

An Introduction to High Availability for Power Systems Running IBM i

Introduction

Every company faces critical hours when system downtime is unwelcome—whether it's planned or unplanned. One company's important hours might only be from 9-to-5, while for another it's 24x7. Increasingly, shops that were able to accommodate some periods of downtime for backups and system maintenance are finding that this window is quickly shrinking, or has disappeared altogether, because of increased system demands. Because of the need to keep systems available for increasing amounts of time, companies are realizing that a system failure or a site disaster would create an enormous disruption and expense, especially if it went on for longer than a few hours. For many companies, exposure to this amount of potential downtime has become unacceptable. Shops that thought they weren't candidates for a high availability (HA) solution are now realizing an urgent need to start looking at their options.

This white paper provides an introduction to High Availability for Power Systems running IBM i. It is for companies that want to understand this technology and evaluate whether such a solution can become a cost-justifiable component of their backup and recovery strategy.

One caveat: Depending on the size and complexity of your information systems, there are many other factors to consider besides the installation of an HA solution when trying to reduce your vulnerability to planned and unplanned downtime. HA clearly is a significant component in an overall data recovery/system availability strategy, but it often takes a variety of software and even hardware components to provide maximum protection against all exposures to downtime.

Before looking at the details of high availability, let's take a quick look at the cost of downtime and some of the primary disaster recovery strategies that are used to mitigate this cost.

The Cost of Downtime

Management is often amazed when they total all the direct and indirect costs of downtime for their company. At first, they may assume that if a system is down for several hours or even a day, it is certainly a big inconvenience and a tolerable risk—as long this kind of downtime is a rare occurrence. However, once numbers are plugged into the following rule-of-thumb formula, they are often shocked.

Take the average sales lost during an hour of system downtime during business hours, then add the total hourly wage (including benefits) of all employees that are idle during that hour of downtime. Now multiply this figure by the estimated number of hours of system downtime during a year. Finally, multiply the result by two to take into account the costs of this lost employee productivity, lost business reputation, and lost business—both now and in the future—from your lost customers.



Unplanned Downtime vs. Planned Downtime

The IBM Power Systems server running IBM i is considered to be one of the most reliable business systems in the industry. IBM Technical Document 22053139 pins Mean Time Between Failure (MTBF) for this system at greater than 100,000 hours. This statistic only addresses the likelihood of a server failure and does not take into consideration problems that might occur outside the box.

Other major contributors to downtime include:

- Power Failures
- Network Failures
- Site Damage
- Human Errors
- Malfeasance

Another sobering trend to consider is this: Courts are increasingly holding firms liable for losses caused by computer failures. According to Disaster Recovery Journal, litigation is becoming increasingly common and companies are feeling the financial strain of defending themselves against corporate lawsuits.

The rising cost of downtime has caused many companies to consider HA that may not have considered it before.

Despite the potentially dire consequences of unplanned downtime, less than 10% of all downtime can be attributed to unplanned events, and only a fraction of that is due to a site disaster. The other 90+%—the kind that companies face on a regular basis—are those caused by system maintenance tasks, including:

- Data backups (nightly, weekly, and monthly saves)
- Reorganization of files—to reclaim disk space and improve performance
- Vendor software upgrades & data conversions
- IBM software release upgrades & PTFs
- New application software installations
- Hardware upgrades
- System migrations



Regardless of the cause of downtime, what matters most is reducing or eliminating the risk of downtime during critical hours of operation.

Disaster Recovery Strategies

Tape backup devices are still widely used by small to medium businesses (SMB). Tape-based strategies usually include periodic saves of the entire system, daily incremental tape saves of changed or otherwise critical data, and then storing these tapes safely offsite. Because of the legendary reliability of IBM i-based Power Systems servers, most companies think that this is sufficient. However, if a failure occurs that requires reloading entire applications from tape, it is not unusual for the data recovery time to be up to 48 hours or longer, depending on the time it takes to repair or replace hardware, restore data from tape, and manually recreate all transactions since the last good tape save. And keep in mind that it is not unusual to run into media errors when restoring from tape.

There are a growing number of companies who have put a pencil to the real cost of this downtime and as a result have introduced additional layers of protection to reduce data recovery time. Many options exist to reduce the recovery time; some of these include:

Journaling – An IBM i process that efficiently monitors any change made to data and objects. In the event of a system failure, data can often be recreated without the need to manually re-key it. More details about journaling later in this white paper.

Disk Protection – Installing disk drives (DASD) that perform parity protection or disk mirroring to help prevent the chance of data loss in the event of a disk drive failure.

Downtime from planned system maintenance events cause IT shops the most pain.

Cloud Recovery Services – Protected third-party cloud-based recovery sites, available on a subscription basis, where data changes made between tape saves are transmitted (data vaulting), and/or backup tapes are restored on a comparably configured system, which acts as a backup system after the loss or failure of a production system.

High Availability (HA) – True HA on Power Systems running IBM i consists of designating a second machine as a backup system, enabling communication between these two systems, then implementing programs that replicate all changes to critical objects on the production system to a backup system.

If a failure or system maintenance event occurs, users are moved to this second ‘mirror’ machine where they can resume business without the loss of data. In general, high availability provides the most efficient way to mitigate most planned and unplanned downtime events.

The Components of High Availability

Every HA solution has four primary components:

- 1) System-to-system communications
- 2) Data replication processes
- 3) System monitoring functions
- 4) Role swapping capabilities (moving from the production system to the backup system)

An HA solution configures, automates, and monitors many standalone technologies that are available on IBM i.

System-to-System Communications

The first step of the HA process is to establish communications between your production and backup Power Systems servers. Existing LANs and WANs are an easy way for two machines to communicate with each other, especially when moving large amounts of data between the machines.

Setting up TCP/IP communications is fairly simple, but the challenge comes in determining the amount of bandwidth required between the two systems to handle the volume of data to be replicated.

Certainly, you can have the second machine in the same location as the first and then directly connect the two machines; however, you may decide to locate a second machine on another floor or in another building, which adds a significant disaster-recovery advantage.

If you are buying a second machine just for the purpose of high availability, there is a real advantage in locating it in another building across town or across the country. By doing this, if a site disaster occurs in the building where the primary machine is located, the second machine will not be vulnerable to the same disaster.

Data Replication Engine

Once communication is established between systems, the next component needed is an engine that replicates or mirrors transactions between the production and the backup machines, and does it as closely to real time as possible. All data replication engines in IBM i-based Power Systems server HA solutions use the journaling function of IBM i to monitor for changes to data and move those changes to the backup environment. All HA solutions harvest journal entries during the data replication process; however, HA solutions either harvest the journal entries from the production system or from the backup system. HA solutions that harvest from the production system use their own proprietary process to harvest and send these journal entries to the backup system (Figure 1). HA solutions that harvest from the backup system use a process within the operating system called remote journaling which takes care of the transmission of journal entries from the production system to the backup system.

It is important to note that prior to beginning an ongoing replication process, the objects that are to be replicated need to be copied first to the backup system. In other words, if you intend to mirror the data from your ERP solution to a backup system, you need to make a copy of all of the application's data file objects, and restore them on the backup system to establish a baseline. In fact, if you intend to run your applications on your backup system in the event of a system failure, or during planned maintenance, you will also need a current copy of all of the application's objects on the backup machine. Of course, if the application is from a third-party, this may require a separate license. Most software vendors grant additional licenses at no extra charge for this specific purpose.



The amount of time required to fully recover a system from tape is far greater than most people expect.

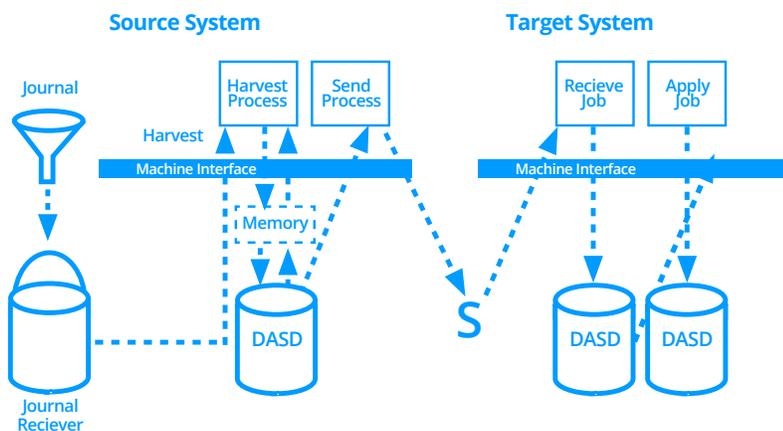
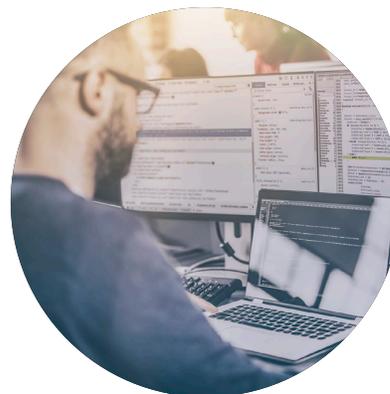


Figure 1: Diagram of an HA topology that harvests data using local journaling.

Remote Journaling in a Data Replication Engine

The remote journaling function of the IBM i operating system transmits and writes—at very high speeds—an identical copy of a journal entry to a duplicate journal receiver on another connected system. When used as the engine for replicating data in a high availability solution, remote journaling works very efficiently since this process occurs at the level of the operating system—beneath the machine interface.



In contrast, a replication engine that harvests journal entries on the production system must have its own process to transmit the journal entries, which typically happens less efficiently because several processes need to occur outside of the operating system.

The following diagram (Figure 2) illustrates the remote journaling process, showing that as changes are made to application data, journaling detects these changes on the production system, and as journal entries are made, remote journaling automatically replicates and transmits each journal entry to an identical journal receiver on the target (backup) system.

Once the journal entry lands in the journal receiver on the target system, a process within the HA software harvests the journal entry, validates the data, and then applies the changes to the data on the target system, thus bringing it current with the source system.

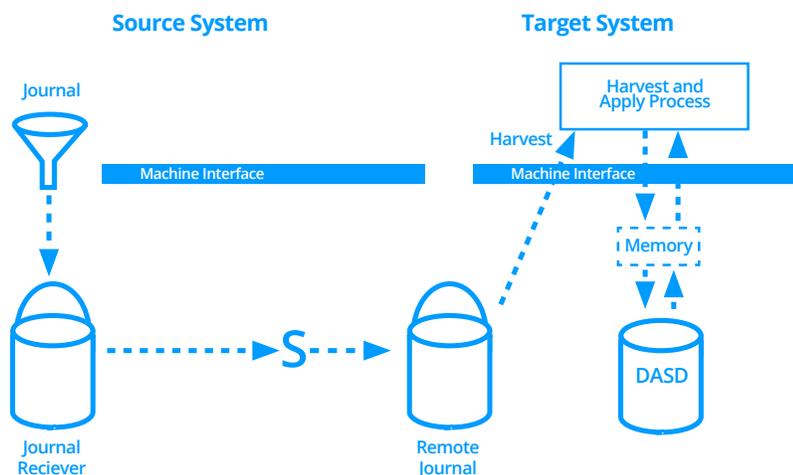


Figure 2: An HA solution must not only mirror objects that can be journaled, but must also be able to detect and replicate changes to non-journaled objects.

Taking a Closer Look at the Remote Journaling Process

When you enable journaling for an object, you essentially initiate a process that 'watches' the object. Journaling consists of two objects: the 'journal' and the 'journal receiver.' When any change occurs to the object that the journal is 'watching,' the journal writes everything about this change in a very efficient way in the journal receiver. Each change that is recorded is called a 'journal entry.' As journal entries pile up in the journal receiver, once the receiver has a predetermined number of journal entries, the journal receiver is 'changed' and a new, empty journal receiver is then associated with the journal. One of the main reasons that the journal receiver is changed is to make groups of journaled data available to be saved offline for later restoration, if needed.

When journaling is used with tape saves in a backup and recovery strategy, if a system failure occurs between tape saves, the journal receivers that were saved to tape can be restored and the journal entries within each can be retrieved and 'applied' to the data in the file. This reintegrates the data changes recorded in the journal entries with the data file, which restores the data in that file nearly back to the state at the point of failure.

The high availability replication process uses journaling in a little different way by sending journal entries to the backup system, which are applied as quickly as possible to duplicate copies of the objects to keep them current with the production system.

Keep in mind that real-time mirroring of changes to objects by any kind of logical HA solution can only be done if the object can be journaled. Currently this includes data files, IFS, data areas and data queues. However, an HA solution must be able to keep other system-critical objects updated on the backup system, including: program objects, spool files, user profiles, device configurations, etc. Typically, these kinds of objects are replicated using an object monitor-and-copy process.

It is also important that any object replication process, whether it uses remote journaling or not, should be able to continue object replication even if an object is renamed or moved, and it should never stop or slow the ongoing replication of objects if an object needs resynchronization with the production system for some reason.

Remote journaling also provides a very efficient, built-in data auditing mechanism that ensures data integrity during the replication process without some of the auditing processes that are required when remote journaling is not used. And keep in mind that with remote journaling, most of the processing overhead occurs on the backup system.



The amount of bandwidth needed for HA depends on the amount of data that is replicated.

Data can be replicated between systems either synchronously or asynchronously. If data is sent synchronously, then control is not returned to the job (that made the data change) on the production node until it is confirmed that the data has been received on the backup node. This of course can negatively impact system performance; however, the criticality of your data may justify this lag time, or it may justify the purchase of a faster machine and/or more bandwidth. Of course, if data is sent asynchronously, programs will continue to run even if journal entries have not been received on the backup.

If your needs require synchronous replication, then it is important that you choose an HA solution that uses remote journaling because only remote journaling can truly handle a synchronous replication process.

For detailed information on the benefits provided by remote journaling in HA solutions, see Chapter 6 of the IBM Redbook: Striving for Optimal Journal Performance on DB2 Universal Database for System i (www.ibm.com/redbooks).

System Monitor

Once data replication processes are in place between systems, you need a mechanism to continuously monitor these processes. As you can imagine, thousands of transactions could be replicated each day—millions in a heavily used application. If there is a glitch in the system-to-system communications, the journaling components, or the journal entry apply processes, one or more objects can easily lose synchronization thereby jeopardizing the data integrity on the backup system.

It is critical, therefore, to have a monitoring process in place that ensures replication integrity; otherwise, in the event of a failure, your ability to reliably use the backup system could be compromised.

A useful monitoring process continuously shows the status of all critical components, and does so in an easy-to-understand format. If any problem arises, it should be apparent on the monitoring screens (or even page a system operator if the severity warrants) and then automatically attempt to correct the problem.

Within the computing world, self-healing capabilities are becoming increasingly common. Self-healing capabilities are crucial to reliability and ease of use for an HA solution for IBM-i based Power Systems servers. For instance, an HA monitor should automatically determine if an object on the backup system is out of synchronization with the same object on the production system. If so, the monitor should self-initiate



the process of re-synchronizing the object by recopying that object from the production machine to the backup and applying all necessary journal entries to bring it current. And it should do this without halting or slowing the ongoing replication of other objects.

Finally, the ability to manage your entire HA environment from any device with a browser, at-a-glance monitoring, intuitive control, and single point monitoring and control should be must-have features. Automatic self-correcting functionality that can resolve most problems as they arise is also a bonus. Together, these features will reduce the time needed to maintain your environment to minutes a day.



Switch-Over

All of the functions of an HA solution described up to this point exist primarily to minimize planned and unplanned downtime by quickly making a fully synchronized, fully functional backup system available to users. If the HA system cannot consistently and efficiently provide this capability, then it is only of limited value.

The process of moving users to a backup system is often called a switch-over because the backup system essentially takes on the role of your production system during the time your actual production system is being maintained or repaired.

It is vital that once you have the components of data replication and system monitoring in place, that you regularly test the switch-over process to verify smooth execution of the process and the integrity of the data on the backup system.

A switch-over generally includes the following processes:

- Monitoring that all objects are currently synchronized between the two systems
- Ending all user and application jobs on the production system
- Ending the replication and monitoring jobs on the production system
- Designating the backup environment as the production environment
- Starting the replication and monitoring jobs on the backup system
- Starting user and application jobs on the backup system

Of course, once you have executed the switch-over, when you are ready to return to the production system, you will need to reverse the process.

The first time you attempt a switch-over, it is not unusual to have to spend extra time to work out the 'kinks' in communications, system addressing, and the ending and restarting of user jobs, interfaces and HA components. This is normal as the requirements of every system are unique due to different types of objects being replicated, as well as the relationship of jobs, objects, applications and interfaces on the particular system.

Keep in mind that a good high availability system will be able to keep a variety of objects replicated in near real time—not just data. Again, you should be able to replicate user profiles, device configurations, spool files, IFS and any other necessary objects. In order to have a successful role-swap, all necessary components must not only exist on the backup system, but they must be current.

The switch-over process of your high availability solution should have sufficient automation built into it so that during a controlled switch-over, most—if not all—components needed for the backup system to assume the role of the production system are automatically activated. This includes all system addressing, as well as all replication and monitoring jobs. If everything is working properly and the process is fine-tuned, your users shouldn't have to wait long before they see a sign-in screen. As you can see, the switch-over process is another area where selfhealing functions are a critical component of an efficient high availability solution.



Evaluating High Availability Solution Vendors

Now that you have an understanding of the need for HA as well as a fundamental understanding of the necessary components of an HA solution for IBM i-based Power Systems, you now have an idea of what to look for in a solution.

To recap, an HA solution requires:

- Efficient replication of objects in as close to real time as possible.
- An easy-to-use system monitor that not only makes it simple to see components that aren't functioning properly or objects that are not in synchronization, but also automatically resets components and corrects out-of-sync conditions. Additionally, the system monitoring process should only require minutes of operator attention each day.
- An easy-to-execute switch-over process that automates the processes of monitoring synchronization, ending necessary jobs on the production system, and starting all necessary jobs on the backup system. In addition to the above, vendors of high availability solutions should be able to provide you with names of customers who can confidently attest that they:
 - Perform regular, successful tests of the switch-over process—ideally on a monthly basis.
 - Have had their HA solution successfully 'fail-over' to the backup system as the result of a system failure on their production system.
 - Consistently experience a high level of support from the HA vendor. For a solution as critical as high availability, the quality of customer support should be one of the deciding factors when choosing a solution.

Any glitch in the system-to-system communications, the journaling components, or the journal entry apply processes will cause one or more objects to lose synchronization and jeopardize data integrity.

Summary

In today's world of ever-expanding hours of business operation and increasing reliance on data availability, it is becoming easier for companies of all sizes to justify the cost of an HA solution.

When evaluating HA, it's important that you understand its critical components, including how the solution fits into your particular environment, the degree of automation that is built into the system, and what it will require of your IT staff resources to manage the solution.

About Syncsort

Syncsort is the global leader in Big Iron to Big Data software. We organize data everywhere to keep the world working – the same data that powers machine learning, AI and predictive analytics. We use our decades of experience so that more than 7,000 customers, including 84 of the Fortune 100, can quickly extract value from their critical data anytime, anywhere. Our products provide a simple way to optimize, integrate, assure and advance data, helping to solve for the present and prepare for the future. Learn more at syncsort.com.

© 2018 Syncsort Incorporated. All rights reserved. All other company and product names used herein may be the trademarks of their respective companies.