InterSystems High Availability Solutions

Version 2020.3
2021-02-04

InterSystems Corporation 1 Memorial Drive Cambridge MA 02142 www.intersystems.com

InterSystems High Availability Solutions
InterSystems IRIS Data Platform Version 2020.3 2021-02-04

For Support questions about any InterSystems products, contact:

InterSystems Worldwide Response Center (WRC)
Tel: +1-617-621-0700
Tel: +44 (0) 844 854 2917
Email: support@InterSystems.com

Table of Contents

InterSystems High Availability Solutions

High availability (HA) refers to the goal of keeping a system or application operational and available to users a very high percentage of the time, minimizing both planned and unplanned downtime. InterSystems IRIS provides its own HA solution, and easily integrates with common HA solutions supplied by operating system providers.

The primary mechanism for maintaining high system availability is called *failover*. Under this approach, a failed primary system is replaced by a backup system; that is, production *fails over* to the backup system. Many HA configurations also provide mechanisms for *disaster recovery* (DR), which is the resumption of system availability when HA mechanisms have been unable to keep the system available.

This article briefly discusses general HA strategies that can be used with InterSystems IRIS-based applications, then covers issues in InterSystems IRIS HA solutions, provides an HA solutions feature comparison, and discusses using distributed caching with a failover strategy

1 Issues in InterSystems IRIS HA Solutions

Bear in mind the following two significant issues when evaluating potential HA solutions for your InterSystems IRIS® systems:

- Shared storage

  An important principle of HA architecture is the avoidance of *single points of failure*. Most HA solutions rely on a shared storage component; which represents just such a risk; if the storage fails, it is impossible to keep the system available. Storage-level redundancy can mitigate this risk to an extent, but can also carry forward some types of data corruption.

  InterSystems IRIS mirroring, on the other hand, uses *logical data replication* between fully independent primary and backup storage, which eliminates the single point of failure problem entirely and avoids carrying forward most types of corruption.

  When using a solution other than mirroring, therefore, a single storage failure can be disastrous. For this reason, disk redundancy, database *journaling* as described in the "Journaling" chapter of the *Data Integrity Guide*, and good backup procedures, as described in the "Backup and Restore" chapter of the *Data Integrity Guide*, must always be part of your approach, as they are vital to mitigating the consequences of disk failure.

- InterSystems IRIS upgrades

  Many HA solutions allow for planned downtime of a given component system without interrupting overall availability. Most, however, require significant downtime to upgrade the production InterSystems IRIS instance.

  InterSystems IRIS mirroring, however, allows for minimum downtime InterSystems IRIS upgrade when application code, classes, and routines are kept in separate databases from application data. HA solutions other than mirroring, on the other hand, require careful planning of downtime windows for InterSystems IRIS upgrades, or any other maintenance requiring InterSystems IRIS shutdown.

2 No HA Solution

The structural and logical integrity of your InterSystems IRIS database is always protected from production system failure by the built-in features described in the "Introduction to Data Integrity" chapter of the *Data Integrity Guide*: write image journaling, database journaling, and transaction processing. With no HA solution in place, however, a failure can result in significant downtime, depending on the cause of the failure and your ability to isolate and resolve it. For many applications that are not business-critical, this risk may be acceptable.

Customers that adopt this approach share the following traits:

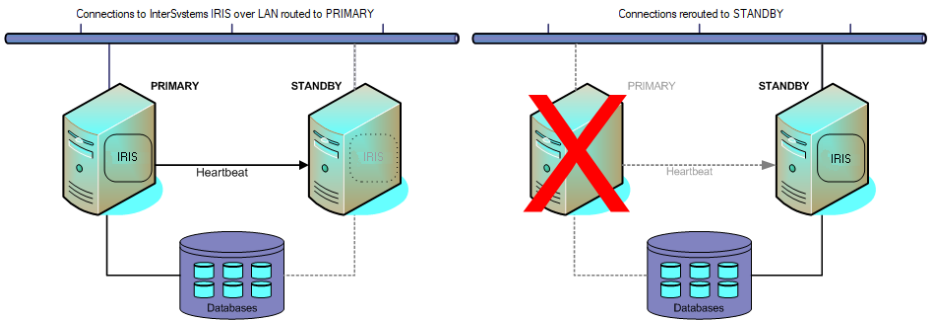
- Clear and detailed operational recovery procedures, including journaling and backup and restore.
- Disk redundancy (RAID and/or disk mirroring).
- Ability to replace hardware quickly.
- 24/7 maintenance contracts with all vendors.
- Management acceptance and application user tolerance of moderate downtime caused by failures.

3 OS-Level Cluster HA

A common HA solution provided at the operating system level is the *failover cluster*, in which the primary production system is supplemented by a (typically identical) standby system, with shared storage and a cluster IP address that follows the active member. In the event of a production system failure, the standby assumes the production workload, taking over the programs and services formerly running on the failed primary. The standby must be capable of handling normal production workloads for as long as it may take to restore the failed primary. Optionally, the standby can become the primary, with the failed primary becoming the standby once it is restored.

InterSystems IRIS is designed to easily integrate with the failover cluster technologies of supported platforms (as described in *InterSystems Supported Platforms*). An InterSystems IRIS instance is installed on the cluster's shared storage device so that both cluster members recognize it, then added to the cluster configuration so it will be restarted automatically on the standby as part of failover. On restart after failover, the system automatically performs the normal startup recovery, maintaining structural and logical integrity exactly as if InterSystems IRIS had been restarted on the failed system. Multiple InterSystems IRIS instances can be installed on a single cluster if desired.

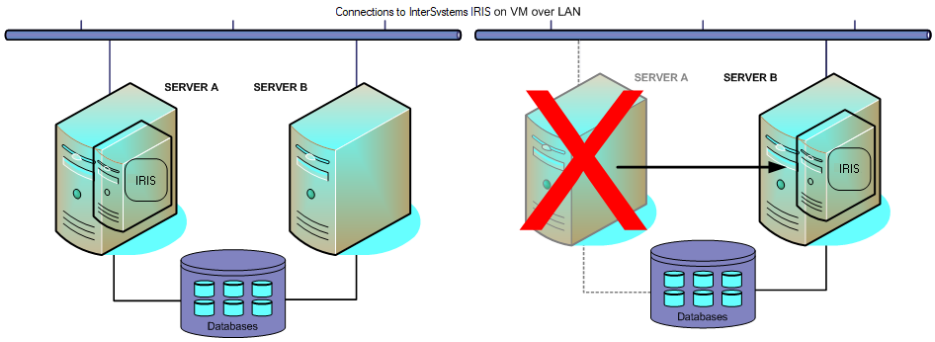
Figure 1: Failover Cluster Configuration



4 Virtualization Platform HA

Virtualization platforms generally provide HA capabilities, which typically monitor the status of both the guest operating system and the hardware it is running on. On the failure of either, the virtualization platform automatically restarts the failed virtual machine, on alternate hardware as needed. When the InterSystems IRIS instance restarts, it automatically performs the normal startup recovery, maintaining structural and logical integrity as if InterSystems IRIS had been restarted on a physical server.

Figure 2: Failover in a Virtual Environment



Virtualization HA has the advantage of being built into the virtualization platform infrastructure, and thus can require very little effort to configure, in some cases none at all. In addition, virtualization platforms allow the planned relocation of virtual machines to alternate hardware for maintenance purposes, enabling upgrade of physical servers, for example, without any downtime.

5 InterSystems IRIS Mirroring

InterSystems IRIS mirroring with automatic failover takes a different approach to HA, relying on logical data replication between fully independent systems to avoid the single point of failure risk of shared storage and ensure that production can immediately fail over to an alternate InterSystems IRIS instance in almost all failure scenarios—system, storage, and network.

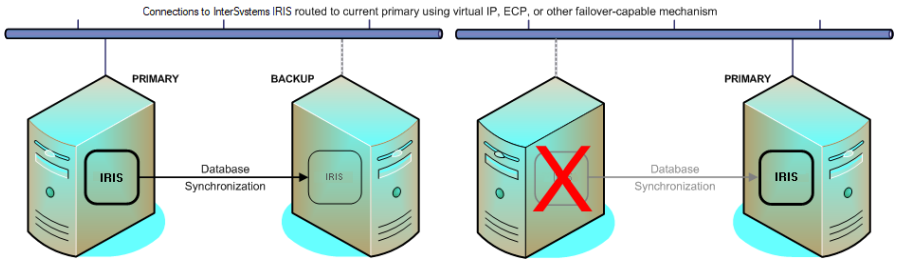
In an InterSystems IRIS mirror, one InterSystems IRIS instance, called the *primary failover member*, provides access to the production databases. Another instance on a separate host, called the *backup failover member*, communicates synchronously with the primary, retrieving its journal records, acknowledging their receipt, and applying them to its own copies of the same databases. In this way, both the primary and the backup always know whether the backup has the most recent journal files from the primary, and can therefore precisely synchronize its databases with those on the primary.

When this is the case, the mirror can quickly and automatically fail over to the backup in the event of a primary outage with no loss of data. A third system, the *arbiter*, helps the backup determine whether it should take over when the primary becomes unresponsive. A mechanism such as a virtual IP address shared by the failover members or a distributed cache cluster redirects application connections to the new primary. The failover process takes just seconds; many users won't even notice it happening. And because the backup has its own copies of the databases, even a total failure of the primary and its storage does not render the databases unavailable. In fact, even when the backup is missing the most recent journal data, the backup's *mirror agent* can retrieve it from the primary's host, if it is still online.

Once you restore the former primary to operation following failover, it becomes the backup and its databases quickly catch up with those on the new primary, returning the mirror to full HA capability. You can then return the systems to their original roles or maintain the new arrangement.

Mirrors can also include *disaster recovery (DR) async* members, which are asynchronously maintained copies of the primary; a DR async can be promoted to failover member, for example to become backup when a failed primary cannot be restored to operation quickly, or (if physically separate) for disaster recovery when an outage, such as a data center failure, brings down both failover members. Finally, mirrors can contain *reporting async* members, which maintain asynchronous copies of the production databases for business intelligence and data warehousing purposes.

Figure 3: InterSystems IRIS Mirror



Mirroring can also be used together with virtualization platform HA to create a hybrid HA approach, under which the virtualization platform responds to unplanned system or OS-level outages while mirroring handles all planned outages and unplanned database outages (including both InterSystems IRIS outages and storage failures) through automatic failover.

For complete information about InterSystems IRIS mirroring, including its DR capabilities, see the "Mirroring" chapter of the *High Availability Guide*.

6 HA Solutions Feature Comparison

The following table provides a very general feature comparison mirroring, clustering, and virtualization as HA solutions.

Feature	InterSystems IRIS Mirroring	OS-Level Clustering	Virtualization Platform HA
Failover after machine power loss or crash	Handles machine failure seamlessly.	Handles machine failure seamlessly.	Handles physical and virtual machine failures seamlessly.
Protection from storage failure and data corruption	Built-in replication protects against storage failure; logical replication avoids carrying forward most types of corruption.	Relies on shared storage device, so failure is disastrous; storage-level redundancy optional, but can carry forward some types of corruption.	Relies on shared storage device, so failure is disastrous; storage-level redundancy optional, but can carry forward some types of corruption.
Failover after InterSystems IRIS shutdown, hang, or crash	Rapid detection and failover is built in.	Can be configured to fail over after InterSystems IRIS outage.	Can be configured to fail over after InterSystems IRIS outage.
InterSystems IRIS upgrades	Allows for minimum- downtime InterSystems IRIS upgrades.*	InterSystems IRIS upgrades require downtime.	InterSystems IRIS upgrades require downtime.

Feature	InterSystems IRIS Mirroring	OS-Level Clustering	Virtualization Platform HA
Application mean time to recovery	Failover time is typically seconds.	Failover time can be minutes.	Failover time can be minutes.
External file synchronization	Only databases are replicated; external files need external solution.	All files are available to both nodes.	All files available after failover.

* Requires a configuration in which application code, classes, and routines are kept in databases separate from those that contain application data

7 Using Distributed Caching with a Failover Strategy

Whatever approach you take to HA, a distributed cache cluster based on the Enterprise Cache Protocol (ECP) can be used to provide a layer of insulation between the users and the database server. Users remain connected to the cluster's application servers when the data server fails; user sessions and automatic transactions that are actively accessing data during the outage pause until the data server becomes available again through completion of failover or restart of the failed system.

Bear in mind, however, that adding distributed caching to your HA strategy can increase complexity and introduce additional points of failure.

For information about distributed caching, see "Horizontally Scaling Systems for User Volume with InterSystems Distributed Caching" in the *Scalability Guide*.