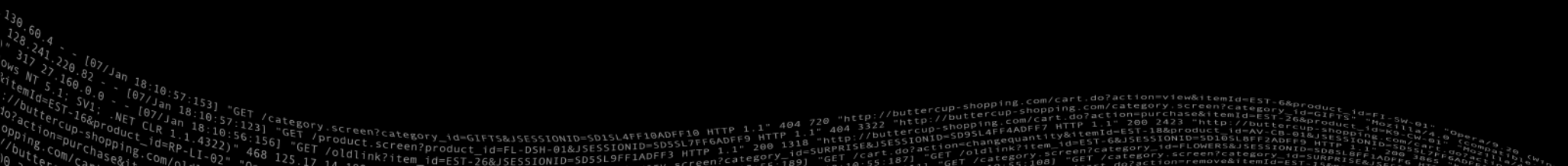
SmartStore

Next-Gen Architecture to reduce TCO

splunk> .conf18

Agenda

- Why we built it
- How it works
- Getting data in
- Searching data



splunk> .conf18

Why we built it

Existing deployments

- Use multiple Splunk Indexers with clustering
- Each Indexer has Compute and fixed Storage
  - Compute : Necessary for indexing and searching data
  - Storage : Necessary for storing indexed data
- Prune older data to free storage for newer data

Why we built it

Want to increase storage?

- Add additional Indexers
  - Higher operational complexity
  - Underutilized compute
  - Higher TCO
  - Need to rebalance data - expensive

Why we built it

Introducing SmartStore

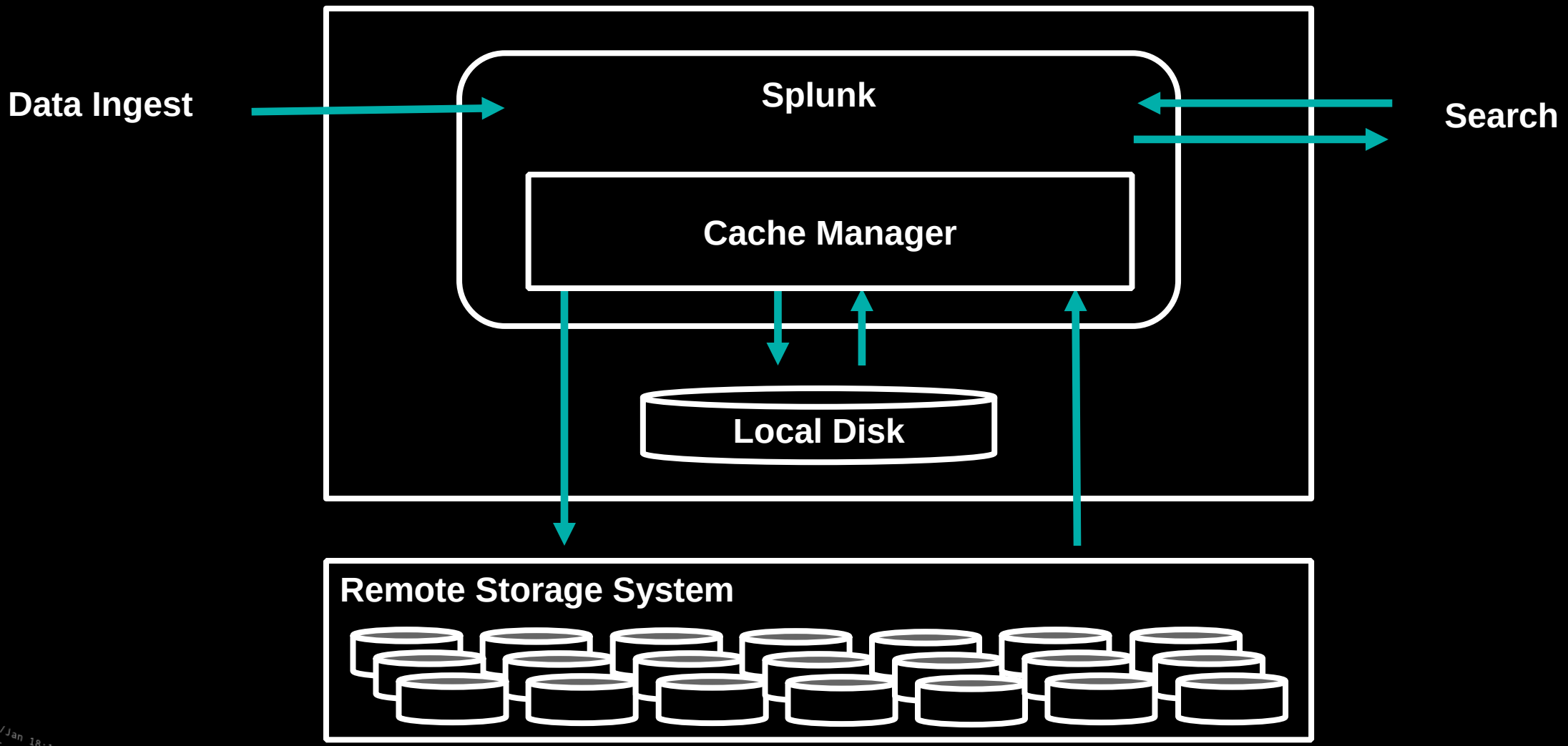
- Decouples Compute and Storage
- Stores data on Highly Available (HA) Elastic Remote Storage
- Caches data on Indexer's Disk Storage

Why we built it

Benefits

- Independently scale Compute and Storage
  - Provision Indexers based on compute requirements
  - Scale Storage without additional Indexers
    - Lower operational complexity
    - HA Remote Storage cheaper compared to Indexer’s Disk Storage

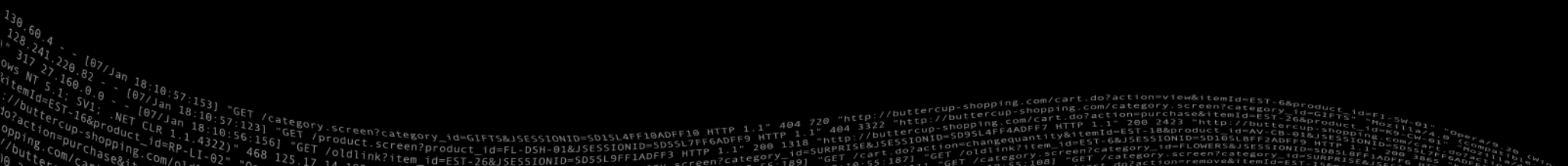
How it works



How it works

Cache Manager

- Pulls data on-demand
- Relies on a reliable Remote Storage (such as Amazon S3)
  - High availability
  - High elasticity
- Is what makes SmartStore “smart”
- Works like magic ☺

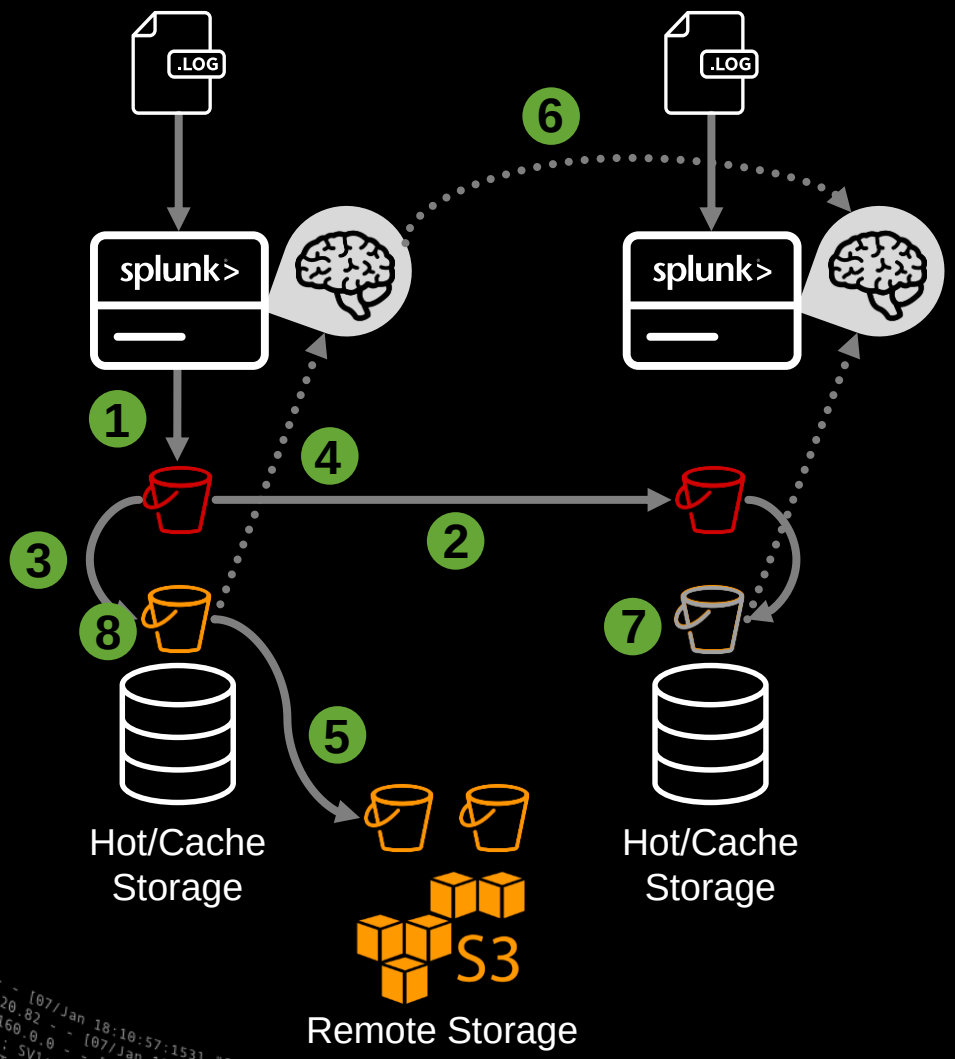


How it works

Cache Manager

- Majority of searches happen over recent data
- Majority of searches use same data
- Uses above observations to cache data efficiently

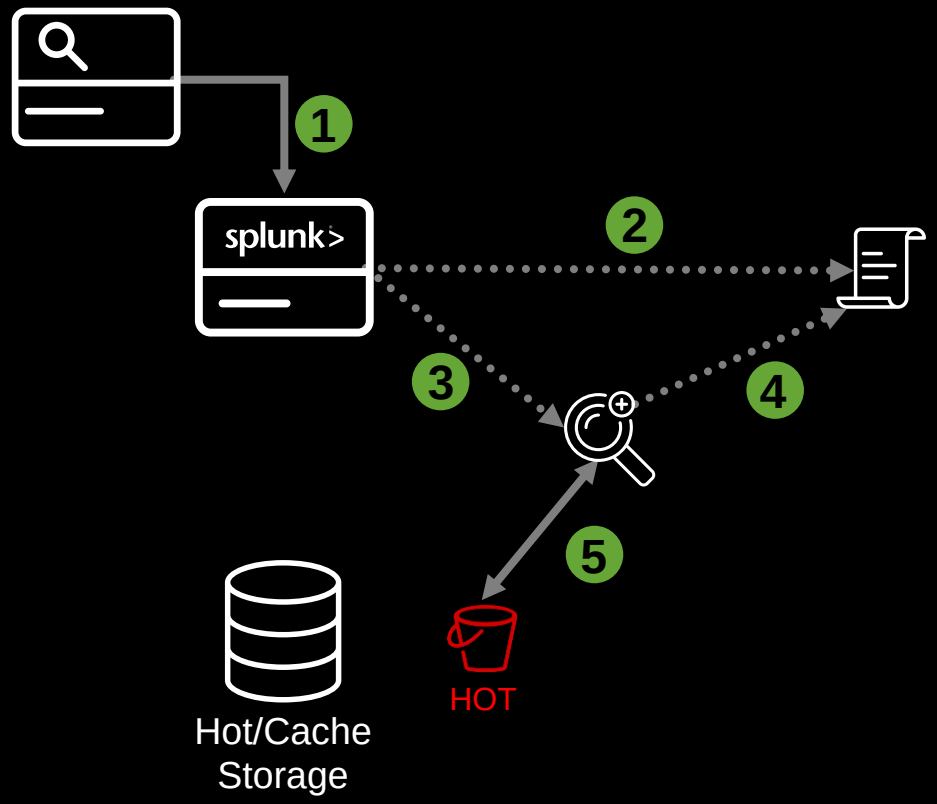
Getting data In



1. Data is written to a hot bucket on source indexer
2. Hot bucket streamed to target indexer
3. Replication completes - buckets roll to warm
4. Buckets are registered with their cache managers
5. Cache manager on source uploads the bucket
6. Source notifies target about the upload
7. Target deletes local contents of the bucket
8. Local contents remain on the source until evicted

Searching data

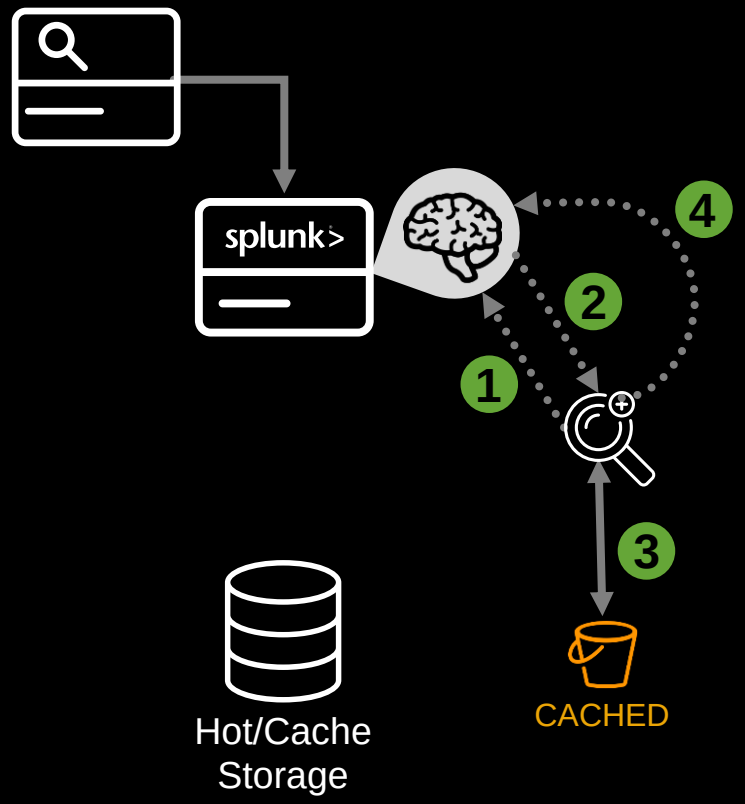
Hot buckets



1. Search request is received
2. Indexer generates a list of relevant buckets to be searched
3. Search process is spawned
4. Spawned process reads the bucket list
5. Hot buckets are searched in the same manner as “classic” search

Searching data

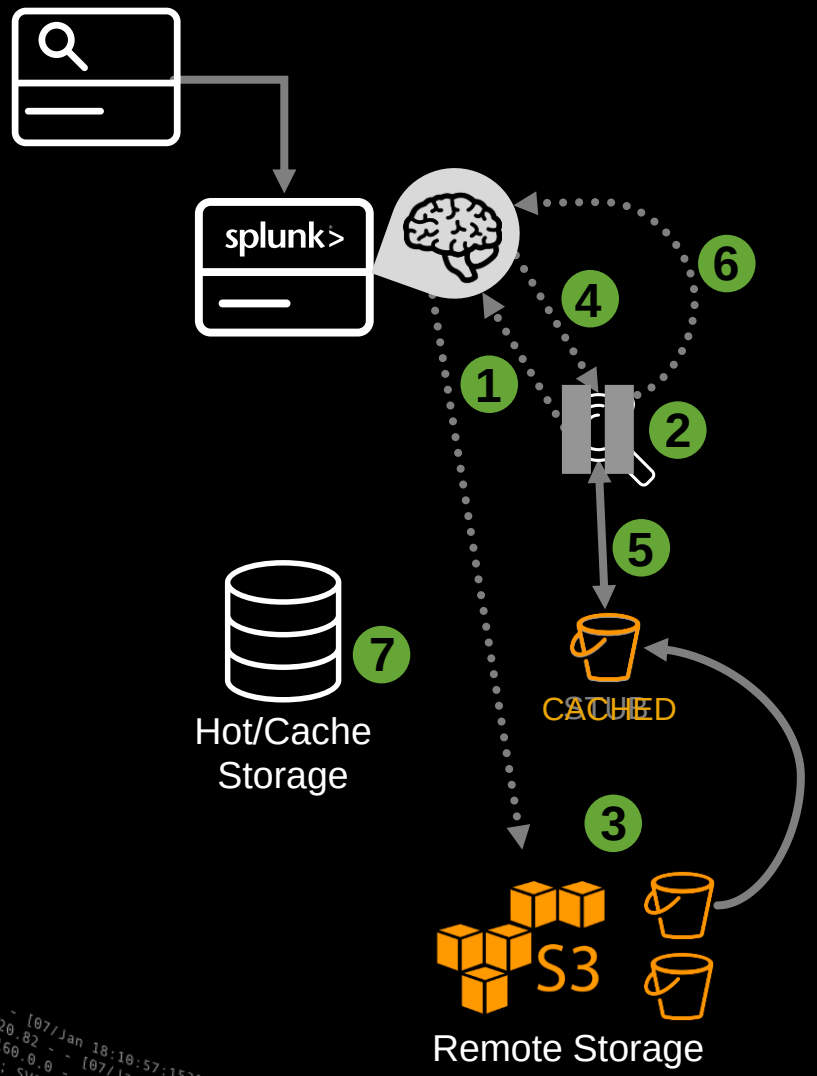
Cached buckets



1. Search process requests Cache manager to open the bucket
2. Cache manager tells the search process that the bucket is local and available for search
3. Search process searches the bucket
4. Search process requests Cache manager to close the bucket

Searching data

Remote buckets



1. Search process requests Cache manager to open the bucket
2. Search process waits
3. Cache manager downloads the bucket from the remote storage
4. Cache manager tells the search process that the bucket is local
5. Search process searches the bucket
6. Search process requests Cache manager to close the bucket
7. Bucket remains in cache until evicted by the Cache manager

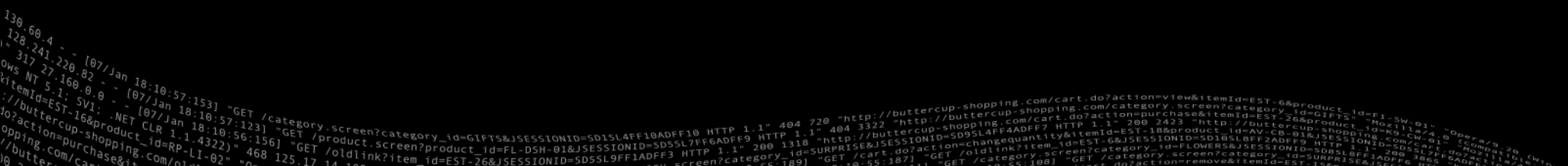
Observations

- Majority of buckets are present on Remote Storage
- Entire bucket does not need to be replicated
- Bucket metadata replication is enough
- Indexer can download bucket contents on demand

… and one last thing

Failed indexers < Replication factor

- Need to send entire data today
- Data is present on Remote Storage
- Bucket metadata is enough with Remote Storage
- Recovery is fast - 10000 buckets in 10 secs



splunk> .conf18

… and one last thing

Failed indexers >= Replication factor

- Recovering data was not possible before today
- Always have access to data with Remote Storage
- Can rediscover and recover data

splunk> .conf18

… and one last thing

Bootstrapping

- Re-using data from one cluster on another was not possible before today
- Can point new cluster to same Remote Storage
- New cluster rediscovers data
- Super fast – 20000 buckets in a minute on a 4 node cluster

Summary

- Enables scaling of storage independently
- Lower TCO
- Faster Cluster recovery time
- Better data resilience from node failures
- Seamless data introduction

Thank You

Don't forget to rate this session in the .conf18 mobile app