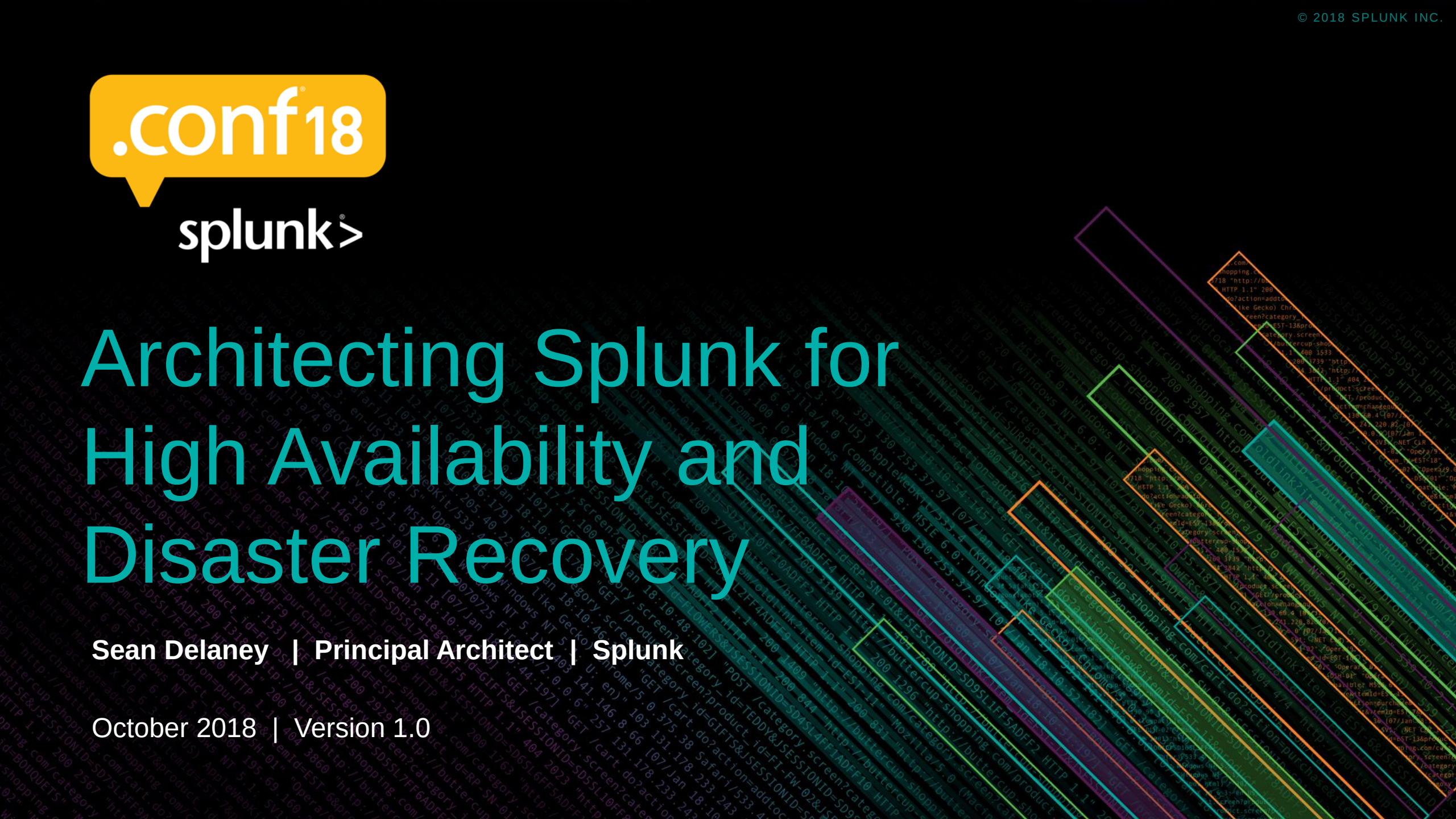


Architecting Splunk for High Availability and Disaster Recovery

Sean Delaney | Principal Architect | Splunk

October 2018 | Version 1.0

Forward-Looking Statements

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 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.

splunk> .conf18

Sean Delaney

- Principal Field Architect
- 7+ years at Splunk
- Large scale deployments
- 9th .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)

DR

What is Disaster Recovery?

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

splunk> .conf18

DR

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

DR

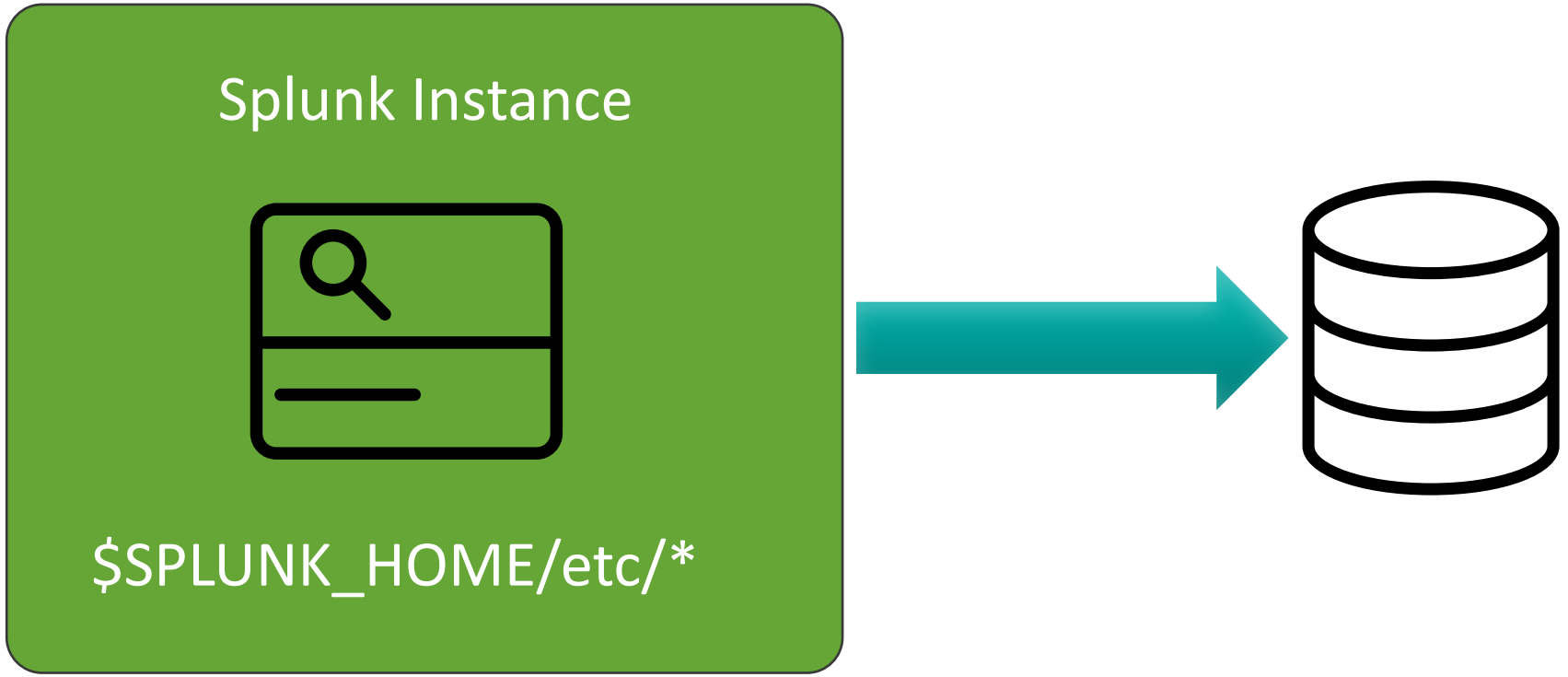
Backup

1	a	Configurations $SPLUNK_HOME/etc/*
	b	Indexes Buckets: Hot*, Warm, Cold, Frozen

splunk> .conf18

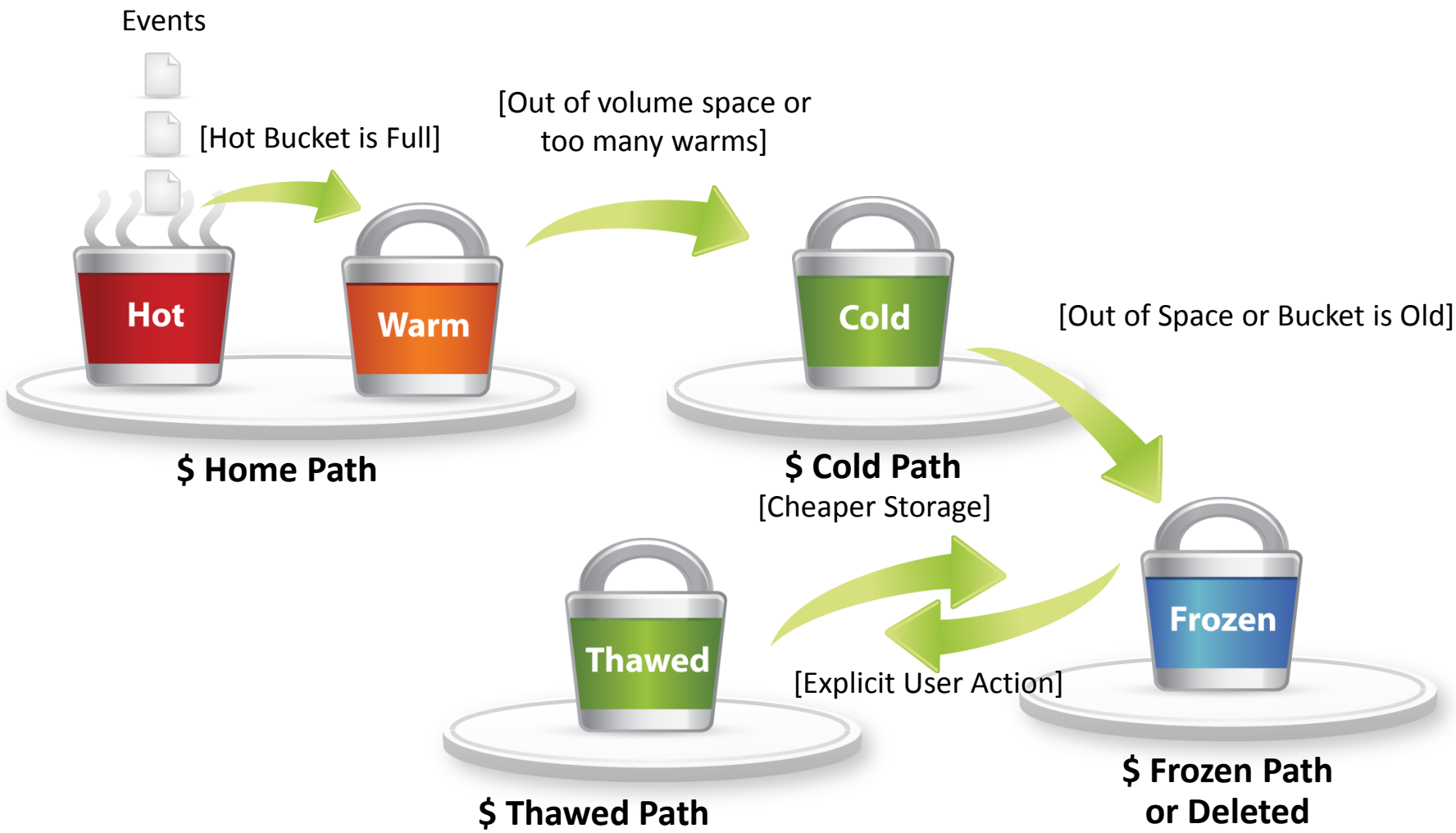
DR

Backup Configurations



DR

Backup: Bucket Lifecycle



DR

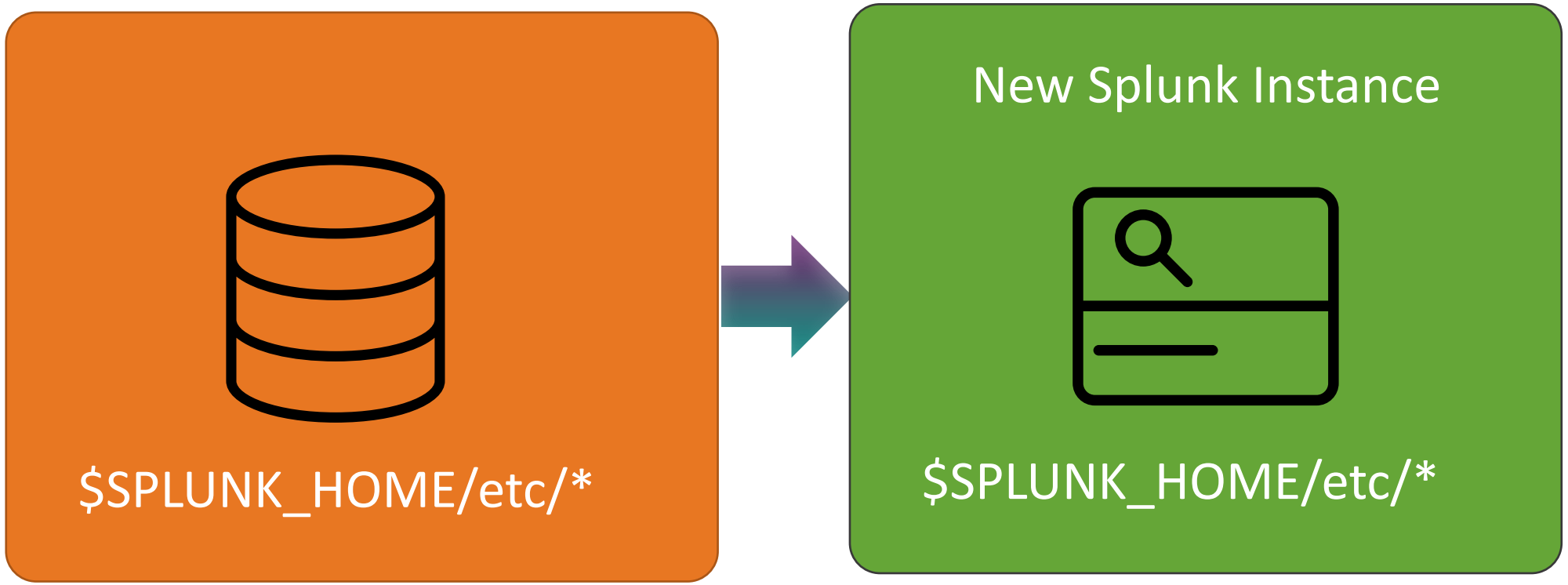
Backup Data

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

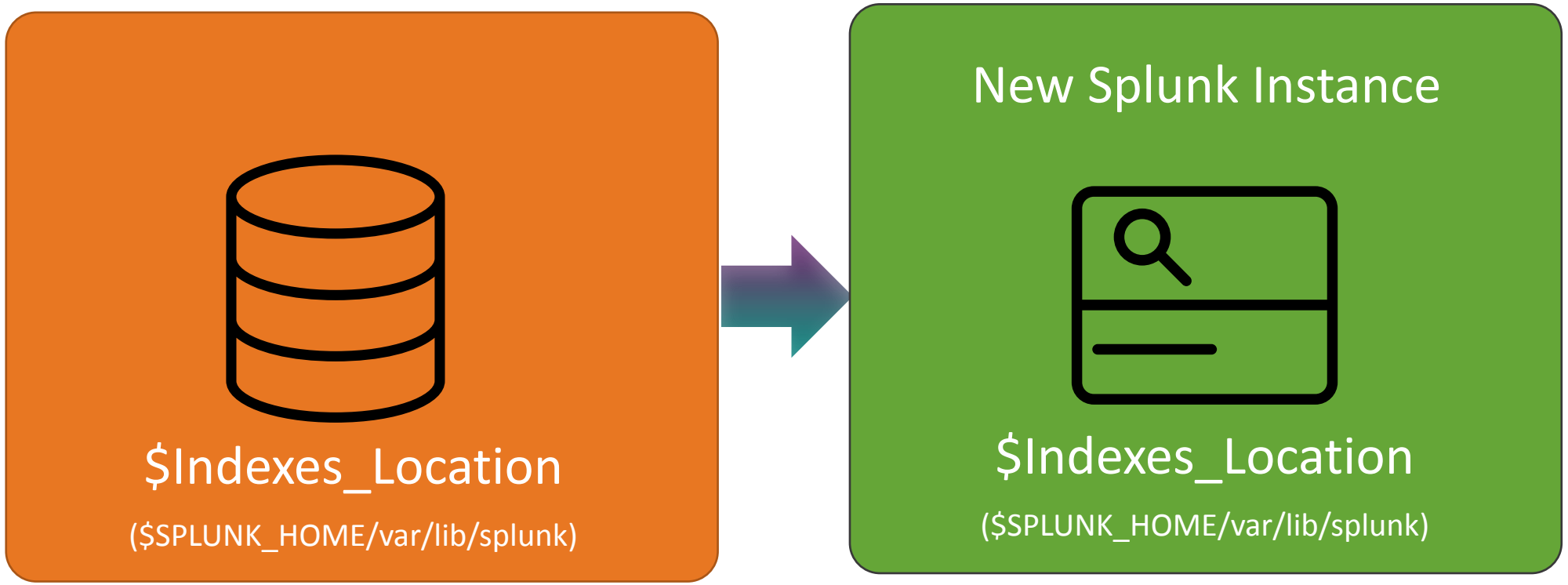
*Unless using snapshot aware FS or roll to warm first (which introduces a performance penalty).

DR

Restore Configurations



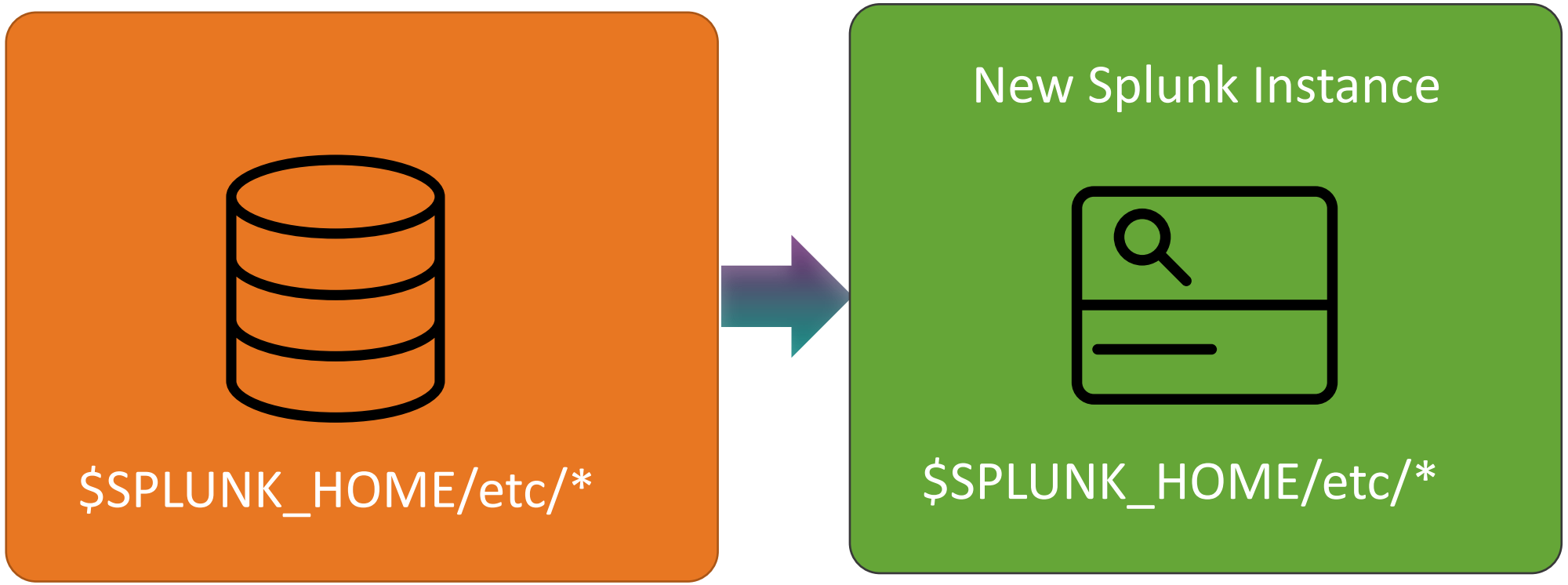
Restore Data



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

DR

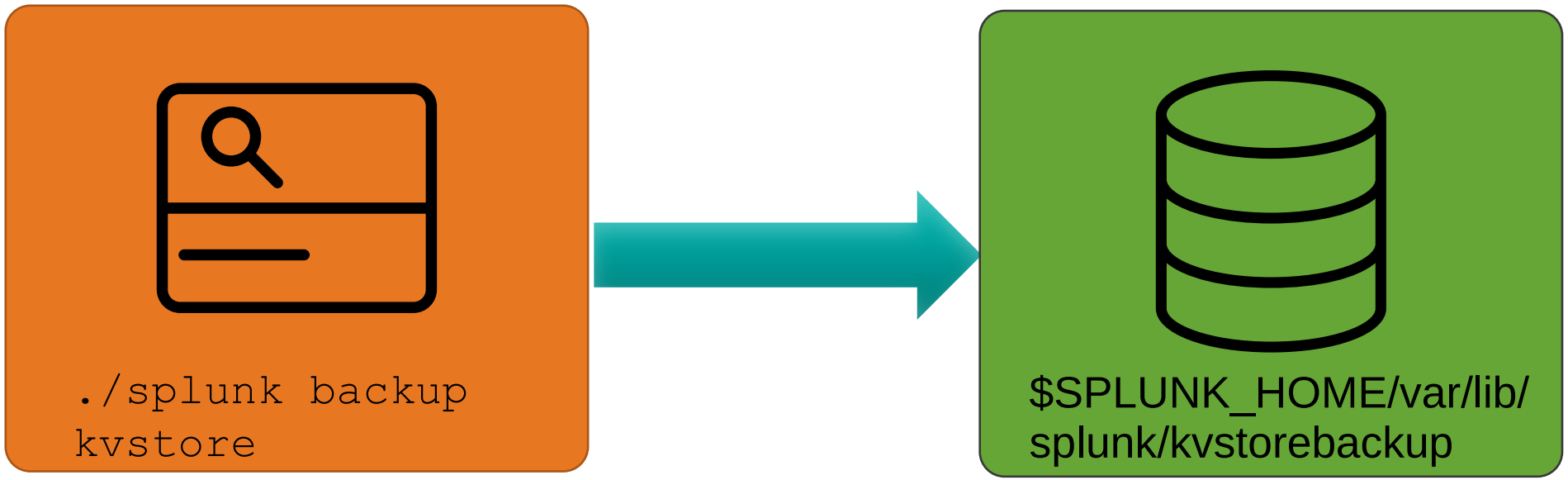
Restore Configurations



DR

Backup KV Store

- **KV store backup scripts added in 7.1, previous versions backup KV Store files**
- **Run on the Searchead to take a stateful snapshot of the KV store**



http://docs.splunk.com/Documentation/Splunk/latest/Admin/BackupKVstore

DR

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>**

splunk> .conf18

DR

Putting Restore Together

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

DR

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)

HA

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”

splunk> .conf18

HA

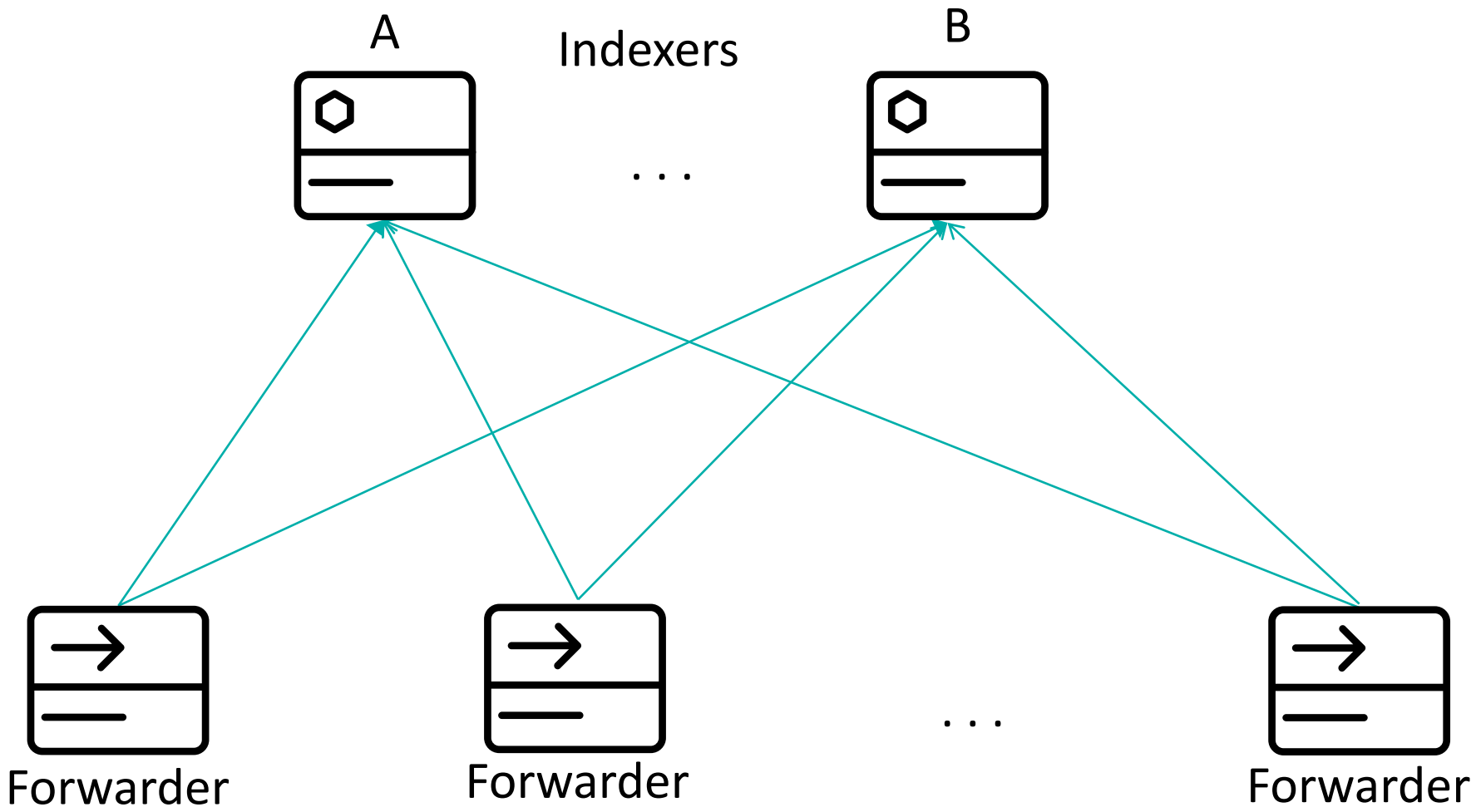
Splunk High Availability

1	Data Collection/Reception
2	Searching
3	Indexing

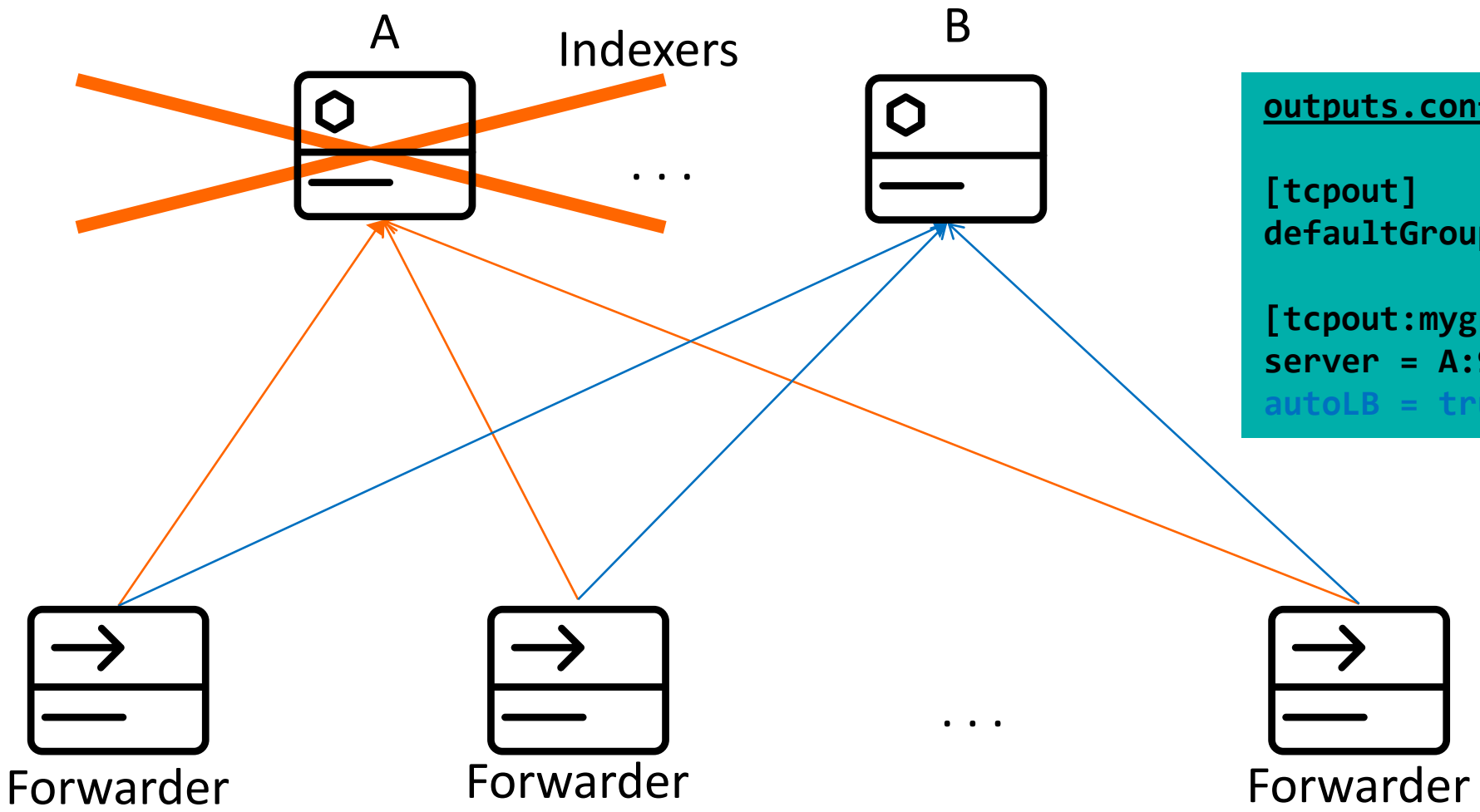
splunk> .conf18

HA

Data Collection



Data Collection



outputs.conf:

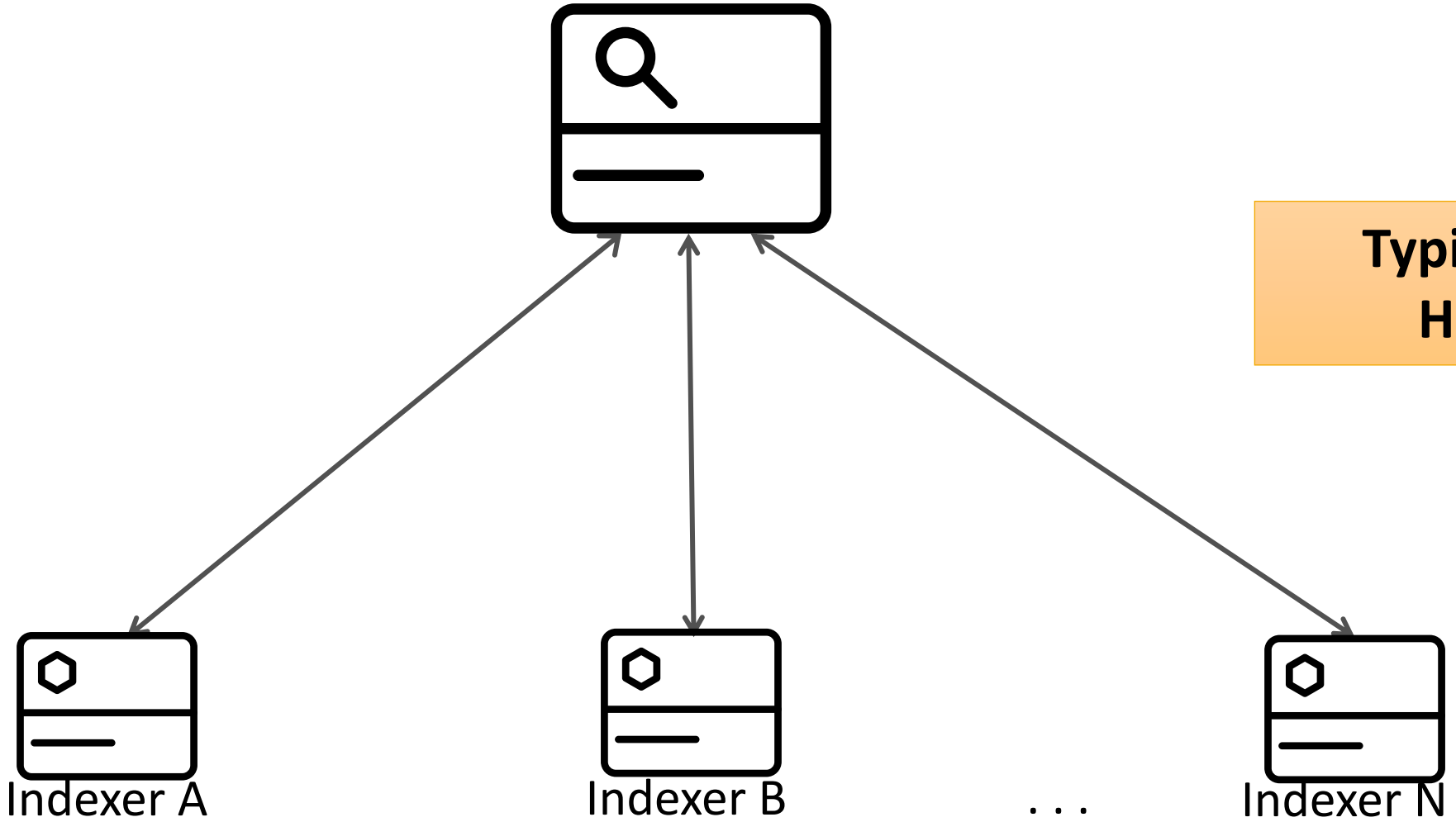
```
[tcpout]
defaultGroup = mygroup

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

Searching

2 Search Head Clustering (SHC)

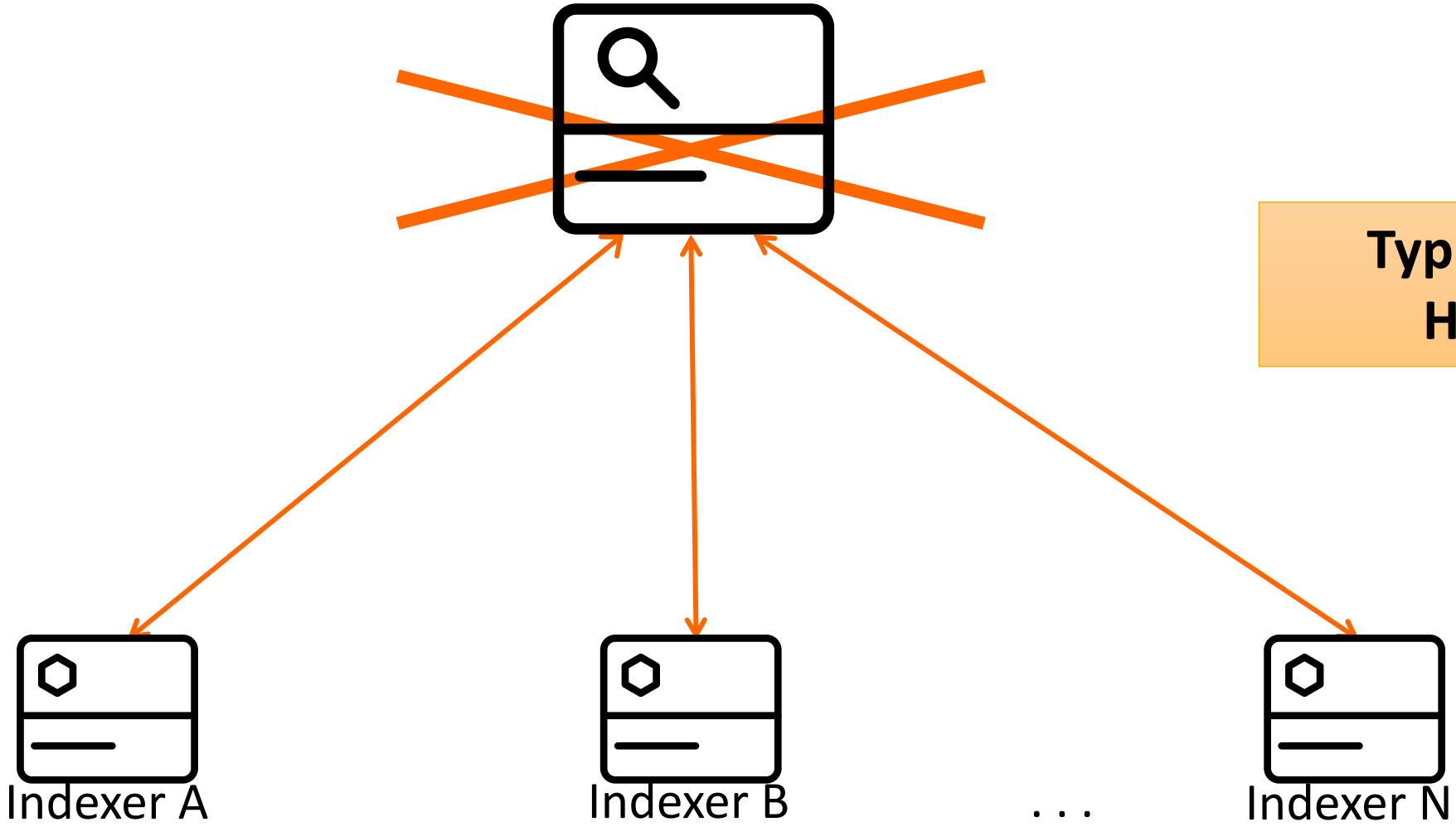
Searching



Typical Search Hierarchy

HA

Searching



Typical Search Hierarchy

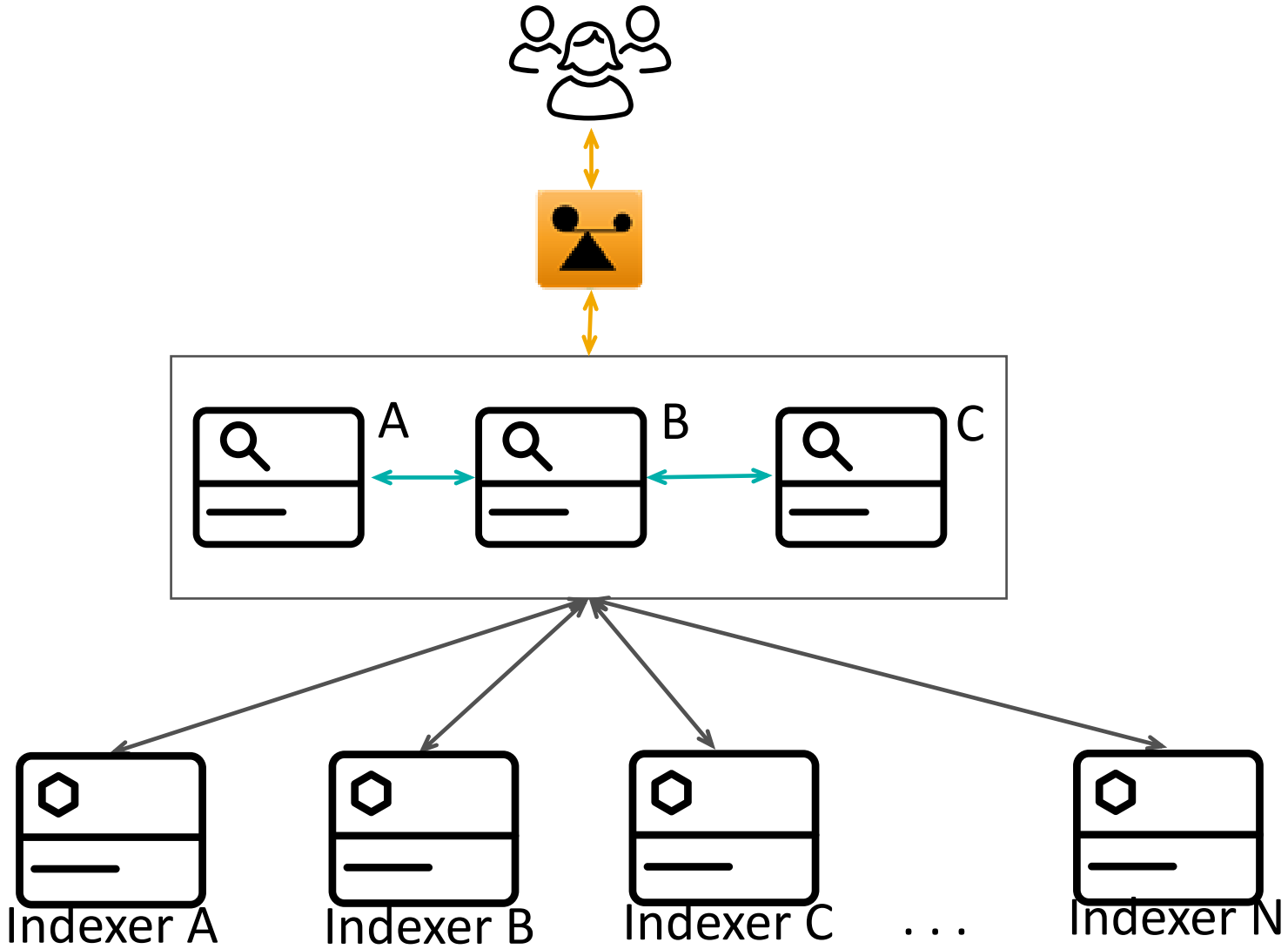
HA

Search Head Clustering (SHC)

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

HA

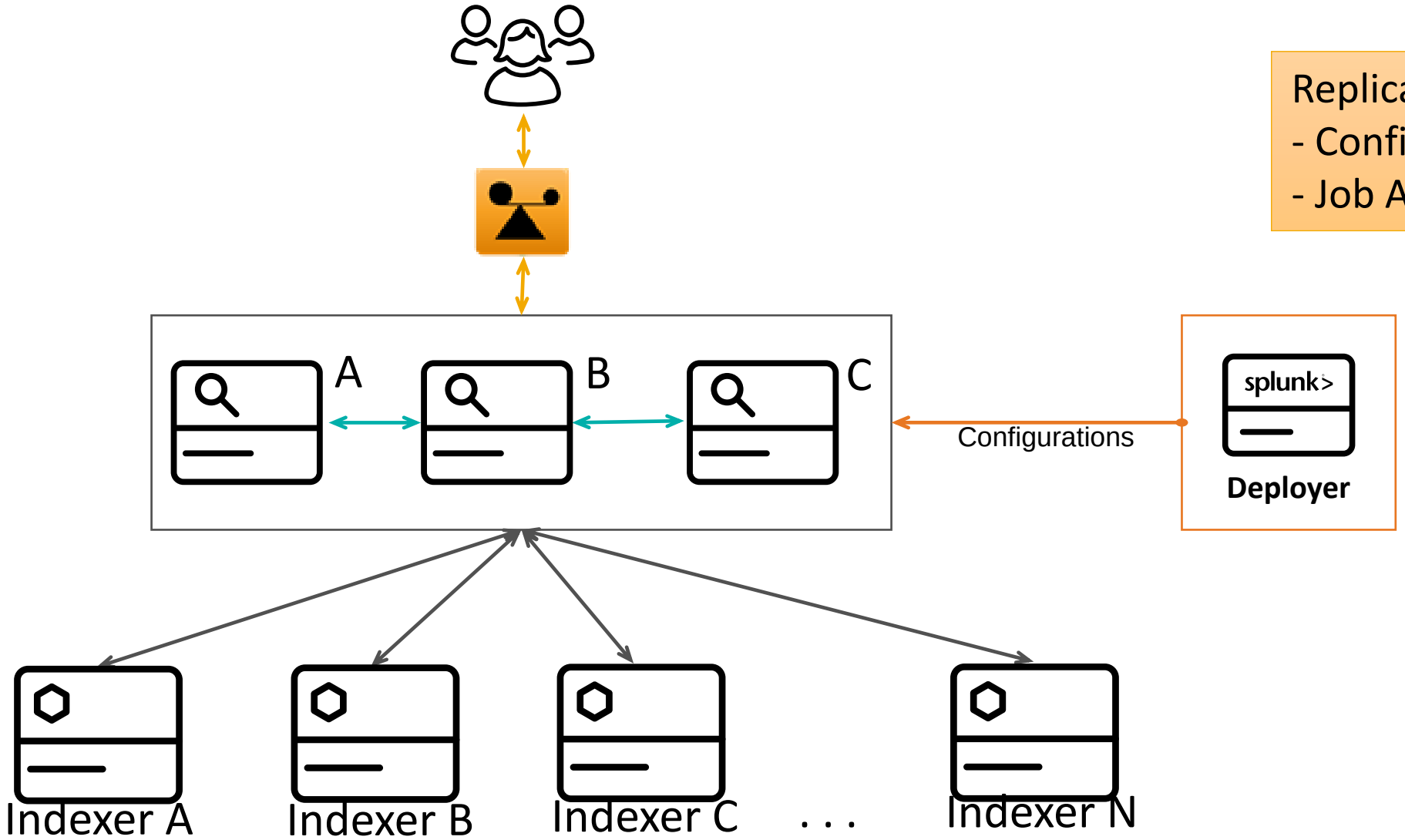
SHC



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

HA

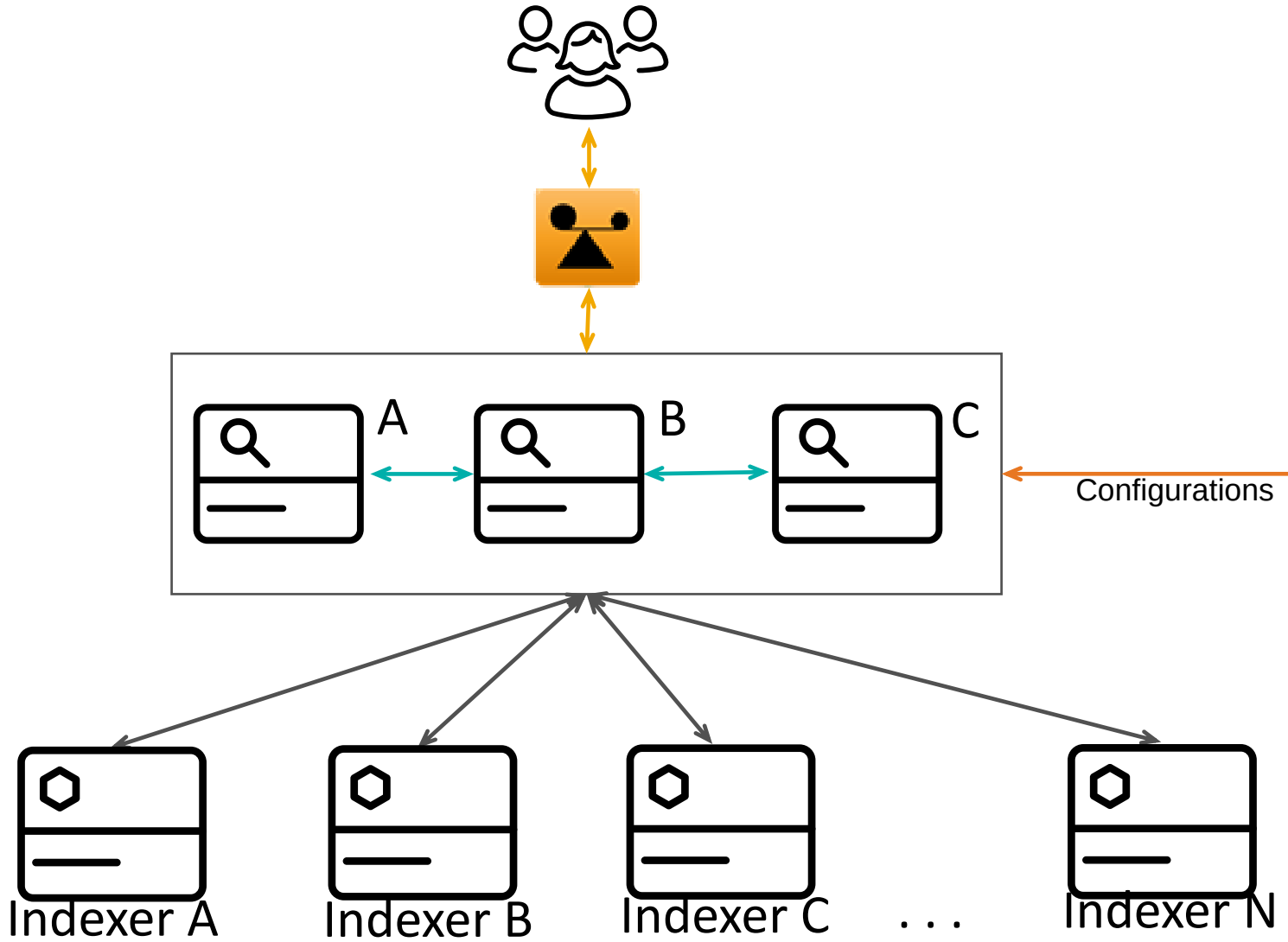
SHC



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

HA

SHC

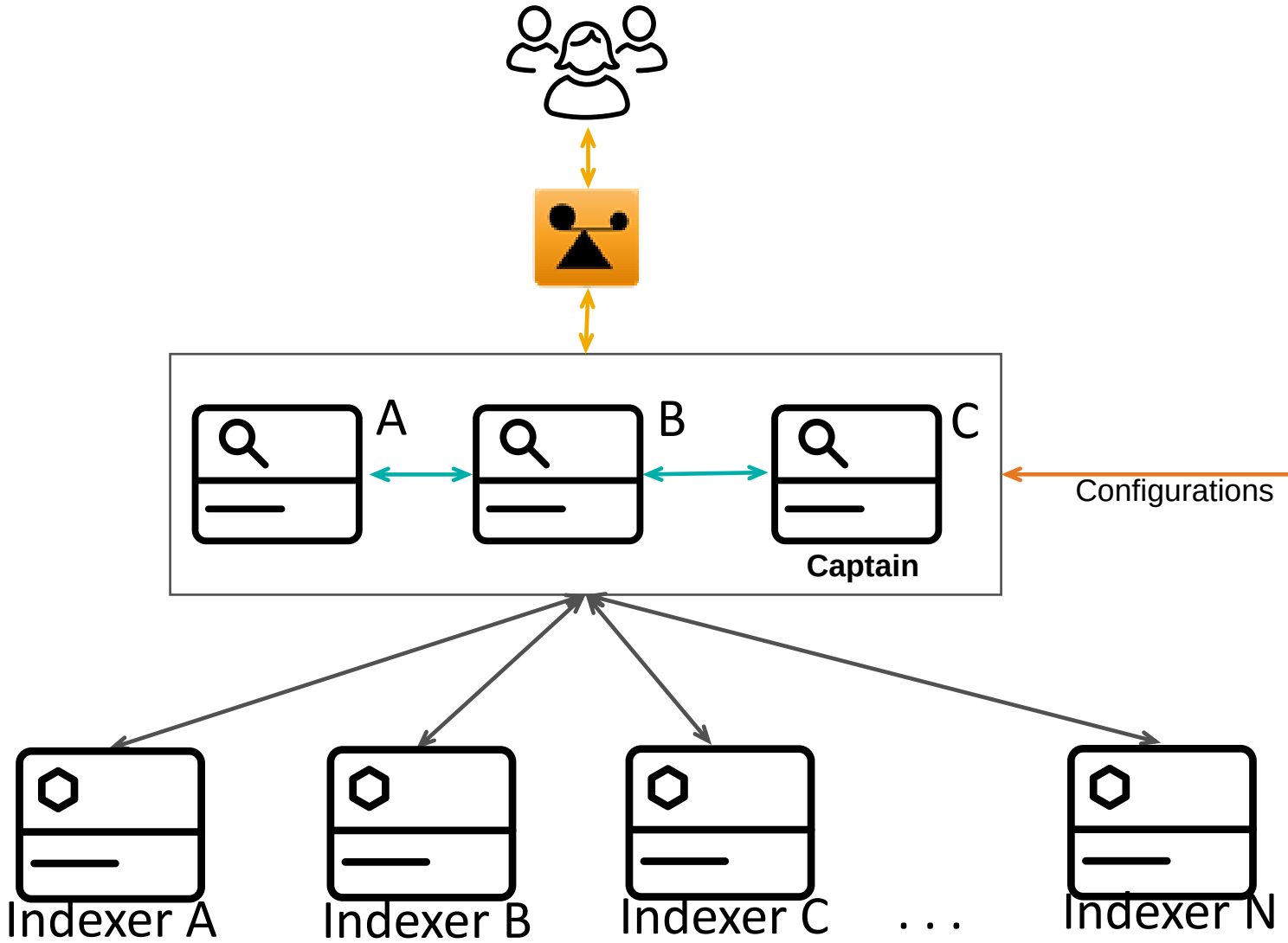


Replication **protocol** syncs:
- Configurations
- Job Artifacts

Deployer ensures identical deployed configurations

HA

SHC



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

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

HA

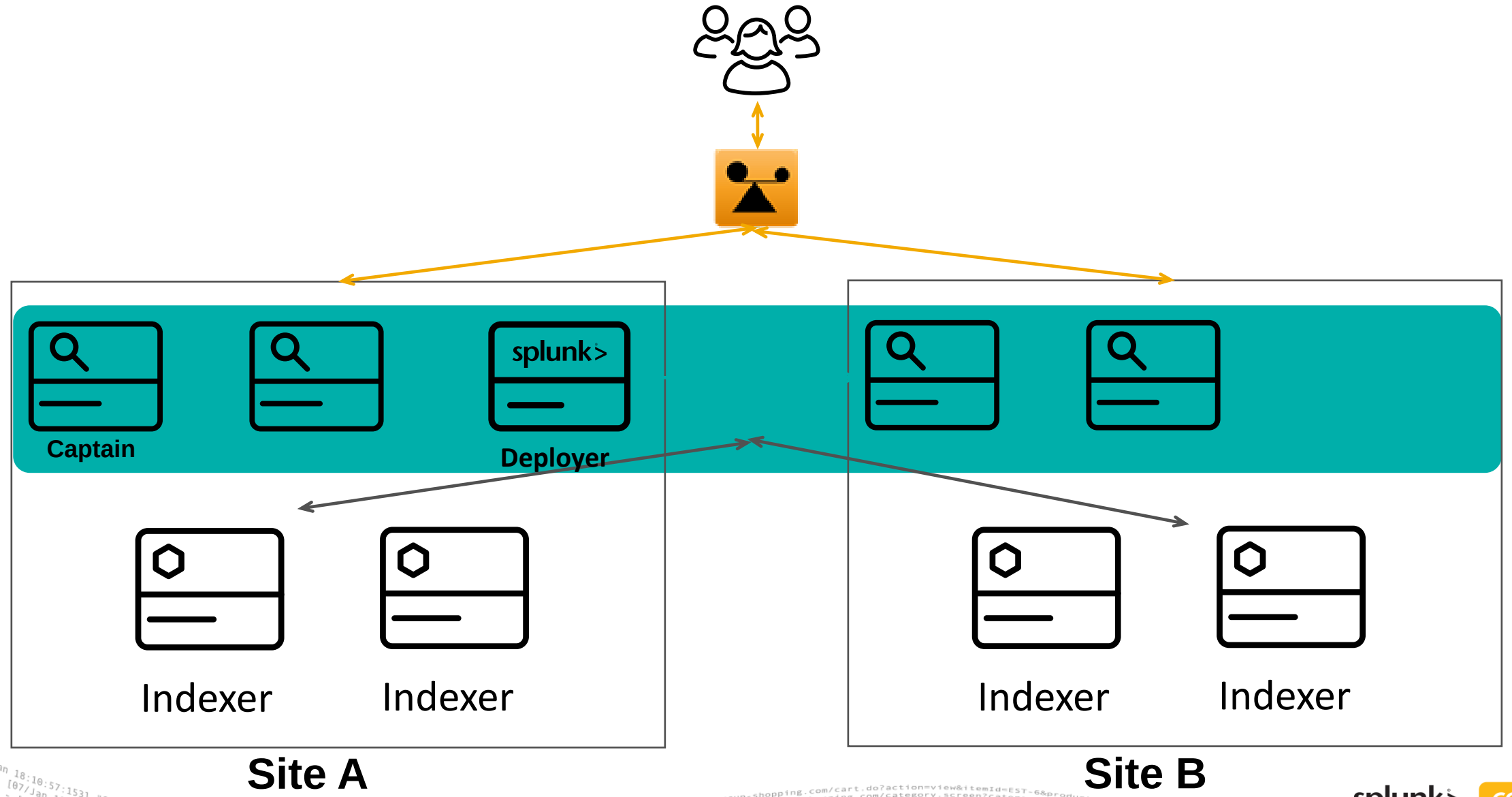
SHC Operation - High Level

- 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.
  - 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.

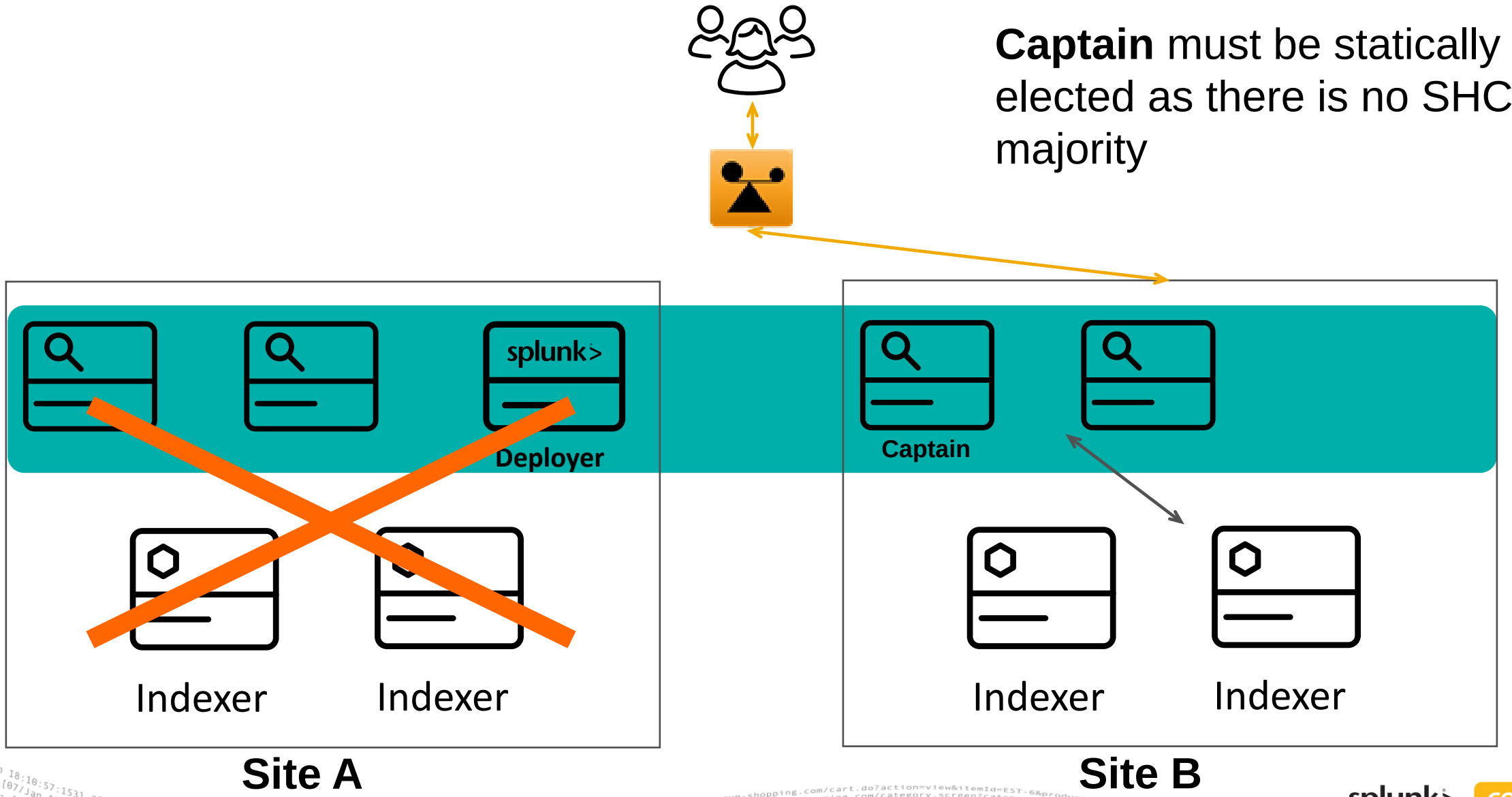
SHC Operation - High Level

- 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

Stretched SHC

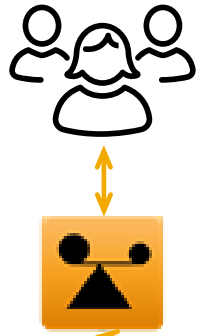


Stretched SHC

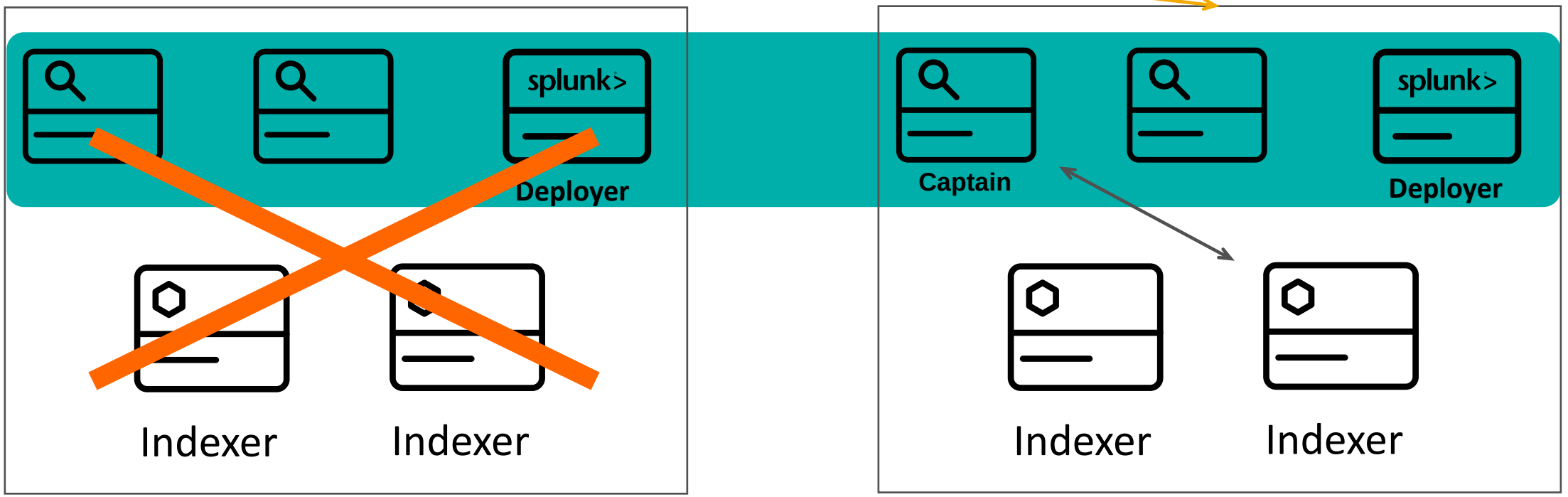


Captain must be statically elected as there is no SHC majority

Stretched SHC



Deployer launched in second site, if SCH updates required



3

Indexer Clustering

splunk> .conf18

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

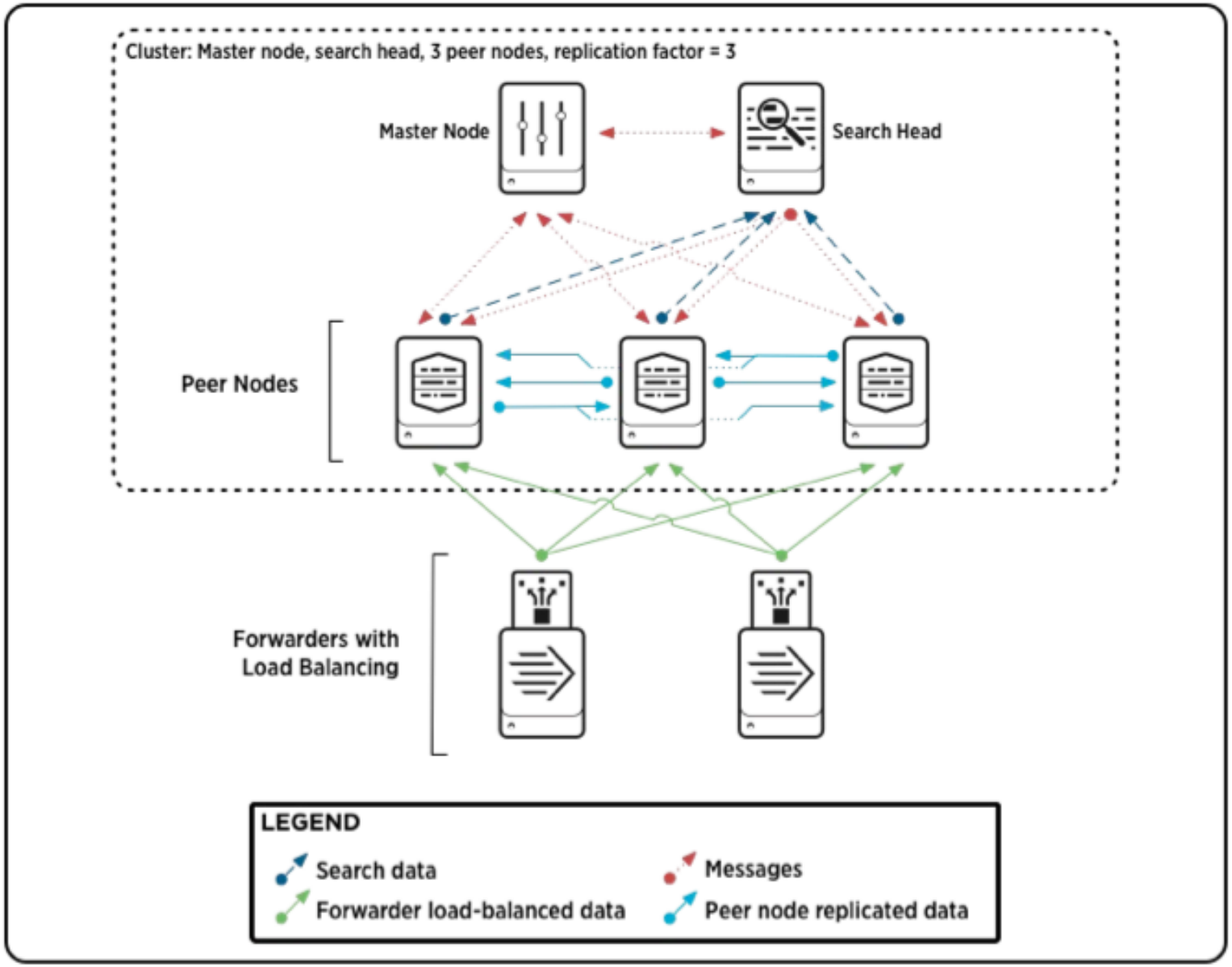
- Extra storage
- Slightly increased processing load.

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

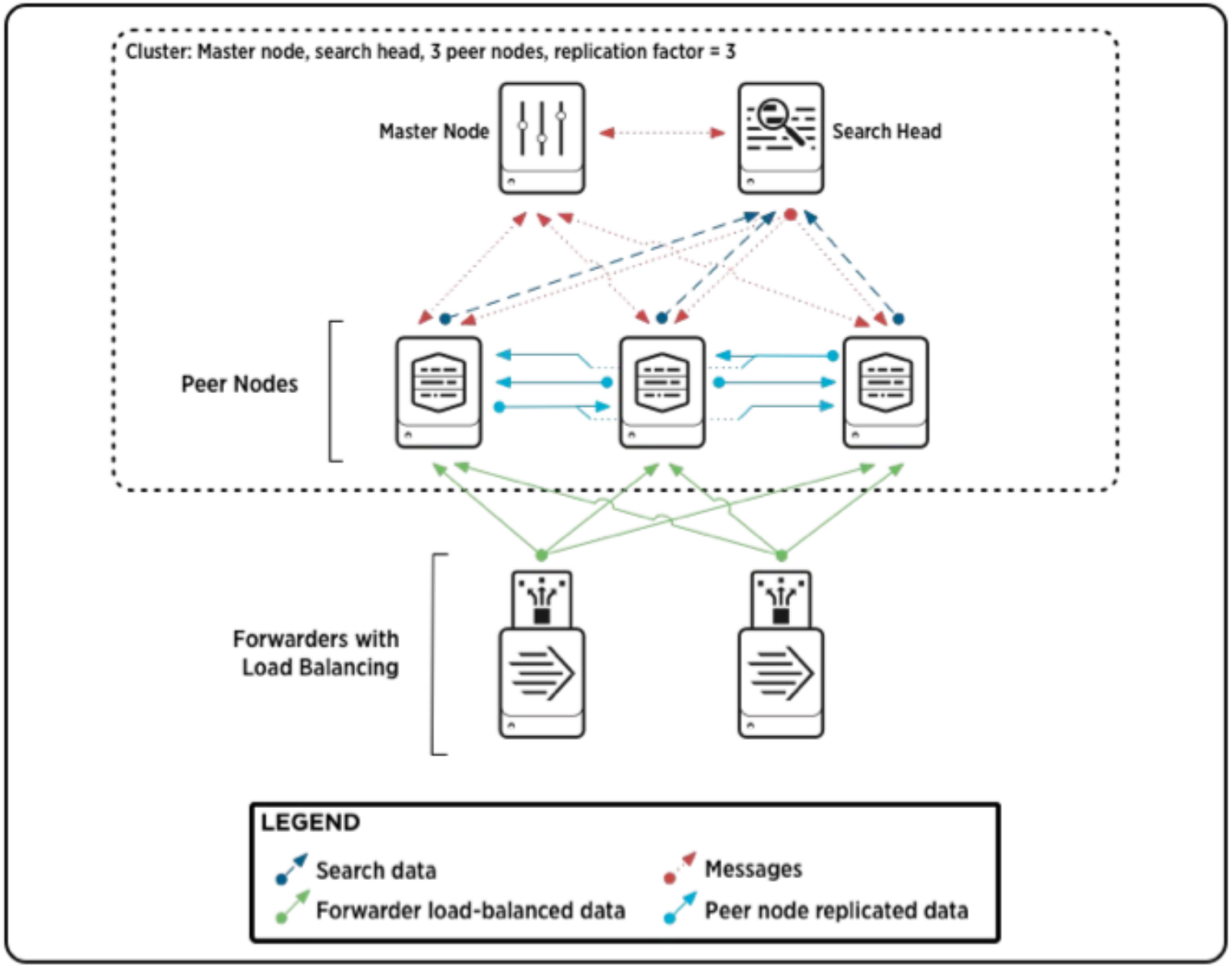
HA

Single Site Cluster Architecture



Credit: Splunk Docs Team

HA

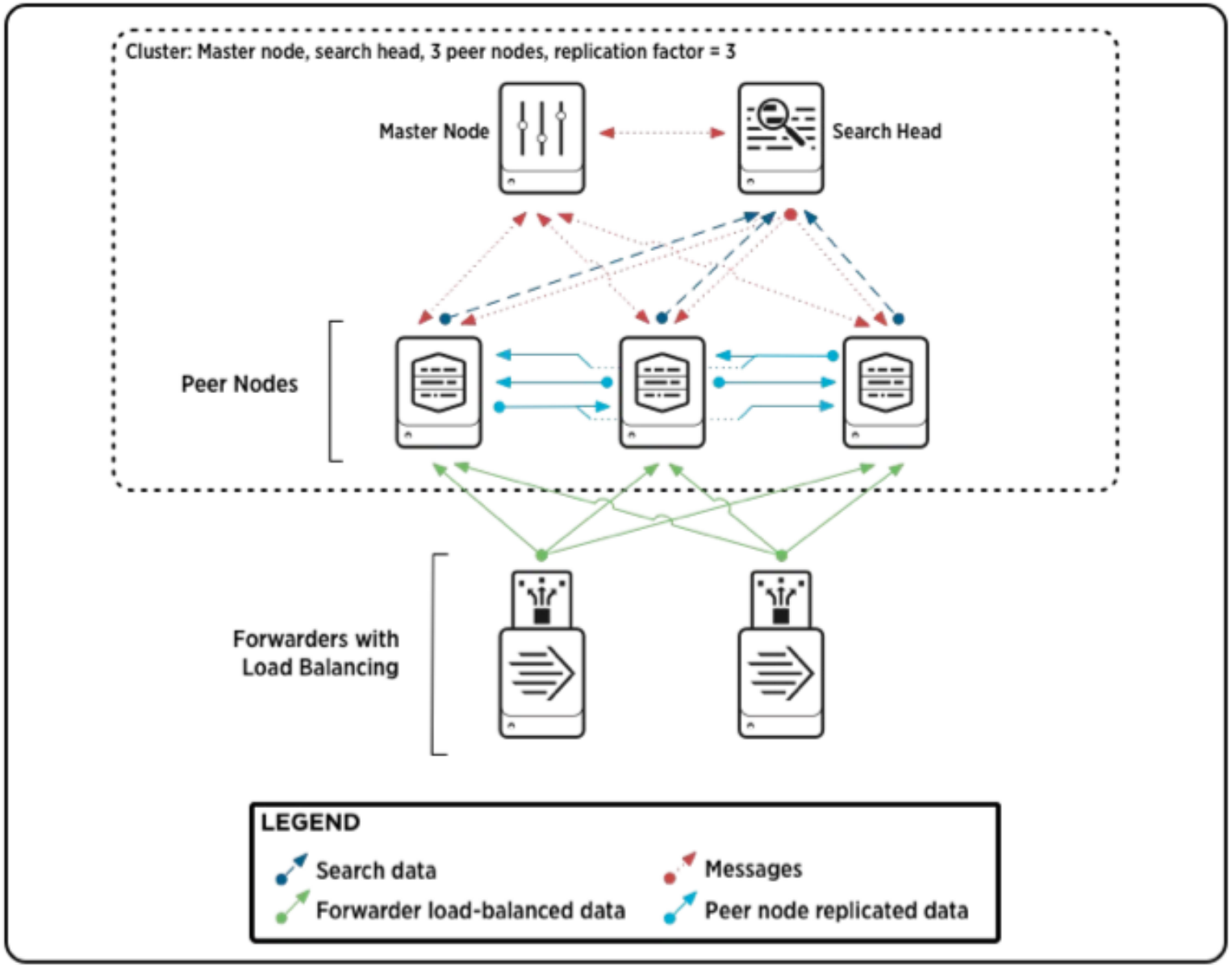


Credit: Splunk Docs Team

Replication Factor (RF)

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

HA



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

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

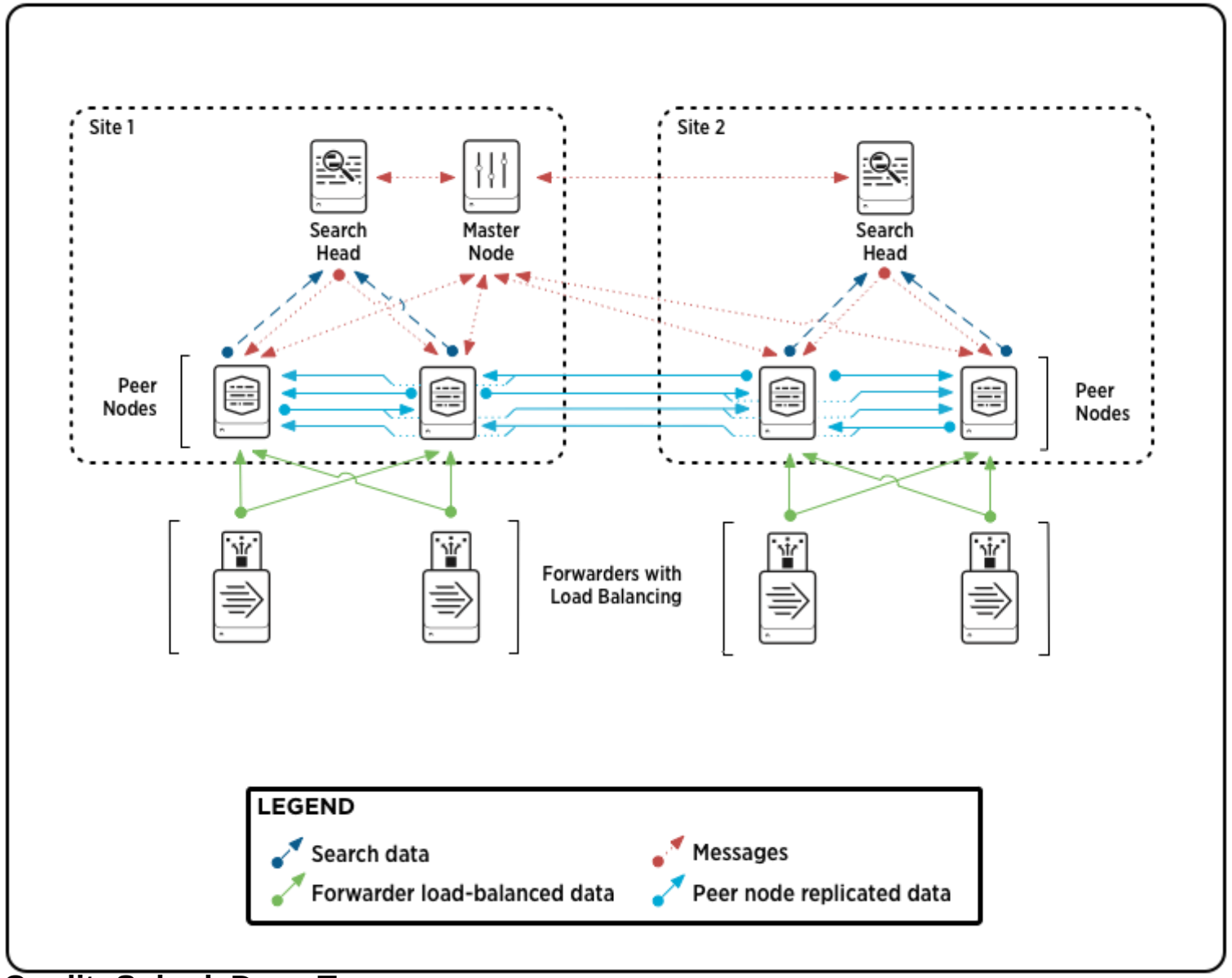
Clustered Searching

- 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

Multi Site Cluster Architecture



Credit: Splunk Docs Team

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

Rolling Restarts and Upgrades

splunk> .conf18

Rolling Restarts/Upgrades

- New features in 7.1.0
- Searchable Rolling Restarts
  - Perform a restart of an Indexer Cluster without effecting Search
- Upgrade a Splunk Clustered Deployment (SHC and Indexer Cluster) without downtime
  - Rolling upgrades apply to versions 7.1 and above (7.1 -> 7.2)

Searchable Rolling Restart

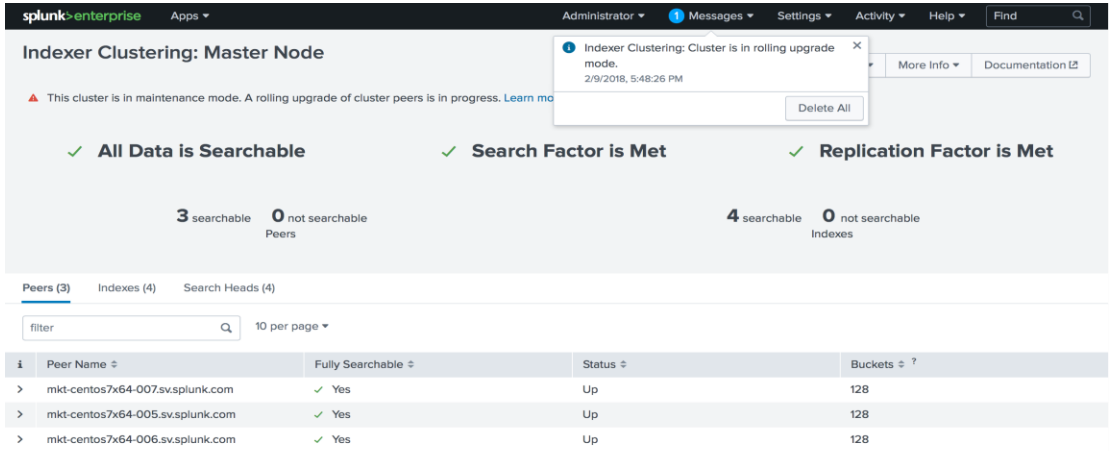
- The master restarts peers one at a time.
- The master runs health checks to confirm that the cluster is in a searchable state before it initiates the searchable rolling restart.
- The peer waits for in-progress searches to complete, up to a maximum time period, as determined by the decommission_search_jobs_wait_secs attribute in server.conf. The default for this attribute is 180 seconds. This covers the majority of searches in most cases.
- Searchable rolling restart applies to both historical searches and real-time searches.

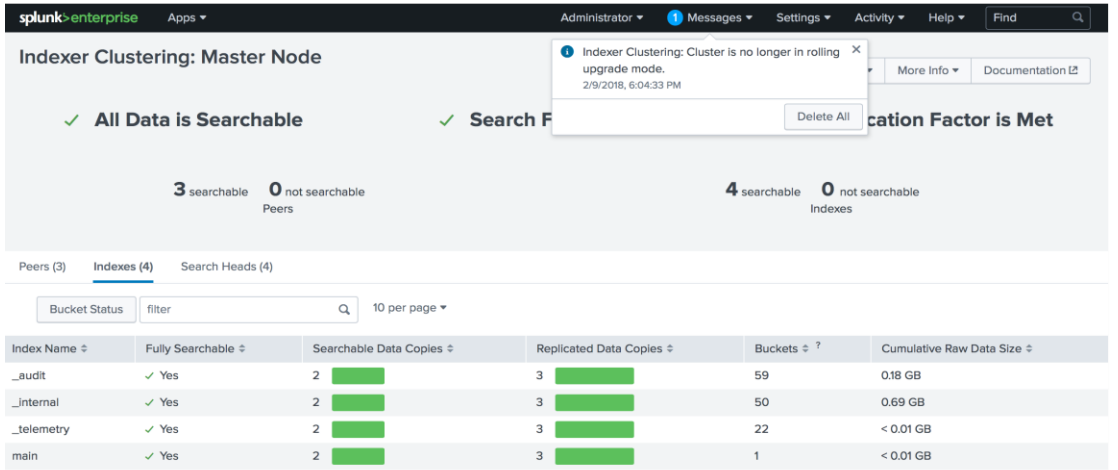
- `splunk rolling-restart cluster-peers -searchable true`

Searchable Rolling Upgrades

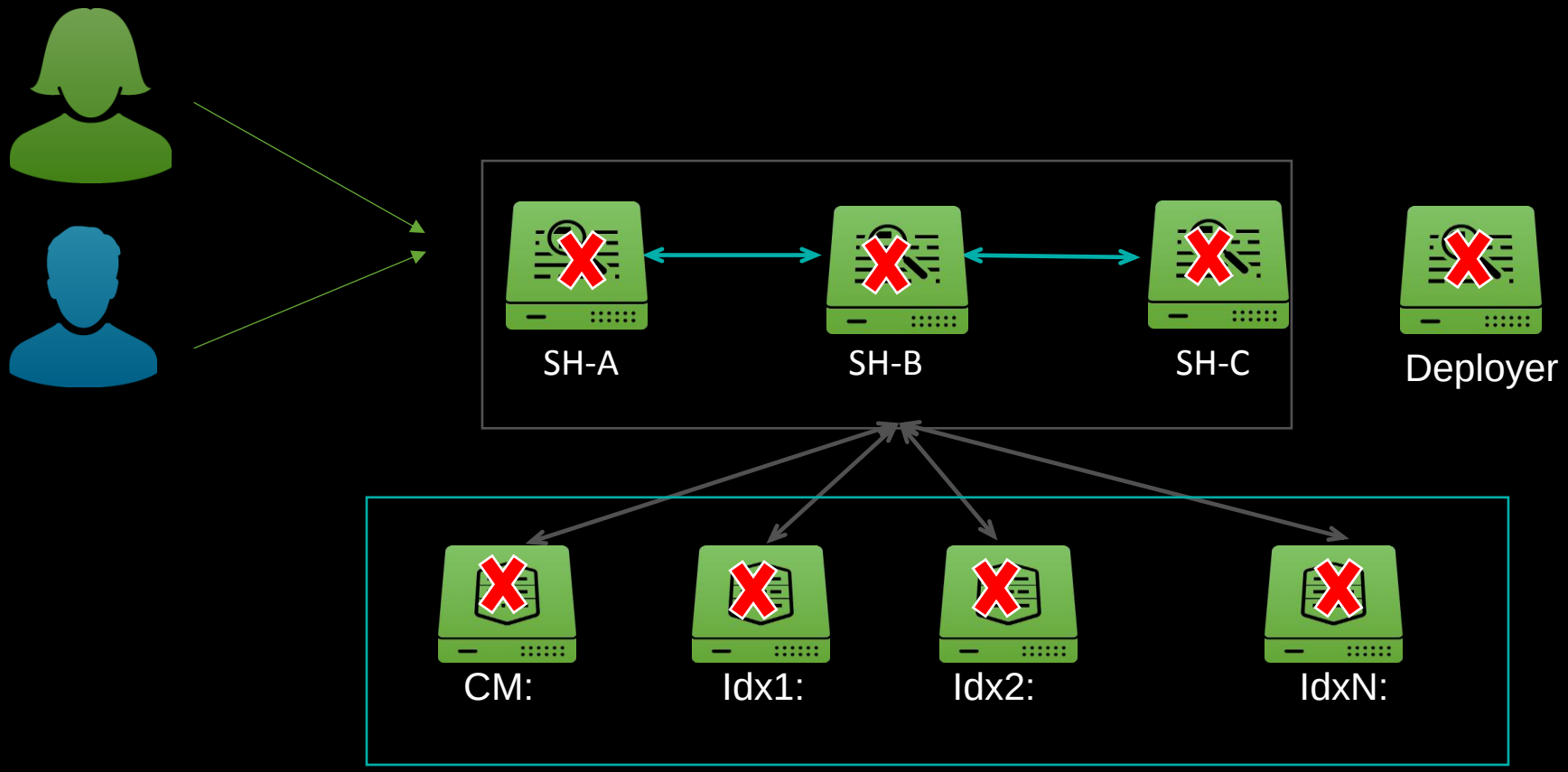
Rolling Upgrade Steps:

1. Perform pre-flight indexer cluster health checks
   - splunk show cluster-status --verbose
2. Upgrade the cluster master
3. Initialize rolling upgrade on the indexer cluster
   - splunk upgrade-init cluster-peers
4. Select a cluster peer and gracefully shutdown the cluster-peer
   - splunk offline
5. Upgrade the selected cluster-peer and restart
6. Repeat steps 4 and 5 for all cluster peers
7. Finalize rolling upgrade on the indexer cluster
   - splunk upgrade-finalize cluster-peers





Searchable Rolling Upgrades



1. Stop the cluster master
2. Upgrade and restart cluster master
3. Rolling upgrade of Search Heads
4. Upgrade SHC deployer
5. Upgrade each indexer ensuring search continuity. Repeat for all indexers
6. Repeat step 5 for multi-site deployments

Deliver business continuity by making search highly available

Smart Store

splunk> .conf18

High Availability

Data vs. Search

- Data Resiliency

Will my data be protected if something goes wrong?

  - SmartStore relies on the remote store to provide data resiliency
    - SmartStore and indexer clustering will not protect data in the remote store
    - When used on SmartStore enabled indexes, indexer clustering will only protect hot buckets

- Search Resiliency

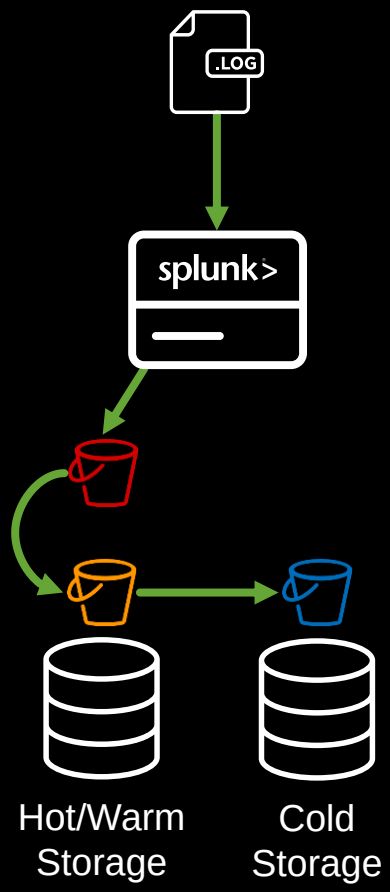
Will my search results be complete if something goes wrong?

  - Indexer clustering must be used to provide highly available search
    - SmartStore does not natively provide search HA

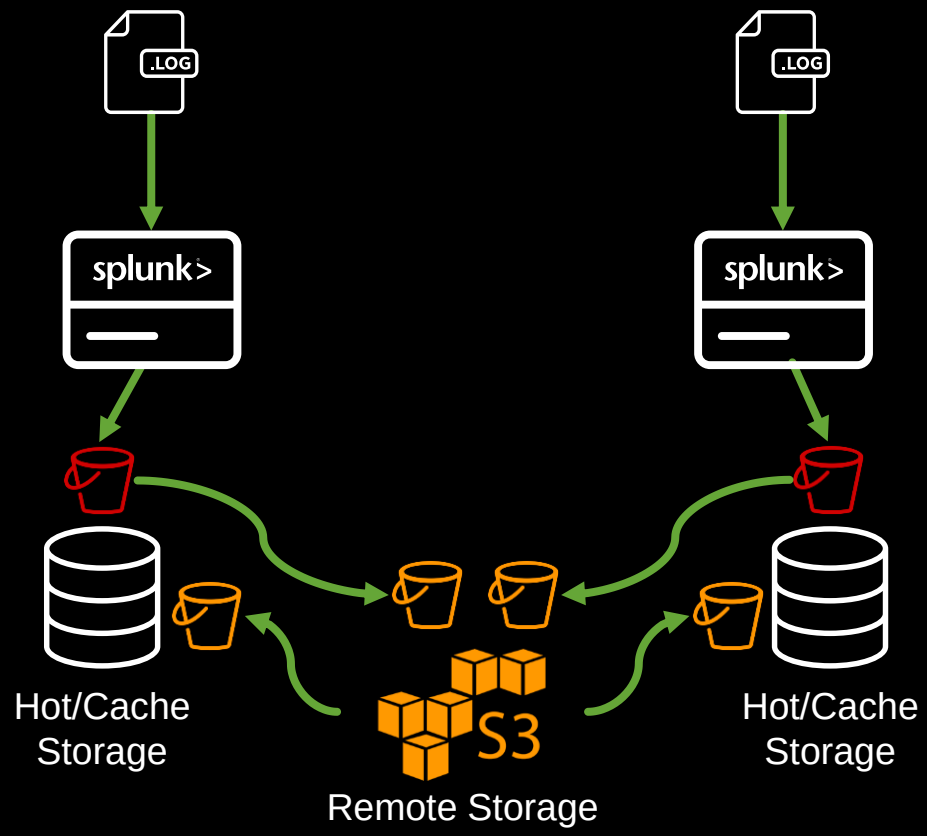
Architecture

Components

Classic Architecture

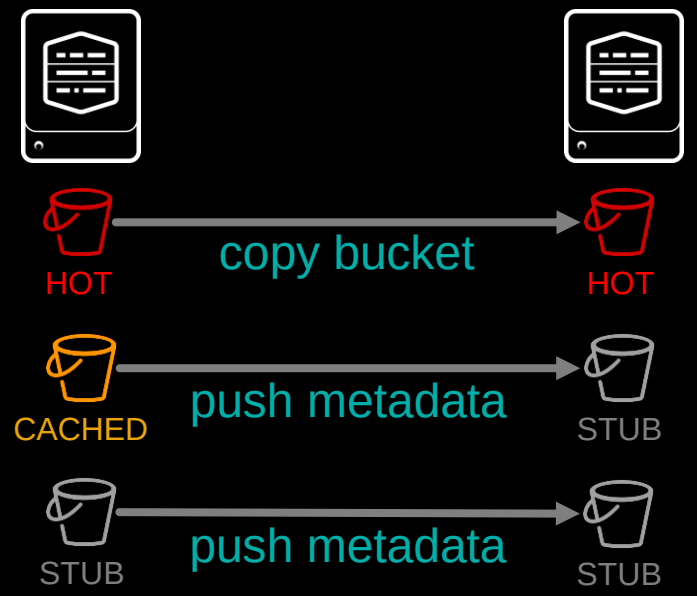


Smart Store Architecture



Failure Scenarios

Failed peers < replication factor



Available copies of a bucket exist in the cluster

- Cluster master initiates a fix-up operation
  - Peers with an existing copy of buckets are instructed to copy them to other peers until RF/SF is met
    - Hot bucket: full copy
    - Cached bucket: metadata push
      - Peers don’t need the full contents of cached buckets, as it can be fetched from the remote store
      - Metadata replication is done with a POST to a REST endpoint on the target peer(s)

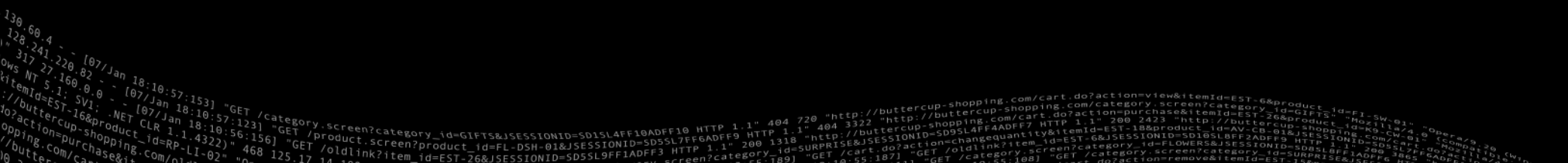
Retention

Options

Retention for SmartStore enabled indexes is managed globally

- Retention for “classic” indexes is managed on a per-indexer basis

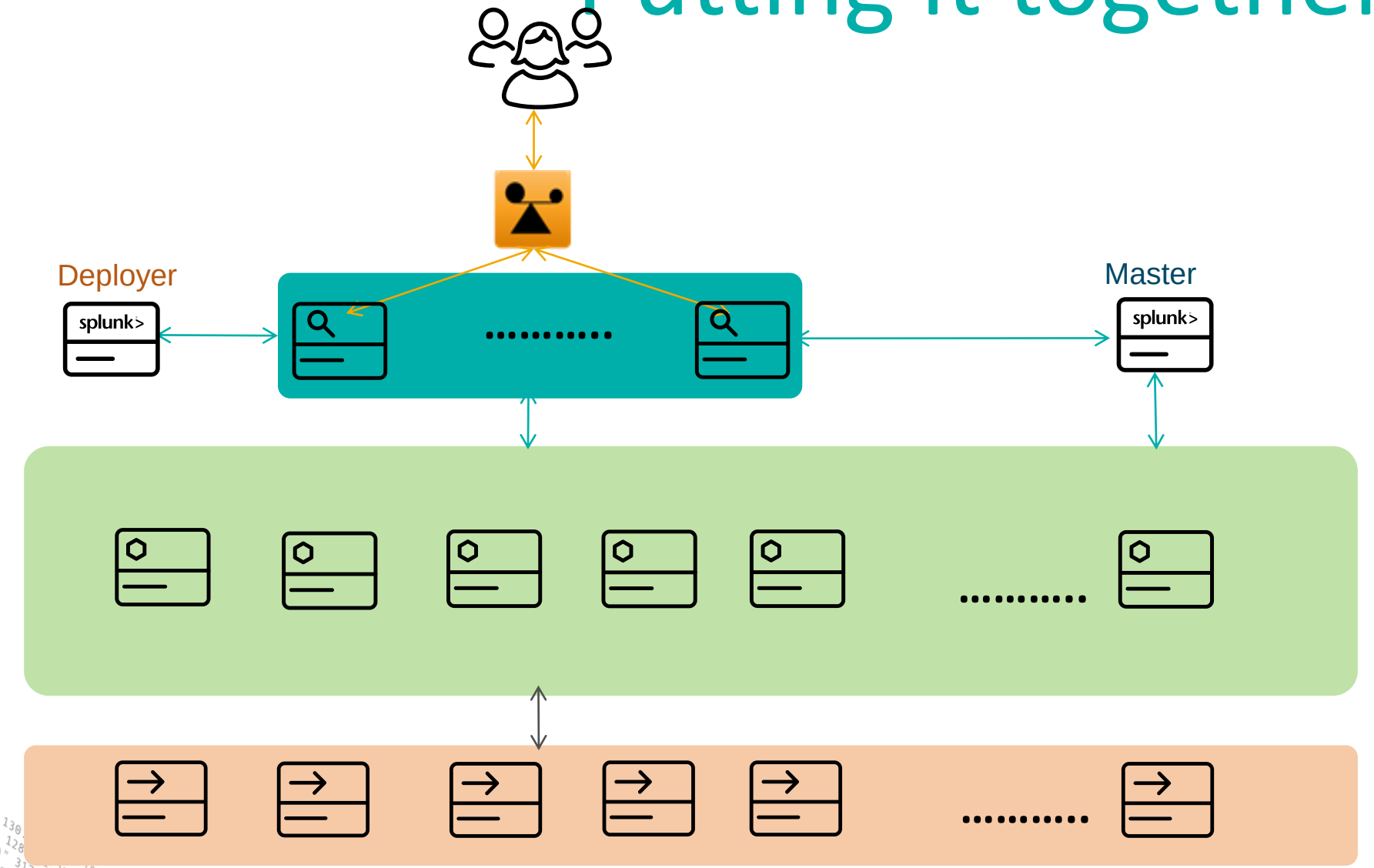
- Retention Options
  - Total Index size in MB
    - Oldest bucket is purged when [maxGlobalDataSizeMB] is reached
  - Age of events
    - Bucket is purged when the oldest event reaches [frozenTimePeriodInSecs]



splunk> .conf18

Final Words

Putting it together



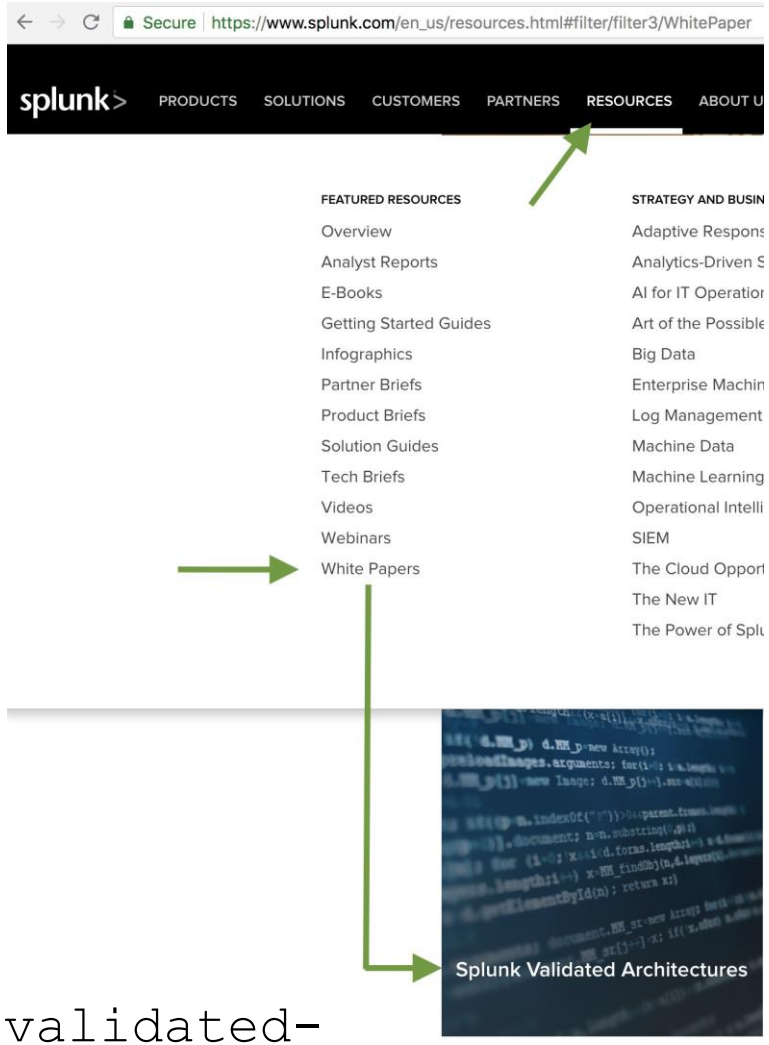
END

Top Takeaways

- 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

SVAs – Splunk Validated Architectures

- Recommended and supported Splunk deployment topologies based upon the following design pillars:
  - Availability
  - Performance
  - Scalability
  - Security
  - Manageability



https://www.splunk.com/pdfs/white-papers/splunk-validated-architectures.pdf

END

Resources

- Splunk Validated Architectures
  - https://www.splunk.com/pdfs/white-papers/splunk-validated-architectures.pdf

- Docs
  - High Availability
  - http://docs.splunk.com/Documentation/Splunk/latest/Deploy/Useclusters
  - http://docs.splunk.com/Documentation/Splunk/latest/Deploy/Indexercluster
  - https://docs.splunk.com/Documentation/Splunk/latest/DistSearch/AboutSHC

  - Rolling restarts and upgrades:
  - http://docs.splunk.com/Documentation/Splunk/latest/DistSearch/SHCrollingupgrade
  - http://docs.splunk.com/Documentation/Splunk/latest/Indexer/Userollingrestart#Use_searchable_rolling_restart
  - http://docs.splunk.com/Documentation/Splunk/latest/Indexer/Searchablerollingupgrade

Q&A

splunk> .conf18

Thank You

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