Successful System Recovery using Ignite-UX

Table of Contents

Abstract

Ignite-UX for HP-UX addresses the need for system administrators to perform operating system installations, deployment, and recovery, often on a large scale. It provides the means for creating and reusing standard operating system configurations. Additionally, Ignite-UX delivers the ability to archive operating system configurations and to use these archives to replicate systems, with the added benefit of speeding up the process. Ignite-UX also permits various customizations, and is capable of both interactive and unattended operating modes.

Executive summary

As a systems administrator developing a disaster recovery strategy, you have to determine how often recovery archives should be created, how long to retain them, and what media is most appropriate: tape (using `make_tape_recovery`) or network (using `make_net_recovery`). Unfortunately, these decisions are not straightforward and you must consider a variety of issues unique to your environment and circumstances.

The information contained in this white paper is intended to aid you in developing a recovery archive strategy. The issues discussed are related so you should read the document in its entirety before coming to any decision. You then need to determine the greatest impacts on a system-by-system (or application-by-application) basis and use that to shape your strategy.

Recovery archive creation considerations

Typically, a recovery archive is created when the system is quiescent and can be executed in single-user mode. This is, however, not a requirement of Ignite-UX, but rather, the choice of the system administrator. In addition, the creation of a recovery archive depends on how it is integrated into your backup strategy.

For more information on how to create recovery archives in single-user mode, refer to *make_tape_recovery*(1M) and *make_net_recovery*(1M).

CAUTION:
HP does not recommend the use of any component of Ignite-UX as part of any backup strategy. The Ignite-UX product is not an enterprise backup utility and therefore does not provide any of the features associated with Enterprise Backup and Recovery utilities such as HP StorageWorks Data Protector (for example, the recovery of individual files).

Ignite-UX is intended for the cloning, recovery, and installation of systems only. Because recovery archives can be created when applications are executing, any application data[1] included in a recovery archive should be treated with suspicion as it could be inconsistent.

[1] The term "application data" is used extensively in this white paper. In this context, it means all files (special or regular) and their contents, directories, permissions, access control lists, and data contained within raw devices (if used) that are used by the system. In other words, all files not associated with the HP-UX operating system are considered application data.

Application backups allow you to recover application file systems and data. Typically, they are scheduled at times when the application is down or quiescent, or in a state where a valid backup can be executed effectively. Your application backup strategy can vary; for example you could run daily full[2] backups, or weekly full backups and daily incremental[3] backups, or a full weekly backup with the database down, coupled with daily multiple backups of the database log files for replay in the event of a complete system recovery.

The following topics are discussed in this paper:

- Documentation
- Rapidity of change
- Creation frequency
- Retention
- Budgetary and other constraints
- Service level agreements
- Testing recovery archives

Documentation

Analyzing the technical documentation that is created when changes occur in your environment is very valuable in deciding how often you should create a recovery archive. Ask yourself whether you will be able to replicate all the changes that have been applied to a system. Critically examine the documentation that is produced with each change and decide whether the changes that were made can be replicated from the documentation. Keep in mind that you can give rise to major problems if you decide now that changes can be replicated, but then discover later that some cannot because the quality of parts of the documentation is poor and therefore not repeatable.

If your change process produces high-quality technical documentation, you may be able to increase the interval between the creation of recovery archives. You should consider, however, the resources required to reapply changes when you recover a system. If you do not have high-quality documentation, or if the resources required to use such documentation are prohibitive in your environment, you must then set the period between recovery archive creation ideally to one week or at most a few weeks to minimize the number of changes that may be lost during a recovery.

Rapidity of change

Systems that are considered to be in a maintenance mode and have very few changes implemented may be able to span many months between generations of recovery archives.

Those systems that are more dynamic may require the creation of recovery archives in a much shorter timeframe: days or, at most, weeks, depending on the changes that are implemented.

Most systems fit somewhere in between these two extremes with periods of large change followed by quiet periods: for example, application, database or operating system upgrades followed by longer periods of relative quiet during which the system only undergoes maintenance of the application and periodic planned patching of the operating system.

[2] All data associated with the application are backed up.

[3] Files that have changed since the last full backup are saved into the backup. A restoration of application data involving incremental backups would require you to recover the last full backup of the application and then the latest incremental backup. There can be varied levels of incremental backup and the previous comments discuss only one level of incremental backup.

The three categories of broadly different systems are as follows:

- Sunrise — those systems that are just starting out the use life-cycle and are undergoing rapid change. These systems probably require the most frequent recovery archives creation: every few days to at most a few weeks. Systems in this category are usually pre-production or test, or have just entered production and are being closely monitored.
- Mid-life — those systems that are in the middle of the use life-cycle, are past the initial intense implementation effort, and are mostly stable, though may still undergo significant change. The majority of these systems probably require fixed, periodic recovery tapes to be created about every month or two. Additionally, *ad hoc* recovery archives should be created before and after major system changes to facilitate a back-out strategy in the event that the changes are unstable and must be removed; this includes major patch changes to the operating system. The creation of *post hoc* recovery archives also allows you to recover to a known condition in the event of a major system outage after any major system changes.
- Sunset — those systems that are at the end of the use-life cycle and rarely change. These systems require fewer recovery archives to be created: about every three to six months, with *ad hoc* recovery archives created before and after significant changes.

Creation frequency

The following table summarizes general recommendations regarding how often recovery archives should be created based on the quality[4] of your change process documentation in relation to each system use life-cycle category.

Table 1

Quality of Change Process Documentation	Frequency for Sunrise Systems	Frequency for Mid-life Systems	Frequency for Sunset Systems
Non-existent (0%[5])	Weekly or more often	Every 2 weeks	Monthly
Poor (33%)	Weekly or more often	Every 2 weeks	Monthly
Average (50%)	Weekly or more often	Monthly	Bi-Monthly
Excellent (80%)	Weekly	Monthly	Quarterly
Complete (100%)	Weekly	Bi-Monthly	Semi-Annually

Important:
Table 1 is intended as a guide only and should be adjusted to suit your business needs for recovery (business resumption planning). There may be other factors that reduce the above time periods: for example, a lack of resources to reapply changes after a recovery, which in a sunrise system may mean creating recovery archives every 1 to 2 days if the change process is dynamic.

[4] Remember that quality is measured by the ability to locate all of the changes that have been effected on a system in a given period of time, as well as the ability to replicate these changes using the change process documentation.

[5] This is the percentage of changes that could be replicated from change process documentation.

Retention

Typically, you should retain recovery archives using similar retention periods as those used for your application backups. This is because there may be a time delay between finding a problem and needing to recover a system to a known state.

Retention periods are best explained with the following example.

Example

- – Weekly recovery archives may have a retention period of 4 weeks.
- – Every 4th recovery archive (monthly) may have a retention period of 6 months.
- – Every 12th (quarterly) recovery archive may have a retention period of one year.
- – Every 24th (semi-annual) recovery archive may have a retention period of two years.

To clarify this example further, Table 2 provides a pictorial view of these retention periods.

Table 2

Week #	1 month	6 months	1 year	2 years
1	▨			
2	▨			
3	▨			
4	▨	▨		
5	▨			
6	▨			
7	▨			
8	▨	▨		
9	▨			
10	▨			
11	▨			
12	▨	▨	▨	
13	▨			
14	▨			
15	▨			
16	▨	▨		
17	▨			
18	▨			
19	▨			
20	▨	▨		
21	▨			
22	▨			
23	▨			
24	▨	▨	▨	▨

This example provides for multiple generations of recovery archives. At the end of 24 weeks, you have recovery archives that are 1-, 2-, 3-, 4-, 8-, 12-, 16-, 20-, and 24-weeks old. Continuing the retention cycle forward another 24 weeks, you have recovery archives that are 1-, 2-, 3-, 4-, 8-, 12-, 16-, 20-, 24-, 36- and 48-weeks old (excluding any *ad hoc* recovery archives). Older generations of recovery archives become important if there is a problem with more recent recovery archives whose use prevents you from recovering a system.

This introduces *ad hoc* (created before and after major system changes) recovery archives should have a retention period of 3-6 months. When choosing a retention period, you must choose one that supports your business objectives.

The following issues for network recovery archives:

- Typically, you may keep two recovery archives for a given system. This means that your backup cycle for the Ignite-UX server (or other location where the recovery archives reside) must allow for the retention of recovery archives that follows the retention cycle you have defined.
- If the recovery archives are not kept on the Ignite-UX server, the system on which they do reside must have its backups synchronized with the Ignite-UX server. It is pointless to create a recovery archive unless you can recover the archive configuration onto the Ignite-UX server, as well as the matching network recovery archive to the place where it is kept.

Retention periods for recovery archive tapes are simple since the tapes can be managed just as any other tape in your tape retention strategy.

Caution:
You should review the Ignite-UX White Paper, *Successful System Cloning using Ignite-UX*, for issues that could prevent you from successfully recovering systems using recovery archives (tape or network).

Budgetary and other constraints

There are additional constraints that further determine the solutions that are possible for recovering your systems. In general, the constraints are budgetary rather than technical. These constraints and your business requirements dictate the solutions that are feasible and most effective for your environment. The topics that follow describe some of the constraints that you may encounter.

Disk-space cost

The issue of disk-space covers several different areas and assumes that you are using `make_net_recovery` to create your recovery archives. First, there is the cost of disk space dedicated to recovery. Second, there is the amount of disk space that is available per system for recovery archives. Lastly, there is the amount of recovery archives required per system.

Suppose, for example, that you have 100 systems connected to an Ignite-UX server for recovery purposes and you need to save six recovery archives for each of the systems. Assuming that the average size of the recovery archives is 2.5 GB, approximately 1500 GB of disk space is required for recovery archive storage on the Ignite-UX server. If you are only able to purchase 300 GB for recovery archive storage, then you have a dilemma: some systems may be limited to only one recovery archive while other, more important systems will have multiple archives.

A possible hybrid solution this dilemma would be to use both the network and tape for recovery archive creation and storage.

The cost of the disk space is not the only cost. Due diligence dictates that the recovery archives must also be saved onto tape regularly so that they can be stored at an off-site location to allow for the recovery of the systems and/or the archives if the Ignite-UX server is lost or the site is lost (the archives can be recovered at another site). This means that even using `make_net_recovery` shares most, if not all, of the same costs as those described in the Tape cost topic.

A tape retention cycle that adheres to the appropriate retention periods for a potentially large number of recovery archives on tape requires a great deal of disk space. In addition, it requires fast, high-capacity tape drives to backup the data. These backups call for a retention cycle that matches your recovery requirements, and you must have a method to easily determine the contents of each tape, including the recovery archives stored on each.

Tape cost

The cost of tape-per-megabyte is much less than the cost of disk space. Furthermore, it allows for off-site storage to provide a recovery solution in a loss-of-site disaster. There are cost factors involved in tape use that are not limited to the cost of the tapes.

Consider the following potential costs:

- Each system must have its own tape drive because `make_tape_recovery` cannot access tape drives connected to another system.
- It is not possible to boot from Fibre Channel tape drives so writing a recovery archive to one means you must understand how to recover a system using dual-media recovery. (For more information, refer to the "Tape Recovery with No Tape Boot Support" section of the *Ignite-UX Administrators Guide.*) This would remove the need to have a tape drive per system.
- It is not possible to boot a virtual partition (vPar) from a tape drive, which means you have a choice to make in a vPars environment when using tape. Creating recovery archives for vPars on tape requires the tapes to be recovered sequentially[6].
- There is the (much smaller) cost of cleaning tapes for tape drives that require them.
- There is the cost of someone changing tapes if tape drives are attached to individual systems.
- The cost of offsite storage for the tapes, as well as the retrieval of the tapes in an emergency, must be considered.
- There is the cost of local storage for the tapes.
- There is the recovery speed: faster tape drives are quicker but they cost more (as does the media). You need to balance the cost of the drive and media versus the speed of recovery[7].

Tape recovery has different costs than network recovery. Some of the costs are not strictly related to tape recovery because if the tape drives are used for other things, the cost of cleaning tapes and someone changing tapes may already be factored into the cost of operating the systems.

Assuming that each of 100 systems already had a DDS5 tape drive, that a recovery archive fits onto one tape, that each tape cost US$4.31[8], and that there were at most six recovery archives kept on tape, then the per system cost would be 100 x 6 x US$4.31 = US$2586.00 (excluding

[6] If you have most systems creating network recovery archives to one vPar and create a recovery tape for that vPar, you can then recover it from tape, recover the latest recovery archives from the tape, and then recover the other vPars systems. The alternative is to use an Ignite-UX server external to the vPars system.

[7] The speed described relates to more than just the speed of the tape drive. A slow, older system may not be able to perform stream reading from the tape drive, which causes the recovery to be very slow even though the tape drive may be very fast. You should test recovery times to ensure you identify the time needed to recover a system.

[8] This information was derived from the online purchase of DDS3 media (per tape) from the HP Small and Medium Business store on August 31, 2004, excluding the cost of shipping and any applicable sales tax.

any other cost). This is significantly lower than the cost of a disk array to hold the equivalent amount of data (1500 GB) on disk.

Network

When deciding whether you should use network or tape recovery, you need to review your network infrastructure.

For network recovery, you must have the available bandwidth on the Ignite-UX server to create recovery archives and recover systems without adversely affecting production applications. If your Ignite-UX server is shared with an application, you need to ensure that creating recovery archives or recovering systems does not affect the application, or you must understand that the performance of the application may be degraded while creating recovery archives or recovering systems.

When creating network recovery archives or when recovering systems while all of the network traffic goes through the one network interface on the Ignite-UX server, then the more concurrent recovery archives being created, the more the network interface becomes saturated. Ignite-UX does not place a large memory load[9] on a system, only potential network congestion issues.

It is possible that with a lot of systems creating recovery archives to the Ignite-UX server at the same time, the time required to create any one archive will be much higher than if you were to create a lower number of recovery archives concurrently. This is because the Network File System (NFS) traffic being sent to the Ignite-UX server may cause network congestion on the Ignite-UX server, or the Ignite-UX server may not be able to process the NFS traffic as fast as it is being sent. If you have time constraints when creating recovery archives (a fixed time window), you should monitor for this occurrence and take appropriate steps to ensure the time is not exceeded.

You should be aware that Ignite-UX commands do *not* control the routing of the traffic between the Ignite-UX server and client. The IP address or hostname that you specify as the Ignite-UX server controls this traffic. If the Ignite-UX server is multi-homed and you want to control the network interface to use, you can *only* control it with the Ignite-UX server name or IP address using the `–s` option of `make_net_recovery`. You should inspect the routing tables on the client system to see how the traffic will be routed from the client system to the Ignite-UX server. If you use the Ignite-UX Graphical User Interface (GUI) to create network recovery archives, you cannot give the hostname or IP address of the Ignite-UX server because the Ignite-UX GUI always gives the official hostname of the Ignite-UX server as an argument to the `–s` option.

It is important to understand that, with a multi-homed Ignite-UX server, if the client has to route traffic to a different subnet, it impacts other traffic flowing between the subnets. If the Ignite-UX server has no network interface on the same subnet as the client, you must provide a boot helper or perform dual-media recovery (booting from CD first and then contact an Ignite-UX server for recovery. For more information regarding either of these topics, refer to the *Ignite-UX Administrators Guide*.

An overloaded network infrastructure can be very expensive to replace or upgrade to technologies that provide more bandwidth (for example, using switches that support auto-port aggregation or replacing a 100Base-T backbone with Gigabit Ethernet).

[9] Low-end and uniprocessor systems may, however, see a large CPU impact on the system when creating a recovery archive over the network as the archive is, by default, `gzipped` when created.

Processing speed

You should be aware that when creating a recovery archive on lower-end and uniprocessor systems (for example: rx1600, rx2600, rp24xx, and rp3410 systems), a considerable amount of CPU time is required to perform Ignite-UX tasks.

When writing a network recovery archive, the system compresses the `tar` or `cpio` archive being written to the NFS file system using `gzip`. There is a considerable amount of CPU time spent compressing this data. The CPU overhead and the impact of NFS traffic on a system can slow other applications down considerably. On low-end and uniprocessor systems, it is not recommended that you create recovery archives while running production applications.

Higher-end systems are less adversely affected when creating recovery archives; however, systems at the edge of performance problems may experience performance problems.

Note:
You should take extreme care when running applications or other software on a system that requires a significant amount of resources while attempting to create a recovery archive at the same time.

Input/output (I/O) usage

When running `make_tape_recovery` or `make_net_recovery`, either command can create a significant amount of disk I/O. This can impact an application as the I/O is executed through the system's buffer cache. The I/O performed by Ignite-UX competes with other applications for the buffer cache, thus reducing the application's performance.

It is possible that in a situation of high I/O usage by the application, contention for physical I/O can be created between it and Ignite-UX. This is not a problem when there is no application data or programs in the root volume or disk group, unless the system is also paging to swap spaces in the root volume or disk group.

Creation time

If there is a fixed time to create a recovery archive, you should review the other topics discussed in this white paper that can affect the amount of time it takes to create a recovery archive.

Using past experience, you should decide what amount of time is reasonable to complete the creation of the recovery archive. If the actual creation time extends past this point, you should evaluate the other topics discussed in this white paper to determine which may be affecting your recovery archive creation. In addition, if the amount of data included in the recovery archive has greatly increased, the expected completion time should be adjusted accordingly.

Service level agreements

The most overriding requirement that you have is to adhere to the Service Level Agreement (SLA). Since part of a normal SLA is restoration time in the event of an outage, you can use this restoration time as the basis of your recovery strategy.

For example, if your SLA requires that the system be recovered within 12 hours, you must develop a recovery strategy that can be implemented in 12 hours or less.

Typically, the SLA is more important than the cost of implementing a recovery strategy because the cost of implementing the SLA should be included in the total, agreed-upon cost (in capital and recurring costs).

Testing recovery archives

It is critically important that recovery archives are tested when created, although it is not necessary to test every recovery archive. If your business resumption plan relies on recovery archives in any way, your plan cannot be considered completely tested until you test the recovery archives every quarter or half-year, depending on how important it is that the archive is recoverable.

To ensure you understand the issues that impact the success of system recovery, HP recommends you read the *Successful System Cloning using Ignite-UX* white paper prior to testing or creating any recovery archive.

Archive recovery essentials

The following three topics are essential to understanding archive recovery:

- Files that must not be restored from backup tapes
- Recovering your applications
- Operating system files to be recovered

Files that must not be restored from backup tapes

Operating systems files with the `is_volatile` SD-UX attribute set to `false` must never be recovered from application backup tapes, as based on the attribute's definition in *sd*(4):

```
is_volatile

     Defines whether the file can be modified or removed.
```

If set to `false`, the file is not intended to be removed or modified by the administrator of the system. The SD-UX product determines system configuration based upon when the recovery archive was created. If files or directories are out of sync with the system configuration, certain SD-UX patches may not install correctly, and it may not be possible to determine the cause of some problems (especially if mixed and incompatible patches are recovered from backup tapes onto the system).

You can use the `swlist` command with the option `–a is_volatile` to report the setting of `is_volatile`, which is only kept at the file level, `-l <file>`.

Files for which `is_volatile` are set to true are intended to be modified by the system administrator rather than by patching. These files can be safely restored from backup, for example `/etc/passwd`. You must careful when recovering file systems and ensure that you do not accidentally recover parts of HP-UX that should not be recovered from your backup tapes.

Recovering your applications

As part of your backup and recovery strategy, you should have already identified which files do or do not need to be recovered to get the applications operational after a full system recovery. This is extremely important when application files are placed into what are normally operating system directories such as `/usr/bin`, `/etc`, or `/usr/lib`.

If the recovery archive contains any application files, you should ignore these files. The data or program files belonging to the application are likely to have changed. Always recover your application programs and data using your application backup solution. You may consider excluding application programs and data from all recovery archives you create.

Important:
It is necessary that you have identified precisely which files comprise your applications, including any operating system configuration files that require modification to support the operation of the application.

Operating system files to be recovered

To recover the operating system, you would normally recover very specific files such as the following:

```
/etc/passwd
/etc/group
```

In addition, you would recover other configuration files that you may have modified since the last time that you created the recovery archive. If you are recovering applications from different systems on to one system, you must identify and resolve duplicates in the `/etc/passwd` and `/etc/group` files (ideally, you have standards for enforcing the uniqueness of a UID and GID in your environment). This is not relevant if you use a central repository for user information such as LDAP, NIS or NIS+ although in a business resumption situation you would need to get the central repository working before any applications would be operational.

You *would not* recover files such as the following:

`/etc/lvmtab`

`/etc/rc.config.d/*`

`/dev` or any subdirectories or files

`/etc/ioconfig` or `/stand/ioconfig`

After cloning a system to perform your business resumption plan and after changing the IP address of the system because it was at a different location, you would not recover any files from your backup tapes like `/etc/hosts` as it would contain old IP address information. If you know you have changes in `/etc/hosts` that must be merged into the file currently on the system, recover the file to a different directory, then make the changes manually. Similarly, if you had to change the hostname of the system during the recovery, you should avoid recovering files that contain the old hostname.

Summary

A recovery archive strategy is not complete until is has been effectively designed, tested and proven to operate successfully. The topics presented in this white paper for your consideration can aid you in this effort and provide useful guidelines that can be applied to your environment.

For more information

The following relevant documents are available online at the HP Technical Documentation Web site at http://www.docs.hp.com/:

Ignite-UX Administration Guide

Successful System Cloning using Ignite-UX White Paper

HP-UX 11i v[1|2|3] Installation and Update Guide

HP-UX 11i v[1|2|3] Release Notes

HP-UX System Administrator's Guide

Managing Systems and Workgroups: A Guide for HP-UX System Administrators

Some or all of these documents are available on the Instant Information media and in printed form.

Product information regarding Ignite-UX for HP-UX is available at the HP Software Depot at

http://www.docs.hp.com/en/IUX/

MPN 5992-2902, 04/2008