest can be replicated to the sales application database from the corporate order database. There, the sales application has local and rapid access to the customer information it needs, even if the corporate order system becomes unavailable. The configuration of Shadowbase Streams for application interoperability is similar to that shown in Figure 2.

It should also be noted that Shadowbase Streams enables application interoperability via direct application-to-application event-distribution mechanisms. A description of this can be found in Gravic's companion white paper, *HPE Shadowbase Streams for Application Integration*.

Offloading Reporting and Querying

Database querying and reporting can be processing-intensive functions. They require significant processor and disk resources. Sometimes, their volumes can grow so large as to seriously affect the performance of a production system.

[4]For more information on RTBI, please read our white paper: *The Evolution of Real-Time Business Intelligence and How to Achieve It Using HPE Shadowbase Software*.

In this case, Shadowbase Streams can be used to move the query and reporting load off the production system. The production system database is replicated by Shadowbase Streams to a separate system that can handle these functions. Reports and queries are then satisfied by the separate reporting system. This approach has the added benefit of eliminating the typically disk-intensive and lengthy querying of the database from the operational (and often quick-response OLTP-driven) nature of the production system. The configuration of Shadowbase Streams for separate reporting and querying is similar to that shown in Figure 2 and Figure 3.

HPE Shadowbase Streams for Data Integration Architecture

Shadowbase Streams for data integration uses the HPE Shadowbase data replication engine for change data capture and to implement its functions. The engine transmits data changes from a source application's database to a target database. As shown in Figure 8, the engine is driven by the transaction log of the source database. The transaction log is created by the transaction manager on the source system. For instance, in HPE NonStop systems, the transaction log is the Transaction Management Facility (TMF) Audit Trail. In Oracle systems, Shadowbase Streams uses Oracle Log Miner to retrieve the events from the Redo log. For Sybase, Shadowbase Streams uses Sybase replication to feed the database changes into the Shadowbase engine. For SQL Server, it can use database triggers to capture the database changes.

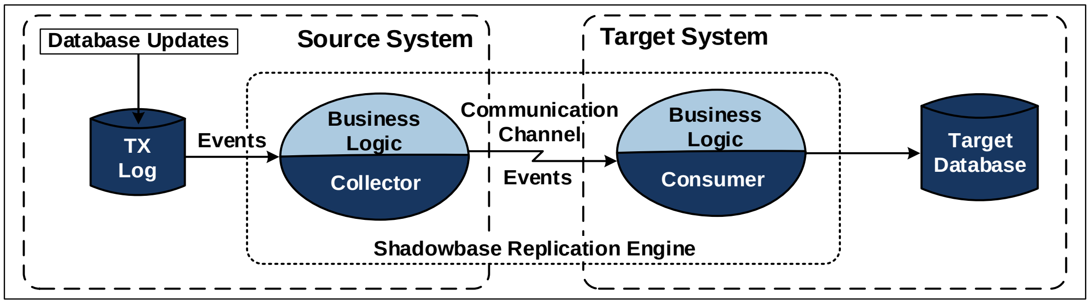

Figure 8 – The HPE Shadowbase Data Replication Engine

The transaction manager records every change made to the source database in its transaction log so that the database can be reconstructed if it becomes corrupted for any reason. The log is therefore an instant source of database changes generated by source applications. By following the transaction log, Shadowbase replication can read every database change in real-time as it is applied to the source database. It can select those changes that represent events of interest, format them according to the requirements of the target application, and pass them to the target application. If there is no transaction log maintained by the source database environment, Shadowbase replication can either receive database changes from triggers added to the source database, or it can follow a change log created by the source application.

Shadowbase architecture comprises a Collector process on the source system and a Consumer process on the target system. The Collector follows the source system's database changes, selecting those that represent events of interest to the target system. It sends these changes to the Consumer via a communication channel. The Consumer formats the changes and applies them to the target database. In most configurations, Shadowbase technology will directly read, process, and deliver the data changes from the source to the target environment; Shadowbase software can optionally be configured to queue these changes to disk during replication should that capability be desired.

HPE Shadowbase replication performs whatever data transformation is required to change the database changes as read from the transaction log into a usable format for the target database via business logic in the Collector and in the Consumer. This business logic is provided by the data-conversion scripts and user exits described earlier. Source database changes can be filtered and only those of interest replicated to the target. Fields can be remapped from a source schema to a target schema. Data values can be reformatted. Fields can be eliminated, combined, or aggregated. Related data can be added from a local database. In short, any enrichment of the data required by the target system can be performed by Shadowbase technology.

The HPE Shadowbase data replication engine is heterogeneous. It supports a wide variety of source and target systems and databases, as shown in Figure 9. (Always check with Gravic for the most up to date listing of supported platforms, databases, and environments.) Shadowbase technology can apply data from any supported source system to any supported target system.

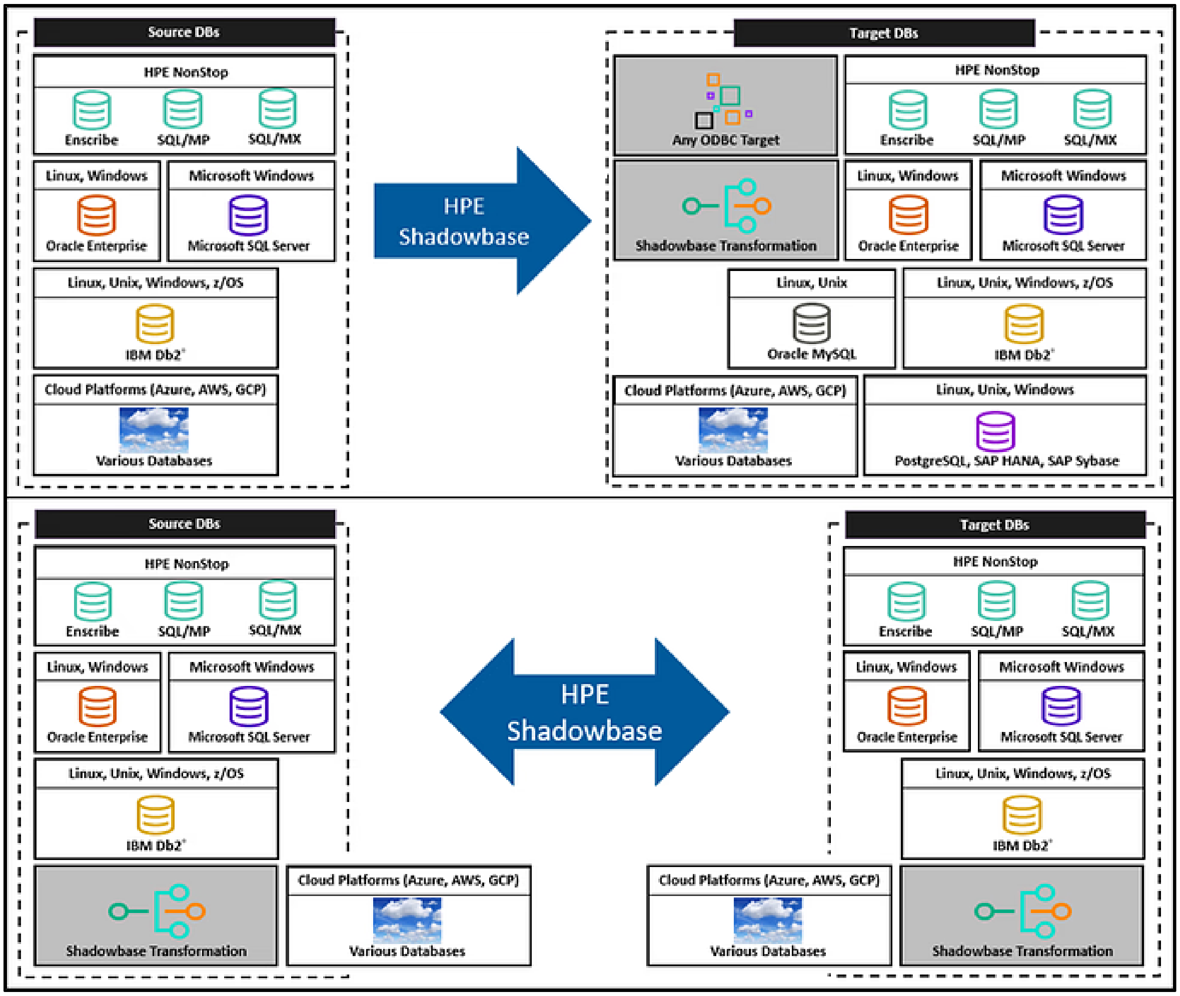
Source DBs
HPE NonStop
Enscribe
SQL/MP
SQL/MX
Linux, Windows
Oracle Enterprise
Microsoft Windows
Microsoft SQL Server
Linux, Unix, Windows, z/OS
IBM Db2
Cloud Platforms (Azure, AWS, GCP)
Various Databases
HPE Shadowbase
Target DBs
Any ODBC Target
HPE NonStop
Enscribe
SQL/MP
SQL/MX
Shadowbase Transformation
Linux, Windows
Oracle Enterprise
Microsoft Windows
Microsoft SQL Server
Linux, Unix
Oracle MySQL
Linux, Unix, Windows, z/OS
IBM Db2
Cloud Platforms (Azure, AWS, GCP)
Various Databases
Linux, Unix, Windows
PostgreSQL, SAP HANA, SAP Sybase
Source DBs
HPE NonStop
Enscribe
SQL/MP
SQL/MX
Linux, Windows
Oracle Enterprise
Microsoft Windows
Microsoft SQL Server
Linux, Unix, Windows, z/OS
IBM Db2
Shadowbase Transformation
Cloud Platforms (Azure, AWS, GCP)
Various Databases
HPE Shadowbase
Target DBs
HPE NonStop
Enscribe
SQL/MP
SQL/MX
Linux, Windows
Oracle Enterprise
Microsoft Windows
Microsoft SQL Server
Linux, Unix, Windows, z/OS
IBM Db2
Cloud Platforms (Azure, AWS, GCP)
Various Databases
Shadowbase Transformation

Figure 9 – HPE Shadowbase Supported Platforms, Databases, and Operating Environments

The HPE Shadowbase replication engine used by Shadowbase Streams supports both asynchronous and synchronous replication. With asynchronous replication, change data is sent to the target system after the changes have been made on the source system. In rare circumstances, it is possible for data to be lost in the event of a failure. For some applications this is not a problem, but for others, the data is critical and must not be lost; it must be presented to the target system for action. Shadowbase Zero Data Loss (ZDL), a future technology, uses synchronous replication to solve this problem. No data is changed on the source system unless the data has been safe-stored on the target system, ensuring no data loss, no matter what the failure. Asynchronous replication also allows for the possibility of data-collisions,[5] which may be unacceptable for some applications. Shadowbase synchronous replication also solves this problem (with Shadowbase ZDL+), preventing the data collision from occurring in the first place. Therefore, Shadowbase Streams with synchronous replication is the solution for the most mission-critical applications, where data loss and/or data collisions cannot be tolerated.[6]

When being used to deliver data from a source database to multiple target databases, the HPE Shadowbase engine can be configured as a separate instance on the source system for each target system, as shown in Figure 5. This configuration allows complete autonomy in how the source to target environments are managed. (For example, if one target environment goes offline for a while, it will have no impact on the other target environments.) Alternatively, a single Collector on the source system can drive separate Consumers on multiple target systems. Using this configuration, only one Collector is reading the transaction log instead of multiple Collectors. Whereas this configuration is the most efficient from a reading perspective, large amounts of data will need to queue if one of the target environments goes offline for an extended period.

A key feature of HPE Shadowbase software solutions is that unlike other data replication technologies, there are optionally no disk queuing points in the transfer path. Therefore, Shadowbase technology imposes a minimum time delay on the delivery of data to the target database. Latency times are measured in sub-seconds. Hence, the data in the target database is as current as possible.

Data Transformation

Besides providing the distribution network to deliver change data in real-time to other databases and applications, HPE Shadowbase solutions include powerful capabilities to transform that change data into whatever format is required by the target database/application; the data may be aggregated, disaggregated, filtered, and transformed. The following methods are available:

- **SBMAP** – a scripting “language” that can be used to inform Shadowbase software how to transform source data into target data formats. SBMAP is powerful, sophisticated, and extensible.
- **SBDDLUTL** – a utility that reads an HPE Enscribe DDL record definition and produces a “flattened” (normalized) DDL structure along with an SQL CREATE TABLE statement for the selected target SQL environment. This capability simplifies the replication of unstructured Enscribe data into structured SQL databases. SBDDLUTL includes features to allow manipulation of the source fields when creating the target columns, including dropping and/or renaming fields, transforming field data, and normalizing the non-normalized Enscribe data.
- **User Exits** – enable the inclusion and execution of customized user logic (program code) at various points in the Shadowbase replication stream and provide capabilities that are more complex than the scripting language. User Exits are extremely flexible, enabling almost any kind of data transformation, and can also perform specific field/column-level encryption and data tokenization.
- **DBS Mapping** – a scripting “language” for target-side Other Server platforms, including these capabilities: drop all events for a target table; drop columns and/or certain events for a target table; convert updates to inserts; concatenate (text) columns; and reformat and convert/replace characters.

There are a number of parameter settings that can be set in Shadowbase software to assist with data transformation, (i.e., to convert non-printable binary data to spaces in a character field). However, these settings work on the data record at an aggregate level, and not at a field level. SBMAP and the other methods

[5]In an active/active architecture, a data collision is where the same data field is updated simultaneously on two (or more) copies of a database. When replication of that update between those copies occurs, each update will overwrite the other, resulting in an inconsistent database.

[6] For further information on Shadowbase ZDL and ZDL+, visit https://www.shadowbasesoftware.com/solutions/business-continuity/zero-data-loss/

described above allow for much finer detail of data transformation.

Case Studies

The following case studies of customer production deployments demonstrate how Shadowbase Streams for data integration may be used to integrate diverse applications. Shadowbase Streams is well-suited to this task. It is heterogeneous, able to replicate data between a variety of systems and databases.

A Large Internet Service Provider Distributes Profile Changes in Real-Time

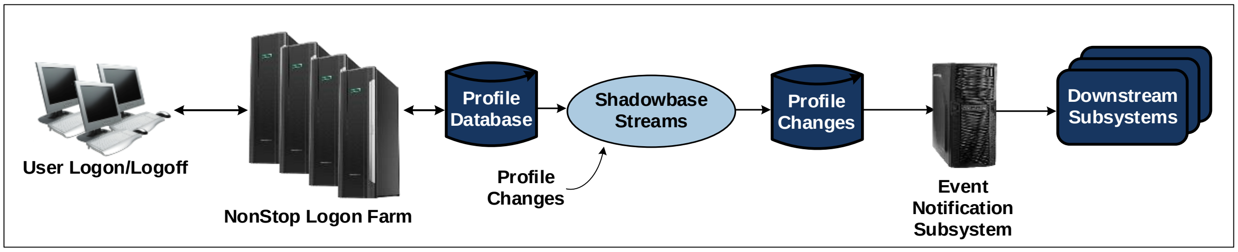

Figure 10 – HPE Shadowbase Streams Distributes Profile Changes in Real-Time

A major internet service provider (ISP) maintains subscriber information in a large HPE NonStop SQL/MP Profile Database managed by its farm of logon servers. The database contains over a billion records. It is so large because there can be multiple profile records per user. For instance, a family might have a separate record for each family member. Furthermore, each person may also have distinct profile records when logged on via their church, youth group, or other organization.

Whenever a user logs on, all of the other subsystems that a user might use in his session are notified. Some subsystems are sent the user's profile information immediately as part of the notification, while other subsystems request it as needed.

A major problem that the ISP faced was that users modified their profiles from time to time. Unless the various subsystems periodically polled the master database, they were not aware of these changes until the users logged off and logged on again. The result was that when users updated their profiles, the changes did not appear to have taken place and users could not see the changes until their next session.

Another issue was that several subsystems obtained user profiles by polling, which is a high-overhead activity and places a significant load on a system. The ISP could have modified all of its applications to broadcast profile changes as they occurred, thereby eliminating polling, but it would have required significant application modifications (and of course testing). The ISP instead decided to replace the need to poll with a profile-change database, which is significantly more efficient.

The ISP chose to install Shadowbase Streams to monitor changes made to the HPE NonStop SQL/MP Profile database and to send the changes as they occurred to a Profile change database. This database is used by an ISP-provided Event Notification Subsystem (Figure 10), which broadcasts the changes in real-time to the subsystems that previously had received the profile information only upon logon, and this system is a great example of an EDA. Other subsystems now receive changes via a publish/subscribe facility, thus eliminating the need for high-overhead polling. The addition of this efficient facility required no application changes.

An additional benefit is that the Profile change database is, in effect, a log of all profile changes. This log allows the ISP to track down changes that were made inappropriately and to selectively "undo" any that it wants to reverse.

Integrating HPE NonStop Trading Applications with Online Trading Applications

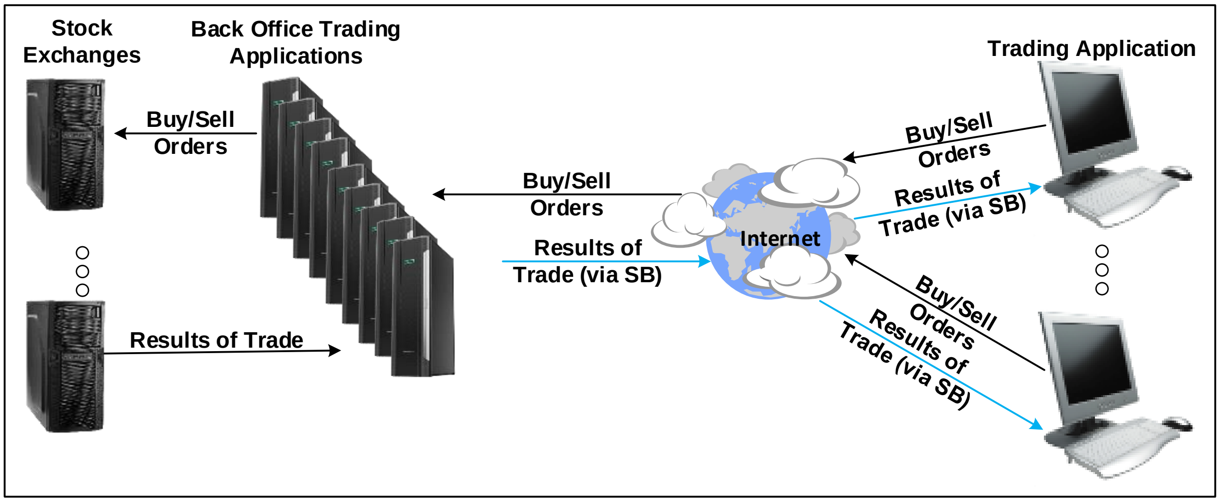

Figure 11 – HPE Shadowbase Streams Integrates HPE NonStop Trading Application with Online Trading Applications

A large service bureau operates a stock trading system. This system comprises a front-end online trading application which customers use to view their stock positions and to input buy/sell orders, and a back-office brokerage application running on an HPE NonStop server which sends the various orders to the appropriate exchange for execution, and which holds the current stock positions for all clients (the database of record).

A user enters a buy/sell order via the online trading application, which routes the request to the back-office brokerage application as shown in Figure 11. Typically, the brokerage application maintains one account with the exchanges from which all trades are done, so the brokerage application aggregates the new order with others and sends them to the exchanges for execution. When the results of the trades are received by the brokerage application from the exchanges, the client's stock position information is updated in the back-office database of record.

HPE Shadowbase Streams is used to capture and replicate stock position changes in real-time from the back-office database to a read-only database used by the online trading system, thereby integrating the brokerage and online trading applications, and enabling clients to display accurate, current, information about their stock portfolios.

Integrating HPE NonStop Trading Applications with Brokerage Data Warehouses in a Service Bureau Environment

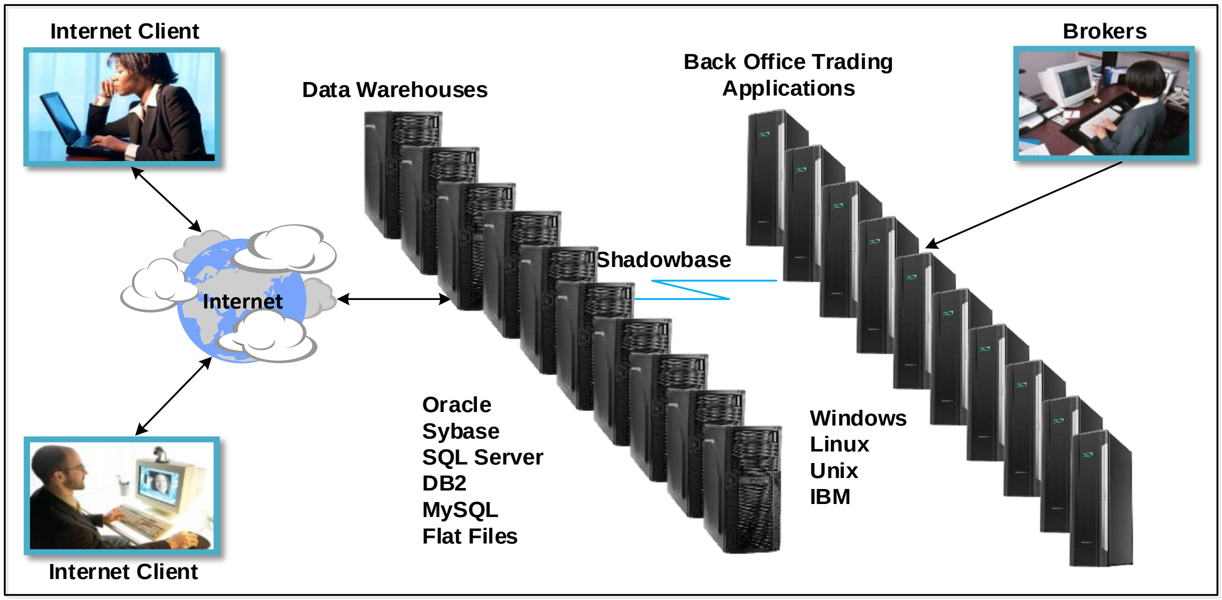

Figure 12 – HPE Shadowbase Streams Integrates HPE NonStop Trading Applications with Brokerage Data Warehouses in a Service Bureau Environment

The back-office trading application discussed in the previous example is also used by many other clients (banks and brokerages), in a service bureau arrangement. These financial institutions have customers of their own, so they also have a need to be able to provide stock trading information to their clients.

In this case, the banks/brokerages have their own in-house data warehouse systems running on many different types of platforms and databases (Windows, Linux, Solaris, HPE-UX, AIX, SQL Server, Sybase, Oracle, DB2, MySQL, etc.) as shown in Figure 12.

Because Shadowbase Streams supports heterogeneous configurations, it is able to replicate the changes made to the trading system database of record running on an HPE NonStop server, to these various data warehouses. Thus, via Shadowbase Streams, the bank or brokerage data warehouse is kept current with its customers actual trading positions, enabling its own online applications to provide the customer's current position when that customer logs on to its internet application.

Each bank/brokerage requires different data to be fed from the back-office trading database, and for the data to be delivered to its data warehouse in a different format. Shadowbase Streams enables each to specify the data it wants to receive (the actual tables and columns that each receives can differ), and the format in which it is to be presented.

Using a data warehouse approach enables the banks/brokerages to store the current state of all their customer positions, as well as maintain a history/archive of all the trades each customer has made over some period of time (possibly decades), whereas keeping this amount of data stored on the back-office database of record would be cost-prohibitive. This case study demonstrates a common usage of Shadowbase Streams for data integration, to feed a data warehouse with real-time changes from an online database.

Integrating HPE NonStop Telco Applications with a Data Warehouse via ETL Technology

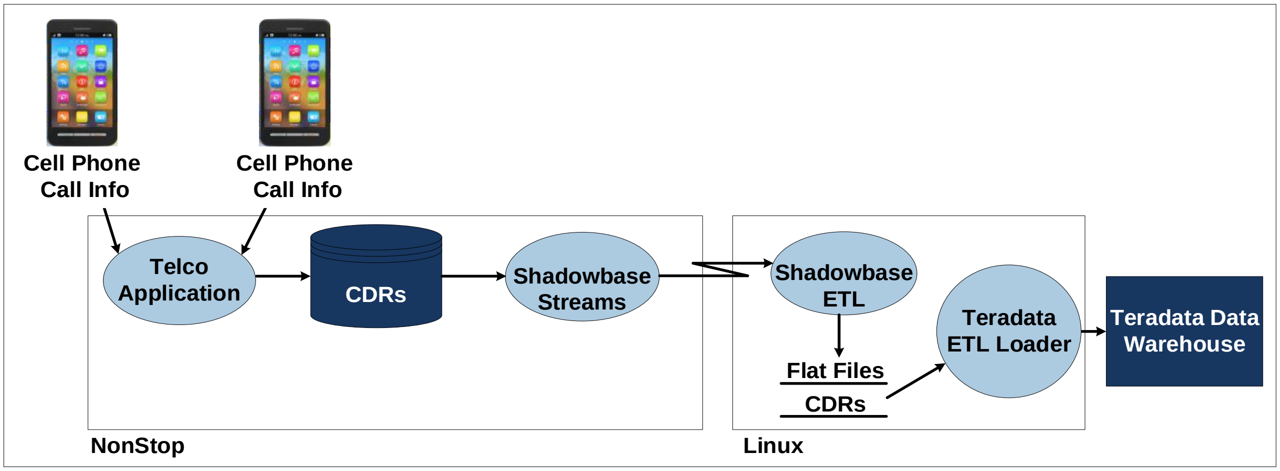

Figure 13 – HPE Shadowbase Streams Integrates HPE NonStop Telco Applications with a Data Warehouse via ETL Technology

A large telecommunications provider has a need to maintain an historical record of billing and call-tracking information for its cellular customers (who called who, when, and to identify calling patterns). The primary source of this information is the call data record (CDR), one of which is created for every phone call. In this case, the real-time CDR information is created by an application running on an HPE NonStop server system. It is expensive and inefficient to maintain the required volume of historical information on the online system, so the company employs a massive Teradata data warehouse for this purpose, feeding it the required CDR information using the Shadowbase ETL Toolkit product (an extension to the Shadowbase Streams technology).

The Shadowbase ETL Toolkit is a set of user exit functions built on the HPE Shadowbase data replication engine to extend the default Shadowbase Streams functionality as shown in Figure 13. Users configure the product specifically for the ETL loader and data warehouse system being used. In this case, Shadowbase Streams follows the audit trail on the HPE NonStop Server system in real-time, picks up the CDR records as they are created, and formats them into ETL-style flat-files (comma delimited for the fields/columns). Approximately every 100,000 CDRs (or 5 minutes, whichever comes first), the HPE Shadowbase engine closes the current flat-file and rolls to a new one and then renames the closed file into a certain format that the Teradata system will recognize. The Teradata ETL loader 'sees' the files appear with the right name, and then loads them into the Teradata data warehouse. This process is then repeated continuously as CDRs are created and saved into the flat-files (as many as 15,000 CDRs per second may be created when the calling circuits are busy). Shadowbase architecture is configured to run with many threads in order to keep up with the creation of the flat files at this CDR rate.

The telecommunications provider is thereby able to retain a massive amount of historical call record information in its data warehouse, while keeping the data current with new CDR data using the Shadowbase ETL Toolkit.

Service Bureau Provides Customer Access to Securities Account Information (plus co-existence with RDF)

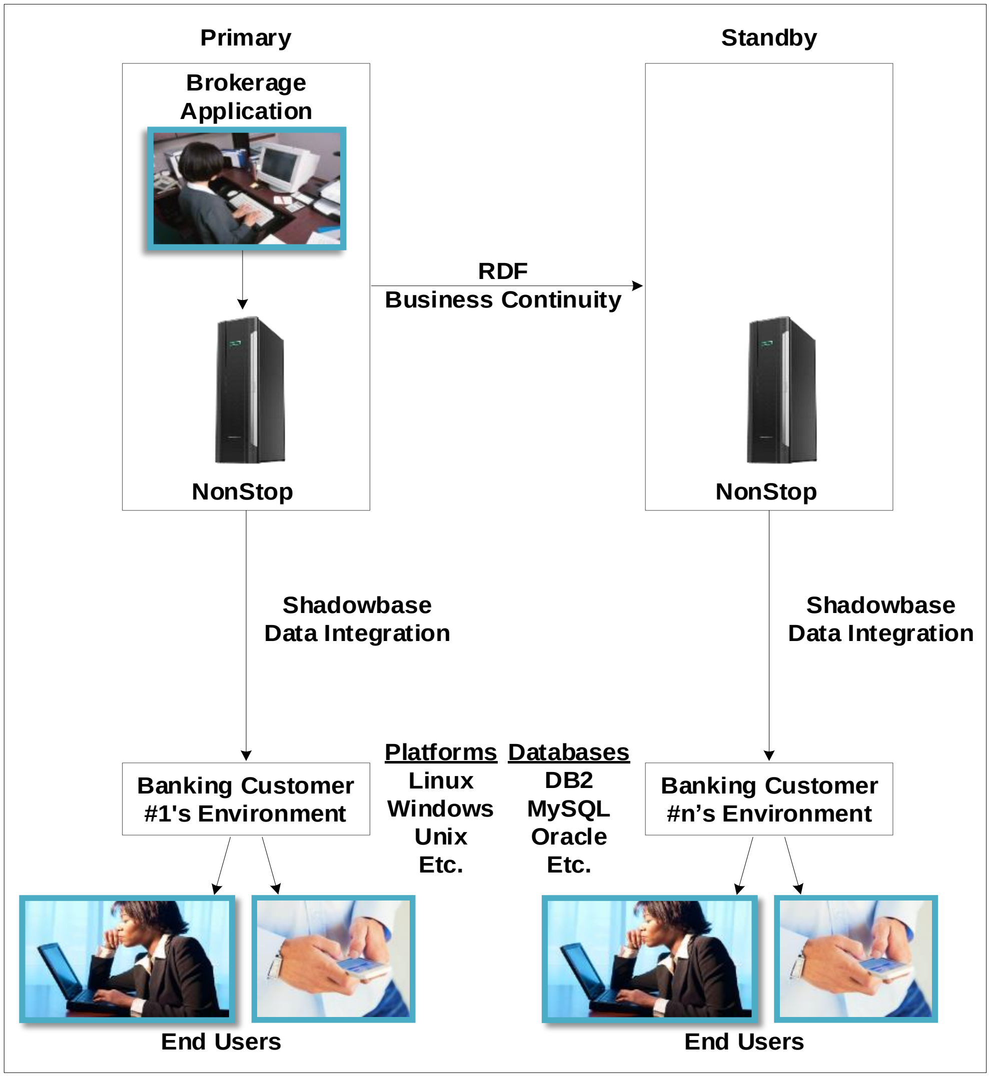

Figure 14 – HPE Shadowbase in a Multiple Replication Engine Environment

A brokerage service bureau provides outsource services for security account balance and trade history to its customers' end users on a pair of HPE NonStop servers, which are connected via RDF for business continuity purposes. The bureau needs to provide 24x7 access to the customers' end users for a variety of devices. It also needs to provide an ad-hoc query and reporting capability to its customers' end users, preferably on *customer-managed* platforms and databases.

These requirements came with some specific restrictions:

- For security reasons, the bureau did not want to allow direct end user access into its NonStop-based database of record
- For performance reasons, the bureau did not want to allow end user ad-hoc reporting on the database of record
- For maintenance reasons, the bureau wanted to avoid managing the customers' target platforms, databases, and environments
- The solution must be able to coexist with the bureau's existing RDF environment

To meet these requirements and overcome the restrictions, the bureau chose HPE Shadowbase data integration to feed heterogeneous databases on separate platforms, and coexist with other replication engines (in this case, RDF) as shown in Figure 14. This solution allows the bureau to provide each customer with the required HPE NonStop source data, while allowing the customer to use privately managed target platforms

and databases of its choice. HPE Shadowbase software selectively replicates the HPE NonStop database changes to each customer's target database in real-time – the existing bureau NonStop environment is not disturbed at all.
Both the bureau and its customers benefited from the choice of HPE Shadowbase data integration for the solution:

- Enabled customers to provide access and value-add processing to their end users in their target environment
- Improved the end user experience with a modern interactive interface and ad-hoc query support
- Allowed the bureau to select either RDF Primary or Standby as the source of the customers' data changes based on loading

Summary

HPE Shadowbase Streams is a member of the HPE Shadowbase suite of products (built by Gravic, sold by HPE). Shadowbase Streams for data integration and Shadowbase Streams for application integration provide the facilities for integrating existing applications, services, or database environments in order to create new functionalities for the enterprise, typically without the need to modify existing application code.

Using Shadowbase Streams, applications that once were isolated now can interoperate in real-time to provide services that were not possible before integration. Critical data generated by one application is distributed immediately to other applications or database environments.

Shadowbase Streams supports many models of data distribution, including maintaining a remote database of critical data, sending critical events to client applications directly via queues or client APIs, publishing data to applications that have subscribed to the data, or responding to poll queries from client applications.

HPE Shadowbase Streams modernizes legacy applications by integrating diverse applications across the enterprise so that new and valuable services that were not possible in the past can be created to enhance competitiveness, to reduce costs or increase revenue, to satisfy regulatory requirements, and to generally improve the user experience.

International Partner Information

Global

Hewlett Packard Enterprise
6280 America Center Drive
San Jose, CA 95002
USA
Tel: +1.800.607.3567
www.hpe.com

Japan

High Availability Systems Co. Ltd
MS Shibaura Bldg.
4-13-23 Shibaura
Minato-ku, Tokyo 108-0023
Japan
Tel: +81 3 5730 8870
Fax: +81 3 5730 8629
www.ha-sys.co.jp

Gravic, Inc. Contact Information

17 General Warren Blvd.
Malvern, PA 19355-1245
USA
Tel: +1.610.647.6250
Fax: +1.610.647.7958
www.shadowbasesoftware.com
Email Sales: shadowbase@gravic.com
Email Support: sbsupport@gravic.com

Hewlett Packard Enterprise Business Partner Information

Hewlett Packard Enterprise directly sells and supports Shadowbase Solutions under the name ***HPE Shadowbase***. For more information, please contact your local HPE account team or visit our website.