

# HPE Shadowbase Repair

## Correct Table Differences

HPE Shadowbase Repair is a feature in the [HPE Shadowbase Compare](#) for SQL NonStop Utility to assist with correcting or repairing SQL/MP or SQL/MX database discrepancies between two tables. Shadowbase Repair works on any two tables in these databases, provided they are both of the same type (MP or MX). It does not require a Shadowbase replication configuration and can be used in any environment where a source and target table need to be verified. Repair is particularly useful in business continuity data replication environments to provide confidence (or certification) that the source and target data matches, and to repair the discrepancies when it does not.



Mission-critical databases are often extremely large, consisting of millions, even billions, or more of data entries. Additionally, these databases are often changing very rapidly, and to improve availability, support OLAP, or to offload reporting, they are often replicated from a source to a target environment.

It is important to know if there are discrepancies between the source and target environments. Shadowbase Compare reports the discrepancies between two sets of SQL tables, which would be an error-prone and time-consuming task to correct if even only a handful of differences were found. With Shadowbase Repair, the user can automatically and quickly correct these differences with just a few simple commands.

With HPE Shadowbase Compare and Repair, the user has full control over selecting which table is considered the *source*, and which table is considered the *target*. Note, changing this frame of reference can be used to control where most of the comparison (and repair) work is performed and can be done based on user preference, system loading, database security or access, or other factors.

## The Shadowbase Compare Ledger Maps Source and Target Value Discrepancies

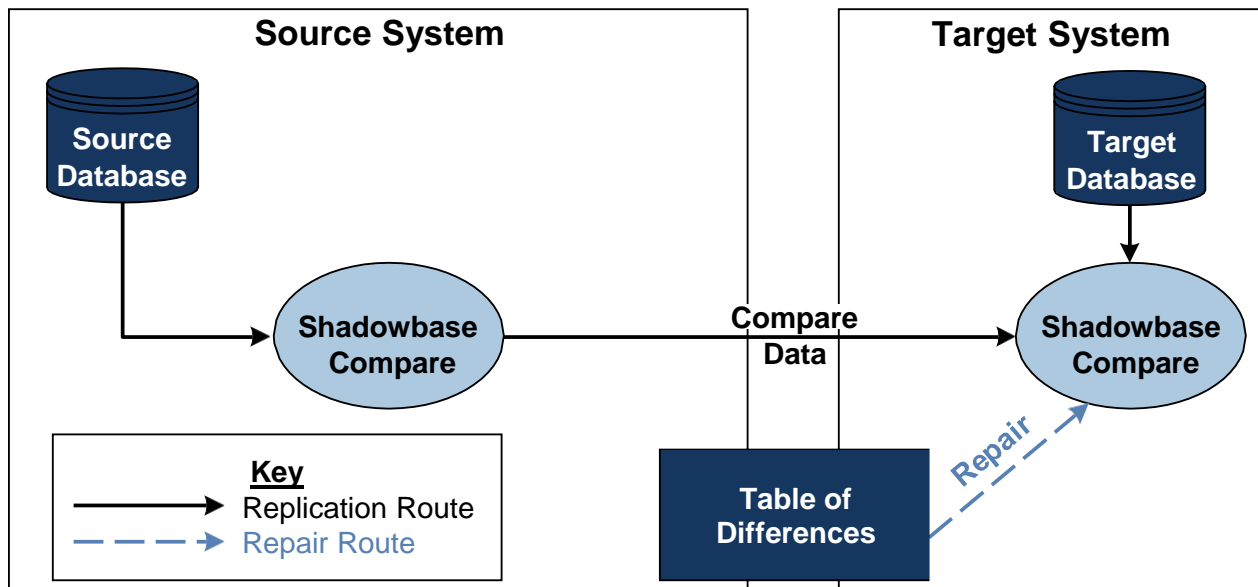


Figure 1 – HPE Shadowbase Compare

HPE Shadowbase Compare acts as a ledger; as it runs, it creates a table of differences, which specifies where the differences are located in each table, and what are the value differences (as shown in Figure 1). Although a replication environment need not be present to run Compare and Repair, Repair uses a frame of reference that defaults to naming one of the tables as the *target*. Compare/Repair will note what changes need to be done to the target to make it match the *source*. (In the following scenario, we assume that this is the case.) For example, data in the source that is not in the target is listed as “missing” from the target; therefore, an INSERT against the target table is required. Data in the target that is not in the source is listed as “extra” data in the target; therefore, a DELETE against the target table is required. Data in both

the source and target that have different column values require an UPDATE for the target table. When a discrepancy is found, the user can specify how to repair the differing tables with a conditional action. Repair references the Compare ledger's information when repairing the data.

With Repair, the options for repairing the discrepancies (in order of suggested usage) include:

- **Option 1:** Update the source data with a “NO-OP” update, using the values in the source table (i.e., setting the source data to the same values). This approach relies on a replication engine to replicate the source update to the target to make the target consistent.
- **Option 2:** Update the target data directly, using the values in the source table. This approach directly updates the target to match the source.
- **Option 3:** Update the target data with a “NO-OP” update, using the values in the target table. This approach relies on a replication engine running in bi-directional mode to replicate the target update to the source to make the source consistent.
- **Option 4:** Update the source data directly, using the values in the target table. This approach directly updates the source to match the target.

### Option 1: Updating the Source Data with a NO-OP Update

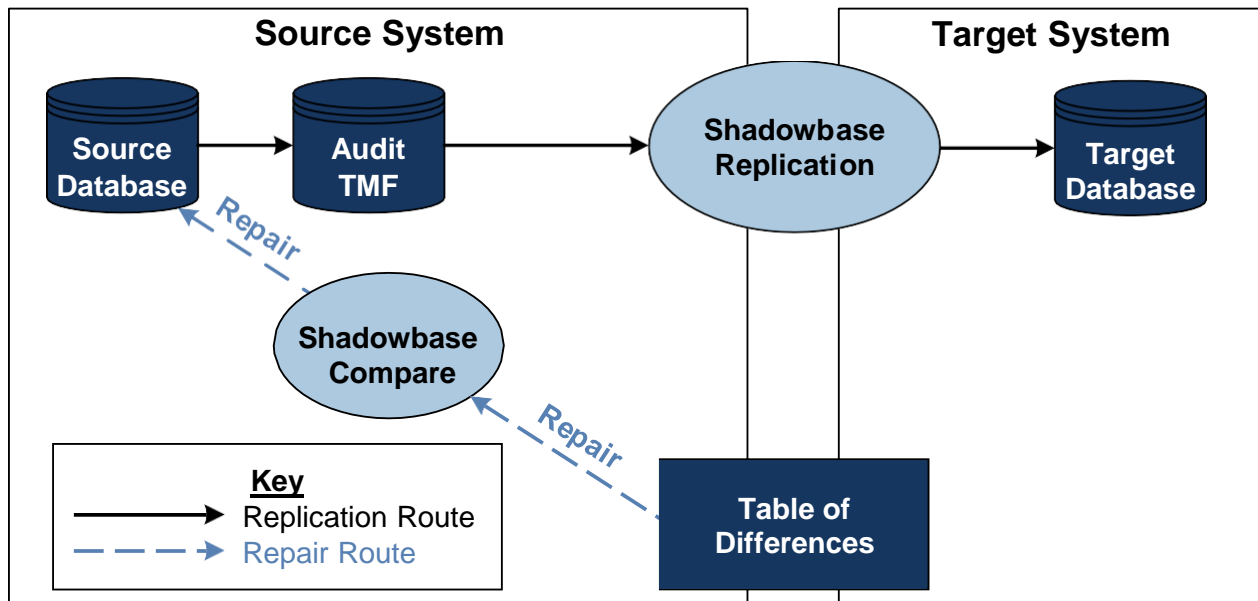


Figure 2 – Updating the Source Data with a NO-OP Update

This approach is suggested when the source and target databases have the same or different format and during active replication. The approach in Figure 2 assumes Shadowbase replication is properly configured, running, and optionally performing data transformations in-flight. (Note, a replication engine must be running in order for the replication of events from the source's audit trail to the target.) If a discrepancy is found between the source table and the target table while the Repair command is running, then it simply updates all of the columns of the source table with the values that already exist (a “NO-OP update”). By updating the source columns with these values, the changes replicate to the target table. This approach is the safest way to conduct the repair when the replication engine is active, and typically should be used if the replication engine is transforming the data in flight.

Option 2: Updating the Target Data Directly

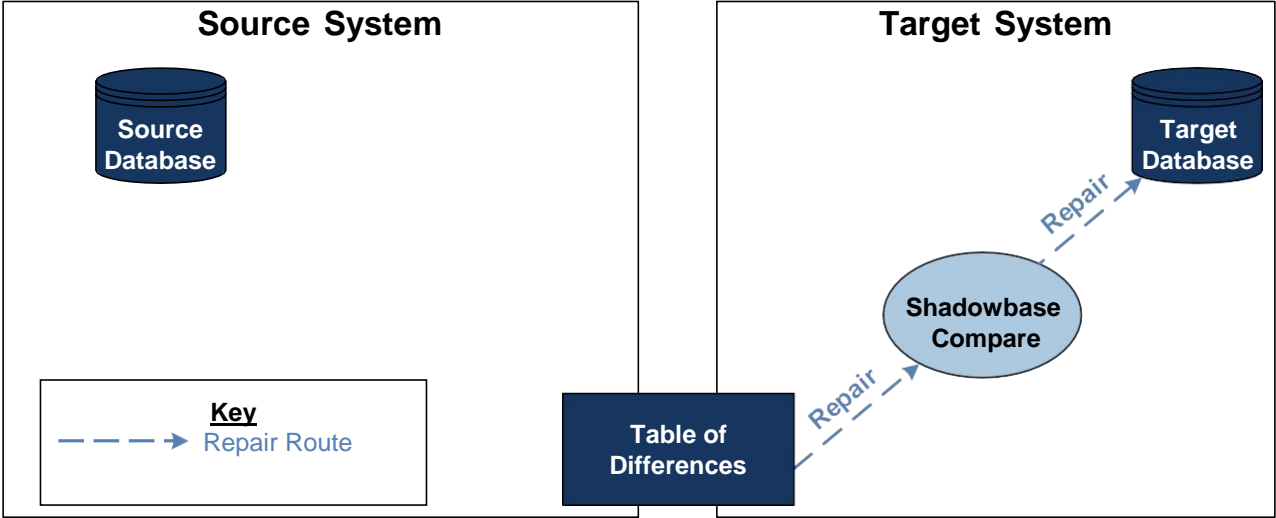


Figure 3 – Updating the Target Data Directly

This approach is suggested when the source and target databases have the same format and when replication is not running. If replication is active, this approach assumes Shadowbase uni-directional replication is properly configured, running, and not performing any form of data transformation, scrubbing, or filtering of the in-flight data, (because when Repair modifies the data, the data avoid this in-flight process).

If a discrepancy is found between the source table and the target table when Repair runs, it updates the current values of selected target columns using the values in the matching source columns (as shown in Figure 3). If necessary, each updated column from source to target can be explicitly specified in order to avoid accidental damage to the user’s data.

Option 3: Updating the Target Data with a NO-OP Update

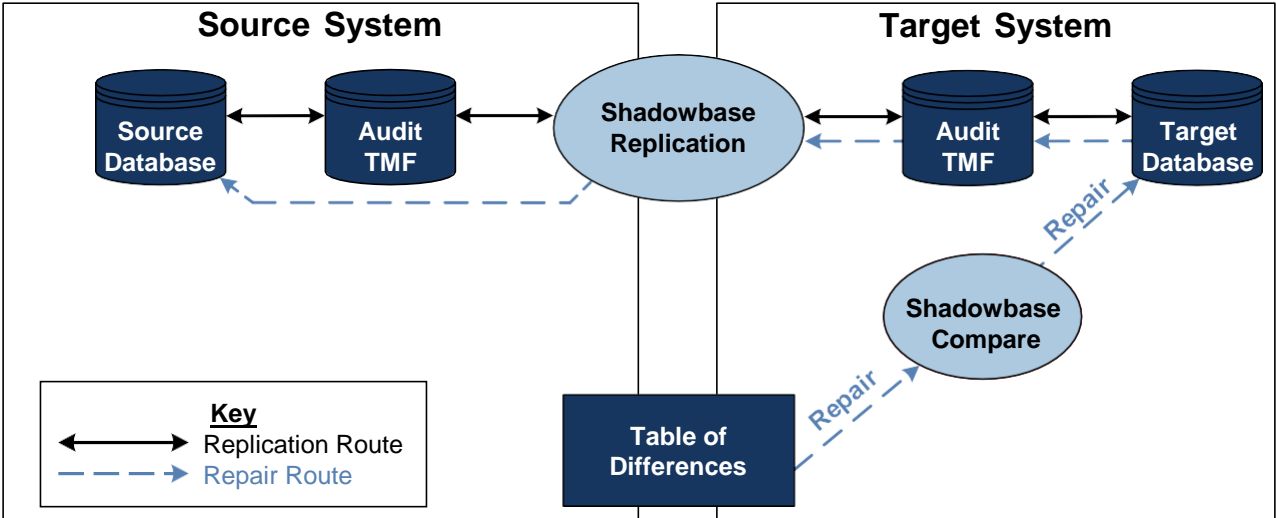


Figure 4 – Updating the Target Data with a NO-OP Update

This approach is similar to the NO-OP case against the source data described above; however, in this case it assumes Shadowbase replication is properly configured and running in a bi-directional environment (as shown in Figure 4). It creates a “NO-OP update,” which replaces the data in the target table with a copy of itself to create an event in the audit trail and leverage the replication stream from the *target* table back to the *source* table.

## Option 4: Updating the Source Data Directly

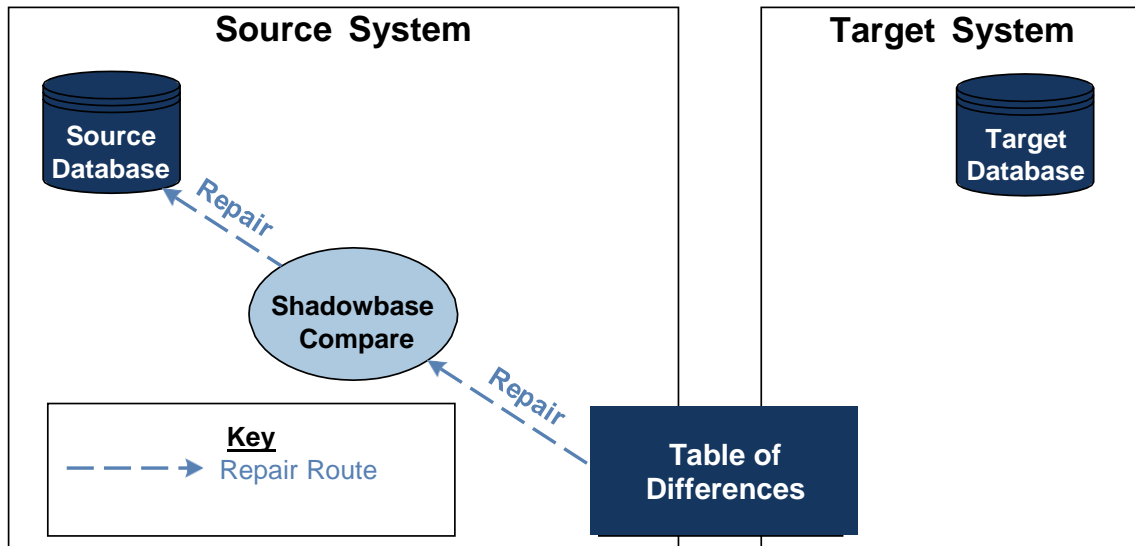


Figure 5 – Updating the Source Data Directly

This approach assumes there is no replication engine running and is symmetrical to “Updating the Target Data Directly.” It directly updates the source table, using the values from the target table (as shown in Figure 5). This approach is least common, since the source data is often considered more current (consistent) than the target data, (but it is available, empowering the user with flexibility and numerous options for unique scenarios).

### Insert and Delete

An insert can be specified if a source or target row is not found, and the missing row can be inserted from the other table. A delete can be specified if a source or target row is found, and the missing row can be deleted from the corresponding table.

## Commands

There are three simple commands in order to run Repair. If Compare is run using these commands, then the differences are recorded and logged. The user can then run Repair to correct these differences.

```
ON DIFF (*) REPAIR DO UPDATE TARGET (*);
ON SOURCE NOT FOUND REPAIR DO DELETE;
ON TARGET NOT FOUND REPAIR DO INSERT ();
```

In practice, every repair operation should have at least three rules to handle each of these operations: whether to update the source or target table in case a discrepancy is found; what to do when a row exists on the target but not the source; and what to do when a row exists on the source but not the target.

## Summary

HPE Shadowbase Compare is a utility that finds the discrepancies between the user’s source and target tables. The Repair command can then be executed to automatically repair any discrepancies between the tables. Before utilizing the Repair command, take caution to carefully consider how the various Repair options will correct the discrepancies. We recommend that the user typically configure the commands so that the target table is properly updated to reflect the source table if any differences are found.

Hewlett Packard Enterprise globally sells and supports Shadowbase solutions under the name HPE Shadowbase. For more information, please contact your local HPE Shadowbase representative or visit our website. For additional information, please view our Shadowbase solution videos: <https://vimeo.com/shadowbasesoftware>.

**Learn more:**

[shadowbasesoftware.com](http://shadowbasesoftware.com)  
[hpe.com](http://hpe.com)

**Contact us:**

Gravic, Inc.  
17 General Warren Blvd  
Malvern, PA 19355-1245 USA  
Tel: +1.610.647.6250  
Fax: +1.610.647.7958  
Email Sales: [shadowbase@gravic.com](mailto:shadowbase@gravic.com)  
Email Support: [sbsupport@gravic.com](mailto:sbsupport@gravic.com)

**Please follow:**



Copyright © 2018, 2022 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.

