An ***InterSystems*** *Technical Overview*

One Memorial Drive, Cambridge, MA 02142, USA
Tel: +1.617.621.0600
Fax: +1.617.494.1631
http://www.intersystems.com

INTERSYSTEMS CACHÉ® DATABASE MIRRORING

Providing a reliable infrastructure for rapid, unattended, automated failover…

CONTENTS

By: Vik Nagjee, Product Manager
vik.nagjee@intersystems.com

TECHNOLOGY OVERVIEW

Traditional availability and replication solutions often require substantial capital investments in infrastructure, deployment, configuration, software licensing, and planning. Caché Database Mirroring (Mirroring) is designed to provide an economical solution for rapid, reliable, robust, automatic failover between two Caché systems, making mirroring the ideal automatic failover high-availability solution for the enterprise.

Key Features and Benefits:

- Economical high availability solution with automatic failover for database systems
- Redundant components minimize shared-resource related risks
- Logical data replication minimizes risks of carry-forward physical corruption
- Provides a solution for both planned and unplanned downtime
- Provides business continuity benefits via a geographically dispersed disaster recovery configuration

In addition to providing an availability solution for unplanned downtime, mirroring offers the flexibility to incorporate certain planned downtimes on a particular Caché system while minimizing the overall SLA's for the organization. Combining InterSystems' Enterprise Cache Protocol (ECP) application servers with mirroring provides an additional level of availability. Application servers allow processing to seamlessly continue on the new system once the failover is complete, thus greatly minimizing workflow and user disruption. Configuring the two mirror members in separate data centers offers additional redundancy and protection from catastrophic events.

Traditional availability solutions that rely on shared resources (such as shared disk) are often susceptible to a single point of failure with respect to that shared resource. Mirroring reduces that risk by maintaining independent components on the primary and backup mirror systems. Further, by utilizing logical data replication, mirroring reduces the potential risks associated with physical replication, such as out-of-order updates and carry-forward corruption, which are possible with other replication technologies such as SAN-based replication.

Finally, mirroring allows for a special *Async Member*, which can be configured to receive updates from multiple mirrors across the enterprise. This allows a single system to act as a comprehensive enterprise data store, enabling – through the use of InterSystems DeepSee™ – real-time business intelligence that uses enterprise-wide data. The async member can also be deployed in a Disaster Recovery model in which a single mirror can update up to six geographically-dispersed async members; this model provides a robust framework for distributed data replication, thus ensuring business continuity benefits to the organization. The async member can also be configured as a traditional reporting system so that application reporting can be offloaded from the main production system.

INTRODUCTION

The cost of system downtime can range from thousands to millions of dollars, depending on the type and length of outage and the type of system impacted. Availability is critical for economic success, and it is the goal of most organizations to provide maximum availability with minimum downtime, both planned (such as scheduled maintenance) and unplanned (such as software or hardware failure).

Table 1 provides an illustration of acceptable downtime based on various availability targets:

Availability %	Downtime – Per Year	Downtime – Per Month	Downtime – Per Week
99.9999% (six 9's)	31 seconds	2.59 seconds	0.605 seconds
99.999% (five 9's)	5.26 minutes	25.9 seconds	6.05 seconds
99.99% (four 9's)	52.6 minutes	4.32 minutes	1.01 minutes
99.9% (three 9's)	8.67 hours	43.2 minutes	10.1 minutes
99% (two 9's)	3.65 days	7.20 hours	1.68 hours
95%	18.25 days	36 hours	8.4 hours
90%	36.5 days	72 hours	16.8 hours

Table 1: The Availability Matrix

Availability solutions are typically designed based on Service Level Agreements (SLAs) applicable to a particular application at a given organization. Figure 1 illustrates the varied cost of providing availability based on the level of availability desired:

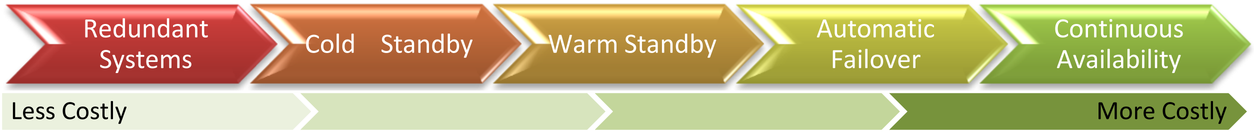

Figure 1: The Levels of Availability

Typical configurations employ redundancy at the hardware and storage level and include cold standby capabilities such as tape and/or online backups. As availability requirements increase, so do the complexities and costs of the supporting systems. Warm standby, or failover, systems typically involve a significant additional capital expense for hardware and software licenses. They often include shared resources, such as shared disk and cluster file systems, which can sometimes be single points of failure. Automatic failover systems involve even more hardware and software (and, therefore, greater expense), and typically involve Storage Area Network (SAN)-based replication technologies that can impose geographic constraints on the availability solution. While continuous availability is the true "nirvana" state, it is very difficult and expensive to achieve. Accordingly, most organizations settle for warm standby system availability architectures.

Caché Database Mirroring (Mirroring) falls within the automatic failover category of system availability strategies. Caché provides mirroring at a fraction of the cost of other database technologies, thus providing an economical, comprehensive, reliable, and robust enterprise solution for database availability.

FUNCTIONALITY REVIEW

A CACHÉ DATABASE MIRROR

As illustrated in Figure 2, a Caché Database Mirror is a logical grouping of two Caché systems, known as failover members, which are physically independent systems connected only by a network. After arbitrating between the two systems, the Mirror automatically designates one of them as the primary system; the other one automatically becomes the backup system.

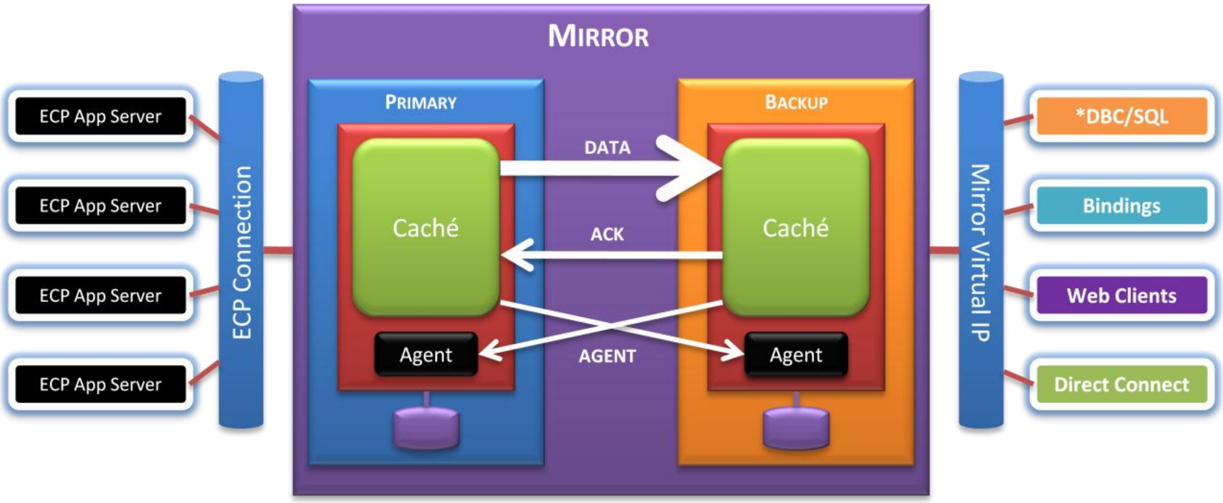

Figure 2: A Mirror

To ensure that the backup system is up-to-date, mirrored databases are synchronized from the primary to the backup failover member in real time. The synchronization is performed over the network[1] in a way that minimizes the performance impact on the primary system. The backup system sends acknowledgments about receipt of mirrored data over a dedicated mirror acknowledgment channel. This indicates, among other things, how up-to-date the backup failover member is. Mirrored databases are only editable on the primary; all mirrored databases on the elected backup system are mounted as read-only, thereby preventing accidental updates to these databases.

Each failover member also include a mirror agent, which provides information on the health of the systems, and assists during the failover process. Each failover member has access to the agent on the other system through a dedicated agent channel.

External clients (language bindings, ODBC/JDBC/SQL clients, direct-connect users, etc.) connect to the mirror through the Mirror Virtual IP (VIP), which is specified during mirroring configuration. The Mirror VIP is automatically bound to an interface on the primary system of the Mirror. The configuration of a Mirror VIP is optional; if not specified, all external clients must connect directly to the running primary, and must have knowledge of both the failover members and their current role within the Mirror.

InterSystems Enterprise Cache Protocol (ECP) application servers have built-in knowledge of the members of the mirror, including the current primary. The application servers, therefore, do not rely on the Mirror VIP, but rather connect directly to the elected primary system.

[1] The data channel is a TCP connection between the primary and backup failover members.

ASYNC MIRROR MEMBERS

As illustrated in Figure 3, mirroring also allows for a special Async Member[2], which can be configured to receive updates from one or more mirrors across the enterprise, thus allowing a single node to act as a comprehensive enterprise-wide data store. The async member provides additional flexibility in that it is possible to choose which mirrored databases from a mirror should be replicated. Alternatively, all mirrored databases from a mirror could be replicated.

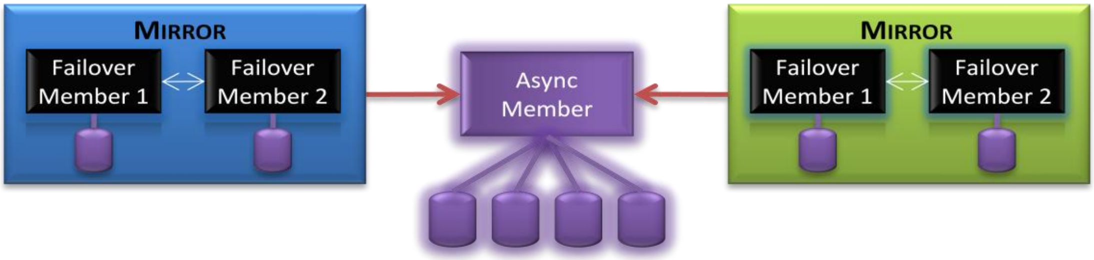

Figure 3: An async member receiving updates from two mirrors

Using an async member as an enterprise data store enables rich reporting, Business Intelligence (BI), and data mining against data from across the enterprise. For example, InterSystems DeepSee™ can be easily deployed on the async member to provide embedded real-time BI so that key performance indicators from across the enterprise can be analyzed quickly and efficiently from a centralized location. Since the async member stays in sync with the mirror(s) to which it is connected, this architecture provides a platform for distributed real-time operational reporting[3].

Finally, as illustrated in Figure 4, it is possible to connect multiple[4] async members to a single mirror, further enhancing the business continuity and disaster recovery plans at an organization by providing a framework for reliable replication across multiple, potentially geographically dispersed, sites.

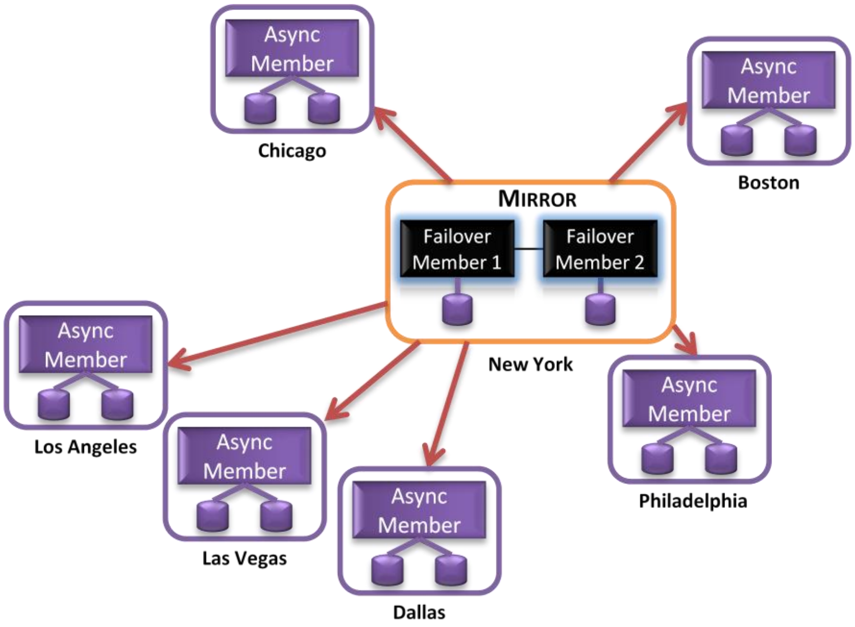

Figure 4: Multiple (6) async members connected to a single mirror

[2] Async members are not candidates for failover; they do not belong to a mirror.

[3] Since the data on the async member is continually updated from changes occurring on the mirrors it is connected to, there is no guarantee of synchronization of updates and synchronization of results across queries on the async member. It is up to the application running against the async member to guarantee consistent results for queries that span changing data.

[4] Up to six async members can be connected to a single mirror

FAILOVER: THE SYSTEM PERSPECTIVE

Whenever possible, mirroring provides rapid, automatic, unattended failover. There are several events that could trigger a failover, such as:

- The backup doesn't hear from the primary within a required interval, which could occur in the case of network problems
- An application or host problem causes Caché to become unresponsive on the primary
- A takeover is initiated by an operator or script

During failover, the backup system ensures it is fully up-to-date before marking itself as the new primary system. The default mirroring configuration prevents errors during takeover, such as split-brain syndrome – a condition whereby both systems concurrently run as active primaries – which could lead to logical database degradation and loss of integrity.

It is also possible for an operator to induce a takeover as part of a planned downtime – for example, to perform hardware or software maintenance on the current primary system. After the planned downtime is complete and the system is brought online, it automatically re-synchronizes within the Mirror.

Finally, it is possible for an operator to temporarily bring the primary system down without causing a failover to occur. This mode can be useful, for example, in the event the primary system needs to be brought down for a very short period of time for maintenance. After bringing the primary system back up, the default behavior of automatic failover is restored.

FAILOVER: THE APPLICATION PERSPECTIVE

On a successful failover, the Mirror VIP (if configured) is automatically bound to a local interface on the new primary. This allows external clients to reconnect[5] to the same Mirror VIP address as before, which greatly simplifies the management of external client programs because they do not need to be aware of multiple database systems and IP addresses. If, however, a Mirror VIP is not configured, external clients will need to maintain knowledge of the two failover members and appropriately connect to the currently running primary.

In an ECP deployment, application servers view a failover as a server restart condition. By design, ECP application servers simply reestablish their connections to the new primary failover member and continue processing their in-progress workload. During the failover process, users connected to the application servers may experience a momentary pause before they are able to resume work. For this to occur, the failover between the two failover members must occur within the configured ECP recovery timeout. If, however, the failover takes longer than this timeout (for example, if the backup was out-of-date during the failover), ECP recovery is initiated (that is, open transactions are rolled back, locks are released, etc.), and new connections to the new primary system are established by the ECP application servers.

[5] The application and connection context are reset because these clients are reconnecting to new systems. Any open transactions are appropriately rolled back. This does not apply to ECP connections as described in the "Mirror" section.

ABOUT CACHÉ

InterSystems Caché® is a new generation of ultra-high-performance database technology. It combines an object database, high-performance SQL, and powerful multidimensional data access – all of which can simultaneously access the same data. Data is only described once in a single integrated data dictionary and is instantly available using all access methods. Caché provides levels of performance, scalability, rapid programming, and ease of use unattainable by relational technology.

But Caché is much more than a pure database technology. Caché includes an application server with advanced object programming capabilities, the ability to easily integrate with a wide variety of technologies, and an extremely high-performance runtime environment with unique data caching technology.

Caché comes with several built-in scripting languages: Caché ObjectScript, a powerful yet easy to- learn object-oriented programming language; Caché Basic, a superset of the widespread Basic programming language, including extensions for powerful data access and object technology; and Caché MVBasic, a variant of Basic used by MultiValue applications (sometimes referred to as Pick applications). Other languages, such as Java, C#, and C++, are supported through direct call-in and other interfaces, including ODBC, JDBC, .NET, and a Caché-provided object interface that allows accessing the Caché database and other Caché facilities as properties and methods.

Caché also goes beyond traditional databases by incorporating a rich environment for developing sophisticated browser-based (Web) applications. Caché Server Pages (CSP) technology allows the rapid development and execution of dynamically generated Web pages. InterSystems Zen™ technology is an implementation of AJAX (Asynchronous Javascript and XML) that is tightly coupled to the Caché database, enabling extremely fast runtime execution for rich Web interfaces. Thousands of simultaneous Web users can access database applications, even on low-cost hardware.

For non-browser based applications, the user interface is typically programmed in one of the popular client-user interface technologies, such as Java, .NET, Delphi, C#, or C++. Best results (fastest programming, greatest performance, and lowest maintenance) are usually obtained by performing all of the rest of the development within Caché. However, Caché also provides extremely high levels of interoperability with other technologies and supports all of the most commonly used development tools, so a wide range of development methodologies are available.

ABOUT INTERSYSTEMS

InterSystems Corporation is a global software technology leader with headquarters in Cambridge, Massachusetts, and offices in 25 countries. InterSystems provides advanced software technologies for breakthrough applications. **InterSystems Caché**® is a high performance object database that makes applications faster and more scalable. **InterSystems Ensemble**® is a seamless platform for integration and the development of connectable applications. **InterSystems HealthShare™** is a platform that enables the fastest creation of an Electronic Health Record for regional or national health information exchange. **InterSystems DeepSee™** is software that makes it possible to embed real-time business intelligence in transactional applications.

For more information, visit InterSystems.com.