SDC

STORAGE DEVELOPER CONFERENCE

SNIA ■ SANTA CLARA, 2014

Stratus to Cirrus: Avoiding nose-bleeds during upgrades of cloud storage systems

Tom Cocagne

Cleversafe

Purpose of the Presentation

- Upgrades of cloud storage systems should be easy, reliable, and fast background operations that end users never even notice
- Large-scale deployments present unique challenges and greatly compound the severity of traditional issues
- In this presentation we'll go over some of the challenges involved and present techniques for addressing them

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Topics

- Overall challenges associated with upgrades of cloud storage systems
- Orchestrating the upgrade process
- Upgrading individual devices

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Overview

- Essential mechanism for feature and fix delivery
- High Risk
  - Performance
  - Availability
  - Reliability
- High Visibility

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Technical Challenges

- Zero-downtime
- Mixed-mode operation
- Bugs in fielded deployments
- Graceful handling of disk failures
- Latent issues
- Exabyte systems

SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Non-Technical Challenges

- Cross-cutting design issue
- Communication between development and support staff
- Testing of all possible permutations

SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Upgrade Orchestration

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Orchestration Design Principals

- Zero-downtime
- Assume upgrades are destructive
- Verify data integrity after upgrade
- Minimize state synchronization requirements

SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Parallelism

- Essential at scale
  - Each per-node minute expands to a full week on systems with 10,000 nodes
- Must protect availability while still allowing some tolerance for failure
- Configurable thresholds allow administrators to tune the risk/performance tradeoff to suit their environments

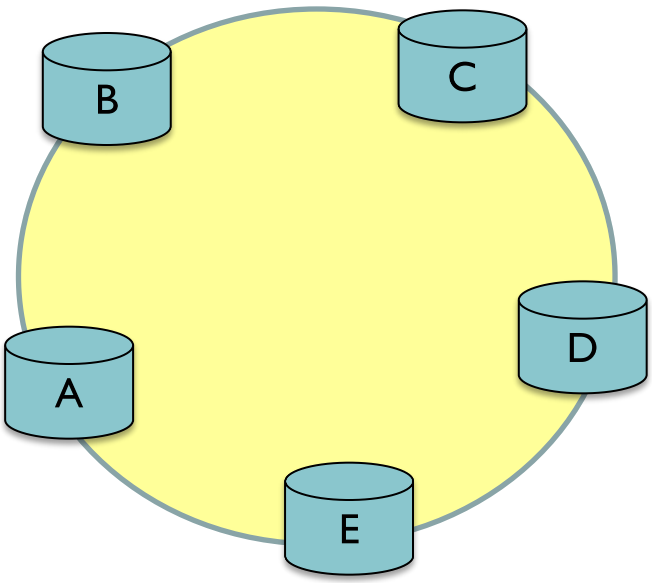
SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Parallelizing Upgrades of Interdependent Systems

Consider a 5-wide storage system with 3x replication.

Each group of three going around the ring defines a replication group.



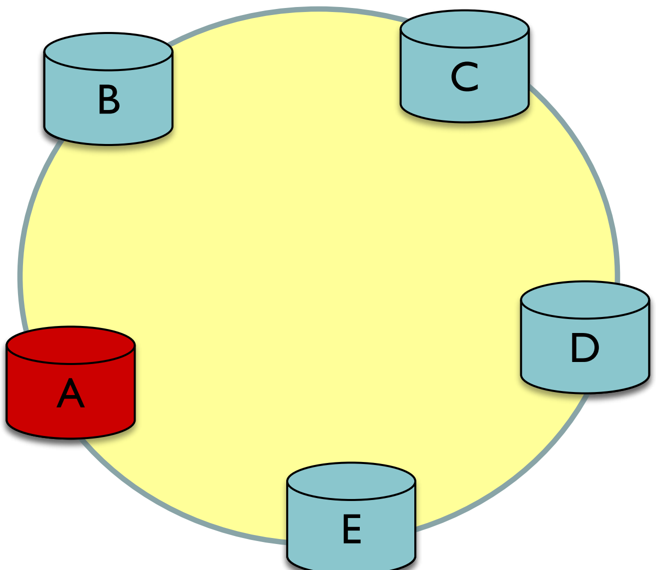
Group	Failure Tolerance
ABC	2
BCD	2
CDE	2
DEA	2
EAB	2

SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Parallelizing Upgrades of Interdependent Systems

Taking down store **A** affects multiple groups.

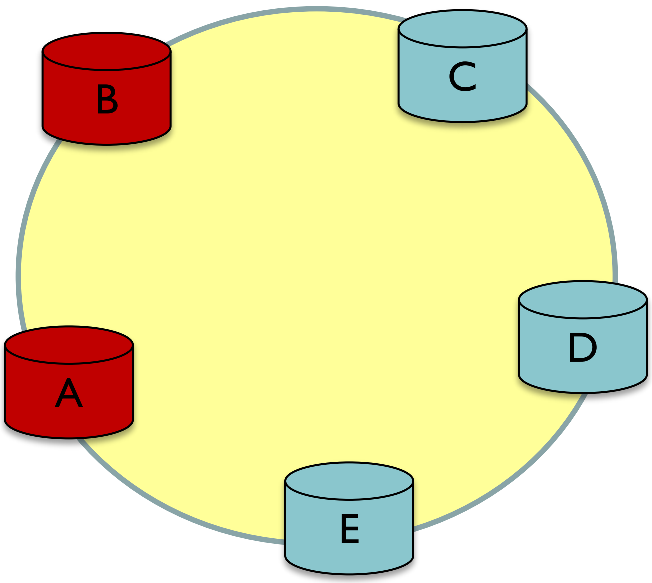


Group	Failure Tolerance
ABC	1
BCD	2
CDE	2
DEA	1
EAB	1

SDC 14
2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Parallelizing Upgrades of Interdependent Systems

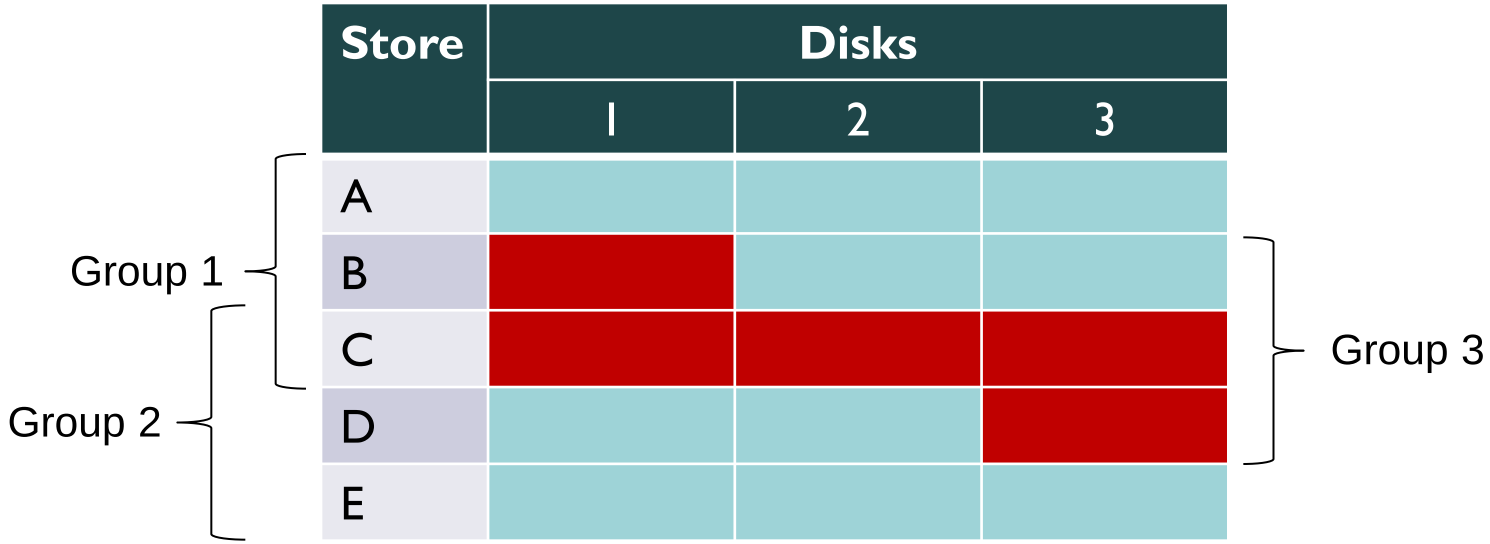
Store **B** can be taken offline as well without sacrificing availability



Group	Failure Tolerance
ABC	0
BCD	1
CDE	2
DEA	1
EAB	0

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

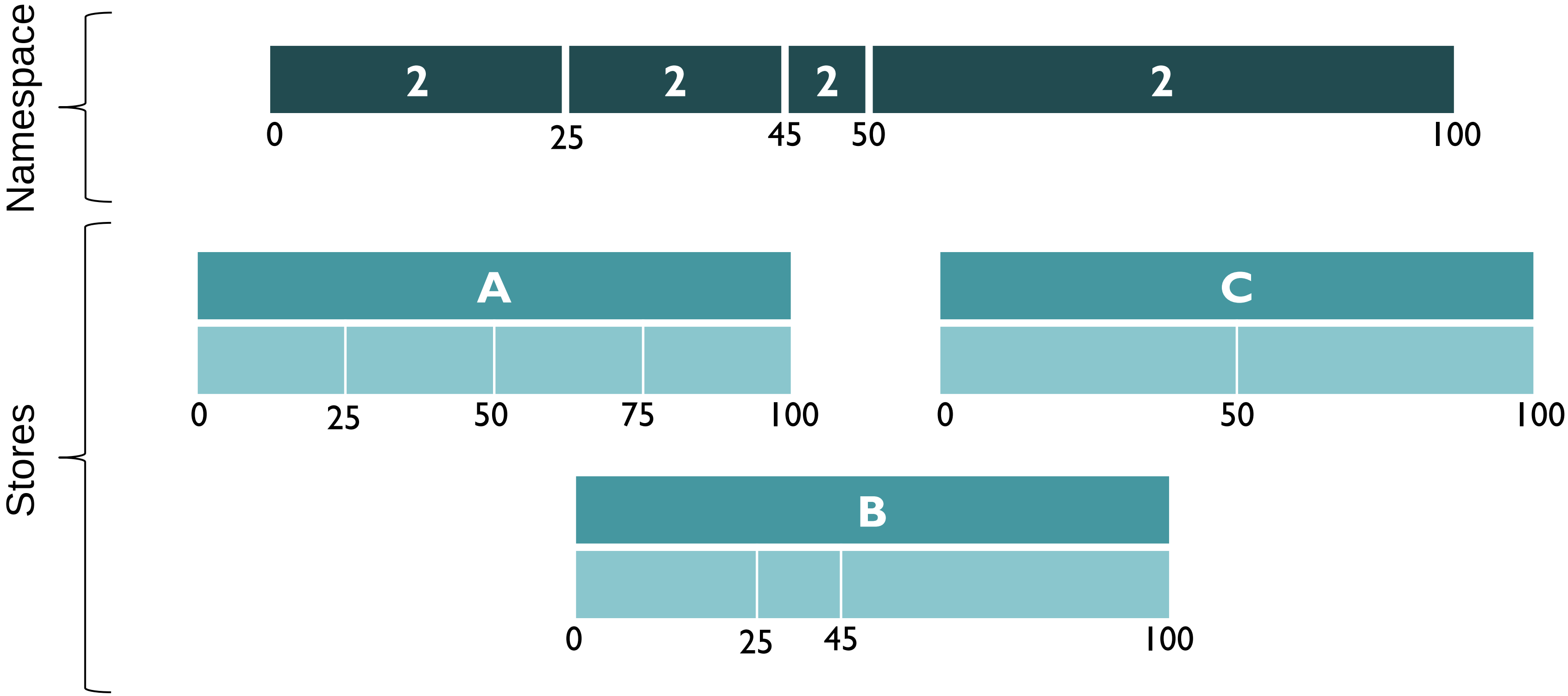
Disk Failures Complicate Parallelization



If store **C** is taken down, all other stores must remain up.

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

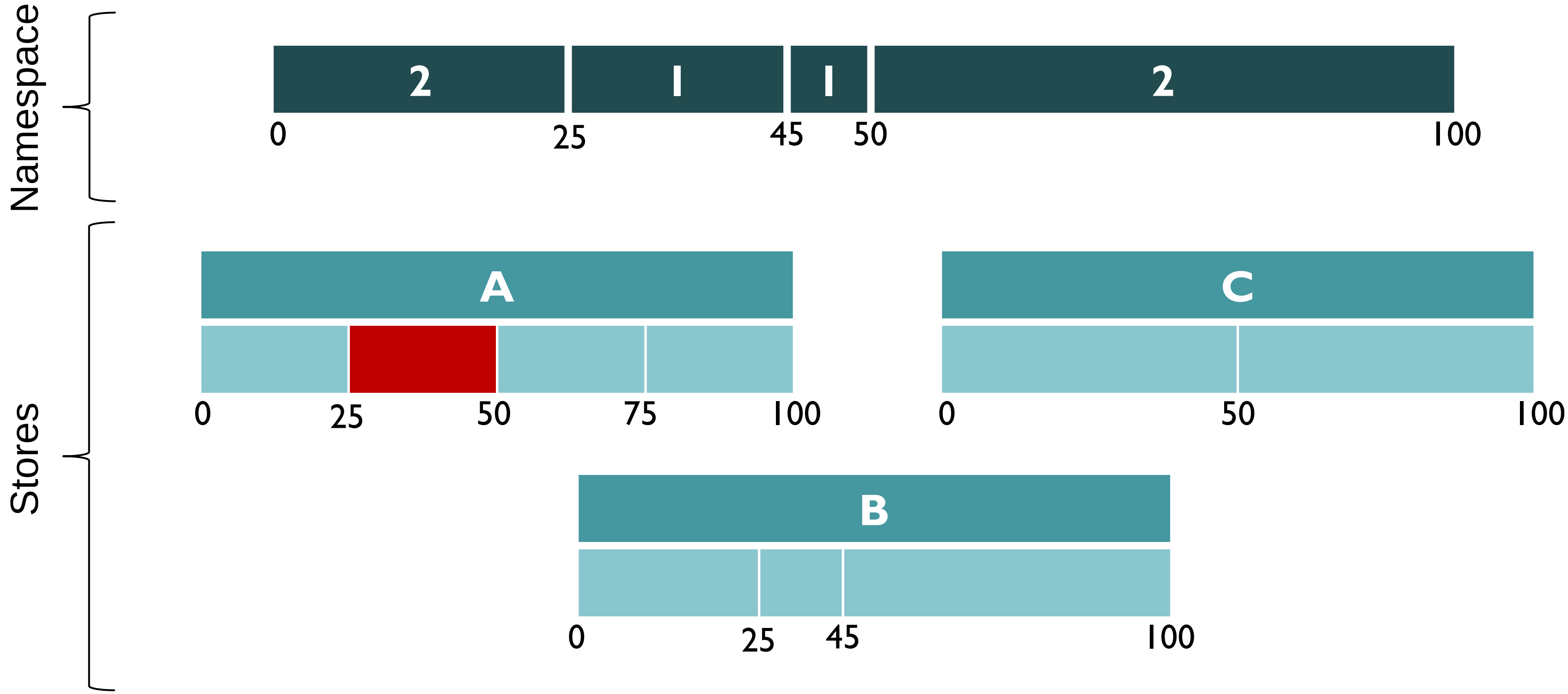
Data Health Model with Disk-Level Granularity



- A, B, and C mirror the same data but have disks of differing sizes.
- Each disk is assigned some portion of the overall namespace

SDC 14
2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

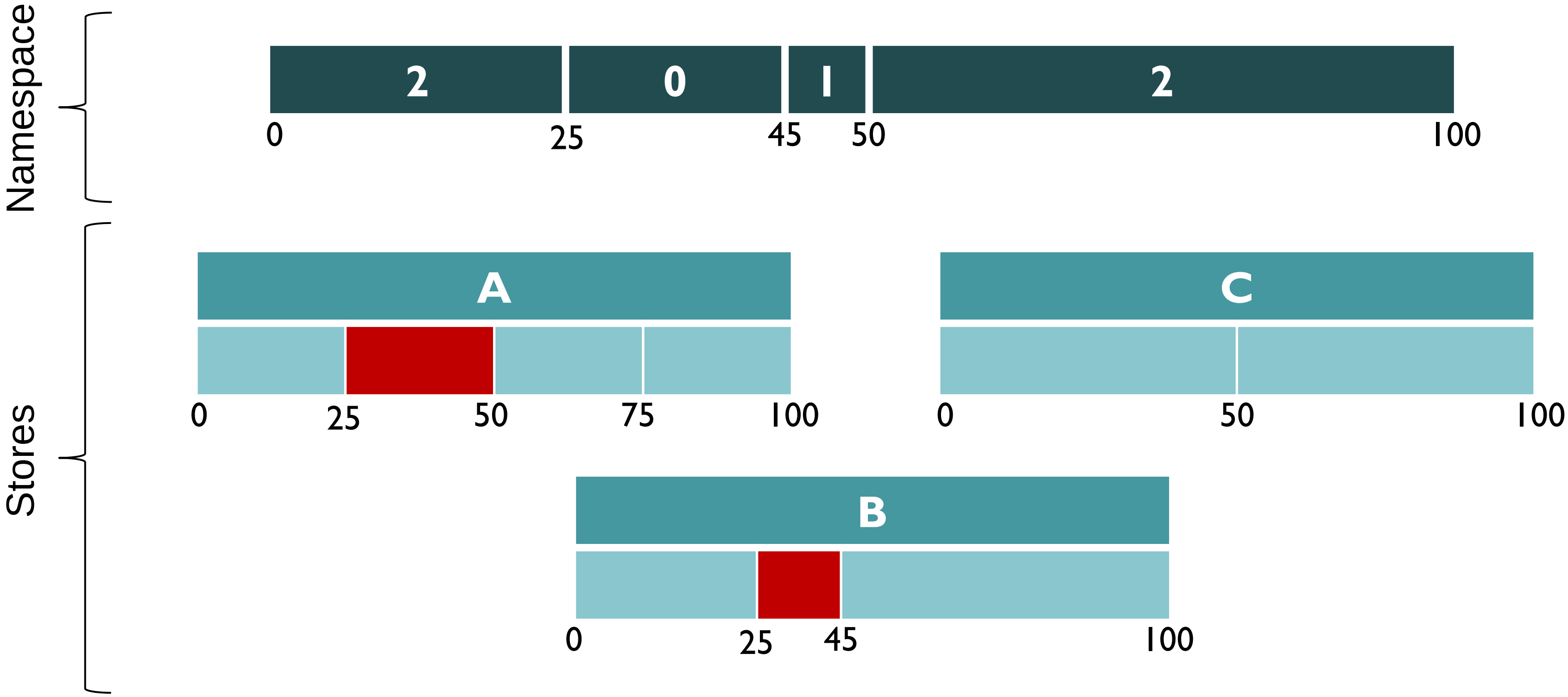
Data Health Model with Disk-Level Granularity



- The namespace is divided into multiple segments
- Different portions of the namespace have different failure tolerances.

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Data Health Model with Disk-Level Granularity



2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

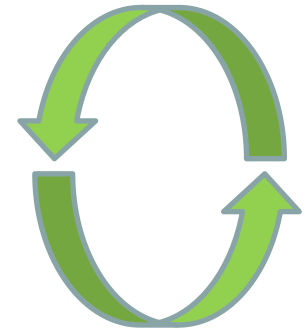
Health-Based Upgrade Candidacy

- As part of determining whether a node can be taken offline, the health of all namespace ranges it hosts must be considered
- The node is not an upgrade candidate if taking it offline would drop any sliver of the namespace below the failure tolerance threshold

SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Upgrade Orchestration Process



- High-Level Orchestration Loop:
  1. Select a node to upgrade
  2. Verify that bringing it down will not violate the failure tolerance threshold for any portion of the namespace
  3. Mark all ranges hosted by all disks on the node as being unhealthy
  4. Upgrade the device
  5. Mark all ranges hosted by disks that survived the upgrade as being healthy

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Health-Based Orchestration Benefits

- If too many disk failures accumulate, for any reason, the upgrade process halts
- Once the unhealthy ranges are repaired, the upgrade process resumes
- Protects against correlated disk failures
- Built-in parallelization
  - Just keep bringing hosts down for upgrade until the failure tolerance threshold is reached

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Device Upgrades

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Device Upgrade Design Principals

- Assume a hostile operational environment
  - The system really is out to get you
- Minimize operational environment dependencies
  - The vehicle for delivering fixes MUST work
- Minimize potential need for manual interaction
  - Exceedingly rare is all-the-time at scale
- Minimize upgrade time
  - Minutes expand to weeks at scale

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Built-In Upgrade Mechanism

- Suffers from the N-1 problem
  - In order to take advantage of a fix/feature when upgrading to version N, the foundation for it must first be delivered in version N-1
  - Delays delivery of fixes
- Limits flexibility in working around bugs
- May require manual intervention to prevent problems during upgrade

SDC 14

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Decoupled Upgrade Mechanism

- Upgrade is driven by a separate component that is downloaded at the time of the upgrade
  - May employ workarounds for known issues in previous versions
  - May be independently maintained and updated outside of the normal product cycle
- Mostly, though not entirely, avoids N-1 problems

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Idempotent Phasing

$$f(f(x)) = f(x)$$

- Breaking the upgrade process into a series of Idempotent Phases ensures consistency in the event of crashes/power outages

- Common architectural pattern used by most modern configuration management systems such as Chef & Puppet

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Environmental Problems

An extremely non-exhaustive list:

- Shutdown Hangs
- Process Pileups
- Memory Exhaustion
- High Load
- Filesystem Corruption
- File Permissions
- Boot hangs
- Limping Hardware
- Disk I/O Errors
- Manual "Tinkering"
- Network outages
- Corrupted downloads

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Mitigation Techniques

- Pre-Checks
  - Examine systems for conditions that will definitely cause upgrades to fail and disqualify them as upgrade candidates if any such conditions are found
  - Insufficient disk space, read-only disks, version mismatches, etc.

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Mitigation Techniques

- Proactively look for potential problems and fix them if possible
  - e.g. bad file permissions
- Assume every system call will hang
- Wrap all operations with beyond-reasonable timeouts
  - e.g. if it takes more than 5 minutes to get a list of all running processes, something is very, very wrong

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Mitigation Techniques

- If all else fails, reboot the box
  - Safe for idempotent architectures
  - *May* clear the problem
  - Be sure to capture as much state information as possible for post-mortem analysis

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.

Questions & Answers

Tom Cocagne
tcocagne@cleversafe.com

2014 Storage Developer Conference. © Cleversafe, Inc. All Rights Reserved.