Architecting Splunk For High Availability And Disaster Recovery

Dritan Bitincka

Principal Architect, 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.

About Me

- Member of Splunk Tech Services
- +5 Years at Splunk
- Large scale and Cloud deployments
- 6th .conf

Agenda

Disaster Recovery

Recover in the event of a disaster

High Availability

- Data Collection
- Indexing & Searching

Maintain an acceptable level of continuous service

Top Takeaways

Disaster Recovery (DR)

.conf2016

splunk>

What Is Disaster Recovery?

Set of processes necessary to ensure recovery of service after a disaster

Disaster Recovery Steps

1

Backup necessary data

Backup to a medium at least as resilient as source

Local Backup vs. Remote

2

Restore

Ensure this works

Backup is worthless without restore

Backup

1

a

Configurations

`$SPLUNK_HOME/etc/*`

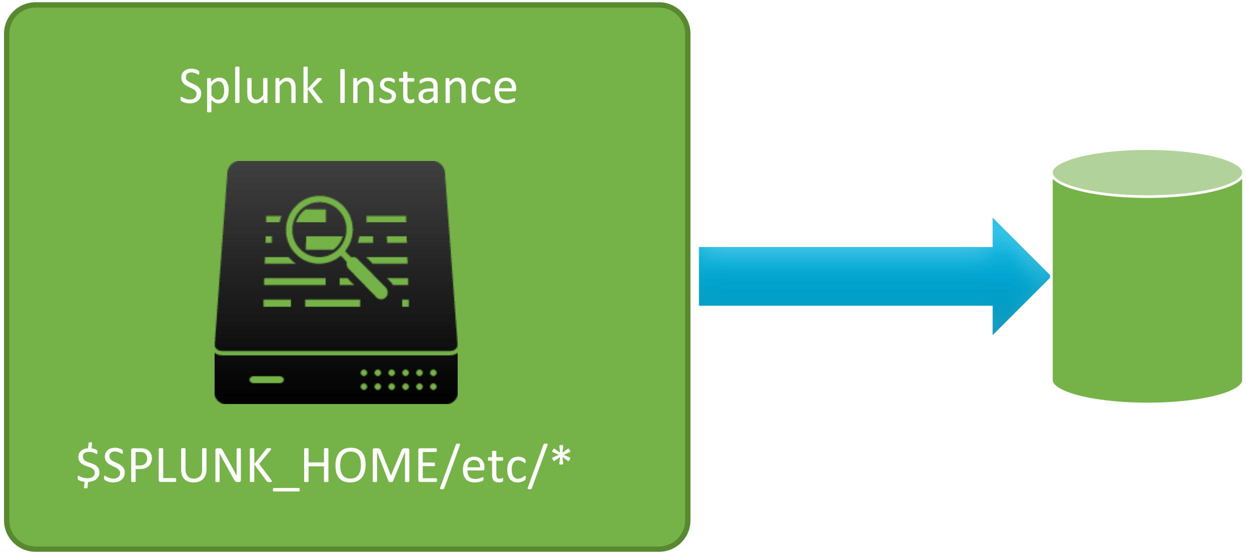
b

Indexes

Buckets: Hot*, Warm, Cold, Frozen

DR

Backup Configurations

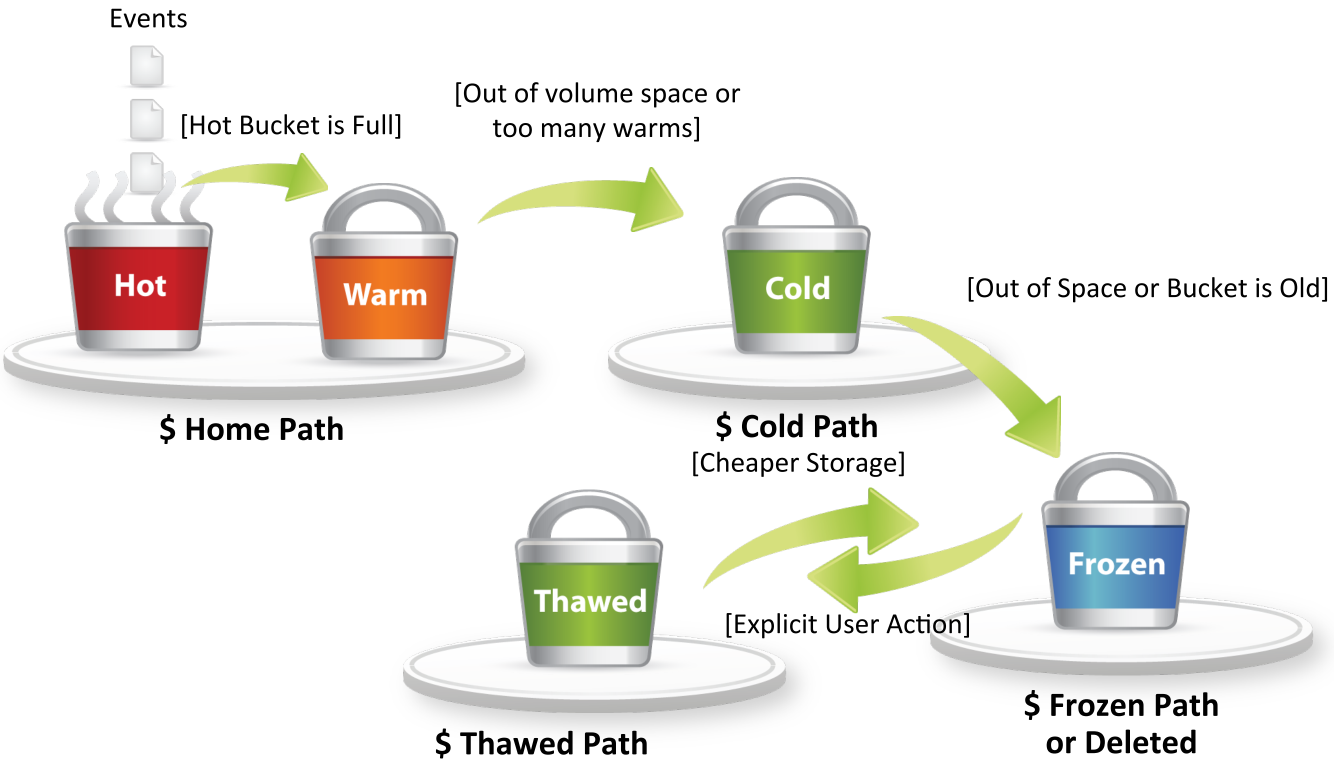


Splunk Instance

$SPLUNK_HOME/etc/*

DR

Backup: Bucket Lifecycle

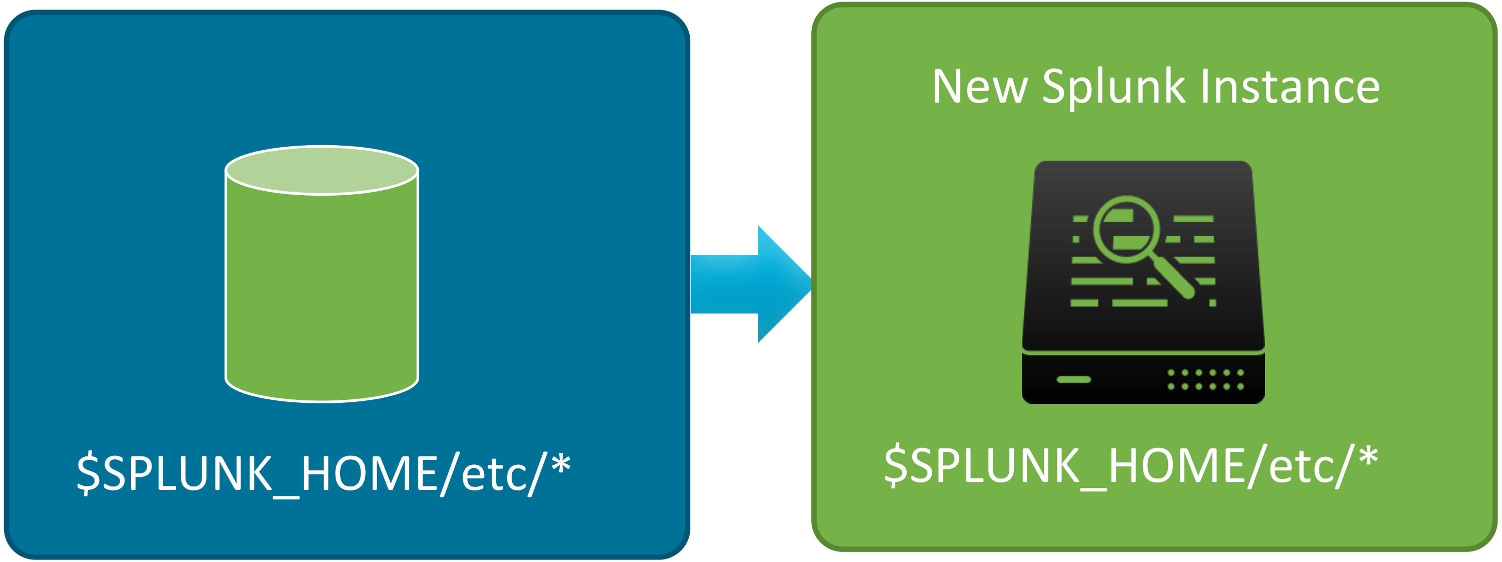


Backup Data

Bucket Type	State	Can Backup?
Hot	Read + Write	No*
Warm	Read Only	Yes
Cold	Read Only	Yes

*Unless using snapshot aware FS (VSS, ZFS) or roll to warm first (which introduces a performance penalty).

Restore Configurations

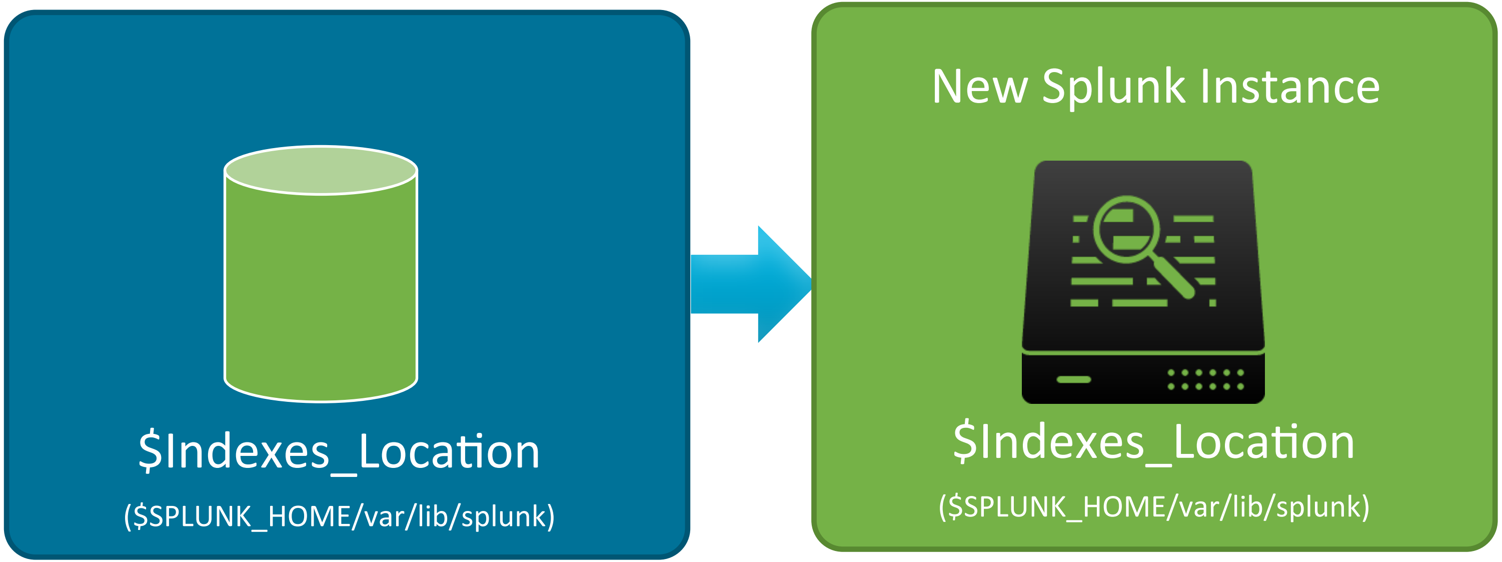


$SPLUNK_HOME/etc/*

New Splunk Instance

$SPLUNK_HOME/etc/*

Restore Data



$Indexes_Location

($SPLUNK_HOME/var/lib/splunk)

New Splunk Instance

$Indexes_Location

($SPLUNK_HOME/var/lib/splunk)

Splunk advises restoring fully from a backup rather than restoring on top of a partially corrupted datastore.

Backup Clustered Data

- **Option 1**: Backup all data on each node
  - Will also result in backups of duplicate data
- **Option 2**: Identify one copy of each bucket on the cluster and backup only those (requires scripting)
  - Decide whether or not you need to also backup index files

Bucket naming conventions

Non-clustered buckets: **db_<newest_time>_<oldest_time>_<localid>**

★Clustered original bucket: **db_<newest_time>_<oldest_time>_<localid>_<guid>**

Clustered replicated bucket copies: **rb_<newest_time>_<oldest_time>_<localid>_<guid>**

Putting Restore Together

2	a	(New) Splunk Instance
	b	Configurations
	c	Data/Indexes

Considerations

Recovery Time and Tolerable Loss

vs.

Complexity and Cost

Other Elements In Your Environment

- Job Artifacts, DM, Collections etc.
- Utility/Management Instances:
  - Deployment Server
  - License Master
  - Cluster Master
  - Deployer

High Availability (HA)

.conf2016

splunk>

What Is High Availability?

A design methodology whereby a system is continuously operational, bounded by a set of predetermined tolerances.

Note: “high availability” !=“complete availability”

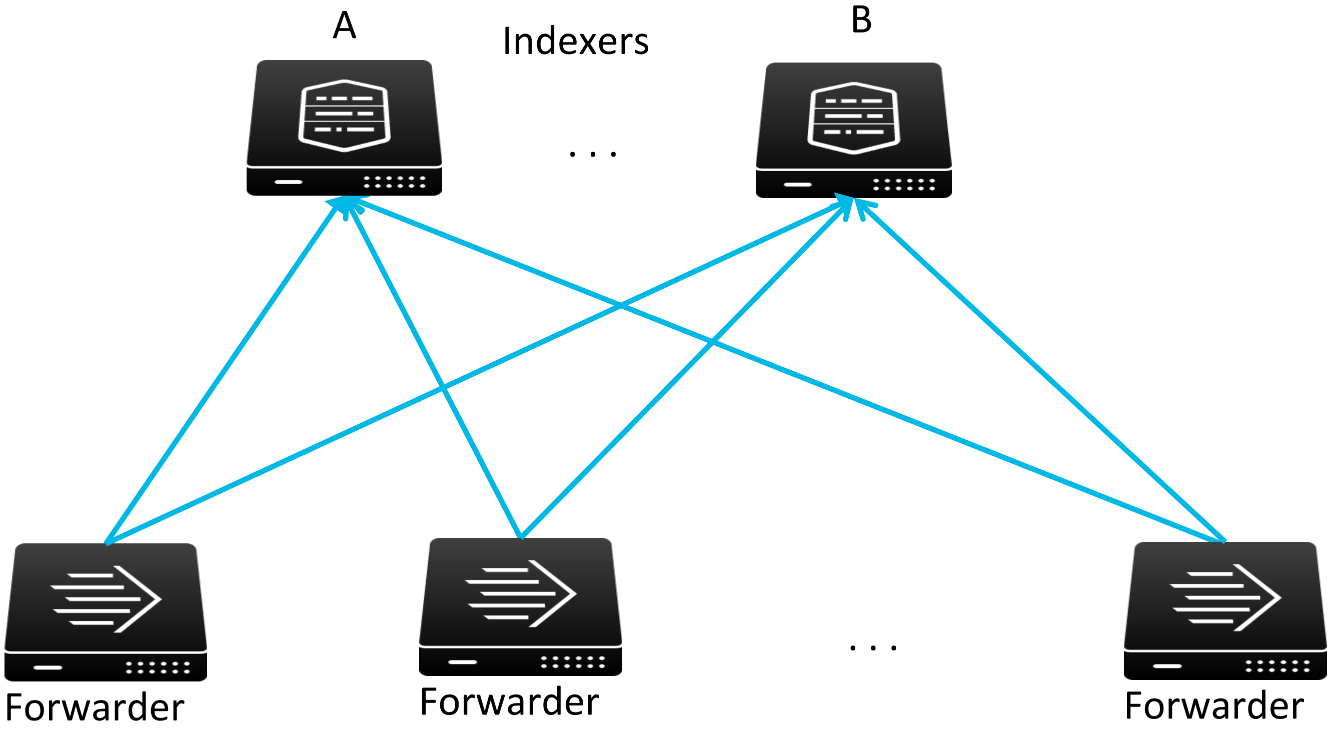
HA

Splunk High Availability

1	Data Collection/Reception
2	Searching
3	Indexing

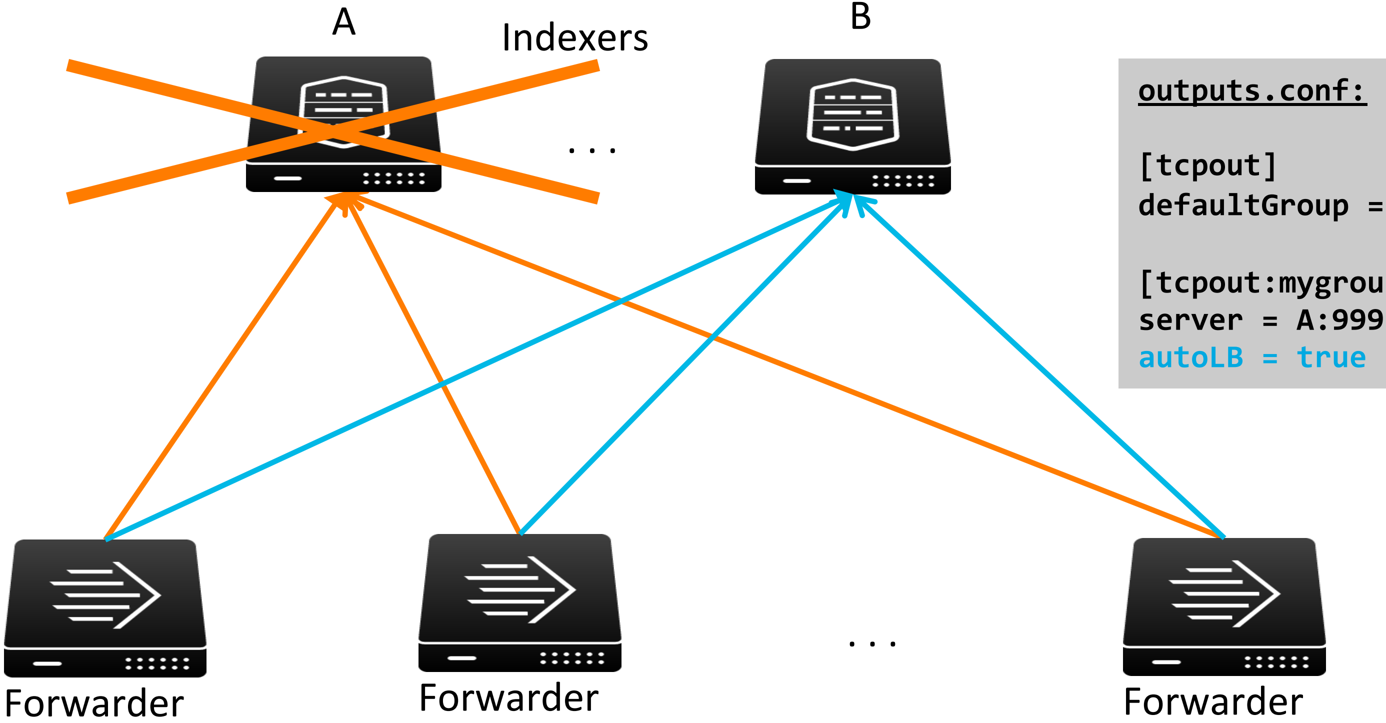
HA

Data Collection



HA

Data Collection



```
outputs.conf:

[tcpout]
defaultGroup = mygroup

[tcpout:mygroup]
server = A:9997, B:9997
autoLB = true
```

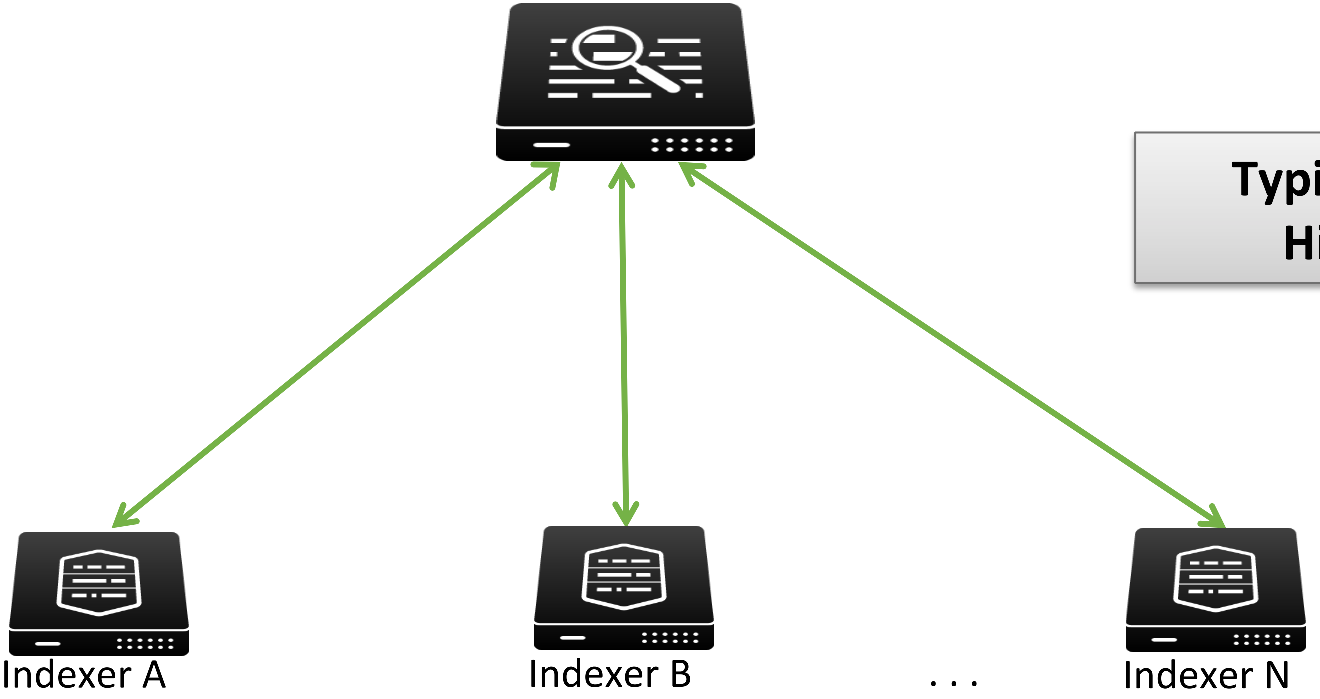
HA

Searching

2	a	Search Head Clustering (SHC)
	b	Search Head Pooling (SHP)

HA

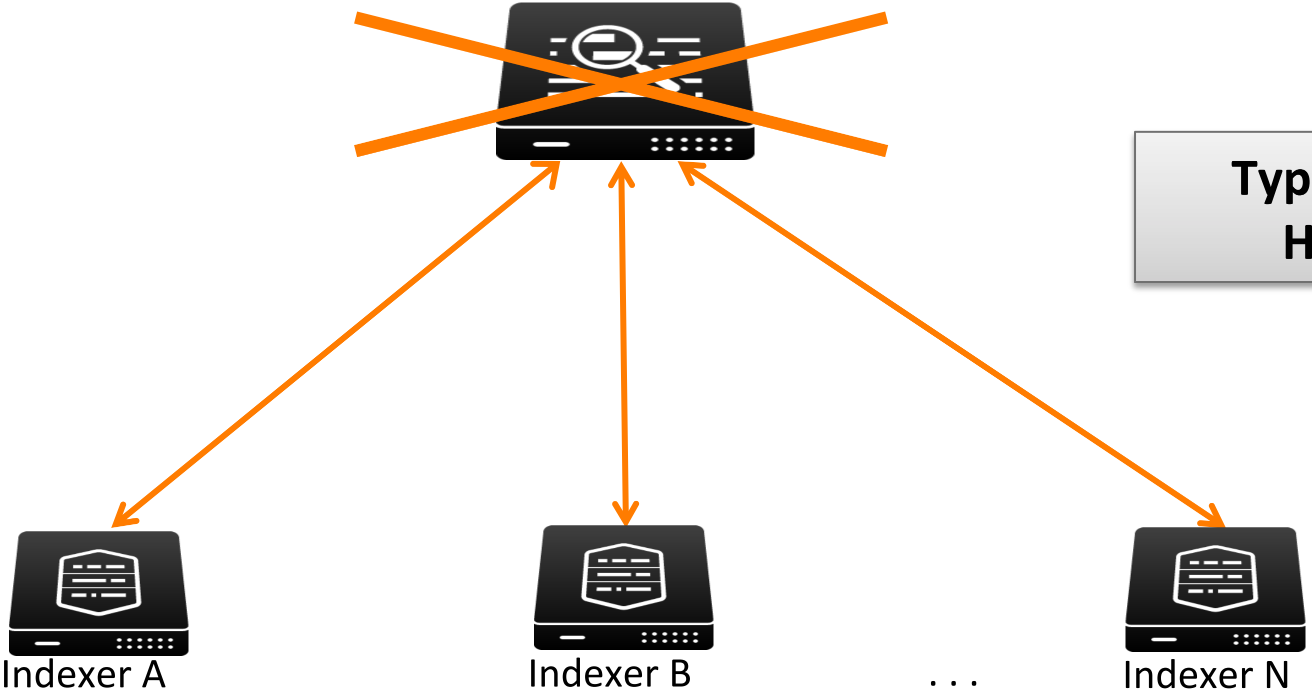
Searching



Typical Search Hierarchy

HA

Searching



Typical Search Hierarchy

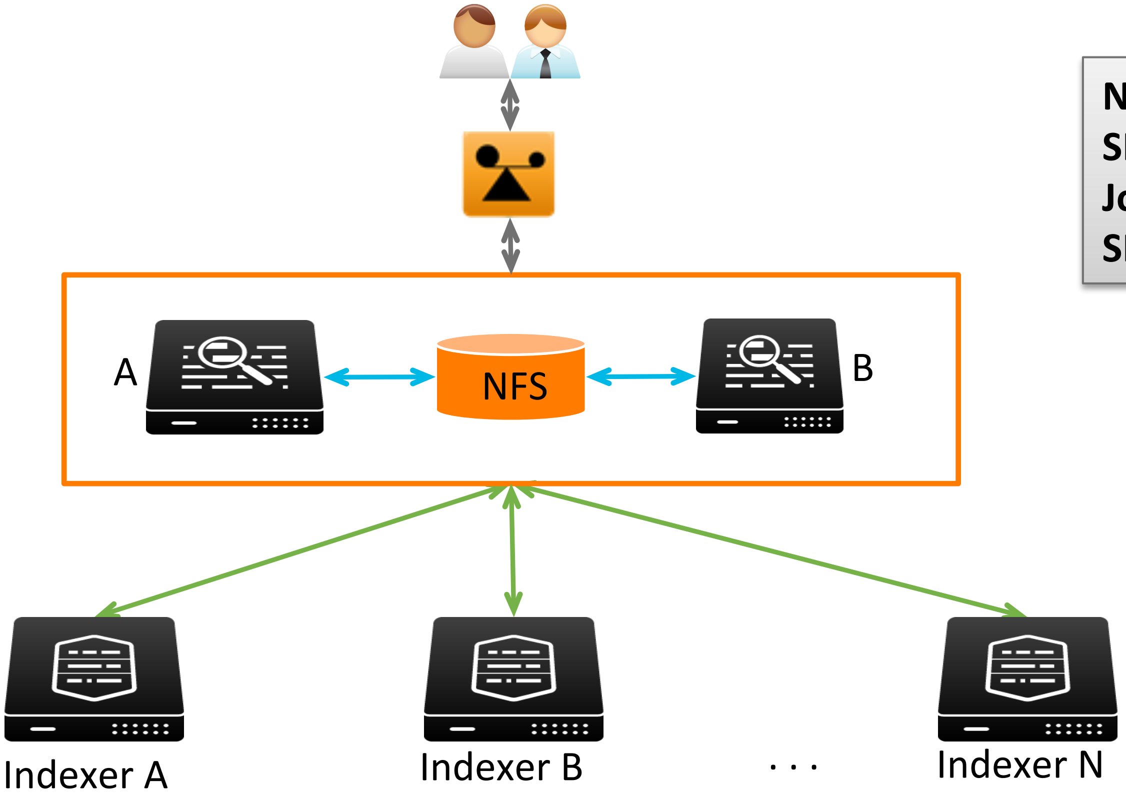
HA

Search Head Pooling

NFS based Search Head Pooling has been **deprecated***

*still works and supported for current Splunk version but plan for its eventual removal.

SHP



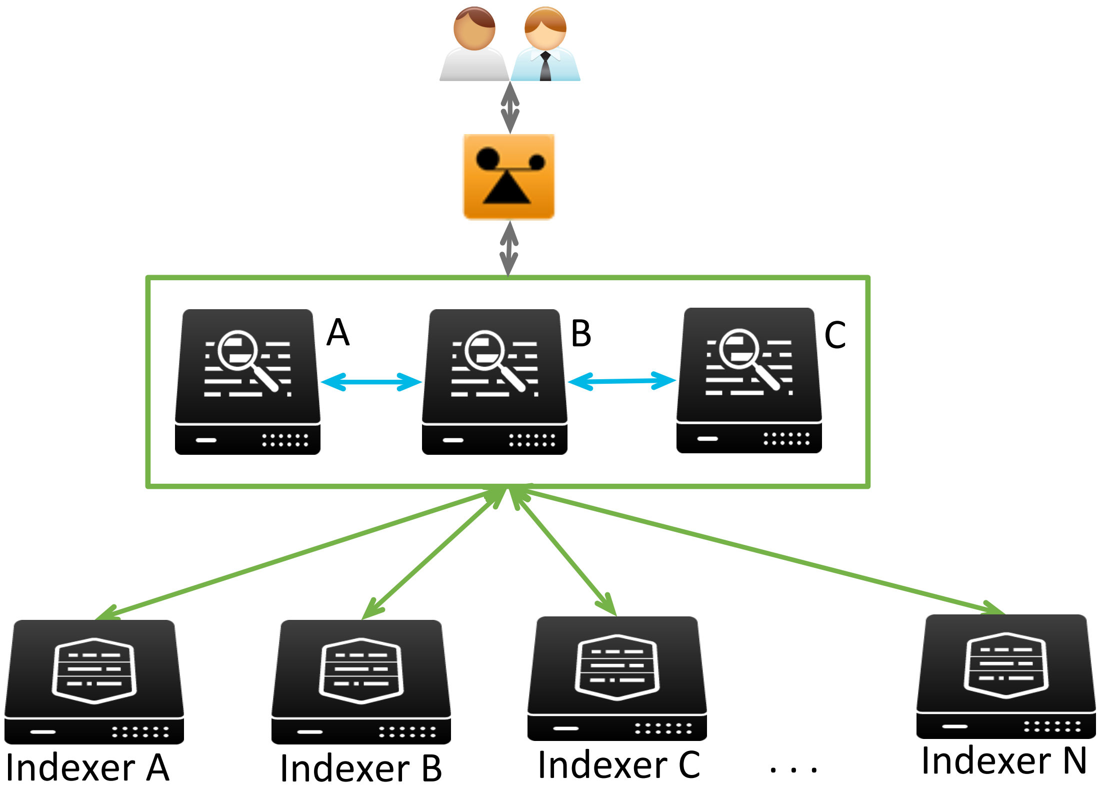
NFS used to sync:
SH Configurations
Job Artifacts
SH Schedulers

HA Search Head Clustering (SHC)

- Improved horizontal scaling
- Improved high availability
- No single point of failure

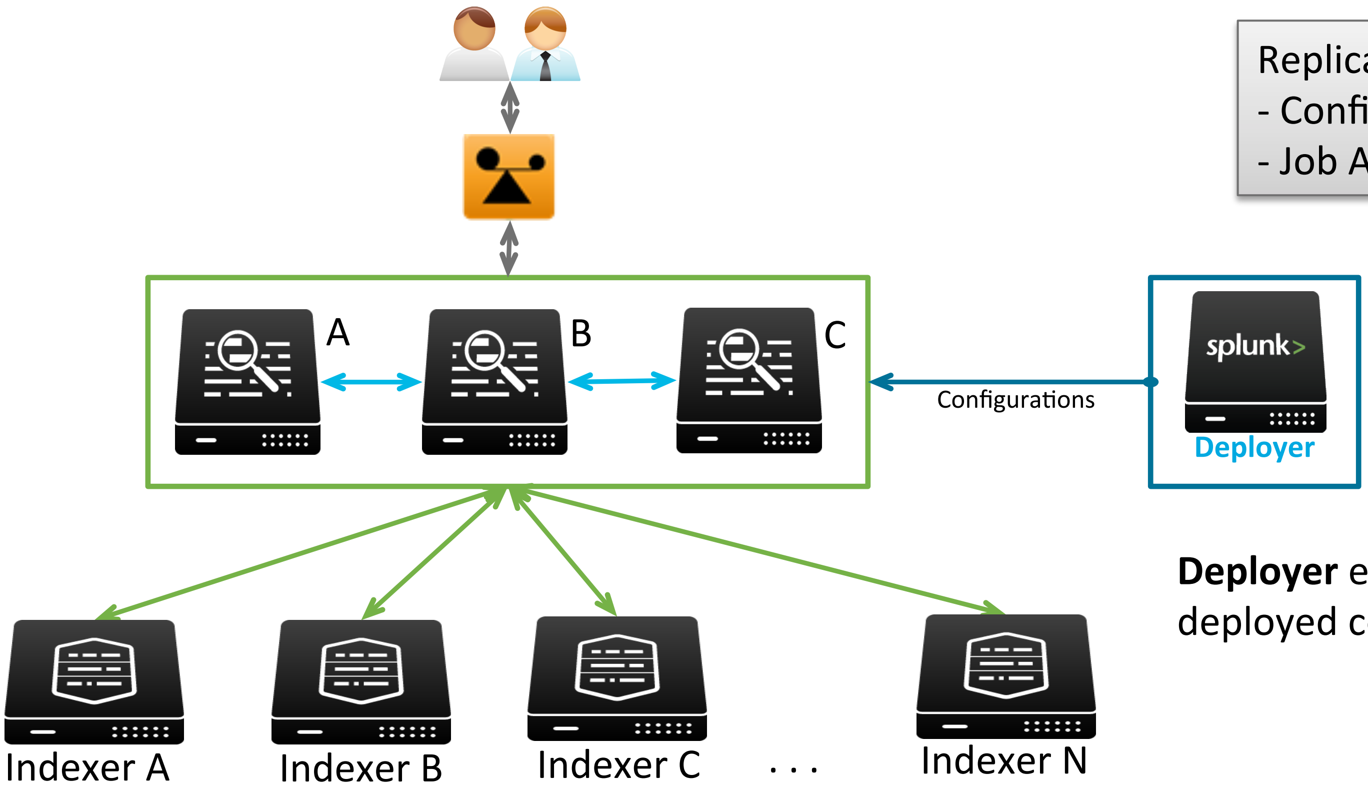
HA

SHC



Replication **protocol** syncs:
- Configurations
- Job Artifacts

SHC

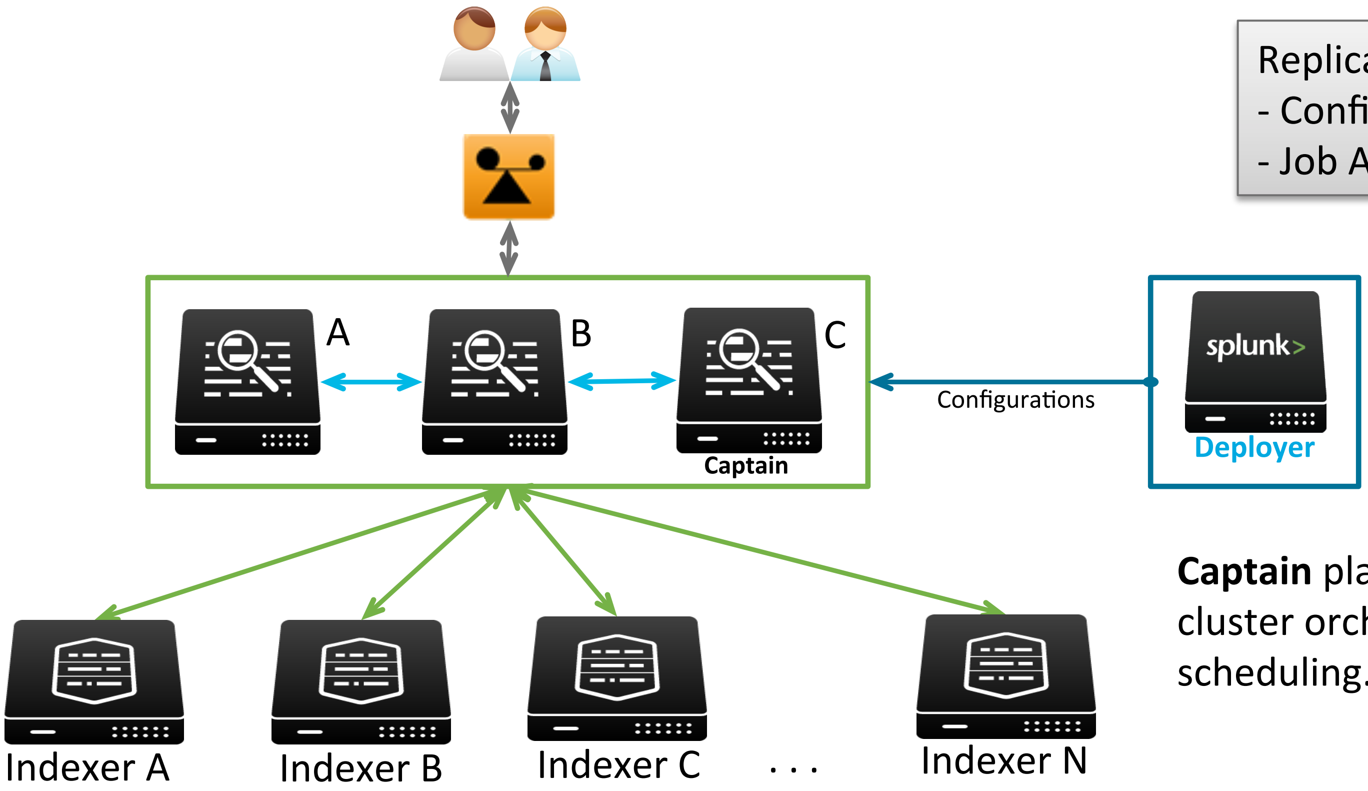


Replication **protocol** syncs:
- Configurations
- Job Artifacts

Deployer ensures identical deployed configurations

SHC

Replication **protocol** syncs:
- Configurations
- Job Artifacts



Captain plays a special role in cluster orchestration and job scheduling.

HA

SHC Operation - High Level

- Deployer ensures all SHC members have identical baseline configurations
  - Subsequent UI changes propagated using an internal replication mechanism
- Job Scheduler gets disabled on all members but the Captain
- Captain selects members to **run scheduled jobs based on load**
  - Selection based on load statistics. Ensures better load distribution vs. SHP
- Captain orchestrates job artifact replication to selected members/ candidates of the cluster
- Transparent job artifact proxying (and eventual replication) if artifact not present on user's SH

SHC Operation - High Level

- Majority requirement and failure handling
  - Surviving majority (>=51%)
- Site-awareness gotchas
  - No notion of **site** in SHC (unlike in index replication)
  - Case for static captain election
- Latency and number of nodes

Deploying SHC

- Same SH version and high speed network (LAN)
  - More storage required vs. stand-alone SHs. Linux/Solaris only
- Needs LB and a Deployer instance (DS or MN can also be used to fulfill this role)
- Select RF per your HA/DR requirements
- Configure Deployer first with a secret key
- Initialize each instance, point them to Deployer, then bootstrap **one** of them to become the cluster captain
- More details on Splunk Docs

HA

Indexing

3

Indexer Clustering

Index Replication

- **Cluster** = a group of search peers (indexers) that replicate each others' buckets
- **Data Availability**
  - Availability for ingestion and searching
- **Data Fidelity**
  - Forwarder Acknowledgement, assurance
- **Disaster Recovery**
  - Site awareness
- **Search Affinity**
  - Local search preference vs. remote

Trade offs

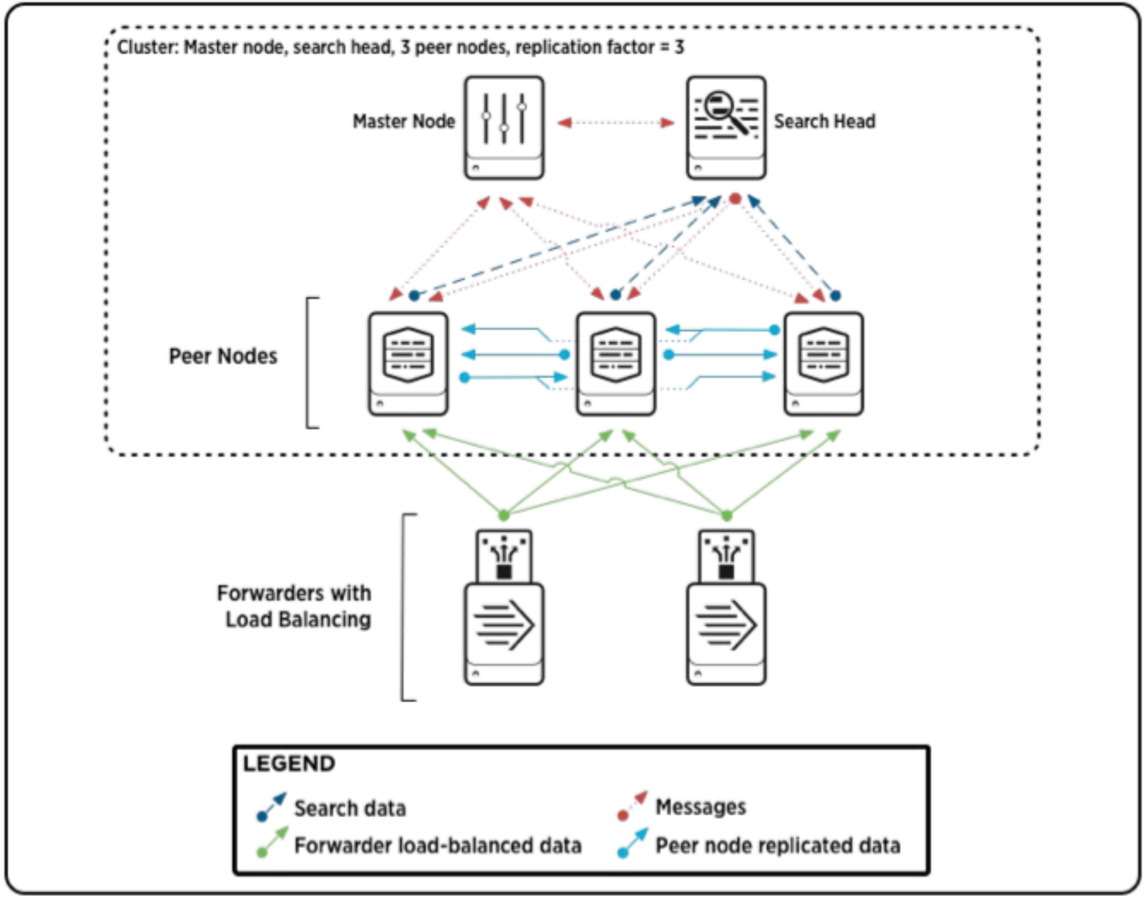
- Extra storage
- Slightly increased processing load

Cluster Components

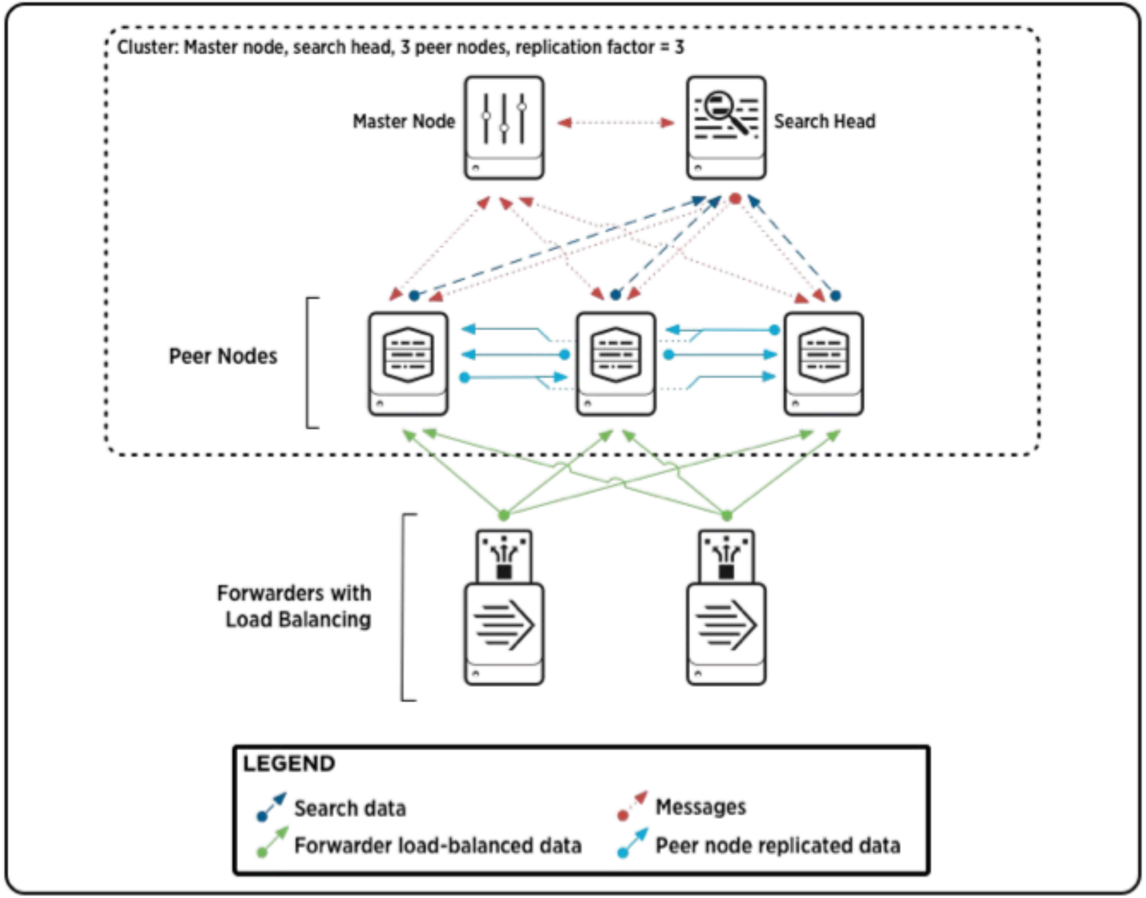
- Master Node
  - Orchestrates replication/remedial process. Informs the SH where to find searchable data. Helps manage peer configurations.
- Peer Nodes
  - Receive and index data. Replicate data to/from other peers. Peer Nodes Number ≥ RF
- Search Head(s)
  - **Must** use one to search across the cluster.
- Forwarders
  - Use with auto-lb and indexer acknowledgement

HA

Single Site Cluster Architecture


Cluster: Master node, search head, 3 peer nodes, replication factor = 3
Master Node
Search Head
Peer Nodes
Forwarders with Load Balancing
LEGEND
Search data
Messages
Forwarder load-balanced data
Peer node replicated data

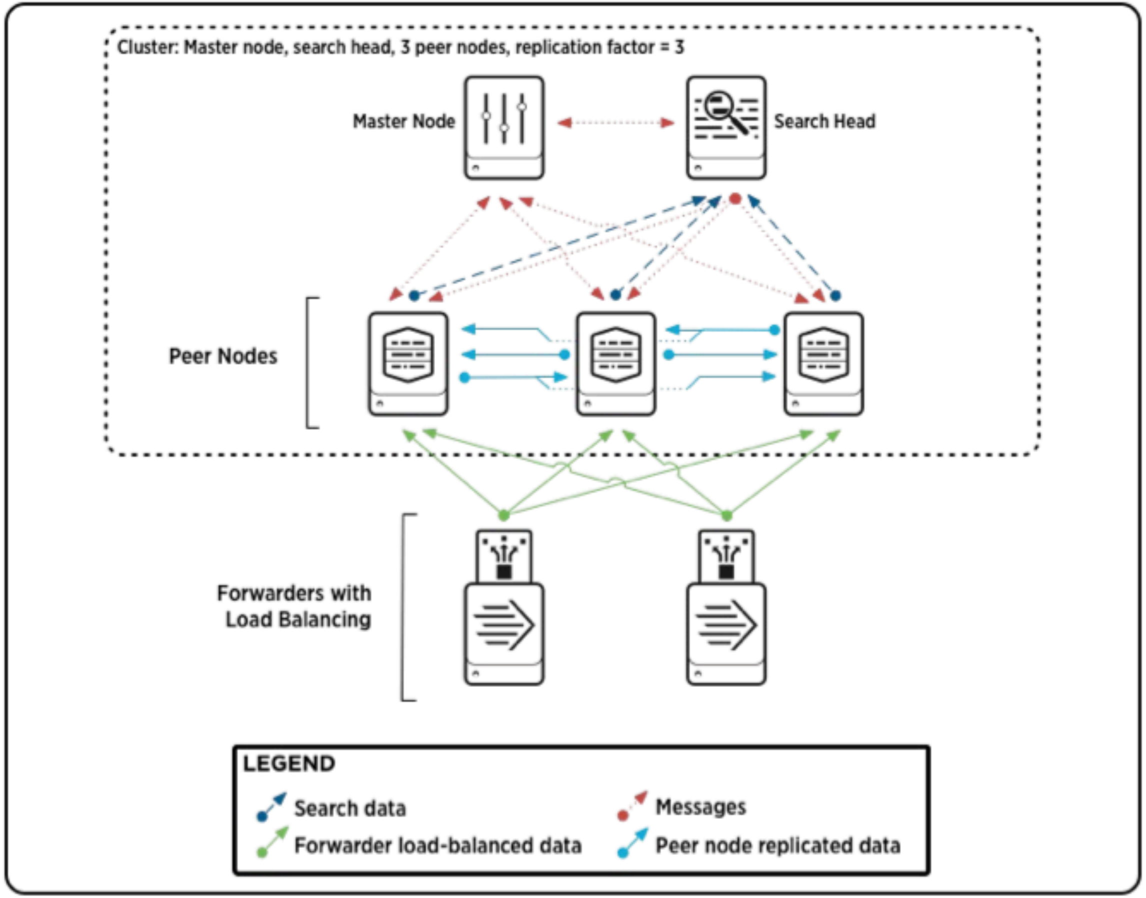
Credit: Splunk Docs Team


Cluster: Master node, search head, 3 peer nodes, replication factor = 3
Master Node
Search Head
Peer Nodes
Forwarders with Load Balancing
LEGEND
Search data
Forwarder load-balanced data
Messages
Peer node replicated data

Credit: Splunk Docs Team

Replication Factor (RF)

- Number of copies of data in the cluster. Default **RF=3**
- Cluster can tolerate **RF-1** node failures


Cluster: Master node, search head, 3 peer nodes, replication factor = 3
Master Node
Search Head
Peer Nodes
Forwarders with Load Balancing
LEGEND
Search data
Forwarder load-balanced data
Messages
Peer node replicated data

Credit: Splunk Docs Team

Search Factor (SF)

- Number of copies of data in the cluster. Default **SF=2**
- Requires more storage
- Replicated vs. Searchable Bucket

HA

Clustered Indexing

- Originating peer node streams copies of data to other clustered peers
  - Receiving peers store those copies
- Master determines replicated data destination
  - Instructs peers what peers to stream data to. Does not sit on data path
- Master manages all peer-to-peer interactions and coordinates remedial activities
- Master keeps track of which peers have searchable data
  - Ensures that there are always **SF** copies of searchable data available

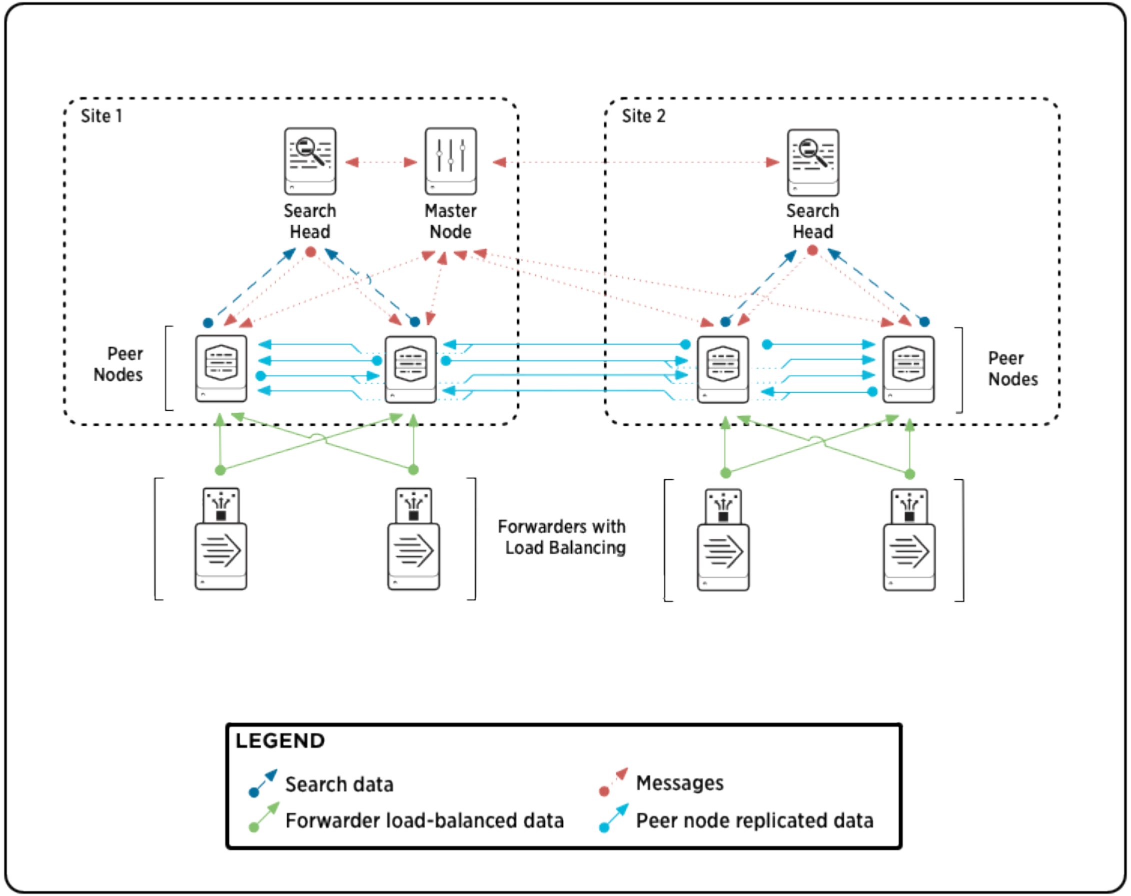
HA

Clustered Searching

- Search head coordinates all searches in the cluster
- SH relies on master to tell it who its peers are
  - The master keeps track of which peers have searchable data
- Only **one** replicated bucket is searchable a.k.a **primary**
  - i.e., searches occur over **primary** buckets, only
- Primary buckets may change over time
  - Peers know their status and therefore know where to search

Multisite Clustering

- Site awareness introduced in Splunk 6.1
- Improved disaster recovery
  - Multisite clusters provide site failover capability
- Search Affinity
  - Search heads will scope searches to local site, whenever possible
  - Ability to turn off for better thruput vs. X-Site bandwidth



Credit: Splunk Docs Team

Multi Site Cluster Architecture

Differences vs. single site

- Assign a site to each node
- Specify RF and SF on a site by site basis

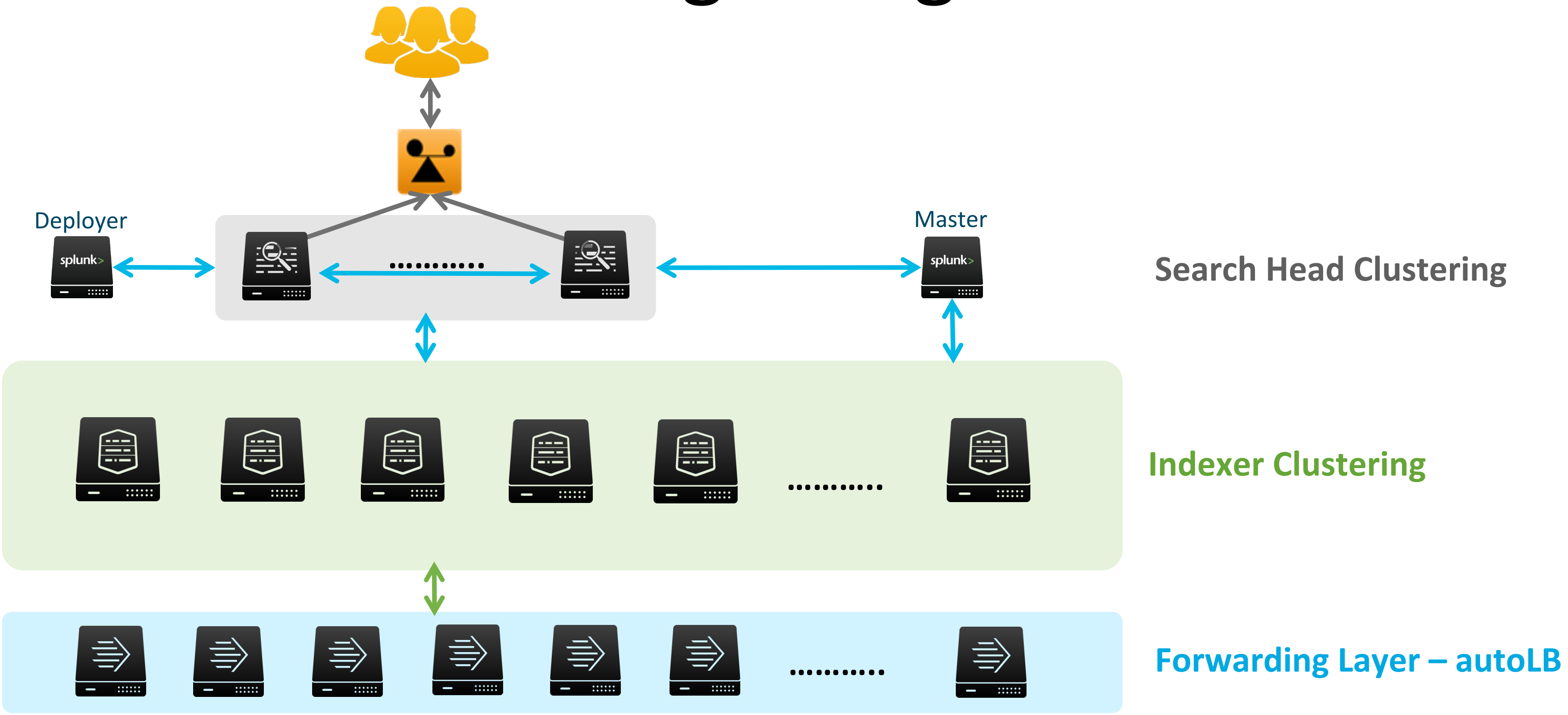
Multisite Clustering Cont'd

- Each node belongs to an assigned site, except for the Master Node, which controls all sites but it's not **logically** a member of any
- Replication of bucket copies occurs in a site-aware manner.
  - Multisite replication determines # copies on each site. Ex. 3 site cluster:

```
site_replication_factor = origin:2, site1:1, site2:1, site3:1, total:4
```

- Bucket-fixing activities respect site boundaries when applicable
- Searches are fulfilled by local peers whenever possible (a.k.a **search affinity**)
  - Each site must have at least a full set of searchable data

Putting It Together



END

Top Takeways

- **DR – Process of backing-up and restoring service in case of disaster**
  - **Configuration files** – copy of $SPLUNK_HOME/etc/ folder
  - **Indexed data** – backup and restore buckets
    - Hot, warm, cold, frozen
    - Can't backup hot (without snapshots) but can safely backup warm and cold
- **HA – continuously operational system bounded by a set of tolerances**
  - **Data collection**
    - Autolb from forwarders to multiple indexers
    - Use Indexer Acknowledgement to protect in flight data
  - **Searching**
    - Search Head Clustering (SHC)
  - **Indexing**
    - Use Index Replication

Q & A

Feedback: dritan@splunk.com

You May Also Like

Jiffy Lube Quick Tune-up for Your Splunk Environment

Best Practices for Deploying Splunk on Amazon Web Services

Deploying Splunk Enterprise on Microsoft Azure Cloud

.conf2016

splunk>

THANK YOU

.conf2016

splunk>