Easing Into Clustering

Lisa Guinn

Sr. Instructor, Splunk

.conf2016

splunk>

Disclaimer

During the course of this presentation, we may make forward looking statements regarding future events or the expected performance of the company. We caution you that such statements reflect our current expectations and estimates based on factors currently known to us and that actual events or results could differ materially. For important factors that may cause actual results to differ from those contained in our forward-looking statements, please review our filings with the SEC. The forward-looking statements made in the this presentation are being made as of the time and date of its live presentation. If reviewed after its live presentation, this presentation may not contain current or accurate information. We do not assume any obligation to update any forward looking statements we may make. In addition, any information about our roadmap outlines our general product direction and is subject to change at any time without notice. It is for informational purposes only and shall not, be incorporated into any contract or other commitment. Splunk undertakes no obligation either to develop the features or functionality described or to include any such feature or functionality in a future release.

Introduction

.conf2016

splunk>

Agenda

- Conceptual Overview: How indexer clusters work
  - Buckets and replication
  - Cluster components
- Clustering without replication - what?
  - Getting the infrastructure management without using the disk space
- Evolving to replication
  - Moving to a "real" replication level
  - Moving to multi-site clustering

This talk only covers **indexer clustering** – *not* search head clustering

About Me

- Splunk Senior Instructor since 2009
- Passionate about solving problems with Splunk
  - #7 on Splunk Answers and proud of it!
- Has a hoodie from every .conf
  - But gave up the goal of owning every Splunk t-shirt
- Find me at the Answers Desk at .conf
  and introduce yourself!

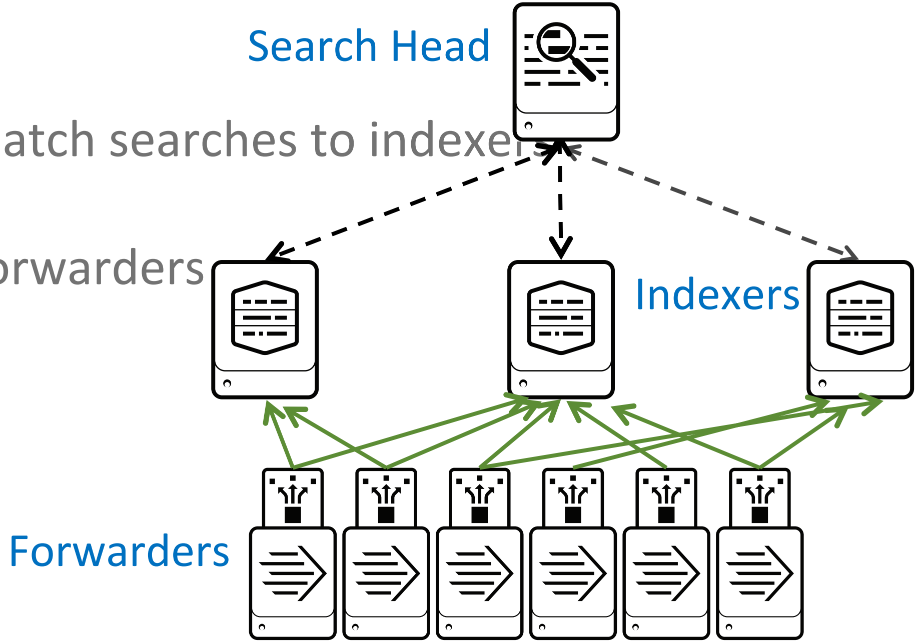
How Indexer Clusters Work

.conf2016

splunk>

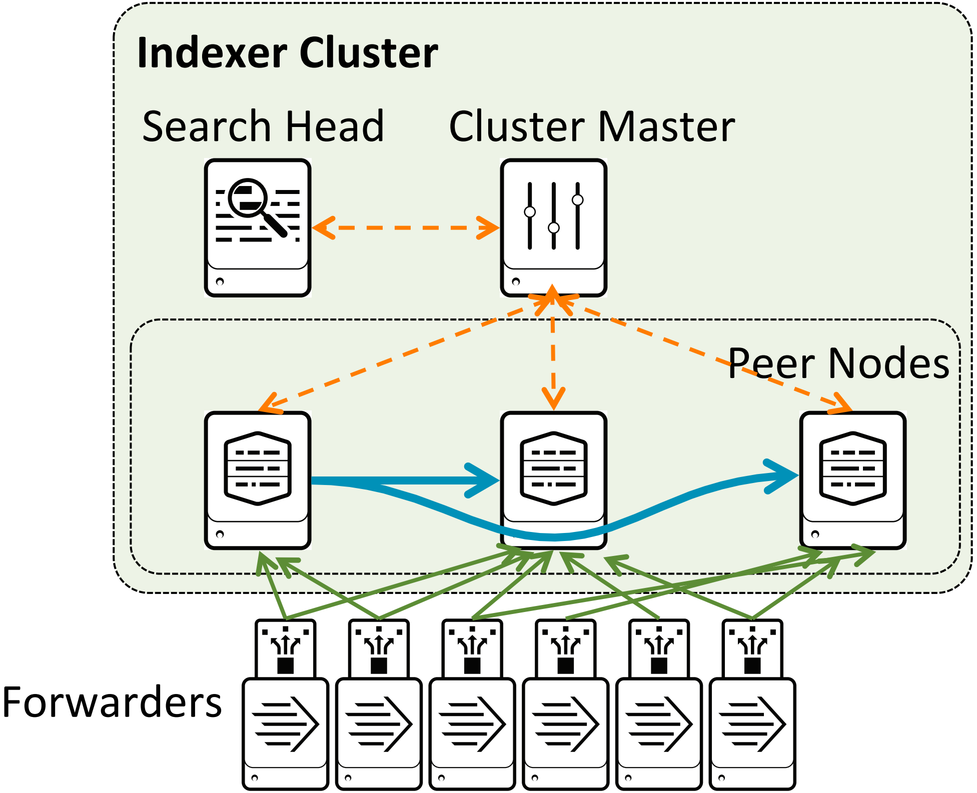
Splunk Distributed Environment

- **Not clustered**
- Search Head
  - Uses distributed search to dispatch searches to indexers
- Indexers
  - Receive and index data from forwarders
  - Respond to search requests
- Forwarders
  - Sends data to peer nodes
  - Uses load balancing (default)



Single-site Indexer Cluster Overview

- Cluster Master
  - Controls replication & recovery
  - Provides list of indexers to forwarders and search heads
- Peer nodes (indexers)
  - Index and search
  - Replicate data to other peers
- Search head
  - Normal Splunk search head
- Forwarders
  - Normal forwarders
  - Load balancing and useACK

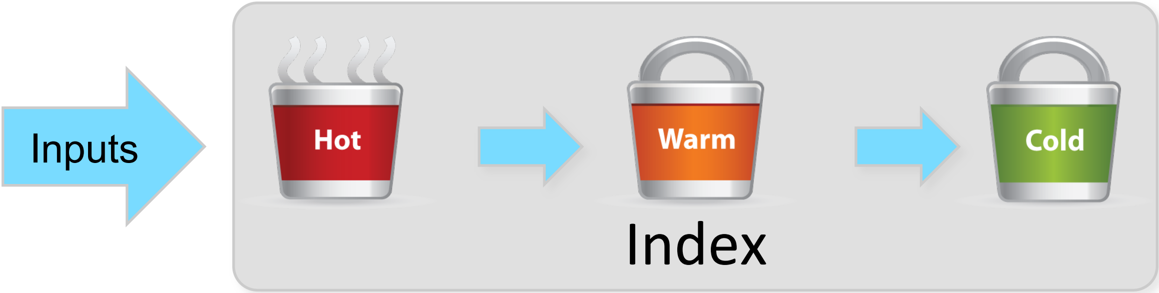


Cluster Master

- Cluster master (CM) provides services to each tier of a cluster
- For Search Heads
  - CM supplies a list of indexers to search
  - Search heads do *not* need to configure distributed search
- For Indexers
  - CM maintains a master set of configuration apps, which are *pushed* to all indexers
  - Deployment server must *not* be used for clustered indexers
- For Forwarders
  - CM provides Indexer Discovery
  - Forwarders do *not* need to maintain a list of indexers in outputs.conf

What's A Bucket?

- Splunk stores data in indexes
- Indexes are composed of buckets
- Indexer clustering replicates buckets



What's In A Bucket?

- rawdata
  - actual event data
  - essential information (host, source, sourcetype, etc.) for each event is included
  - stored in compressed form
- Index files
  - keyword indices
  - bloom filters
  - everything needed for searching
  - usually about 3x the size of rawdata (can vary widely)

Replication and Search Factors

- Peer nodes copy buckets to each other
  - but you choose whether to copy the entire bucket, or just the rawdata
- Replication factor (RF)
  - Specifies how many total copies of rawdata
  - This sets the total failure tolerance level
- Search factor (SF)
  - Specifies how many copies will be searchable
    - Buckets will have both rawdata and index files
  - Determines how quickly you can recover the search capability

How many indexers can go down before

- data is lost?
- users cannot search?

Configuring Splunk Cluster Master

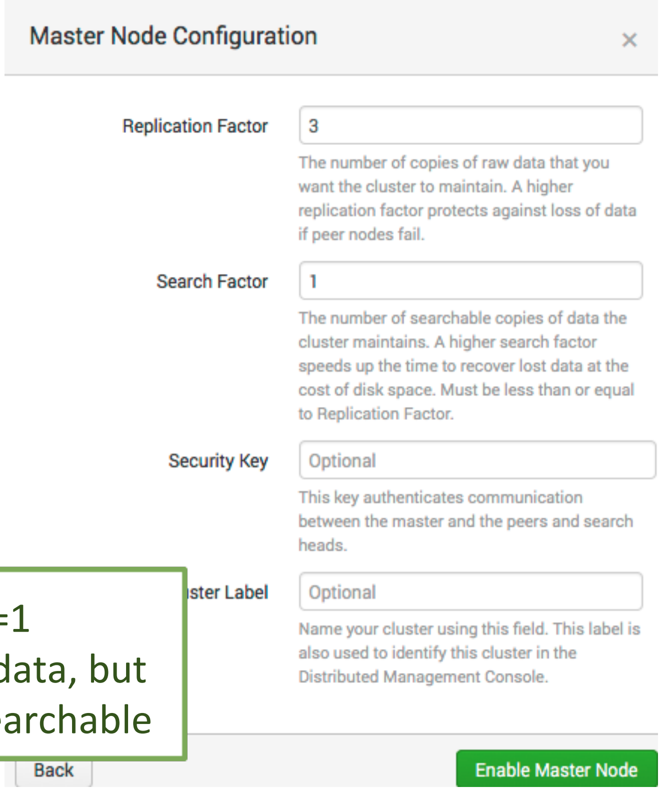
- Settings -> Indexer Clustering
  - Replication Factor
  - Search Factor
  - Security Key
  - Cluster Label

Results in:

SPLUNK_HOME/etc/system/local/server.conf

```
[clustering]
mode = master
replication_factor = 3
search_factor = 1
pass4SymmKey = Hashed_Secret
```

RF=3, SF=1
3 copies of rawdata, but
only 1 copy is searchable



Master Node Configuration

Replication Factor: 3

The number of copies of raw data that you want the cluster to maintain. A higher replication factor protects against loss of data if peer nodes fail.

Search Factor: 1

The number of searchable copies of data the cluster maintains. A higher search factor speeds up the time to recover lost data at the cost of disk space. Must be less than or equal to Replication Factor.

Security Key: Optional

This key authenticates communication between the master and the peers and search heads.

ster Label: Optional

Name your cluster using this field. This label is also used to identify this cluster in the Distributed Management Console.

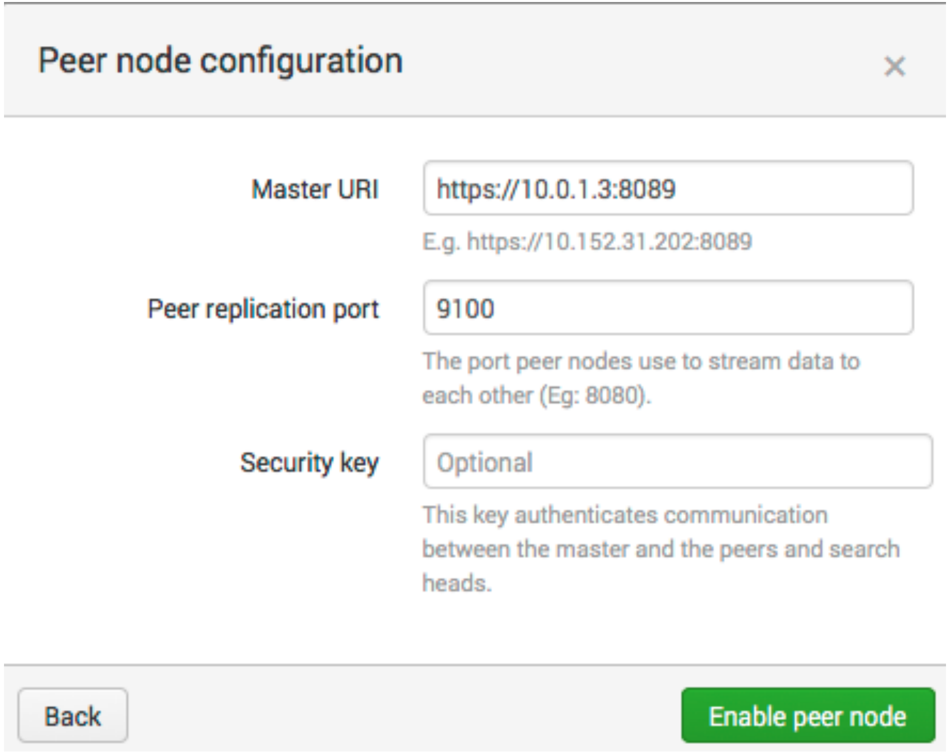
Configuring Splunk Cluster Peers

- Settings -> Indexer Clustering
  - Cluster Master
  - Peer Replication Port
  - Security Key

SPLUNK_HOME/etc/system/local/server.conf

```
[clustering]
mode = slave
master_uri = https://10.0.1.3:8089
pass4SymmKey = Hashed_Secret

[replication_port://9100]
```



Peer node configuration

Master URI: https://10.0.1.3:8089

E.g. https://10.152.31.202:8089

Peer replication port: 9100

The port peer nodes use to stream data to each other (Eg: 8080).

Security key: Optional

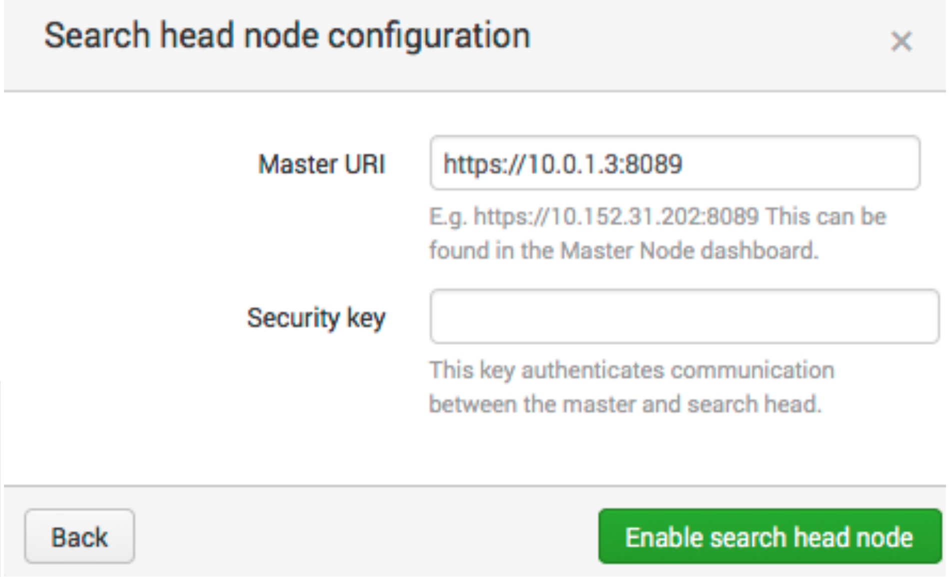
This key authenticates communication between the master and the peers and search heads.

Configuring Search Heads

- Settings -> Indexer Clustering
  - Cluster Master
  - Security Key

SPLUNK_HOME/etc/system/local/server.conf

```
[clustering]
mode = searchhead
master_uri = https://10.0.1.3:8089
pass4SymmKey = Hashed_Secret1
```



Search head node configuration

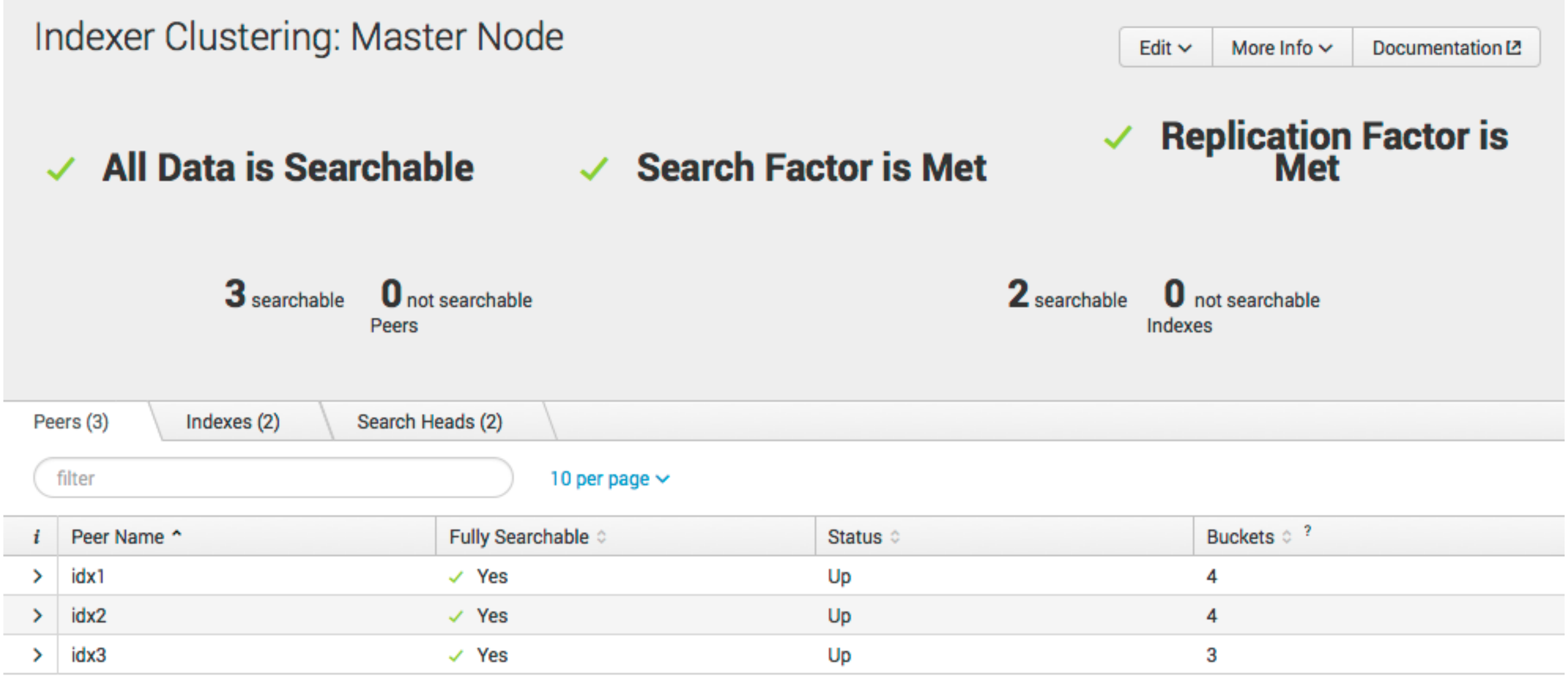
Master URI: https://10.0.1.3:8089

E.g. https://10.152.31.202:8089 This can be found in the Master Node dashboard.

Security key:

This key authenticates communication between the master and search head.

Master Dashboard – Single-site Cluster



Indexer Clustering: Master Node

✓ All Data is Searchable

✓ Search Factor is Met

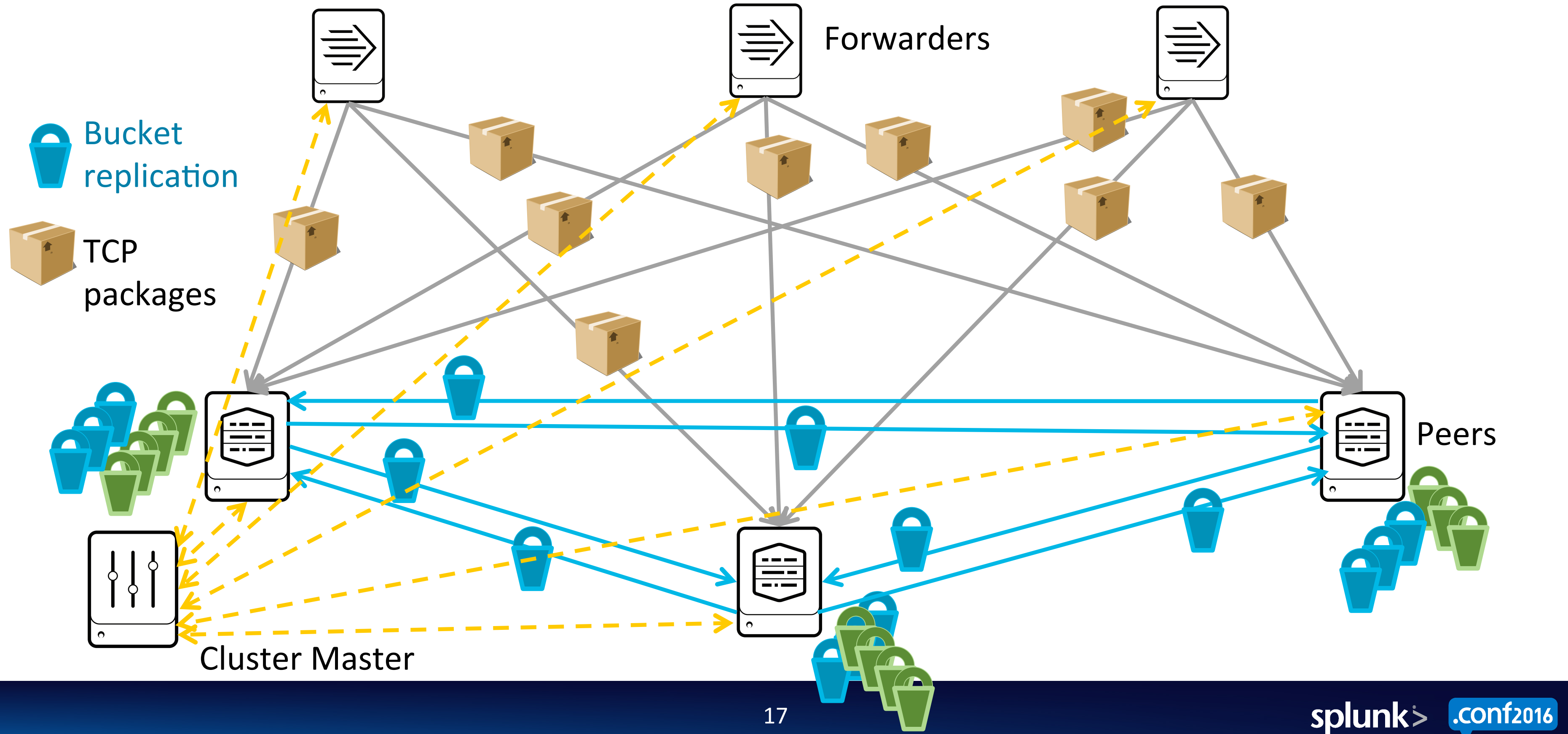
✓ Replication Factor is Met

3 searchable 0 not searchable
Peers

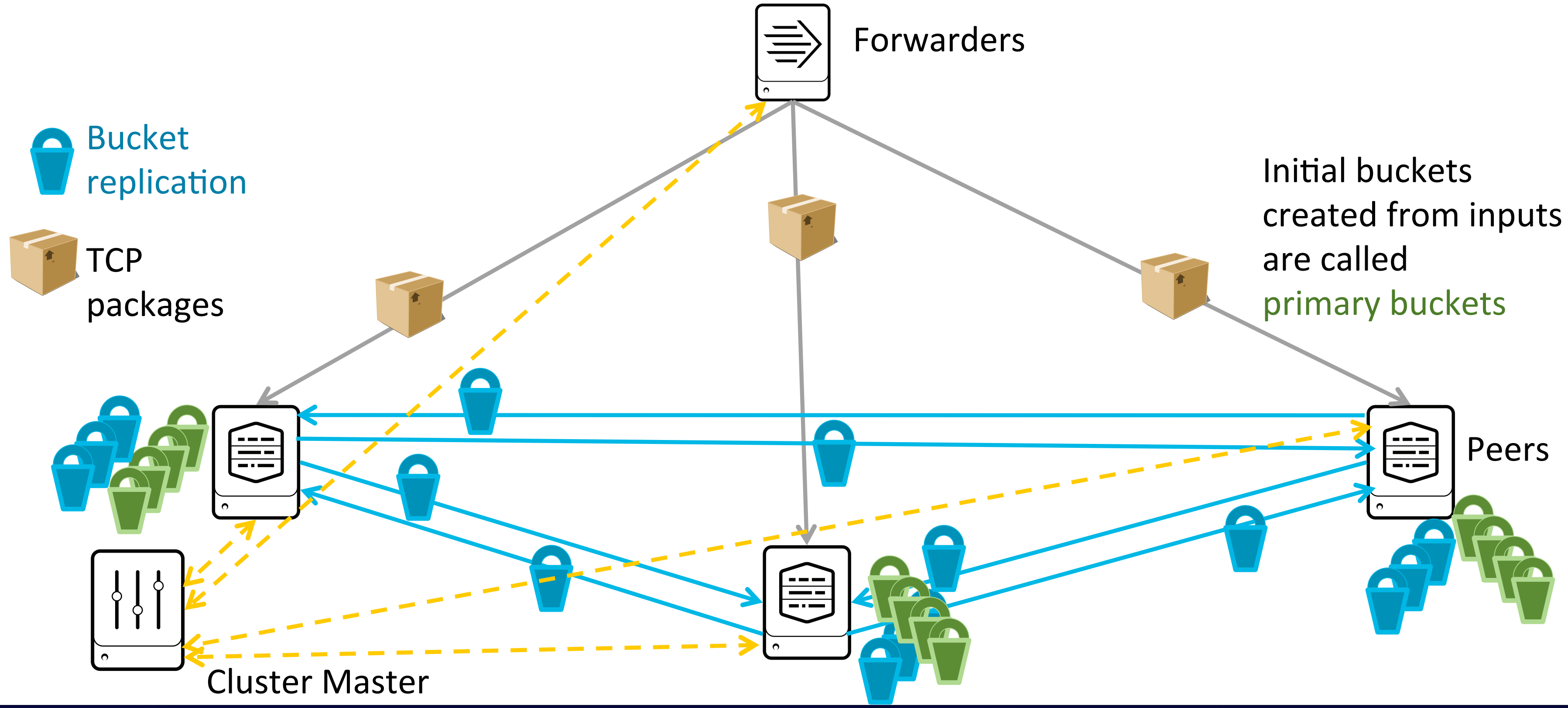
2 searchable 0 not searchable
Indexes

i	Peer Name ˄	Fully Searchable	Status	Buckets ?
›	idx1	✓ Yes	Up	4
›	idx2	✓ Yes	Up	4
›	idx3	✓ Yes	Up	3

Replication in Action - Normal



Replication In Action – Just One Forwarder



Forwarders

Bucket replication

TCP packages

Initial buckets created from inputs are called primary buckets

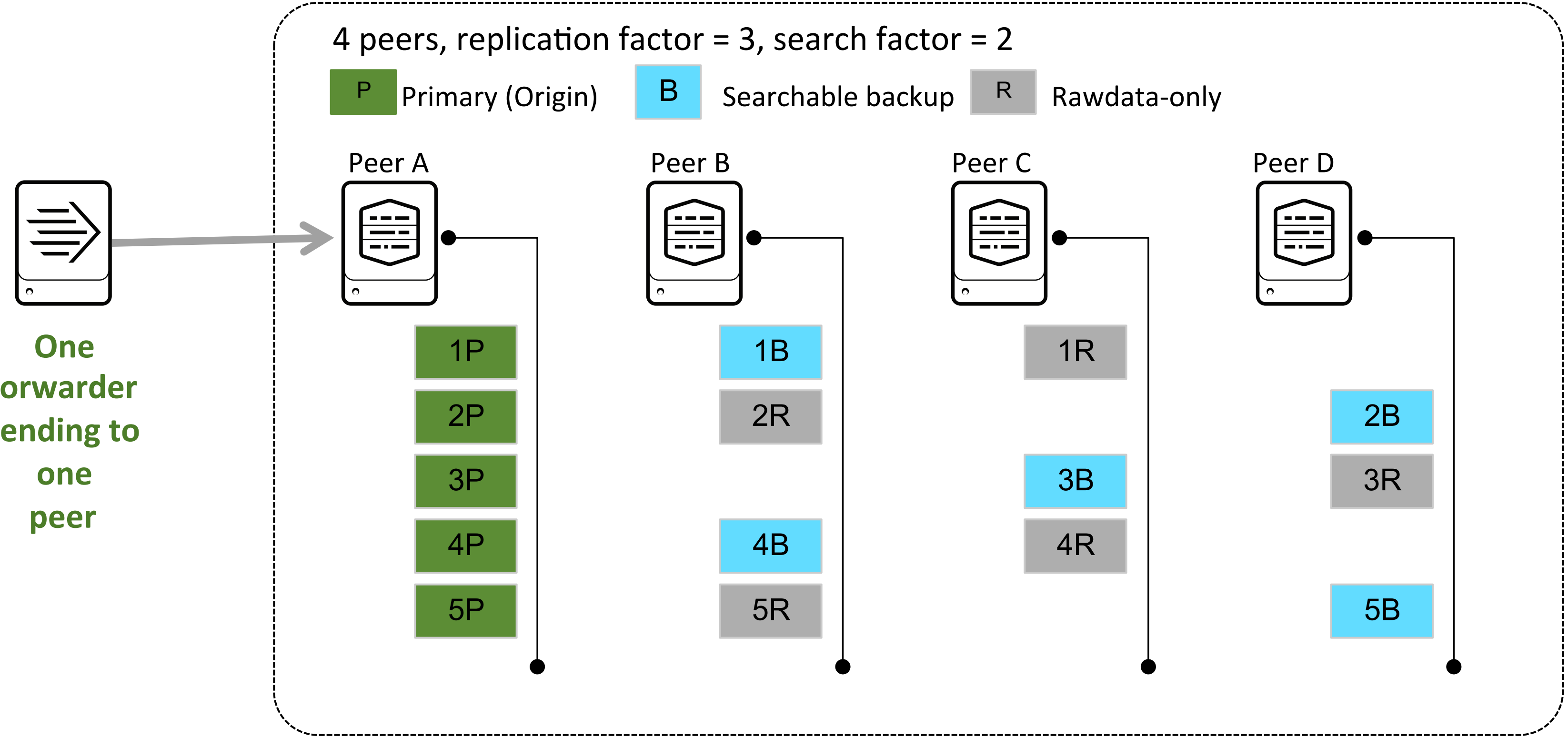
Peers

Cluster Master

Replication In Action – A Simplified View

4 peers, replication factor = 3, search factor = 2

P Primary (Origin) B Searchable backup R Rawdata-only

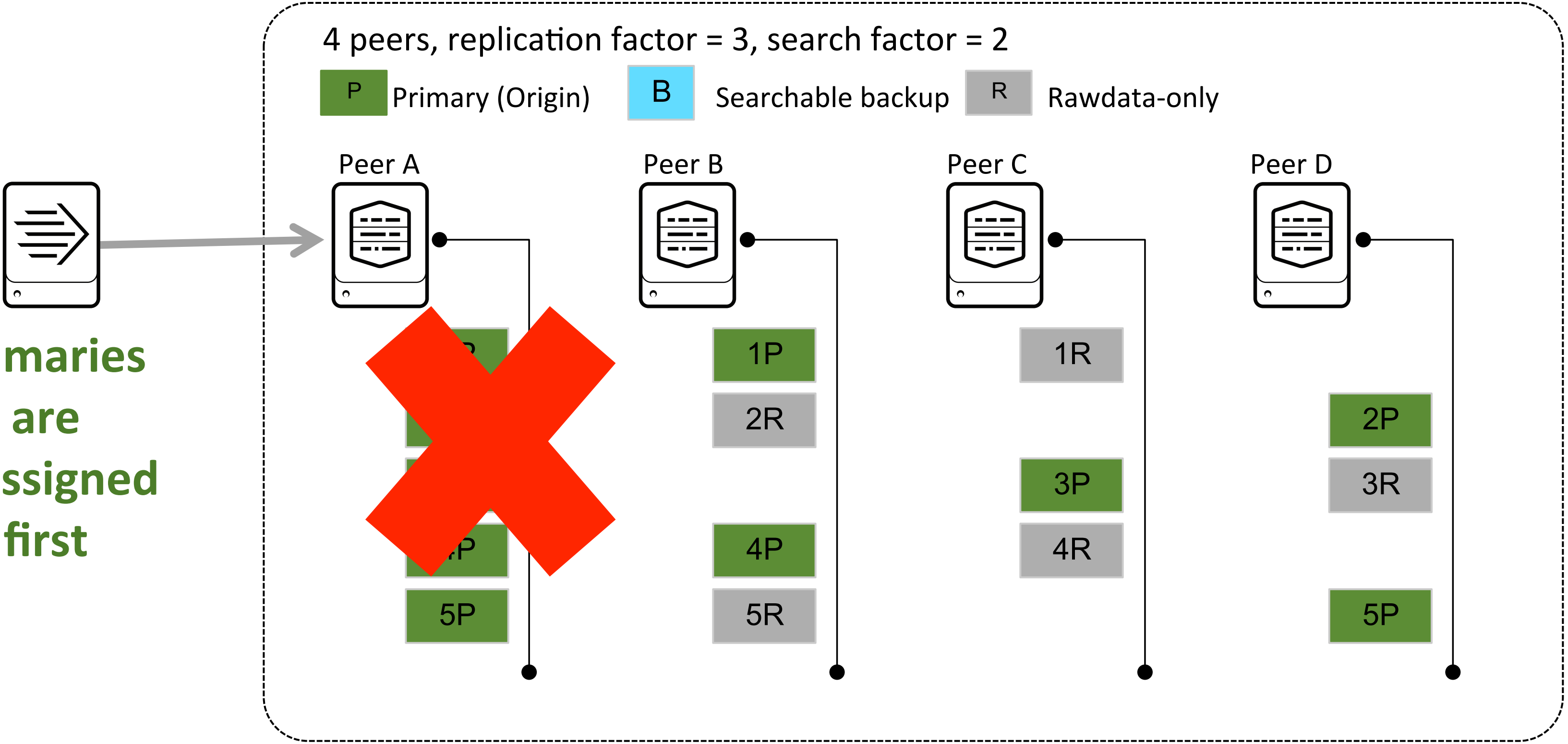


One forwarder sending to one peer

Peer A	Peer B	Peer C	Peer D
1P	1B	1R	
2P	2R		2B
3P		3B	3R
4P	4B	4R	
5P	5R		5B

Peer Loss

Primaries are reassigned first

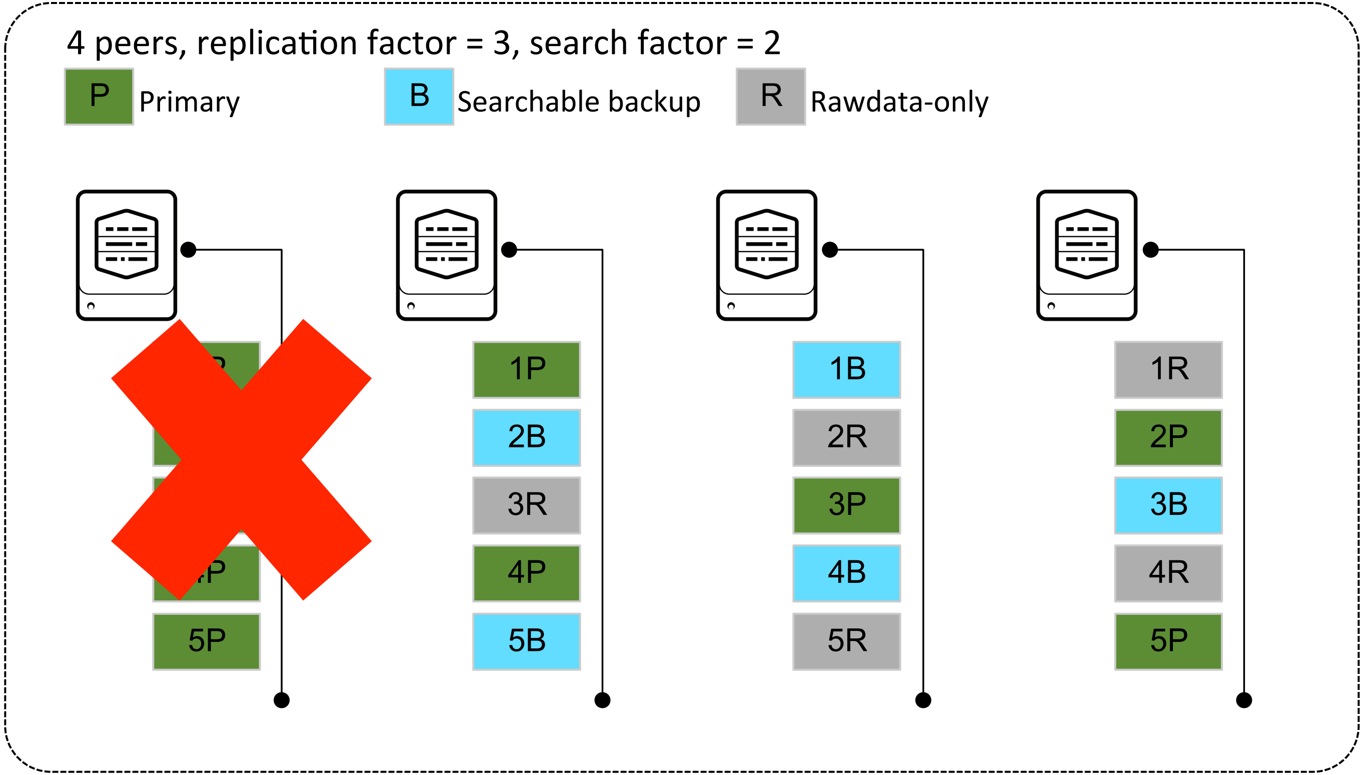


4 peers, replication factor = 3, search factor = 2

Recovery Complete

Complete & Valid

after recovery is complete

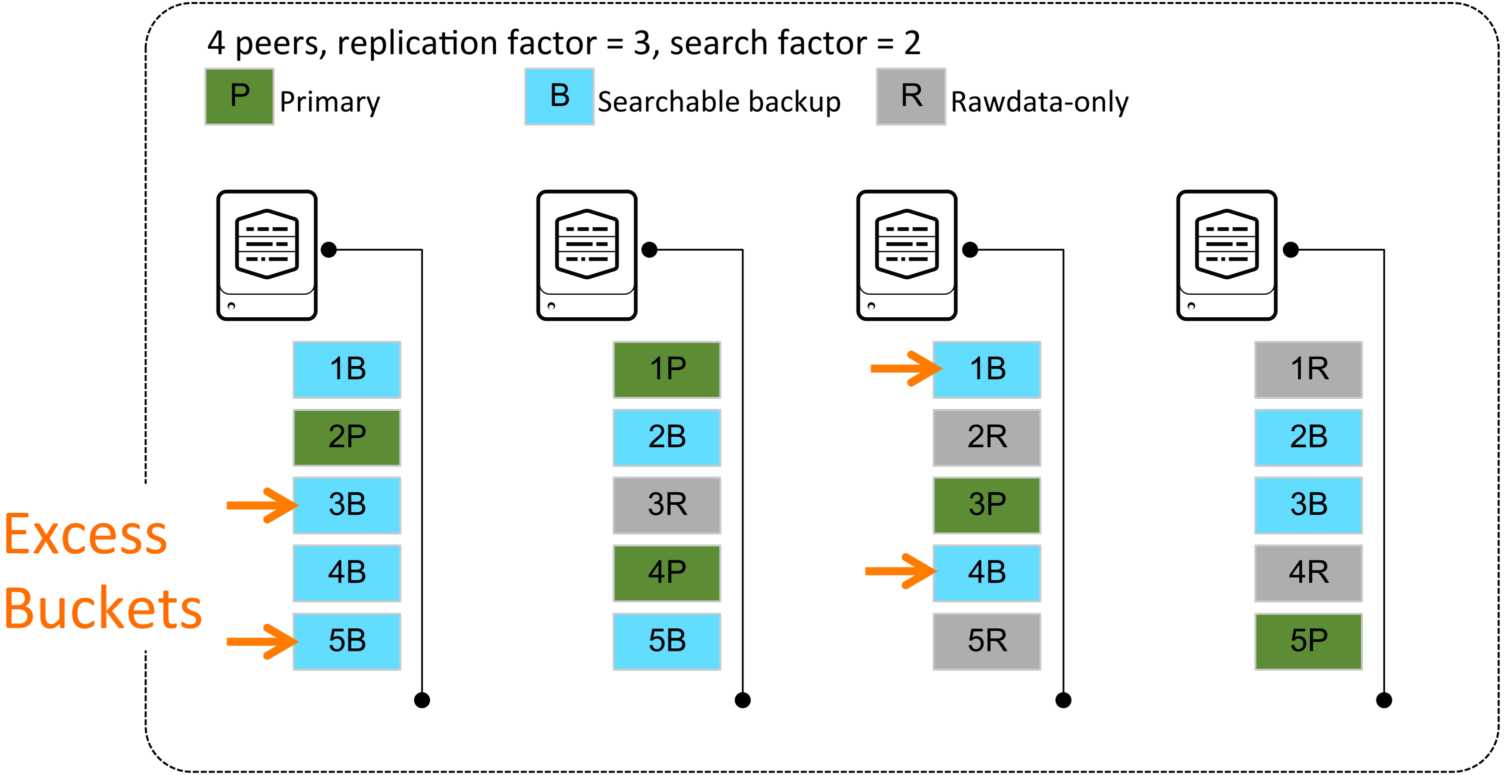


4 peers, replication factor = 3, search factor = 2

P Primary B Searchable backup R Rawdata-only

P	1P	1B	1R
	2B	2R	2P
	3R	3P	3B
4P	4P	4B	4R
5P	5B	5R	5P

Peer Restored



4 peers, replication factor = 3, search factor = 2

P Primary B Searchable backup R Rawdata-only

1B	1P	→ 1B	1R
2P	2B	2R	2B
→ 3B	3R	3P	3B
4B	4P	→ 4B	4R
→ 5B	5B	5R	5P

Excess Buckets

Indexer Clustering Without Replication

.conf2016

splunk>

Cost Of Replication

- Disk space
  - for replicated copies
  - to support recovery if peer(s) lost
- Overhead on peers
  - usually small, but larger factors create more overhead
  - may stress a fully-loaded environment

What if you can't afford replication?

Without Replication

- Set replication factors to 1
- **All the management benefits**
  - Search head queries CM for indexers to search
  - Forwarders query CM for indexers for forwarded outputs
  - Indexer configurations are managed by the cluster master
- **None of the data replication benefits**
  - No improvement in search availability
  - No protection from data loss

Cluster Master:
`SPLUNK_HOME/etc/system/local/server.conf`

```
[clustering]
mode = master
replication_factor = 1
search_factor = 1
pass4SymmKey = Hashed_Secret
```

Increasing Replication

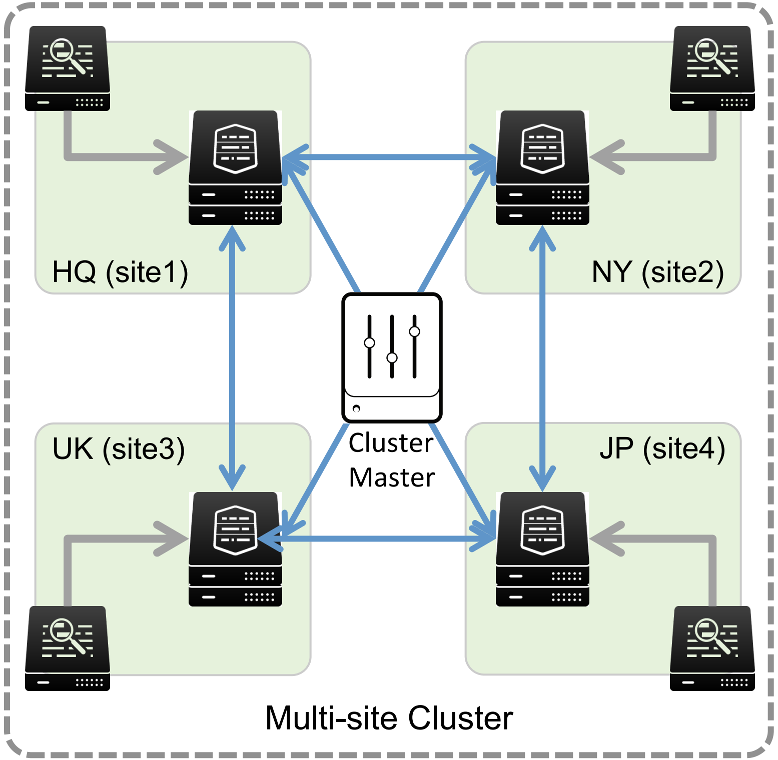
- Seems easy!
- What happens…
  - Cluster goes into recovery mode
  - Disk space used by indexes doubles (in this example)
  - Indexers become busy "catching up" to the new factors
- This is **not** what we intended!
- Multi-site indexer clustering to the rescue!

Cluster Master:
SPLUNK_HOME/etc/system/local/server.conf

```
[clustering]
mode = master
replication_factor = 2
search_factor = 2
pass4SymmKey = Hashed_Secret
```

Multisite Indexer Cluster

- Provides extra layer of data partitioning
  - Indexers are grouped in "sites" (defined by the Splunk Admin)
- Multisite cluster benefits
  - Disaster recovery
    - stores index copies at multiple sites
    - provides automatic site-failover
  - Search affinity
    - searches assigned site as much as possible
    - greatly reduces WAN network traffic



But Why Does It Work For Migration?

- Existing single-site replication buckets are **not** converted
  - Existing buckets are maintained
  - Eventually they will age out from the indexes
- New buckets will follow the new multi-site replication rules
- Therefore, the cluster will *not* enter recovery mode

http://docs.splunk.com/Documentation/Splunk/6.4.2/Indexer/Migratetomultisite

Requirements for MultiSite Indexer Clustering

- One Cluster Master
- Two sites with a minimum of two indexers at each site
- One search head per site
- Remember that you can define "site" in any way that you want!

Cluster Master:
SPLUNK_HOME/etc/system/local/server.conf

```
[clustering]
mode = master
replication_factor = 2
search_factor = 2
multisite = true
available_sites = site1,site2
site_replication_factor = origin:2,total:
3 site_search_factor = origin:1,total:2
pass4SymmKey = Hashed_Secret
```

MultiSite Factors

- Basic configuration
  - site_replication_factor = origin:2,total:3
  - site_search_factor = origin:1,total:2

- More specific configuration
  - site_replication_factor = origin:2,site1:1,total:3
  - site_search_factor = origin:1,site1:1,total:2

Summary

- Everyone should consider indexer clustering
  - protection from data loss
  - maintain search availability
  - management of indexers and forwarders
- Have a plan for growth and change

What Next?

- Ask questions here for a few minutes!
- Visit the Answers Desk and let's chat
  - If not here at .conf, on http://answers.splunk.com

- Other Sessions
  - Indexer Clustering Internals, Scaling, and Performance (Tuesday, 3:15 pm)
  - Search Head Clustering – Basics to Best Practices (Thursday, 1:30 pm)

THANK YOU

.conf2016

splunk>