SDP Standard for journalPrefix

Perforce Professional Services

Version v2020.1, 2020-12-09

Table of Contents

Preface

This document describes the SDP Standard for the journalPrefix configurable in Perforce Helix Core.

This is related to the SDP Server Spec Naming Standard.

Please Give Us Feedback

Perforce welcomes feedback from our users. Please send any suggestions for improving this document or the SDP to consulting@perforce.com.

Chapter 1. Overview

The Perforce Helix configurable `journalPrefix` determines where the active journal is rotated to when it becomes a numbered journal file during the journal rotation process. It also defines where checkpionts are created. In the SDP structure, the journalPrefix is set so that numbered journals and checkpoints land on the `/hxdepots` volume. This volume contains critical digital assets that should be reliably backed up should sufficient storage for large digital assets.

Chapter 2. Master Metadata journalPrefix Value

The journalPrefix value used on the master server is of this form, where N is replaced with the SDP instance name:

```
/p4/N/checkpoints/p4_N
```

If the SDP instance name is the default 1, then files with a p4_1 prefix would be stored in the /p4/1/checkpoints directory on the filesytem. Journal files in that directory would have names like p4_1.320.jnl and checkpoints would have names like p4_1.320.ckp.gz.

This journalPrefix value and the corresponding /p4/1/checkpoints directory should be used for the master server. It should also be used for any replica that is a valid failover target for the master server. This includes all *completely unfiltered* replicas of the master, such as standby and forwarding-standby replicas.

> A standby replica, also referred to as a journalcopy replica due to the underlying replication mechaninsm, cannot be filtered. It is commonly deployed for High Availability (HA) and/or Disaster Recovery (DR) purposes.

2.1. Detail on "Completely Unfitered"

A "completely unfiltered" replica is one in which:

- None of the *DataFilter fields in the replica's server spec are used
- The p4 pull command configured to pull metadata from the the replicas P4TARGET server, as defined in the replica's startup.N configurable, does not use filtering options such as -T.
- The replica is not an Edge server (i.e. one with a Services value in the server spec of edge-server.) Edge servers are filtered by their vary nature, as they exclude various database tables from being replicated.
- The replica's seed checkpoint was created without the -P ServerID flag to p4d. The -P flag is used when creating seed checkpoints for filtered replicas and edge servers.

Note that the SDP sync_replica.sh script can be deployed on unfiltered replicas of the master.

Chapter 3. Second Form of `journalPrefix` Value

A second form of the `journalPrefix` is used when the replica is filtered, including edge servers. The second form of the `journalPrefix` value incorporates a shortened form of the *ServerID* to indicate that the data set is specific to that *ServerID*. Because the metadata differs from the master, checkpoints for edge servers and filtered replicas are stored in a different directory, and use a prefix that identifies them as separate and divergent from the master's data set.

> Filtered replicas are a strict subset of the master server's metadata. Edge servers filter some database tables from the master, but also have their own indepdent metadata (mainly workspace metadata) that varies from the master server and is potentially larger than the master's data set for some tables.

The "shortened form" of the *ServerID* removes the `p4d_` prefix (per the SDP Server Spec Naming Standard. So, for example an edge server with a *ServerID*` of `p4d_edge_uk` would use just the `edge_uk` portion of the *ServerID* in the `journalPrefix`, which would look like:

```
/p4/N/checkpoints.edge_uk/p4_N.edge_uk
```

Chapter 4. Scripts for Maintaining the offline_db

The SDP has two scripts for the offline_db, daily_checkpoint.sh and sync_replica.sh.

The daily_checkpoint.sh is used on the master server. When run on the master server, this script rotates the active journal to a numbered journal file, and then maintains the master's offline_db using the numbered journal file immediately after it is rotated.

The daily_checkpoint.sh is also used on edge servers and filtered replicas. When run on edge servers and filtered replicas, this script maintains the replica's offline_db in a manner similar to the master, except that the journal rotation is skipped (as that can be done only on the master).

The sync_replica.sh script maintains the offline_db on the unfiltered replicas. It rsyncs the checkpoints from the master, and replays them to the local offline_db. This keeps the offline_db of the replica current, which is good to have should the replica ever need to take over for the master.

INFO: For HA/DR and any purpose where replicas are not filtered, we promote that replicas of type standby and forwarding-standby displace replicas of type replica and forwarding-replica.

Chapter 5. SDP Structure and `journalPrefix`

On every server machine with the SDP structure where a `p4d` service runs (excluding broker-only and proxy-only hosts), a structure like the following should exist for each instance:

- A `/hxdepots/p4/N/checkpoints` directory
- In `/p4/N`, and symlink `checkpionts` that links to `/hxdepots/p4/N/checkpoints`, such that it can be referred to as `/p4/N/checkpoints`.

In addtion, edge servers and filtered replicas will also have a structure like the following for each instance that runs an edge server or filtered replica:

- A `/hxdepots/p4/N/checkpoints.ShortServerID` directory
- In `/p4/N`, and symlink `checkpionts.ShortServerID` that links to `/hxdepots/p4/N/checkpoints.ShortServerID`, such that it can be referred to as `/p4/N/checkpoints.ShortServerID`.

The SDP `mkdirs.sh` script, which sets up the initial SDP structure, initializes this structure on initial install.

Chapter 6. Goals of this Standard

Some design of goals this standard:

- Make it so the `/p4/N/checkpoints` folder is reserved to mean checkpoints created from the master server's full metadata set.
- Make the `/p4/N/checkpoints` folder be safe to rsync from the master to any machine in the topology (as may be needed in certain recovery situations for replicas and edge servers).
- Make it so the SDP `/hxdepots` volume can be NFS-mounted across multple SDP machines safely, such that two or more edge servers (or filtered replicas) could share versioned files, while writeing to separate checkpoints directories on a per-ServerID basis.
- Support all replication uses cases, including support for 'Workspace Servers', a name referring to a set of edge servers deployed in in the same location, typically sharing `/hxdepots` via NFS. Use of Workspace Servers can be used to scale Helix Core horizontally for massive user bases (typically several thousand users).