HPE Shadowbase
SOLV and ETL Toolkit Product Suite — Online Database Loading, Copying, Synchronizing, and Unloading

A Gravic, Inc. White Paper

Executive Summary

Before a database is useful, it must first be loaded with data (for example, when installing a new backup system for business continuity, or when performing a system/software upgrade or migration). The Shadowbase Online Loading Software (SOLV) Utility and Shadowbase Extract, Transform and Load (ETL) Toolkit™ are products designed to help perform this task.

Together, the SOLV and the ETL Toolkit support various architectures to load a database:

- SOLV (Online) Loading
  - Merged Loading
  - Block loading
- Bulk (Offline) Loading

SOLV

SOLV can load audited and non-audited HPE NonStop Enscribe files and HPE NonStop SQL tables into any Shadowbase-supported environment and database target combination. It is also compatible with both HPE NonStop Guardian and Open System Services (OSS) environments. The source and/or target databases can be open for reading and updating while the load (or copy) occurs, even while regular replication is occurring.

SOLV can coordinate loads with HPE Shadowbase change data capture replication, organizing and merging the source database changes with the data being loaded, which keeps the target fully synchronized with the source during the load. Keeping both databases synchronized is critical when eliminating planned downtime via zero downtime migration (ZDM).

For non-audited HPE NonStop Enscribe and SQL data: SOLV supports point-in-time *Snap-Shot Loads* (also known as *refreshes*). Using the SOLV solution, the user can periodically load (or refresh) the entirety (or a portion) of the non-audited file or table into the target environment.

Shadowbase ETL Toolkit

The Shadowbase ETL Toolkit is used to integrate a Shadowbase source environment with almost all target environments. It supports HPE Shadowbase Application Integration by providing a suite of adapters to easily integrate streaming data from the source data to the target environment.

Available adapters include:

- Integrating with KAFKA
- Integrating with IBM MQ environments
- JSON output format
- Flat-files (CSV, fixed position, tab-delimited) for subsequent ETL loading
- Custom Shadowbase User Exits to interface Shadowbase environments with almost all third-party APIs

The Shadowbase ETL Toolkit is extremely flexible. Users can directly feed or stream data from a Shadowbase source environment into a Kafka or IBM MQ target environment/API, produce JSON formatted output, or produce flat files, and then use an existing vendor's ingest program for an initial load or to load change data. The Toolkit also supports periodic "refreshes," incremental loading using a micro-batch update approach to keep the databases synchronized. To support additional target environments, Shadowbase software provides its User Exit customization support to allow integrating the source data with almost all target environments or third-party APIs.

When coupled with SOLV, the Toolkit can be used to extract select data from a source database, transform, filter, and cleanse it into the proper format, and then deliver it into any of the target environments listed above. This capability is useful for integrating the source environment with a target database or environment that is not directly supported by Shadowbase replication. The user can load the normalized/flattened data with Shadowbase ETL and SOLV into a new previously inaccessible target database!

Table of Contents

Executive Summary **2**
SOLV *2*
Shadowbase ETL Toolkit *2*
HPE Shadowbase Online Database Copying, Loading, Synchronizing, and Unloading Software (SOLV) **5**
SOLV Online Loading *5*
Key Features of SOLV *5*
Supported Online Database Loading Architectures **6**
Methods for Offline and Online Database Loading **6**
Bulk (Offline) Loading *6*
Database Status: Source – Online; Target – Offline *6*
Transaction Replay is Essential to Avoid Data Corruption *6*
SOLV (Online) Merged Loading **8**
Run SOLV in Parallel with Shadowbase Replication *8*
SOLV and Shadowbase Replication Merge Load and Change Data *8*
Hybrid of the Bulk (Offline) and SOLV Merged Loading Methods **9**
SOLV (Online) Block Loading *9*
HPE Shadowbase Online Loading and Verification (SOLV) for Offline Loading **9**
HPE Shadowbase SOLV for HPE NonStop Open System Services (OSS) **10**
End of File (EOF) Chasing *10*
HPE Shadowbase ETL Toolkit **11**
Available Adapters Include *11*
Key Features of the ETL Toolkit *11*
The HPE Shadowbase ETL Toolkit and SOLV can be Combined *11*
SOLV "Snapshot" Loads *12*
Key Considerations of SOLV *12*
SOLV's Data Consistency Options *12*
SOLV Leverages the Shadowbase Data Replication Engine in Certain Situations **12**
Data Replication During the Load *12*
Target Database is Current with Source's Database Changes During the Load *12*
Write Loading Routines or Mappings Only Once *12*
Typical SOLV Unloading Scenarios **13**
Flattening a Database *13*
Load During Replication *13*
HPE Shadowbase SOLV Manager (SOLVMGR) *14*
Summary **15**
International Partner Information **16**
Gravic, Inc. Contact Information **16**

Table of Figures

Figure 1 – SOLV Online Loading Integrated with HPE Shadowbase Change Data Replication 5
Figure 2 – Bulk (Offline) Loading Method 6
Figure 3 – SOLV (Online) Merged Loading Method 8
Figure 4 – SOLV (Online) Block Loading Method 9
Figure 5 – HPE Shadowbase Online Loader (SOLV) Offline Loading 10
Figure 6 – HPE Shadowbase SOLVUTIL for OSS Regular Files 11
Figure 7 – SOLV Unloading with HPE Shadowbase ETL Output 13
Figure 8 – SOLV Unloading Integrated with HPE Shadowbase Replication and ETL Output 14
Figure 9 – HPE Shadowbase SOLVMGR Injecting Flat File Data into the Replication Engine 15

Table of Tables

Table 1 – Bulk (Offline) Loading Pros and Cons 7
Table 2 – SOLV (Online) Merged Loading Pros and Cons 9
Table 3 – SOLV (Online) Block Loading Pros and Cons 9

HPE Shadowbase Online Database Copying, Loading, Synchronizing, and Unloading Software (SOLV)

The SOLV utility enables applications to access an online source or target database while a load occurs.

SOLV Online Loading

SOLV loading utilizes special patented features that allow it to properly merge the data being loaded with the data being replicated. Note that SOLV loading can also occur without affecting HPE Shadowbase replication or while replication is not actively running.[1] More specifically, SOLV loading can be used with any replication engine, or without a replication engine, since SOLV can coexist with other applications and products that access and update the database. In this case, SOLV acts like a stand-alone data transformation process, reading and converting the source data formats into target data formats, and applying that information into the target environment.

The SOLV utility can load audited and non-audited HPE NonStop Enscribe source files and HPE NonStop SQL tables into any supported environment and database target combination.

Figure 1 illustrates SOLV in an environment where data replication is actively running.

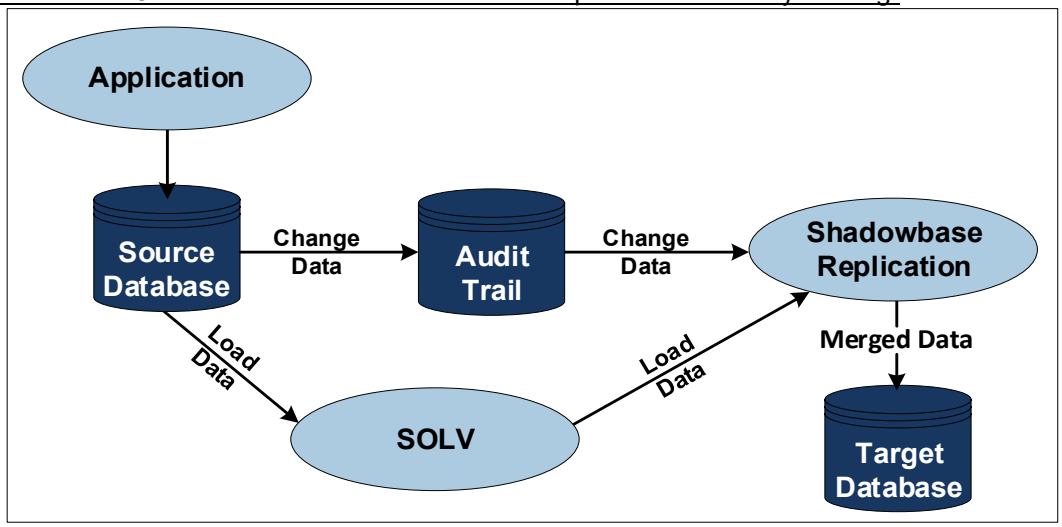


Figure 1 – SOLV Online Loading Integrated with HPE Shadowbase Change Data Replication

Key Features of SOLV

- Perform non-disruptive initial bulk loads and periodic refreshes
  - Source and target databases can be opened for read/write activity during loads
- Extract, transform, filter, and cleanse data before loads
  - Use the same (or different) conversion routines as Shadowbase replication
- Synchronize both databases during and after loads
  - Coordinate loads with Shadowbase change data capture replication
    - Replication operations can continue during loads
  - Perform loads from a source's specific point-in-time
- Leverage SOLV's efficiency
  - Perform loads without needing to drain a queue of events
  - SOLV automatically and dynamically adjusts load processing to minimize system resource utilization
- Completely and efficiently load online source data into a target environment with one operation
- Utilize SOLV's offline loading feature, if desired

[1] If a solution uses an existing replication engine, then care must be taken during the load to either pause the replication engine, or otherwise ensure that the load path stays ahead of the replication path during the load.

Supported Online Database Loading Architectures

- *HPE NonStop to HPE NonStop* – This is common during a Zero Downtime Migration, and for initial disaster recovery system installation and configuration.
- *HPE NonStop to Other Servers* – This is common for initial loading, refresh loading, and for other miscellaneous types of heterogeneous loads.

Methods for Offline and Online Database Loading

Bulk (Offline) Loading

This loading technique uses various disk management utilities. These utilities are typically provided as part of an operating system or database software, such as:

- Backup/Restore,
- Copy/Load,
- FUP/SQL DUP,
- PAK/UNPAK,
- etc.

Database Status: Source – Online; Target – Offline

During the loading process, the source database may be online (even for updating), however, the target database is temporarily *offline and unavailable* for use. After the target database is loaded, it must be made consistent by replaying the source database's transactions that were replicated during the load before it can be used (
Figure 2, step #4).

Transaction Replay is Essential to Avoid Data Corruption

In a Shadowbase data replication environment, the user must replay the new, queued transactions that were applied to the source after the initial target load. Then, the target database can be brought back online.

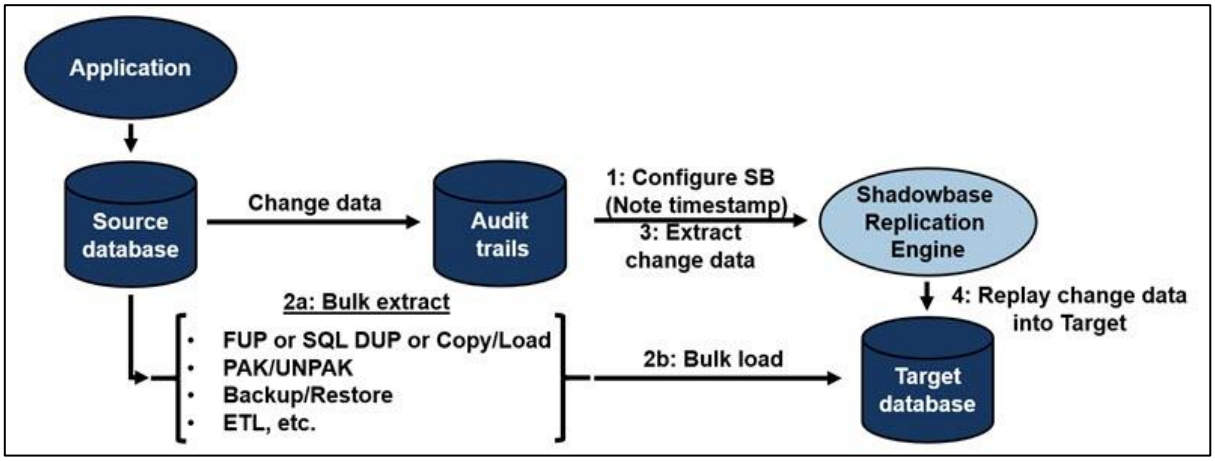


Figure 2 illustrates the bulk offline loading method.

Figure 2 – Bulk (Offline) Loading Method

1. A timestamp needs to be noted at the exact point the bulk extract begins so that Shadowbase replication can start queueing transactions that will later be replayed (step #1).
    a. This timestamp could simply be the wall clock time when the "Bulk extract" is performed
    b. However, a more accurate and reliable method is to use Shadowbase to set the restart point when the "Bulk extract" is started, in three easy steps:
        i. Start the Shadowbase data replication engine and configure it between the source and target systems.

1. Do not start replication yet (*you have been warned*).

ii. Set a Shadowbase *restart point* when the "Bulk extract" starts (step 1).

iii. After the "Bulk extract" and "Bulk load" is started, run, and completed in steps 2a and 2b, Transaction Replay (step 4) can simply be started from this Shadowbase restart point (from step 1).

2. During the time the load is taking place (steps 2a and 2b) the source system is processing transactions, and the database audit trail (for example the TMF audit trail) is filling up with transaction entries (called "change data").
   a. The audit trail disk space is not infinite.
      i. It is possible that while the "Bulk load" (step 2b) is still in process, the audit trail may fill-up and "roll[2]," and transactions may be lost.
         1. (They will be deleted from the audit trail; hence, they will neither be found in the audit trail by Shadowbase software when the replay process begins, nor will they be applied to the target database.)
      ii. To prevent this serious data loss situation from occurring, while the load is proceeding, Shadowbase software can be configured to replicate the source data changes from the audit trail into its own queue files, using the *Shadowbase Queue Manager* from where they will be replayed later, thereby preventing the loss of any data if the audit trail rolls off the system and is scratched by TMF.
3. Once the "Bulk load" has completed (step 2b), the transactions in the change log are replayed by Shadowbase to re-synchronize the source and target database (steps 3 and 4).
4. A best practice for performance purposes is to not create indices nor alternate key files on the target database until *after* the transaction replay process is complete.
   a. Performing lots of change data replication with the indices in place can lead to locking issues on the indices, which slows down replay of the queued transactions.

Table 1 discusses the pros and cons of the bulk (offline) loading method.

Pros	Cons
• Bulk Loading is very fast. • It typically performs block-oriented operations (not record or row-oriented operations).	• The target database is offline and unusable until after the Bulk Load. • The source transactions need to queue and then be replayed into the target after the Bulk Load completes. • The target database is typically inconsistent until the Change Data replay is complete. • Often, indices or alternate keys are not created/loaded on the target until after the Bulk Load completes.
• The source database will remain active for updating during the load.	• The source and target databases must be very similar. • Typically, no (or minimal) data filtering, transformation, or cleansing is allowed. • Often, this method is only available for homogeneous, like-to-like source and target databases.

Table 1 – Bulk (Offline) Loading Pros and Cons

[2] When an audit trail "rolls" it means that the existing audit trail is has reached its data storage capacity, and a new audit trail must be created and used. Under certain conditions, TMF will "scratch" an entire audit trail, and reuse that disk space by creating a new audit trail. If this "scratched" audit trail was not previously extracted and successfully replicated to another location, then that data may be lost forever.

SOLV (Online) Merged Loading

Run SOLV in Parallel with Shadowbase Replication

This technique takes advantage of the benefits of the Shadowbase SOLV utility, a key aspect of which is the ability to run the SOLV loading process in parallel with Shadowbase data replication from the source to the target database, enabling the target database to remain online and removing the transaction replay step after the load completes.

The procedure is straightforward[3]:

1. simply configure and start Shadowbase data replication between the source and target databases,
2. and then commence the SOLV loading process.

SOLV and Shadowbase Replication Merge Load and Change Data

Close integration between SOLV and Shadowbase data replication software handles the merging of load and change (replay) data into the target database, ensuring the target database remains consistent with the source database while the load occurs. Once the load is complete, there are no queued transactions to replay.

Figure 3 provides an overview of this process.

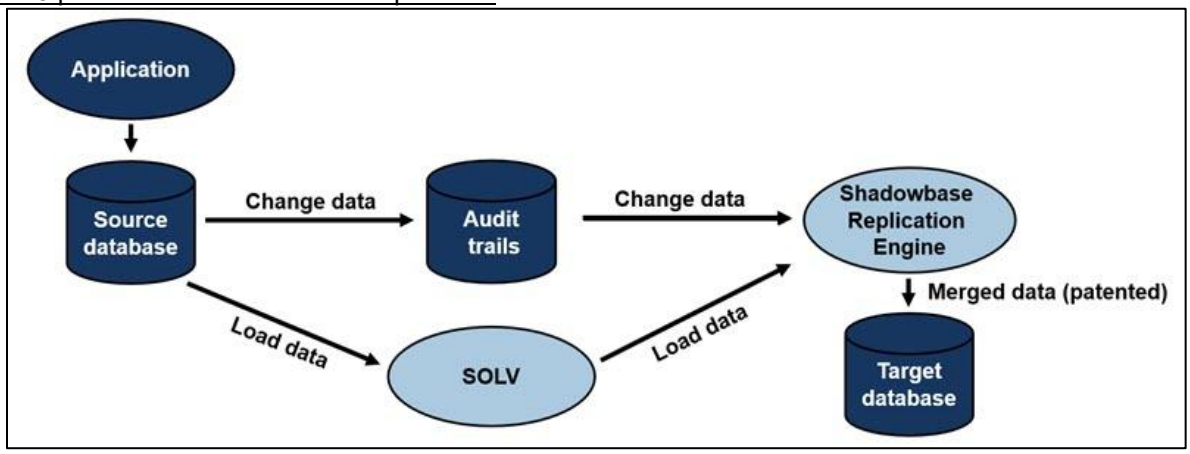


Figure 3 – SOLV (Online) Merged Loading Method

Table 2 discusses the pros and cons of the SOLV (online) merged loading method.

Pros	Cons
• The target database is online and consistent with the source database while the load operation occurs. ○ The SOLV load and transaction replay happen in parallel.	• Slower than bulk (offline) loading as it inserts by record/row instead of block.
• No waiting for replay of queued transactions. ○ Once the load is finished, the target database is immediately caught-up and synchronized.	

• Source and target databases can be very different.
  ○ Data filtering, transformation, and cleansing is possible.
  ○ Useful for homogeneous *and* heterogeneous source and target databases.	

[3] Alternatively, simply start SOLV, note the time that the SOLV load started, and set Shadowbase data replication to begin before that time.

Table 2 – SOLV (Online) Merged Loading Pros and Cons

Hybrid of the Bulk (Offline) and SOLV Merged Loading Methods

SOLV (Online) Block Loading

This technique is similar to SOLV merged loading as described above, except that the transaction replay and merging of the load data into the target database do not take place in parallel. This hybrid approach combines the flexibility of using SOLV (for heterogeneous data transformation, etc.), with the speed of bulk loading. As is the case for bulk loading, Shadowbase data replication can be configured to drain audit trail entries into its queue files to avoid the possibility of lost data as a result of audit trail rolls.

This method is a hybrid of the "Bulk (Offline)" and "SOLV (Online) Merged" loading methods:

1. SOLV (rather than a bulk loading utility) is used to first load the target database by loading a block of records or rows at a time.
2. Simultaneously, the source database changes are queued by Shadowbase software, and after the SOLV load is complete, then replayed into the target database, making the target database consistent.

Figure 4 provides an overview of this process.

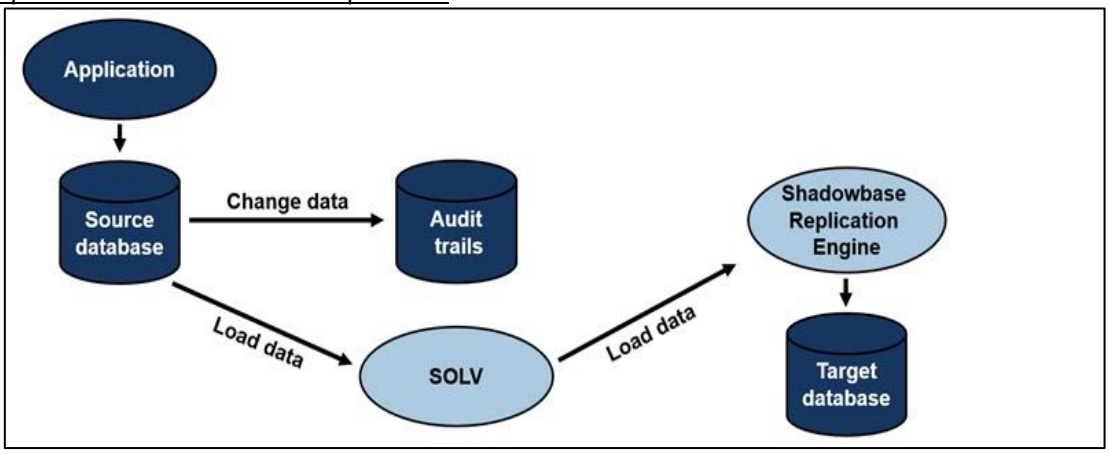


Figure 4 – SOLV (Online) Block Loading Method

Table 3 discusses the pros and cons of the SOLV (online) block loading method.

Pros	Cons

• Target database is online the entire time.
• Source and target databases can be similar formats or very different.
  ○ Data cleansing, filtering, and transformation is possible.
  ○ Useful for homogeneous and heterogeneous source and target databases.
	• Slower than bulk (offline) loading, but faster than SOLV (online) merged loading.
	  ○ Target is inconsistent until the replay of queued transactions completes (which has to be manually replayed after the load).

Table 3 – SOLV (Online) Block Loading Pros and Cons

HPE Shadowbase Online Loading and Verification (SOLV) for Offline Loading

SOLV also has a function for offline loading. In this context, offline means the source database is not being actively changed by the application during the load; if it is, then the change data are either being queued or ignored. Figure 5 depicts a typical SOLV *offline* loading architecture.

Figure 5 shows how SOLV Offline Loading works.

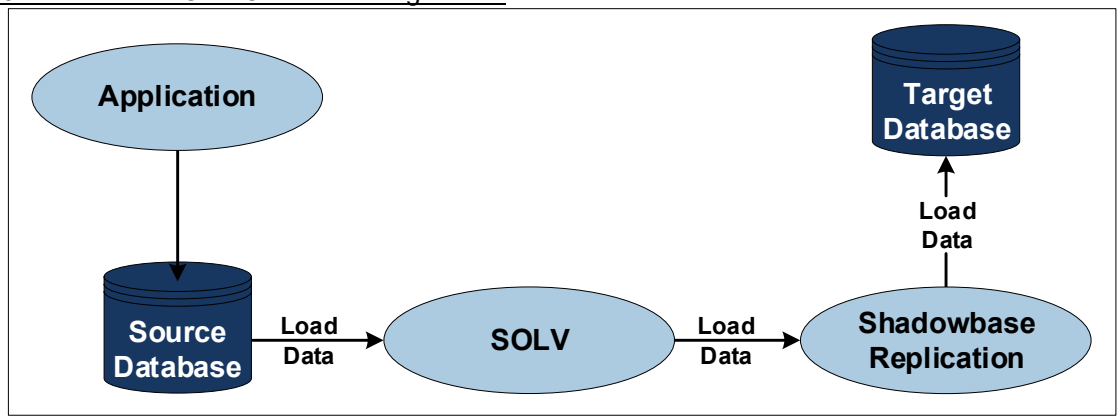


Figure 5 – HPE Shadowbase Online Loader (SOLV) Offline Loading (Change Data Replication is not Active)

1. In Figure 5, an application posts its changes to a source database.
2. SOLV reads a copy of the source data at a point-in-time and optionally performs data transformations on it.
3. Then, the Shadowbase replication engine prepares and applies the data into the target database.

For this approach, a point-in-time copy of the source database (load data) is taken, and applied into a target database environment by SOLV and Shadowbase replication.

This form of loading is *offline* because any source application changes made to the source database after the load starts (and after the data has been loaded) will not subsequently be replicated into the target database. This form of loading is most suitable when the source database is not actively being updated, or when a point-in-time snapshot of the source database is required. SOLV allows specifications for the entire source file/table (or even key ranges) for the load. Source data filtering (e.g., by partition, or data content) is also supported.

HPE Shadowbase SOLV for HPE NonStop Open System Services (OSS)

SOLV for OSS supports loading/copying OSS *regular* files and validating data consistency. It can even create a backup copy of the data and synchronize it with the source. It leverages the SOLVUTIL process (a low-level Shadowbase replication process that runs in the Guardian space) to copy a particular file (or set of files) from the source system to the target. These source and target files can be on the same or different systems. SOLVUTIL can run on either the source or target system.

End of File (EOF) Chasing

This method of replication is called *file chasing* – or more accurately *end of file (EOF) chasing* – because SOLVUTIL will replicate any data appended into the source file to the target file, applying (or "appending") it to the end of the target file.

SOLVUTIL checks if there is data to be replicated, sleeps for a short period, and then wakes up and checks if there is more data to replicate. Any newly appended source file data is read and replicated by SOLVUTIL into the target file, applying it to the end of the target file. When SOLVUTIL reaches EOF on the source, it repeats this process. This process continues the entire time the source application is appending to a source file.

SOLVUTIL contains a function to automatically track when the application switches from one source file to another (e.g., the source log file rolls to a new log file), and starts replicating into a new target file. To validate the data, the SOLVUTIL process can be given two filenames and it will compare the contents of both files (Figure 6).

Figure 6 illustrates how SOLVUTIL compares OSS Regular Files.

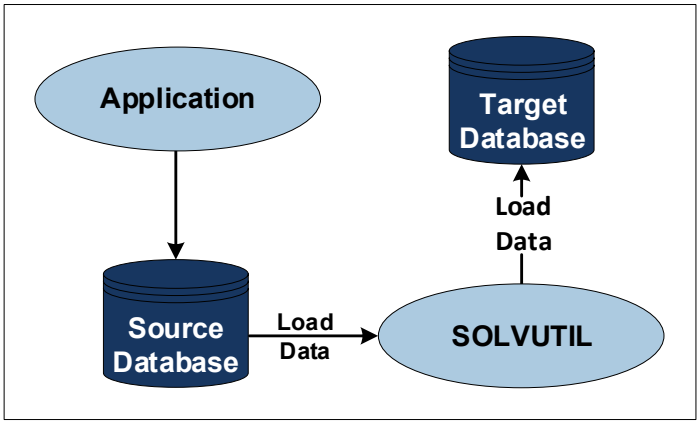


Figure 6 – HPE Shadowbase SOLVUTIL for OSS Regular Files

The SOLVUTIL process recognizes and highlights two key situations:

- *Mismatched Data* – It provides the user with the address of the first mismatch, and options to perform a loading session to correct the target's mismatched data.
- *Row Not Found* – This situation occurs when a row of data is missing in the target table. It provides the user options to perform a loading session to insert values from the source to resolve the situation.

HPE Shadowbase ETL Toolkit

The HPE Shadowbase ETL Toolkit is particularly useful for feeding high-volume quantities of loading data into a data warehouse, and when feeding row-based data into columnar databases. The Toolkit extends HPE Shadowbase Application Integration to provide a suite of adapters to easily integrate the source data with the target environment.

Available Adapters Include

- Integrating with KAFKA
- Integrating with IBM MQ environments
- JSON output format
- Flat files (CSV, fixed position, tab-delimited)
- Custom Shadowbase User Exits to interface Shadowbase with almost all third-party APIs

Key Features of the ETL Toolkit

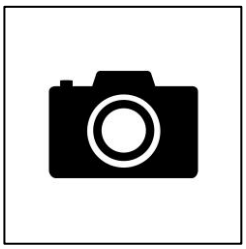
- Provides Shadowbase user exit examples in a set of 'C' language functions that can be adapted for specific scenarios
- Trickle-feeds replicated-events via a loading and UN-loading
- Provides snapshot, point-in-time bulk loading and UN-loading
- Extracts, transforms, filters, and cleanses data before loading and unloading when used with SBMAP
- Keeps the source or target database online for read-write activity while loading and unloading
- Leverages the SOLV features for ETL processing

The HPE Shadowbase ETL Toolkit and SOLV can be Combined

The HPE Shadowbase ETL Toolkit combined with SOLV loading capabilities allow for reading and injecting events from flat files into the Shadowbase replication engine for real-time processing in various architectures. When they are combined, the Toolkit can be used to extract select data from a source database, Shadowbase replication can transform/cleanse it into the proper format, and save it into flat files that the ETL loading utility can understand and load into a target database environment not directly supported by Shadowbase replication.

Alternatively, the ETL Toolkit can produce flat files, comma-separated value (CSV), fixed-position, and tab-delimited formats of database data or change events that can then be subsequently processed by the ETL Toolkit or a different vendor's ETL program or application process.

SOLV "Snapshot" Loads



SOLV can independently read all or parts of a source database's audit trail (or change log), and periodically produce *snapshot* (or *point-in-time)* loads into a target environment. This ability flattens the data, and periodically loads it into the new target environment in an incremental approach via micro-batching. In this architecture, any changes made to the source database during and after the load are typically *not* replicated to the target database. During a SOLV load, the source database data, reflecting all committed transactions *up to that point*, is loaded into the target database.

Key Considerations of SOLV

1. SOLV can be configured to only read and load *consistent* source database data. In this mode, SOLV will *not* read through locked or otherwise inconsistent data, including data with in-process transactions.
2. A typical load takes time, and any changes that occur during the load may not be stored in the target if that data was already loaded. In other words, the target may not contain the latest data that was updated while the load occurred.
3. To ensure that the entire target matche s the source after the load completes, the source application needs to temporarily *stop* updating the source database when the snap-shot load is taken, or the "lock file during the load" option must be selected.
    a. Otherwise, the change data that occurred during the load needs to be subsequently loaded into the target database to re-synchronize it with the source. Shadowbase data replication can provide this feature.

SOLV's Data Consistency Options

SOLV offers several *data consistency* level options. These consistency options allow the user to tailor the loading parameters to the particular data consistency needs. These options can affect data concurrency while the load takes place. By default, the data being loaded is locally consistent (unlocked and consistent).

However, SOLV also supports an inconsistent load level (that reads through locks) as well as a file/table level lock consistency mode, where it locks the entire source file or table before the load begins. These options ensure that the point-in-time load of the target is consistent with the *entire* source when the load completes.

SOLV Leverages the Shadowbase Data Replication Engine in Certain Situations

Data Replication During the Load

ETL and SOLV fully cooperate with the Shadowbase data replication engine to ensure that the data being replicated and the data being loaded are properly serialized when applied to the target database. ETL and SOLV leverage the Shadowbase core (the main data replication engine) for certain services. If the engine is being used for data replication of the source file or table being loaded to the target, then Shadowbase ETL and SOLV can integrate with it to load the target database while change data replication is occurring.

Target Database is Current with Source's Database Changes During the Load

The target database is always kept current with the source's database changes during the load. Hence, while the load occurs, the Shadowbase core properly keeps that part of the target database (that has already been loaded) synchronized with any subsequent changes that the application makes to the source database. This approach is particularly useful for high-volume applications with a large database because change data replication is not suspended while the load takes place, (*unlike other data loading products*). Instead, the queue of database changes is consumed while the load takes place, and the target is always kept *current* with the source's database changes during the load. The target does not grow stale after the load completes because the Shadowbase core continues to keep the target database synchronized with the source database.

Write Loading Routines or Mappings Only Once

SOLV can utilize the Shadowbase core for powerful data transformation, data filtering, and data cleansing routines and scripts. Many data mappings are performed out-of-the-box. Other more complex data mapping functions must be developed using Shadowbase data transformation technology such as user exits or SBMAP. Then, both the Shadowbase engine as well as ETL and SOLV can use these same functions during the load. The routines or mappings only need to be written and tested once, since they are then used for both database

change replication as well as the loading operation. They do not have to be re-implemented in a separate custom ETL tool, script, or program.

SOLV leverages the Shadowbase technology that allows the source and target database data to be homogeneous (such as like-to-like replication), or heterogeneous (such as loading a non-relational source file into a target SQL table or when loading a data warehouse). Once Shadowbase replication is configured, SOLV automatically uses this information to perform the load.

Typical SOLV Unloading Scenarios

Flattening a Database

In Figure 7, an application is posting changes to a source database, which is being read, parsed, and formatted into a flat file, such as a typical fixed-position (FP) or comma-separated values (CSV) format for a subsequent program, application, or ETL tool to process.

Figure 7 depicts HPE Shadowbase ETL output producing flat files from a source database.

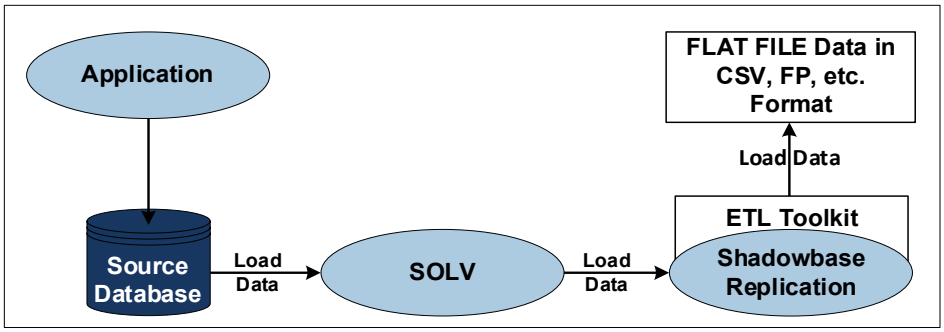


Figure 7 – SOLV Unloading with HPE Shadowbase ETL Output

More specifically, SOLV is reading the source data (of interest, meaning data can be selectively read) from the source database, and sending that data to the Shadowbase engine to transform, filter, and process it (via the embedded Shadowbase ETL Toolkit) into the desired flat file target format. The flat file is produced by the Shadowbase user exit functionality using the Toolkit, typically written as a *consumptive* user exit. A consumptive user exit receives the data changes, and parses them into the desired output format and writes them directly into the target flat file.

Load During Replication

Shadowbase data replication can also be active on the source data being processed, allowing for database changes to be added into the output file. This application has proven popular, for instance, for telco environments to capture their call detail record (CDR) data for loading a downstream data warehouse system.

For example, Figure 8 depicts flattening application change data and merging it with load data.

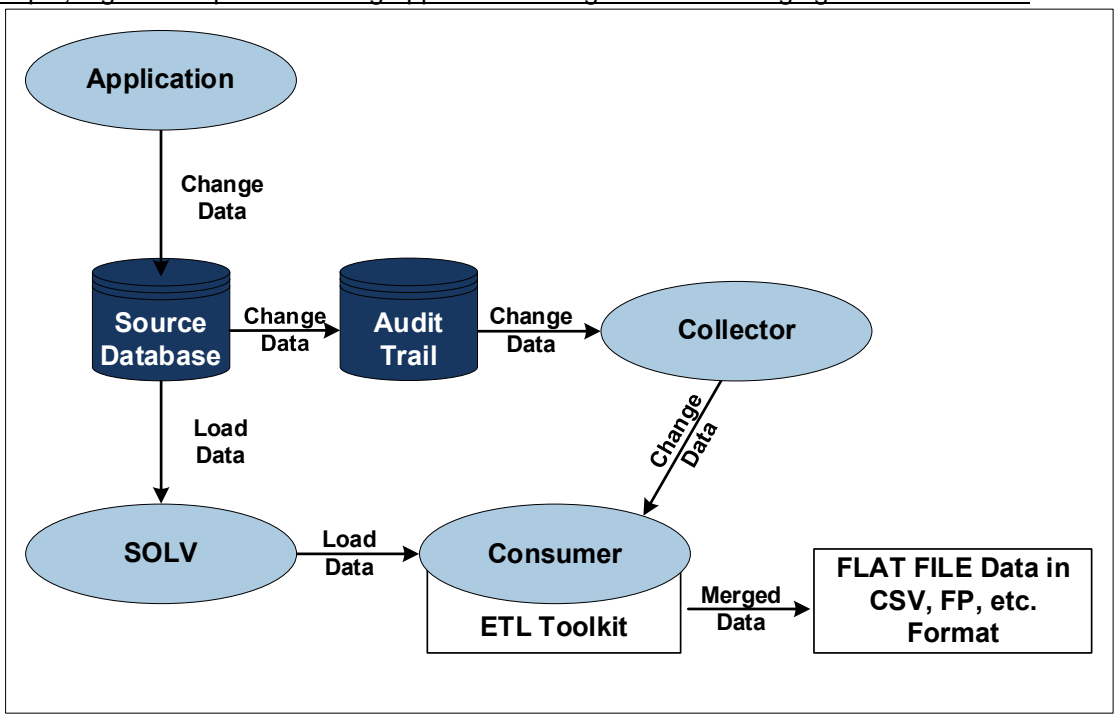


Figure 8 – SOLV Unloading Integrated with HPE Shadowbase Replication and ETL Output

This common scenario is where an application is actively modifying the source database, and SOLV is unloading that source data into a target flat file in CSV or fixed-position format along with any changes made to the source for the data of interest.

HPE Shadowbase SOLV Manager (SOLVMGR)

SOLVMGR (also known as the *HPE Shadowbase File Chaser*) can also process flat files of input data and inject their contained events into the replication engine for subsequent application into a supported database or flat file.

The SOLVMGR is a component of the SOLV loading product that can monitor objects such as:

- filesets,
- application log files,
- queues,
- etc.,

and *chase* the data insertions into these objects, typically at end-of-file (EOF), extracting them for Shadowbase software to process (often referred to as *EOF chasing*). The SOLVMGR will read and prepare that data and inject it into the Shadowbase replication engine for processing. This extends Shadowbase replication so it is capable of replicating *even more* data sources!

Figure 9 depicts a typical HPE Shadowbase ETL scenario for injecting ETL data into the replication engine.

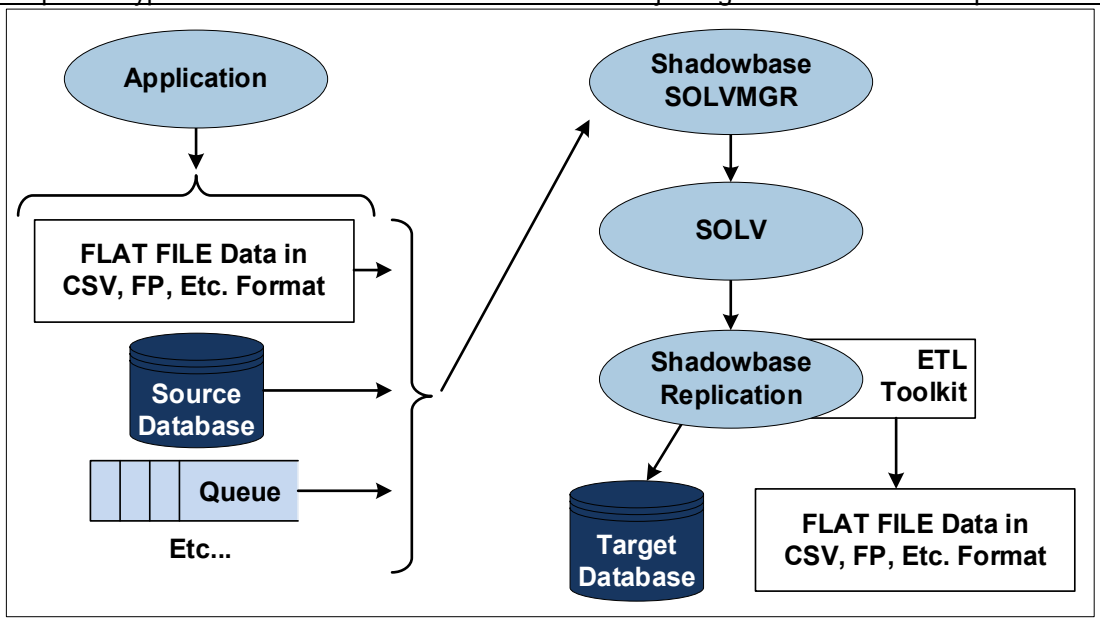


Figure 9 – HPE Shadowbase SOLVMGR Injecting Flat File Data into the Replication Engine

In this figure, the HPE Shadowbase SOLVMGR (also known as the Shadowbase *File Chaser*) is reading data to inject into the Shadowbase replication engine from a variety of formats, including:

- flat files,
- application change data log files,
- middleware queues,
- or other application-generated files.

The SOLVMGR (or, in some cases, a Shadowbase user exit that is directly configured with the Shadowbase replication engine), will read the events as they are generated and inject them into the replication stream for subsequent processing by the engine.

Summary

HPE Shadowbase Online Loader and the HPE Shadowbase ETL Toolkit are utility products designed to perform initial database loads, including any subsequent target database resynchronization, providing online and offline loading (or unloading) of a source database into a target database, which can be open for reading and updating while the load (or copy) occurs.

The SOLV utility plays an important role in eliminating planned downtime via zero downtime migration (ZDM) and also provides an option for replicating non-audited HPE NonStop Enscribe and SQL data. The ETL Toolkit is used to extract database changes or initial load data into flat files for subsequent ETL loading into a data warehouse using comma-separated value (CSV), fixed-position, and tab-delimited formats. The ETL Toolkit supports a variety of features and databases.

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
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.

Copyright and Trademark Information

This document is Copyright © 2019, 2020, 2024 by Gravic, Inc. Gravic, Shadowbase and Total Replication Solutions are registered trademarks of Gravic, Inc. All other brand and product names are the trademarks or registered trademarks of their respective owners. Specifications subject to change without notice.