Shadowbase Synchronous Replication – Product Overview

A Gravic, Inc. White Paper

Executive Summary

Shadowbase Synchronous replication avoids data loss when a total system or datacenter failure occurs.

This white paper reviews the Shadowbase Synchronous replication features and rollout schedule and discusses the main issues with synchronous replication. It also suggests an approach to leveraging this new technology.

Please note that not all features are available in the first release, and delivery dates as well as specific (future) functionality is not guaranteed.

Table of Contents

Table of Figures

1 Business Continuity Fundamentals

In order to fully explain the features of Shadowbase Synchronous replication, it is first necessary to understand some basic business continuity concepts and the relationships between them.

1.1 RPO and RTO

The availability of a system can be characterized by two parameters – its Recovery Point Objective (RPO) and its Recovery Time Objective (RTO), as shown in Figure 1.

- RPO is the amount of data that an application may lose as a result of a system failure. It is the data updated between the time of the last backup (data taken off platform) and the point of failure. RPO is primarily influenced by the business continuity *technology* (asynchronous vs. synchronous replication).
- RTO is the amount of time that the system is down after a failure. It is the time from the point of failure to the time that the system is restored to an acceptable level of service. RTO is primarily influenced by the business continuity *architecture* (Active/Passive vs. Active/Active).

The goal of a business continuity solution is to minimize RPO (lose less data) and maximize RTO (recover faster).

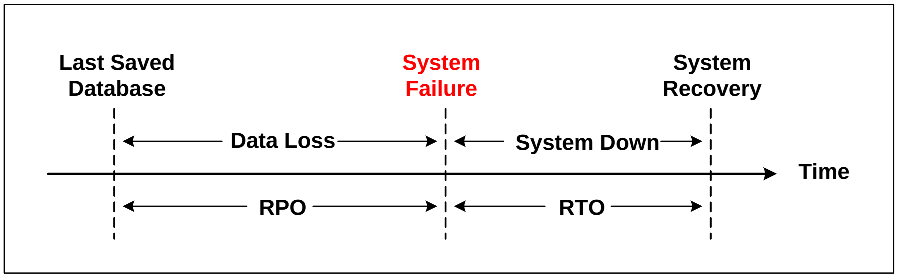

Figure 1 – RPO and RTO Definition

1.2 Business Continuity Architecture

There are two primary business continuity data replication architectures: Active/Passive and Active/Active.

The primary attributes of an Active/Passive architecture are:[1]

- A *highly available* system architecture comprising multiple geographically distributed systems
- All online (transactional) work takes place on a *single* system, the *active* system
- Change (updated) data is replicated from the active system to one or more *passive* systems (*uni-directional* replication)
- If the active system fails, a passive system takes over the online work (one passive system becomes the new active system, via a *failover* process)
    - All users are affected if a failover occurs and they must failover to the passive system
    - This process can be time-consuming and error-prone, leading to failover faults if the failover is not successful
    - Hence, this architecture imposes more risk to the business than Active/Active
- Passive systems cannot be used for online processing (but can be used for read-only or query activities)
- Has an RTO > 0 (recovery takes time, typically minutes to hours)
- Is often referred to as a *disaster recovery* architecture, due to the after-the-fact reconfiguration and recovery aspect

The primary attributes of an Active/Active architecture are:

- A *continuously available* system architecture comprising multiple geographically distributed systems
- Online (transactional) work takes place on *multiple active* systems (two or more), each with its own copy of the database
- Change (updated) data is replicated between all active systems (*bi-directional* replication)

[1]For the purposes of this document, this architecture includes Sizzling-Hot-Takeover (SZT), which is a form of Active/Passive architecture, but with a faster recovery time (RTO → 0) and a reduced risk of failover faults. In all other respects, SZT provides the same characteristics as Active/Passive.

- All nodes are *known-working* (i.e., fully operational and able to assume work of failed systems immediately)
- If an active system fails, other active systems continue processing the extra online workload (providing *continuous application availability*)
  - Fewer users are affected if a system fails
  - Users connected to the failed node are network switched to a surviving node that is already processing requests and is in a *known-working* state
  - Hence, this architecture eliminates the failover risk that Active/Passive systems suffer due to potential failover faults
- All systems can be used for online processing, increasing overall capacity
- Has an RTO = 0 (i.e., recovery is instantaneous; some users see no outage event at all)
- Often referred to as a *disaster tolerant* architecture because application services survive any individual system's failure

1.3 Business Continuity Technology

There are two primary business continuity data replication technologies: asynchronous and synchronous.

The primary attributes of asynchronous replication are:

- Replication activity is decoupled from the application making database changes on the source system
  1. Application does work and commits the database changes
  2. Data changes are read after-the-fact and replicated to the target (passive/backup) database
- Thus, there can be application data updates committed on the source which are not yet replicated to the backup (this time delay is called "replication latency"), and such data which has not yet been replicated can be lost in the event of a failure of the source system (RPO > 0)
- Data collisions[2] are possible with Active/Active architectures; such collisions must either be avoided (e.g., by application or data *partitioning*), or identified and resolved if they do occur

The primary attributes of synchronous replication are:

- Replication activity is synchronized with the application changing the database on the source system
  - Application does work and calls to commit the transaction
  - Changes are not committed on the source system until those changes are replicated to the backup system
- Thus, there cannot be application data updates committed on the source which are not yet replicated to the backup, and no data is lost in the event of a failure of the source system (RPO = 0)
- There are two types of zero data loss sub-architectures:
  - In the first, the source data is merely safestored at the target; it may or may not be applied into the target database before the source is allowed to commit. This sub-architecture is the basis of the Shadowbase Zero Data Loss feature.
  - In the second, the source data is applied into the target database before the source is allowed to commit. This future technology is under consideration.[3]
- Hence, when running in an Active/Active architecture, data collisions may still be possible with a zero data loss architecture, unless the implementation (or a future enhancement) also applies the replicated data into the target database before the source is allowed to commit. In this case, data collisions would not be possible; the collisions become transaction deadlocks, which are much less serious and more easily internally handled by the data replication engine.

2 The Benefits of Shadowbase Synchronous Replication

Asynchronous replication represents state-of-the-art technology, is satisfactory for many applications, and offers excellent levels of protection against outages (especially in SZT and Active/Active configurations).

However, it has limitations:

- Data loss (in Active/Passive, SZT, and Active/Active modes)

[2] Data collisions occur in asynchronous Active/Active architectures when the same database record is updated simultaneously on two (or more) systems in the configuration. The updates are then replicated to the other system(s), overwriting the original changes. Thus, all database copies are made inconsistent and incorrect. Data collisions do not occur with Active/Passive architectures.

[3] Check with Gravic for the possible future availability of this feature.

- May require application/data partitioning (in Active/Active mode), which may result in imbalanced load across systems
- May incur data collisions (in Active/Active mode)

Figure 2 graphically illustrates the relationship between data replication architecture/technology and RTO/RPO. As is shown, an Active/Active architecture has the best RTO, and synchronous replication technology has the best RPO.

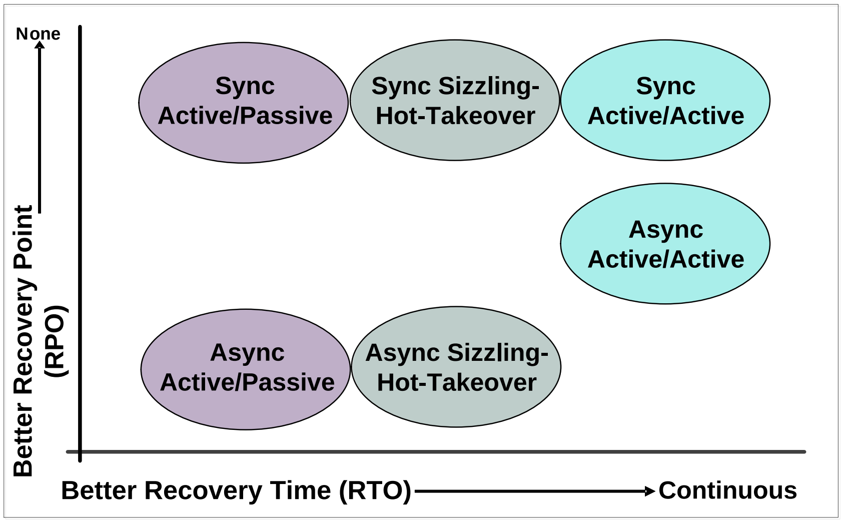

Figure 2 – RPO and RTO for Asynchronous vs. Synchronous Replication

Asynchronous replication is insufficient for the most critical applications – those for which *any* lost data or downtime will incur unacceptable levels of business cost. For such applications, Shadowbase Synchronous replication resolves these issues.

Synchronous replication provides the following benefits compared to asynchronous replication:

- Zero data loss (RPO = 0)
- Data collisions may be possible in Active/Active architectures, unless the synchronous replication engine also applies the replicated data into the target database before the source transaction commits.

Therefore, a business continuity replication solution now exists for those applications where no data loss can be tolerated. Shadowbase Synchronous replication enables the minimum (best) possible values for RPO and RTO for the widest possible range of applications. An implementation using an Active/Active architecture with synchronous replication is the best business continuity solution.[4]

[4] For more information about the pros and cons of various business continuity solutions, see the Gravic white papers: *Fingers Crossed? Or What is Your Business Continuity Plan for the Inevitable?* and *Choosing a Business Continuity Solution to Match Your Business Availability Requirements*.

3 Shadowbase Synchronous Replication – Product Rollout

Shadowbase Synchronous replication capabilities will be incrementally released as follows[5]:

1. Shadowbase Zero Data Loss Release 1 (**Shadowbase ZDL R1**)
    - Built on the tried-and-true Shadowbase Asynchronous technology platform
    - Supports Active/Passive architectures
    - Supports zero data loss for Active/Passive system architectures (uni-directional) by safe-storing data on the target system before the source transaction is allowed to commit
    - Enables *Maximum Availability Mode (MAM)* – If a transaction cannot be safe-stored, Shadowbase replication will failback to Asynchronous Mode, allowing the source application transaction to COMMIT. After the issue is resolved, Shadowbase replication will automatically switch back to Synchronous Mode.
2. **Shadowbase ZDL R2** (future release)
    - Adds bi-directional replication to support Sizzling-Hot-Takeover and Active/Active architectures
    - Data collisions may occur with Shadowbase ZDL when running in an Active/Active architecture. Check with Gravic for the availability of data collision resolution features.
3. **Shadowbase ZDL R3** (future release)
    - Adds support for SQL/MX
    - Adds support for TCP/IP network communications
    - Enables *Maximum Reliability Mode (MRM)* – If the network or target system is down, Shadowbase replication will delay the source application's COMMIT until the data can be safe-stored at the target (or until the application's transaction request times out).
    - Data collisions may occur with Shadowbase ZDL when running in an Active/Active architecture. Check with Gravic for the availability of data collision resolution features.

Figure 3 illustrates the capabilities of each sequential Shadowbase Synchronous replication release.

	Shadowbase ZDL R1	Shadowbase ZDL R2	Shadowbase ZDL R3
Zero data loss?	A/P: Yes SZT: Not supported A/A: Not supported	A/P: Yes SZT: Yes A/A: Yes	A/P: Yes SZT: Yes A/A: Yes
Communications	Expand	Expand	Expand and TCP/IP
Databases	Enscribe, SQL/MP	Enscribe, SQL/MP	Enscribe, SQL/MP, SQL/MX
Are data collisions possible in an active/active environment?	N/A	Yes	Yes

Figure 3 – Shadowbase Synchronous Replication Features by Release

4 Shadowbase Synchronous Replication – Usage Considerations

The following characteristics of Shadowbase ZDL Synchronous replication need to be considered before deployment:

- Potential for Application Latency
- Because transactions may take longer to complete, there may be more concurrent active transactions
- Decide which transactions (and data) to include in synchronous replication
- Decide what to do if network or "target" system is down
- Decide how to handle in-doubt transactions
- May increase the number of source transactions that are aborted
- If running in Active/Active mode, decide how to handle data collisions (ZDL R2 only)
- How to handle upgrades to the ZDL environment

[5] Specifications are subject to change without notice, and delivery dates/timeframes are not guaranteed. Purchasing decisions should not be made based on this material.

Potential for Additional Application Latency

Application Latency is defined as the time between the application committing the transaction (e.g., calling TMF EndTransaction or SQL COMMIT WORK on an HPE NonStop system) and the response being received (transaction committed or aborted). Because replication of the application's data changes to a backup system have to be completed and confirmed before the application's transaction is committed, additional overhead may occur, resulting in a delay (i.e., an increase in application latency). Otherwise, synchronous replication only incurs minimal additional overhead beyond regular asynchronous replication.[6]

Shadowbase ZDL includes an option called *Synchronous Monitor Mode*, which enables users to measure the potential application latency without actually incurring any delay. (The data is still replicated asynchronously while the replication engine monitors what the effects *would be* if synchronous replication were in use.) Synchronous monitor mode should be used to evaluate the effects that synchronous replication will have on your application environment in order to determine if additional system tuning is required before deploying in production.

Potential for more concurrently active transactions

Because transactions may take slightly longer to complete, there may be more concurrently active transactions active in the system. When deploying synchronous replication, the source system must therefore be able to scale to handle more simultaneous transactions while still achieving the same throughput (in transactions-per-second, or TPS).

Since each transaction may last longer, the locks that transaction holds on the data may also last longer; for some applications, this may reduce the concurrency and impose data access delays while a second application waits for the first transaction's locks to clear.

In practice, the source system must have sufficient resource headroom to accommodate more CPU and TMF activity. For example, in a classic Pathway environment, you may need to run more instances of the application server class processes to achieve the same transaction throughput rate.[7]

Deciding what to do if the target system is down (so-called Split Brain Syndrome)

If the source system is unable to communicate with the target system, it is not possible to guarantee synchronous replication. Therefore, it must be decided what action to take in this circumstance. Options include:

- Fall back to asynchronous replication mode (maximizes availability at the expense of possible data loss should a failure occur). Also, if running in Active/Active mode and data collisions are a concern, failover all users to a single system to avoid the potential for data collisions.
- Stop application processing of new transactions (maximizes reliability, or zero data loss, at the expense of application service availability).
- Enter *Pause Mode* – to wait and decide an appropriate action (requires manual intervention via command or API call).
- Consider Maximum Availability Mode (MAM) vs. Maximum Reliability Mode (MRM)

Maximum Availability Mode (MAM) vs. Maximum Reliability Mode (MRM)
MAM and MRM are two modes that can be selected for overall Shadowbase ZDL processing.

In MAM, providing application access/services is considered more important than safestoring the data. When in this mode, Shadowbase ZDL considers maintining application availability (the ability to process transactions through to COMMIT completion) to be paramount. Hence, the replication engine *may* fall back to asynchronous mode if the data cannot be safestored on the target environment within supplied timeout thresholds, meaning that there is a potential for data loss when in MAM fall-back mode. Therefore, any source data not fully safestored on the target or in-transit can be lost if a catastrophic failure of the source occurs while running in asynchronous mode.

[6]It is only at transaction commit time that synchronization occurs; prior to that point, application data changes are replicated asynchronously with minimal impact to the application. This patented Shadowbase innovation is known as *Coordinated Commits*.

[7]Applications that scale well are a basic tenet of properly designed transaction processing systems. Synchronous replication leverages this characteristic to achieve the same overall processing rate even though each individual transaction may take longer to complete. Therefore, applications that cannot scale well may not perform well when deploying synchronous replication.

More specifically, at source transaction commit time, Shadowbase ZDL attempts to safestore the source application's transaction data at the target before voting to allow the source transaction to commit. If Shadowbase ZDL cannot complete the safestore of the transaction's data for any reason (e.g., timeout), it votes for the source application transaction to commit and switches back to asynchronous replication. When the issue(s) is resolved, Shadowbase ZDL will automatically recover to synchronous replication mode.

In MRM, data integrity and safestoring the data is considered more important than application processing. When in this mode, Shadowbase ZDL considers maintaining zero data loss to be paramount, possibly to the detriment of application availability. Hence, Shadowbase ZDL holds its vote of the source application's transaction commit until the data is safestored at the target. If, for any reason, the synchronous replication safestore at the target cannot be completed, Shadowbase ZDL pauses the source application's commit (by not voting to allow the transaction to commit) until Shadowbase ZDL can safestore the data at the target environment. Both the source and the associated target transactions remain active, holding locks, until Shadowbase ZDL can be assured. If it cannot be assured before a configurable timeout expires, both transactions will abort.

Deciding how to handle in-doubt transactions

An in-doubt transaction is one that has been fully processed on the source environment (i.e., the application has called commit), and the replication engine has fully delivered all changes to the target environment and has also voted to commit the source transaction. The transaction becomes in-doubt at the target if the replication engine does not receive a final directive that the source transaction committed or aborted (e.g., the communication line between the source and target goes down).

In this case, Shadowbase replication on the target system does not know whether the transaction was committed or aborted by TMF at the source. By default, it will hold in-doubt transactions, awaiting user direction. There are also configuration options to unilaterally commit or abort in-doubt transactions.

Deciding which transactions to include in synchronous replication

Not all data is created equal; some is more valuable than others. Therefore, Shadowbase replication provides means to select which data is replicated synchronously vs. asynchronously. In a practical sense, this means that in the event of source system failure, the synchronous data that committed on the source will be present on the target environment and the asynchronous data may not be. Hence, the best candidates for asynchronous data are the data items that can be lost or otherwise recreated (e.g., mostly static data).

It is possible to include all transactions in synchronous replication or to filter based on user requirements (e.g., by application program name, process name, user id, etc.). In addition, Shadowbase configuration options allow the selection of individual files and/or tables to be replicated synchronously or asynchronously. In a future release, APIs may also be provided to programmatically control which transactions are replicated synchronously on a per transaction basis.

Shadowbase ZDL may increase the number of source transactions that are aborted

In an Active/Passive synchronous architecture, an increase in the number of source transactions that are aborted will occur if Shadowbase ZDL cannot perform synchronous processing and if the customer has chosen "maximum reliability" mode. In this case, a split-brain condition has arisen, and until the situation is resolved, all new transactions will be paused and eventually aborted.

In an Active/Active synchronous architecture, an increase in the number of source transactions that are aborted will also occur if Shadowbase replication encounters data collisions. In this case, Shadowbase ZDL will abort one of the transactions involved to avoid the data collision.

In either architecture, when an aborted transaction arises, the solution is to resubmit the aborted request/transaction. This process is similar to the processing for any application transaction that is automatically aborted by TMF (for reasons of timeout, certain media failures, deadlock, etc.).

5 Shadowbase Synchronous Replication – Deployment Procedures

Gravic is available for Professional Services to help ensure its customer's projects run smoothly.

Before putting a Shadowbase Synchronous replication environment into production, the possible impacts should be managed using a phased implementation following these steps:

1. In a test environment, create and tune a copy of the intended ZDL application in Shadowbase Asynchronous mode
2. Enable Shadowbase ZDL Sync Monitor Mode
3. Activate Shadowbase ZDL Synchronous Replication
4. Once all customer SLAs (Service Level Agreements) are met, migrate the test application into production

Step One – In a Test Environment, Create and Tune a Copy of the Intended ZDL Application in Shadowbase Asynchronous Mode

Shadowbase Synchronous replication is built on its existing Asynchronous technology platform. Hence, the first step in deploying Shadowbase Synchronous replication is to create, tune, and implement a Shadowbase Asynchronous replication environment for the data that that is to be replicated synchronously. Become familiar with Shadowbase features, including monitoring and managing, problem identification and resolution, shutdowns plus failures and restarts, failovers and failbacks, etc. Establish an asynchronous application performance baseline by gathering statistics that can be used as a benchmark when tuning Shadowbase Synchronous replication.

Existing customers with a well-tuned Shadowbase Asynchronous production version of the intended Synchronous application can start by copying this environment to a test system.

Step Two – Enable Shadowbase ZDL Sync Monitor Mode

Enabling *Sync Monitor Mode* allows Shadowbase to measure its impact on the application's transaction speed *without actually performing synchronous replication* or affecting the existing Shadowbase Asynchronous replication. This provides a "best practices estimate" for expected performance and application/transaction latency after synchronous deployment. Note: no application latency is introduced in this mode, it is only simulated and computed.

Test in this mode, tune (using MEASURE, Shadowbase STATS, ESTATS, etc.) as necessary, and study the projected replication latency and application latency[8]. Share these test results with HPE/Gravic for tuning pointers.

Step Three – Activate Shadowbase ZDL Synchronous Replication

Turn on Shadowbase Synchronous replication (i.e., exit Synchronous Monitor Mode) and monitor the performance (throughput) and application latency under various loads (including peak). Tune the system as necessary. Be sure to test failure and recovery modes, as well as failover and failback to become familiar with the differences between async replication processing and sync replication processing in each of these scenarios. Report any issues.

Update existing documentation and procedures, specifically note how these tests and procedures differ from a pure async replication environment.

Step Four – Once All Customer SLAs are Met, Migrate the Test Application into Production

Once all testing of the Shadowbase Synchronous replication environment is complete and the environment is performing acceptably, put the system into production. Continue to monitor the environment to ensure SLAs are met. The application can remain online while starting synchronous replication (however, starting sync mode while taking a brief application outage is easier), but a restart to the existing Shadowbase environment may be required.

[8]Note: Refer to the *Managing Latency Technical Brief* for more information on how to monitor and manage replication and application latency.

6 Shadowbase Synchronous Replication – Current Restrictions

There are some current restrictions on the usage of Shadowbase Synchronous replication which must be considered[9]:

- Supported platforms – HPE NonStop servers L-Series and newer only (including both NonStop X and Virtualized NonStop)
- SQL/MX – Not supported in synchronous mode (however, asynchronous mode is fully supported); adding support for synchronous mode is in-plan (check with Gravic for availability)
- Dependencies – TMF SPRs required: L21.06 (or later); or L20.10 + T8606L01^ANR (or later); or the appropriate set of SPRs applied to L19.08 (or later)[10]
- Interoperability – NonStop to NonStop platforms only; heterogeneous architectures (for example, NonStop to Other Servers) are under consideration (discuss your needs with Gravic)
- DDL operations – CREATE table, DROP table, etc. are not currently supported by Shadowbase ZDL. Workarounds include manually implementing the command(s) on each side or using the Shadowbase Asynchronous replication's DDL Command Replication (DCR) feature.

7 Shadowbase Synchronous Replication – Summary

Shadowbase Synchronous replication resolves the key issue with asynchronous replication: no data loss (RPO = 0) on source system failure.

In a nutshell, by removing this restriction, the minimum possible outage and data loss for the widest possible range of applications is now a reality. Shadowbase Synchronous replication is the only choice for the most business critical applications where even milliseconds of lost data or downtime is unacceptable. Shadowbase ZDL provides unique, differentiating product capabilities for the HPE NonStop market!

[9] Contact Gravic or your Shadowbase reseller regarding these Shadowbase ZDL features.

[10] Contact Gravic or your Shadowbase reseller for more details and verification

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.

Disclaimers

This document describes major enhancements to the Shadowbase suite of data replication products. The new features described herein and the availability of these products are not guaranteed; dates and functionality are subject to change without notice. Check with your HPE account team or Gravic for the latest feature set.

This product does not guarantee that you will not lose any data; all user warranties are provided solely in accordance with the terms of the product License Agreement. Each user's experiences will vary depending on its system configuration, hardware and other software compatibility, operator capability, data integrity, user procedures, backups and verification, network integrity, third party products and services, modifications and updates to this product and others, as well as other factors. Please consult with your supplier and review our License Agreement for more information.