Perforce Software, Inc.
510.864.7400
consulting@perforce.com
www.perforce.com

Perforce Server Deployment

Best Practices

Document Version 1.0

April 9, 2009

Table of Contents

1 Introduction

The Perforce Server Deployment Package (SDP) is one of several offerings from Perforce Consulting Services. For customers bringing a new Perforce server online, we can send a consultant to ensure that things get started on the right footing by optimizing the Perforce installation and automating many of the administrative tasks. It entails a 2-day engagement, which can be done remotely or on site. The second day includes training for Perforce system administrators.

Existing Perforce servers can also be converted to the SDP structure, enabling customers to readily take advantage of offline checkpoints, email notification, and in general get on a supported solution.

The SDP is focused primarily on the System Admin and IT perspective of Perforce administration. In preliminary discussions we help validate that Perforce is provisioned with adequate resources. The on site portion of the training consists of deploying Perforce and training.

The Server Deployment Package is offered as a stand-alone package. It is also a part of our larger FastStart Package (currently in the making), a 5-day package that builds on the SDP, adding other services typically needed when customers just start out with Perforce. The FastStart Package bundles the SDP, 2 days of Perforce User Training, and one day of ad hoc consulting, which typically involves task like designing a branching strategy and Perforce directory structure standard for the organization.

2 Overview of the Server Deployment Package

2.1 What it Is

The Server Deployment Package consists of:

1. A set of documented recommendations for preparing server hardware to receive Perforce, primarily in the form of Volume Layout.
2. A standard procedure for deploying a p4d process and related automation.
3. A set of utilities for managing Perforce that reflect best practices evolved over the years. A few examples include:
    a. Automated server start/stop scripts.
    b. Automated daily/weekly checkpoint operations.
    c. Offline checkpoint capability.

Many experienced admins have independently solved some of the same problems solved by the SDP scripts; these problems have been solved hundreds of times.

The Server Deployment Package utilities are officially supported by Perforce Support. After initial deployment by Perforce Consulting Services, basic support for the utilities is covered under your standard support agreement.

2.2 What it is NOT

The Server Deployment Package is not a collection of every cool Perforce administrative tool or utility we've come across.

We balance functionality vs. maintainability when making updates, with some emphasis toward simplicity and stability.

3 Volume Layout

The following table indicates volume layout recommendations for a typical Perforce server.

Table 1: Volume Layout for Perforce

Partition Name	Sample Name	Contents and Backup Notes	Performance Considerations, Sample Storage Options
Operating System	/	Backup per local policy. Contains no Perforce files.	
Metadata	/metadata	Backup, but **exclude db.* files**. Contains active Perforce databases, offline copes of active Perforce server software, and license files. P4ROOT is here. Keep disk space utilization at or below 65% to allow for recovery procedures.	Optimize I/O for *random* read/write. Expect 10x to 100x the I/O demands compared to the depot storage volume. RAID 1+0 on XFS, using 15K RPM drives, or even solid state storage may be appropriate.
Depot Data	/depotdata	Contains contents of versioned files, in compressed binary and RCS reverse delta formats, checkpoints & inactive journals, and admin utilities and scripts. Backup entire contents of this filesytem using rsync, Robocopy, or other replication solution to spare machine and/or DR site.	Typically more *sequential* read/write. RAID5/6 on XFS is often appropriate, 7200 RPM drives, cost-saving SATA. Faster I/O solutions may be helpful if many binaries are versioned.
Logs	/logs	Contains server logs and active journal.	High performance demands. RAID 1+0 and XFS with 15K RPM drives filesystems appropriate.

Notes:

- Disclaimer: Generalizations here reflect the nature of the Perforce application and typical usage. Your mileage may vary.
- Performance of the /metadata filesystem is crucial to overall Perforce server performance. Performance of the /logs filesystem is also important.
- If the /logs volume does not exist, the logs and journal will go on the /depotdata volume.
- The /logs must not go on the /depotdata volume if the /depotdata volume is being replicated real-time.
- It is crucial that the Perforce databases be on a separate volume than the Perforce journal and checkpoints, as these contain redundant information. Do not combine these onto one volume.

- During creation of any Perforce depot, the Map field in the depot spec must be modified to assign the storage area to the /depotdata volume. The SDP includes a trigger to ensure that the Map filed is set correctly.
- If an SAN is used, we might recommend the /metadata volume be set up as a local volume, and all other partitions be set up on the SAN. This recommendation may vary depending on various factors, such as number of Perforce users, degree of automation (which can make 10 Perforce users seem like 100), and I/O performance of your SAN hardware. If your SAN hardware delivers I/O performance comparable to (or faster than) direct attached storage, then putting the /metadata volume on your SAN is appropriate.
- In extreme scalability scenarios (for example, 1,000+ users on a single server), solid state disks should be considered for the metadata volume to provide maximum scalability and performance.

3.1 Virus Scanning and Backup Utilities

- Avoid allowing any software other than the Perforce server itself from touching the db.* files on the metadata volume, including backup and virus scanning utilities, as this can cause data corruption.
- Virus scanning software should not be installed on a Perforce server. Virus scanners can cause data corruption, and in some cases misclassify valid files as viruses.
- Scanning the contents of the /depotdata volume is reasonable and a good practice in most cases. However, virus scanning systems should be set in 'detect and notify' rather than 'detect and delete' mode, so that administrators can take appropriate action (e.g. 'p4 obliteate') if a virus file is somehow checked into Perforce.

3.2 Emerging Storage Technologies

There are exciting things going on in the storage market with solid-state disks technologies. Major players in the storage industry are slated to deliver iSCSI FC SAN devices that support solid state disks in Fall 2008. Other technologies promise to revolutionize the enterprise storage landscape, offering unheard of throughput with more IOPS than ever, decreasing price points, and vastly increasing storage capacities.

Perforce tends to be more I/O bound than anything else. In our Performance Lab, we have been testing some SSD technology vendors. As one would expect, Perforce performs very well when solid-state storage solutions enter the picture.

4 Server Deployment Package (SDP) Structure Goals

The structure of the Server Deployment Package was designed with the following goals in mind

4.1 Apply Volume Layout to optimize Perforce performance and reduce risk of data loss.

Following the volume layout recommendations should result in optimal performance, reduced risk of data loss, and reduced amount of at-risk data even in event of catastrophic failure. (More on this in subsequent presentations).

4.2 Simplify DR -- Only one volume to replicate.

The /depotdata volume contains both versioned file and metadata in the form of checkpoints and journals. Thus, this volume contains all data essential to the mission of disaster recovery. It is common practice to backup the /logs volume as well (excluding the active journal), since doing so can aid in forensic analysis (figuring out what went wrong).

4.3 Provide SysAdmin/IT view (physical layout) and App Admin view (app-centric view).

The Server Deployment Package utilizes symlinks to provide two views of the system. The illustration in Figure 1 below shows the easily navigable Application Admin view, the logical structure of the SDP. Paths are kept short and easy to type. Symlinks present in a single view Perforce databases, versioned files, server logs, offline databases, utilities and auxiliary files.

The color coding in the illustration indicates which volumes that various types of entities are stored on, a view of interest to Perforce SysAdmin/IT.

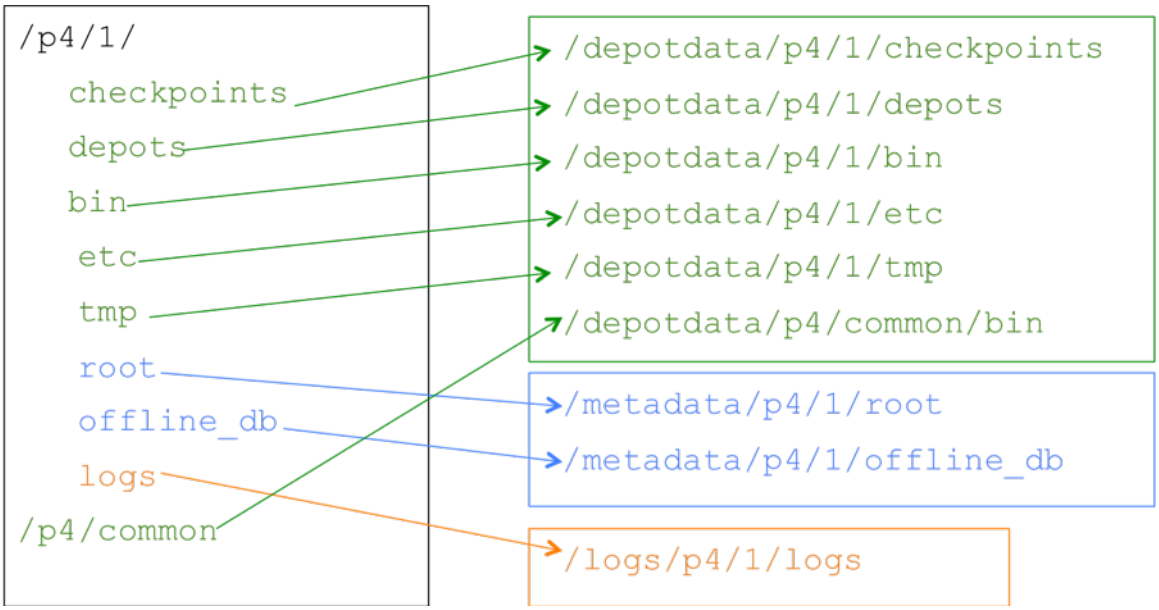

Figure 1: SDP Structure (Logical View)

4.4 Simplify configuration for failover in clustered environments.

The SDP is designed to simply the task of automating failover of Perforce in a clustered environment. Perforce itself is not a "cluster-aware" application. The SD does not provide a complete failover solution, since such solutions are inherently custom, taking into account your specific hardware. However, it does provide a baseline structure that can be readily adapted to complete a failover solution in a given environment.

4.5 Simplify upgrades.

The SDP makes follows some common practices using file naming conventions (e.g. p4d_2008.1.168182) to identify versions of p4d that are available, and symlinks to indicate which executable version is live.

5 Sever Deployment Package Features

The Server Deployment Package provides the following features:

- Disaster Recovery: All data, hardware, software, and Perforce licenses necessary to support business continuity are maintained, refreshed on a regular basis. The frequency of refresh is limited primarily by WAN performance, but generally one or more times a day. Usually depot data transfer is the limiting factor rather than metadata in terms of the number of refreshes possible per day.

- Validation of Usability of Backups: The recovery procedures that would be used in an actual Disaster Recovery scenario are exercised on a regular basis on DR hardware.

- Status Notification: Email notification keeps administrators informed of the status.

- Fast Recovery for Mundane Failures: In the event of failures more mundane that a complete disaster, the Server Deployment Package maintains local and remote copies of data assets that can decrease time to return a primary server to operation after a less-than-disaster failure.

- Optional Remote Verification: Verification of depot data against checksums stored in metadata (originally calculated on the primary server) can be a particularly resource-intensive operation. If the impact of doing a verify is noticeable on a server, or if it takes too long to complete while competing for resources on an active server, the archive verification can be configured to run on the DR server, making good use of otherwise passive but powerful hardware. If verification errors are detected, it may indicate either corrupted archives on the primary server, or possibly just problems transferring data to the DR site.

- Offline Checkpoint Capability: The offline checkpoint capability minimizes downtime for backup operations.

6 Default Protections Table

Defining a protections table is in inherently a custom task, since each organization has its own requirements, corporate culture, and guiding security philosophy. When deploy the SDP, our mission is to come up with an initial 'protect' table, and set the customer on the right path to managing it, raising awareness of performance and other issues. The protections table is accessed almost constantly during Perforce server operation, so emphasis on keeping it simple can help performance.

The initial protect table

- Disable implicit user creation.
- Proactively deny access to the 'remote' user.
- Define Limits.G to provide some sensible defaults for MaxScanRows, MaxResults, and MaxLockTime. Note that this group need not be listed in the protections table.

7 Default Typemap

The SDP provides a basic typemap that incorporates some common and best practices we've come across. Some key tidbits:

- For already-compressed file types (such as .zip, .gz, .avi, .gif), assign a file type of `binary+Fl` to prevent the server from attempting to compress them again before storing them, and to prevent multiple people from checking them out simultaneously.
- For regular binary files, such as Microsoft Office documents and other formats that cannot be easily merged, use the exclusive lock attribute, `binary+l`, to ensure that only one person modifies the file at a time.
- For large, generated text files, assign the `text+C` file type, to avoid causing server memory issues.

8 Applied Common Practices

As part of our standard procedure for bringing a Perforce server online, we go through a checklist of Perforce admin tasks to make a server ready for action.

1. Enable Monitoring.
2. Create & populate a 'spec' depot.
3. Set 'security' counter or use external authentication w/AD, LDAP.
4. Set appropriate logging verbosity.
5. Automate checkpoint creation with weekly & daily tasks.
6. Coordinate checkpoint creation activity with IT backups.
7. Test crontabs and scheduled tasks.
8. Enable the email change reviewer.
9. Apply SCM to the SDP!