High Availability and Disaster Recovery Solutions for Perforce

This paper provides strategies for achieving high Perforce server availability and minimizing data loss in the event of a disaster.

Perforce Consulting
consulting@perforce.com
Contents

Introduction ........................................... 2
Defining Business Objectives ....................... 3
Configuring Hardware for High Availability and Disaster Recovery ............. 4
  Recommended Configurations. .......................... 4
    Basic Disaster Recovery ................................ 5
    Warm Standby plus Disaster Recovery .................. 5
    High Availability Only .................................. 6
    High Availability with Basic Disaster Recovery ........... 8
    High Availability with Disaster Recovery and No Data Loss ........... 9
    Extremely High Availability with Disaster Recovery
      and No Data Loss ....................................... 10
  Disk Volume Layouts for the Perforce Server ..................... 11
  Handling Perforce Failover .............................. 12
Additional Reading ..................................... 12
Glossary .................................................. 13
Introduction

To protect your business and to ensure that your users have ready access to your Perforce servers, you must plan a strategy for high availability and for recovery from a hardware failure or natural disaster that disables your Perforce installation.

To determine and implement high availability and disaster recovery solutions, you must first define your business objectives, then purchase and deploy the hardware that is required to achieve those objectives. This document outlines some options for Perforce installations. It provides general information on configurations that offer varying levels of availability while minimizing data loss resulting from hardware failures or natural disasters.

This document is not intended to replace a specific assessment of any given environment. Perforce Consulting is available to assist you with the design of high availability and disaster recovery solutions for your specific Perforce infrastructure.
Defining Business Objectives

To determine your objectives for disaster recovery and server availability, quantify your budget and the level of risk that your business can tolerate. During analysis, determine the following:

- **Recovery point**: the maximum amount of data loss that your business can tolerate because of an outage, specified in units of time. For example, you might determine that you must be able to recover, at a minimum, all work done through the preceding day.

- **Recovery time**: the maximum amount of Perforce downtime that your business can tolerate.

After determining the requirements for server availability and disaster recovery, you can determine the hardware and processes required to achieve your goals. The following sections describe the essentials and offer a range of options, from simple, low-cost approaches to more elaborate high-availability solutions.
Configuring Hardware for High Availability and Disaster Recovery

To ensure that you can recover well from a hardware failure or natural disaster that affects your Perforce installation, be sure to consider the following:

- **System configuration**: buy and configure server hardware so that downtime and data loss are minimized.
- **Failover**: if the server process or machine fails, how is a backup system activated?
- **Server data storage**: configure the server files to ensure that the Perforce server has optimal access to its files, while physically separating files so that data loss is minimized if volumes fail.

The following sections discuss these points in more detail.

**Recommended Configurations**

Routine hardware planning must cover the following:

- **Daily verification of backups**: ensures that backed-up data is usable. Use an email notification to inform system administrators of the status of the backup system.
- **Frequent verification of versioned files**: ensures that file history can be recovered. This verification is time- and CPU-intensive. To minimize its impact on server performance, use a redundant disaster recovery system instead of the primary server machine to perform verification.
- **Fast recovery from hardware failures**: determines what hardware will replace the primary machine if it cannot be repaired quickly.
- **Minimum downtime for backups**: plan regular backups that minimize the impact to your users.

Regardless of the size of your organization, there are two essentials for ensuring you can recover from severe problems:

- **Hardware redundancy**: Configure at least two identical Perforce server machines, a primary machine and a disaster recovery system.
- **Media backups**: institute backups to tape or another medium and ensure that the backups are reliable.

When your primary Perforce server machine is unavailable, you need to provide your users with access to a substitute machine that can run the Perforce server and has access to up-to-date copies of the Perforce server files (metadata, versioned files, and journal).

Make sure that the redundant server machines have the same hardware, operating systems, and file system. Use consistent file system names and, in Windows environments, consistent drive letters. The ideal redundant system is a dedicated machine running in a data center. For large and/or mission-critical Perforce installations, consider other options to provide high availability, such as custom cluster solutions.

The following sections describe five possible configurations, ranging from a low-cost approach that provides a basic level of assurance to a high-cost approach that assures maximum uptime with no loss of data in the event of hardware failure or natural disaster. The following table provides an overview of these configurations, which are detailed next.
<table>
<thead>
<tr>
<th>Description</th>
<th>Cost</th>
<th>Number of Servers Required</th>
<th>Data Loss</th>
<th>Down Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic disaster recovery</td>
<td>Low</td>
<td>2</td>
<td>Data modified after the last synchronization from the primary to the disaster recovery</td>
<td>Generally low</td>
</tr>
<tr>
<td>Warm standby plus disaster recovery</td>
<td>Low</td>
<td>3</td>
<td>Data modified after the last synchronization from the primary to the disaster recovery</td>
<td>Generally low</td>
</tr>
<tr>
<td>High availability only</td>
<td>Medium</td>
<td>2</td>
<td>Data modified after your last tape backup</td>
<td>Less than 15 minutes</td>
</tr>
<tr>
<td>High availability plus basic disaster recovery</td>
<td>Medium to high</td>
<td>3</td>
<td>Data modified after the last synchronization from the primary system to the disaster recovery</td>
<td>Less than 15 minutes</td>
</tr>
<tr>
<td>High availability plus high availability-disaster recovery and low data loss</td>
<td>High to very high</td>
<td>4</td>
<td>Less than 1 hour of data modifications</td>
<td>Less than 15 minutes</td>
</tr>
<tr>
<td>Extremely high availability plus high availability-disaster recovery and no data loss</td>
<td>Very high</td>
<td>4+</td>
<td>None</td>
<td>Less than 1 minute</td>
</tr>
</tbody>
</table>

The recommendations in this section are based on the following assumptions:

- All hardware and software required for the Perforce infrastructure is functioning properly.
- Your disaster recovery site does not fail at the same time as your primary site.
- SAN devices, which are designed to be insulated from any hardware single point of failure, are reliable.
Basic Disaster Recovery
This low-cost solution, which is far superior to tape backup alone, validates the usability of backups on a daily basis. This solution consists of two identically configured machines, one in the primary data center and another at a remote disaster recovery site. Each machine uses local RAID or SAN storage, configured with the file system layout that is specified in the Perforce Server Deployment Package (see Glossary). An automated process based on readily available software such as rsync or robocopy periodically copies the Perforce server files to the remote system.

The following diagram illustrates the relationship between the production server, and the remote disaster recovery system.

Warm Standby Plus Disaster Recovery
This solution consists of three identically configured machines using local RAID storage, configured with the file system layout that is specified in the Perforce Server Deployment Package. There are two local machines for primary and warm standby, with the third machine at the remote disaster recovery site. Like the basic approach, an automated process based on readily available software such as rsync or robocopy periodically copies the Perforce server files to the secondary system and, from there, to the remote system.

This approach is low-cost and capable of keeping downtime under thirty minutes, assuming your business can tolerate the minimal data loss caused by failing over to the warm standby or disaster recovery system. With this configuration, you risk losing any data modified after the last synchronization between the primary and disaster recovery servers.
The following figure shows the relationship between the primary, secondary (warm standby), and remote systems.

This approach does not provide high availability in failover situations, because the amount of time required to synchronize the data on the warm standby with the primary server's data can vary drastically depending on the nature of the hardware failure and the replication software that is in use. If the drives from the primary system are not damaged, or if you lose only the drive containing the Perforce metadata or journal, the warm standby can take over for the primary machine with no data loss by moving the drives to a machine where they can be accessed again.

High Availability Only
This medium-cost, high availability solution is built around a SAN, and does not provide disaster recovery capability. This solution consists of two identically configured machines connected to a dedicated high-performance SAN, or a SAN hosting the metadata and journal and NAS hosting the versioned files. (NAS is not recommended for hosting Perforce server metadata.)

The two systems are configured as a clustered system with automated failover of the storage layer, but there is no provision for an automated restart of the Perforce server process. The Perforce administrator must check the state of the system before restarting the server. Configure the SAN and NAS solutions with the file system layout that is specified in the Perforce Server Deployment Package. The following figure illustrates this configuration.
Using this approach, you can keep downtime under 15 minutes in the event of a failure, with no loss of data unless the SAN or NAS fails. If the SAN or NAS fails, you stand to lose all data modified since you last backed them up.

**High Availability with Basic Disaster Recovery**

This approach uses SAN with the primary and warm backup systems in a clustered environment to support high availability and disaster recovery, but does not provide for high availability in a failover situation. The primary server's depot data is replicated to the remote disaster recovery system as often as is feasible, which reduces data loss to the interval between replications. Because of the low risk of failover to the disaster recovery system in this configuration, this solution provides a high level of reliability at a reasonable cost.

The following figure shows the relationship between the systems in this configuration.
High Availability with Disaster Recovery and No Data Loss
This solution provides high availability capability at the disaster recovery site. It consists of two identically configured cluster systems: one for the primary server and the other for the disaster recovery site. The depot data volume must be replicated using real-time replication software or hardware solutions. This solution is recommended for large customer installations with a high volume of change.

Though more expensive than the preceding configurations, this approach can keep downtime under 15 minutes. In local failures, no data is lost. In the event of failover to the disaster recovery site, data loss can be kept to less than one hour, perhaps only minutes.
Extremely High Availability with Disaster Recovery and No Data Loss

This top-tier approach, expensive to build and maintain, is intended to ensure near-zero downtime and no loss of data. It consists of duplicate cluster systems, with all data replicated to the disaster recovery system in real time, as shown in the following figure.

The active Perforce journal and versioned files must be replicated in real time, perhaps using file-level replication systems. Real time replication of the journal can affect server performance, because the Perforce server constantly locks and writes the journal, potentially contending with the replication program for access. (Perforce does not provide a supported mechanism for replicating the active journal, but several Perforce customers have implemented their own solutions.)

This approach is recommended only for systems for which there truly is no tolerance for downtime (for example, transactional revenue-generating systems or communications infrastructure). In most environments, even those with thousands of users, SCM systems simply do not warrant the initial and ongoing efforts to develop and maintain custom, zero-downtime-guarantee failover solutions.
Disk Volume Layouts for the Perforce Server

To optimize performance, deploy the Perforce server files on three different volumes, if possible. The Perforce metadata must be on a separate physical volume from the Perforce journal and checkpoints, because they contain redundant information.

The following table provides details about how best to deploy server files.

<table>
<thead>
<tr>
<th>Volume</th>
<th>Performance Considerations</th>
<th>Recommended Storage</th>
<th>Backup Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Metadata</td>
<td>Optimize for I/O performance. Expect 10x to 100x I/O demands compared to depot storage file systems.</td>
<td>RAID 1+0</td>
<td>In high-volume environments, solid-state storage can be suitable. Do not backup or replicate. Keep disk space utilization at or below 65% to allow for recovery procedures.</td>
</tr>
<tr>
<td>Depot Data</td>
<td>Nominal performance needs. Optimize for redundancy and cost-effectiveness.</td>
<td>RAID 5/6 or RAID 1+0</td>
<td>Back up entire contents of this volume and replicate to disaster recovery site.</td>
</tr>
<tr>
<td>Logs</td>
<td>Higher performance needs.</td>
<td>RAID 1+0</td>
<td>Back up, but exclude the active journal.</td>
</tr>
</tbody>
</table>

The primary factor that determines server performance is the throughput of the I/O subsystem where the Perforce metadata is stored. I/O performance demands on the metadata volume volume are typically between 10 and 100 times greater than the volume where versioned files are stored.

If a SAN is used, consider using local storage for metadata and SAN for the versioned files and logs. Considerations include:

- The number of Perforce users
- The degree of automation (which can make 10 Perforce users seem like 100)
- The I/O performance of your SAN hardware
- Your priorities (performance, initial cost, ease of administration, commodity vs. custom solution, etc.)

If your SAN hardware delivers I/O performance comparable to (or faster than) direct attached storage, then put the metadata on your SAN. If you have thousands of users connecting to a single Perforce server, consider solid state disks for the metadata, to provide maximum scalability and performance.

If you intend to use two volumes rather than three for Perforce server data, put logs and depot data on the same volume, unless you are replicating depot data in real time.
Handling Perforce Failover

Though it is possible to automate a failover procedure with Perforce, it is recommended that a trained Perforce administrator initiate any failover procedure. Some customers have had limited success with automated failover solutions. However, in other cases, automated failover processes were triggered erroneously and caused data corruption.

The problem with automating failover is that many things that can go wrong. It is hard for an automated system to know the difference between a temporary network disconnection and a CPU failure on a server. There is no replacement for having a trained administrator familiar with the hardware configuration and Perforce who can analyze a failure, determine the root cause of it, and decide on the best course of action. The best course of action might be to initiate failover, wait for an unrelated network issue to be resolved, or to restore the primary server back to operation.

Developing a fault-tree diagram to comprehend and accurately identify all the things that might go wrong for any given Perforce installation, and defining corrective actions for each, is time consuming. It depends on having the specific software and hardware configuration encoded into mechanisms, which need to be maintained over time as your software and hardware environment changes. It is difficult to prove that a fault-tree diagram and related automation of corrective actions is complete, and impossible to test it without causing downtime.

It is a good idea to develop a failover script that takes care of all aspects of the failover in your environment and can be started manually after you have decided that it is prudent to do so.

Failover procedures include:

- DNS redirection to the hot spare and then starting the Perforce server process on the hot spare (if you are not using cluster software)
- Configuring the standby machine to become the primary (and vice versa). If clustering software and a SAN/NAS storage solution is used for the depot data volume, the Perforce Server Deployment Package makes this step unnecessary, because it is cluster-ready. Required only if you are using a fully-replicated solution for all metadata and versioned file tree data

Ensuring that the metadata is in a known-good state. Replay the journal into the most-recently-restored checkpoint, and make the resulting set of metadata files the active set. The Perforce Server Deployment Package ensures that offline copies of metadata files are always readily available.

- Starting the Perforce server process on the active node.
- Ensuring that the Perforce server process is not running on the inactive node (if accessible).

Additional Reading

For more information about high availability solutions, please see the Perforce Knowledge Base Article #308: “High Availability Solutions”.

Perforce Consulting is available to assist you with the design of high availability and disaster recovery solutions for your specific Perforce infrastructure. For more information on consulting services, please visit www.perforce.com/consulting.
The following terms are used in this document:

**Business impact analysis** The process of quantifying the amount of risk that your business can tolerate in the event of a disaster, such as acceptable downtime and cost.

**Direct attached storage** A storage system that is directly attached to a server or workstation.

**Disaster recovery** The process of restoring full functionality after your Perforce installation has been disabled by a natural or technical catastrophe.

**Extremely high availability** A zero-data-loss approach that replicates versioned files, checkpoints, and archived journal files on a schedule, and replicates the active journal file in real time.

**High availability** Minimizing Perforce downtime in the event of hardware failures.

**Network attached storage (NAS)** A specialized file server that connects to the network. A NAS device contains a basic operating system and file system, and processes only I/O requests.

**Recovery point objective** The maximum amount of data loss that can be tolerated as the result of a failure or disaster.

**Recovery time objective** The maximum Perforce downtime that can be tolerated after a service disruption or disaster, according to the severity of the cause.

**Replication** Duplication of the Perforce server files (metadata, versioned files and archived journals). Replication can be done in real time, based upon the desired level of availability. See also extreme availability.

**Server Deployment Package** A Perforce Consulting package that simplifies planning and deployment of high availability and disaster recovery solutions.

**Single point of failure** Any component of a system that, upon failure, causes an entire system to malfunction.

**Storage Area Network (SAN)** Dedicated networks that connect one or more systems to storage devices, making the storage devices appear to be locally attached to the operating system.