Distributed Asynchronous Object Storage (DAOS)

Catastrophic Recovery

Monday 13-Nov-2023

In the unlikely event …

- Existing data protection and fault tolerance measures are overwhelmed
- Self healing mechanisms may fail (ENOSPC)
- HW/SW bugs
- Human error

Stages - New tools

1. Recover local storage
   - Existing Tools: ipmctl, ndctl, pmempool check, e2fsck, smartctl, debugfs, …
   - **New Tool: DAOS Debugger (ddb)**

2. Address distributed inconsistencies
   - **New Module: DAOS Checker**
   - Pool and Container consistencies

DAOS Debugger (ddb)

- Standalone debug utility
- Runs locally on DAOS Server
- Connects to a single VOS file (Version Object Storage)
- Navigate and update metadata: containers, objects, dkey, akey, recx.
- Examples:
  - Pool Shard metadata
  - Pool/Container Service Replicated Data Base File (RDB)

Some DAOS Background

- Management Service – manages system metadata, creates pools
- Pool Service – manages pool metadata, replicated, pool map
- Container Service – manages container metadata, replicated

DAOS Checker

Pass 0

- **Setup**
  - Verify Management Service and engines' status
  - Select engine as check leader, start engines in check mode

Pass 1

- **Pool List Consolidation**
  - Engines scan and report known pools
  - Inconsistencies: orphan pool, dangling pool, broken Pool Service

Pass 2

- **Pool Membership**
  - Run pool service in check mode
  - Inconsistencies: orphan pool shard, dangling pool map entry

DAOS Checker

Pass 3

- **Pool Cleanup**
  - Check pool labels and stale pool connection
  - Inconsistencies: inconsistent pool labels

Pass 4

- **Container List Consolidation**
  - Engines scan & report known containers
  - Inconsistencies: orphan containers

Pass 5

- **Container Cleanup**
  - Check container labels and stale container handles
  - Inconsistencies: inconsistent container labels

A Few More Things …

- Testing and Fault Injection
- Release
  - Tech Preview in 2.6
  - Production in 3.0
- Future Work
  - Object Scrubbing
  - Middleware


intel