Demand Attach Fileserver

High Availability Clustering Extensions

Tom Keiser
Sine Nomine Associates

Introduction

- Goal is to extend DAFS to support live partition migration
- numerous changes required to VLDB, volume package, and vos
- VLDB changes are done; volume package work is ongoing
- existing code is of prototype quality
- This is a “spare time” development effort

DAFS Refresher

- provides on-demand volume salvaging
- codifies volume state in an FSA in order to move all high-latency operations (e.g. I/O) outside of locks, thus improving concurrency
- VLRU garbage collection mechanism reduces size of attached volume set at shutdown/crash
- extensions to support highly concurrent fileserver shutdown
- serialization and restore of host/callback state during normal restart process

Motivating Problems

- HA clustering of AFS fileservers is difficult with the present tooling
- existing solutions address DR, not HA
  - e.g. vos shadow + vos convertROtoRW
- existing solutions do not address scalability or load balancing
  - scalable N-node fileserver clusters are not practical with the current codebase

Motivating Problems II

- with SANs practically ubiquitous in enterprise environments, it would be nice to be able to move vice partition LUNs without resorting to ineligant techniques such as: partition evacuation, vos syncvldb, etc.
- high-performance cluster filesystem backends present similar problems (e.g. how to balance load in a cluster of fileservers with vice partitions backed with GPFS or Lustre)

Goals

- provide a framework capable of supporting N-node scalable HA fileserver clusters
- initially focus on RW volume serving (more on this later)
- extend the VLDB schema to codify more complex storage hierarchies

Architectural Changes

- partition UUIDs are the core concept
- servers now register mounted disk partition UUIDs in VLDB
- volume package disk partition objects can be dynamically mounted/unmounted
- whole-partition salvages can occur without bringing down the fileserver

Implementation: VLDB

- new vlserver supports several new on-disk data structures which provide additional layers of indirection
- current on-disk vldb schema maps volumes directly onto storage using a $(\text{server}, \text{partition})$ tuple
- new on-disk vldb schema logically maps volumes onto disk partition objects, which then indirectly map onto a server
- new schema tracks list of all partitions currently mounted on each server

Implementation: Volume Package

- disk partition objects become mutable and refcounted
- allow for “mounting” and “unmounting” vice partitions on-the-fly
- allow for whole-partition salvages without fileserver shutdown
- maintain a local table of all known partitions, which are cleared for mounting, and other metadata (e.g. backend type for Derrick’s multibackend prototype)

Implementation: vos

- several new `vos` subcommands are needed:
  - `vos definepartition <server> <partition>` – define a vice partition as mountable on a server
  - `vos removepartition <server> <partition>` – remove a vice partition definition from a server
  - `vos mountpartition <server> <partition>` – mount a vice partition on a server
  - `vos umountpartition <server> <partition>` – unmount a vice partition from a server
  - `vos listpartitions <server or partuuid>` – list VLDB's notion of all vice partitions on a given server, or details of a specific partition given a UUID

Observations

- RO replicas significantly complicate this problem
  - volser does not correctly support multiple replica sites on the same machine
  - possibility to lose HA by migrating all repsites to a single node

Futures

- ideally, an HA clustering solution should support RO replicas
- achieving this goal will likely require a way to codify policy in the VLDB
- ideally, load balancing and policy enforcement would be handled by a centralized autonomic control application
- an optimal load balancer needs to weight its decisions by the number of repsite clone operations necessary to maintain policy in the face of RO site duplication on a node due to partition migration

DAFS HA Clustering Extensions

Questions?