Architecting Splunk Enterprise Deployments

Document Usage Guidelines

- Should be used only for enrolled students
- Not meant to be a self-paced document, an instructor is needed
- Do not distribute

Course Prerequisites

- Required
  - Splunk Fundamentals 1 and Splunk Fundamentals 2
  - Splunk Enterprise System Administration
  - Splunk Enterprise Data Administration
- Strongly Recommended
  - Advanced Searching and Reporting with Splunk
  - Advanced Dashboards and Visualizations

Note

In order to receive credit for this course, each student is expected to complete their own lab exercises.

Course Goals

- Apply best practices for architecting and documenting Splunk Enterprise distributed deployments
- Provide a framework and methodology for Splunk deployments

Course Outline

Module 1: Introduction

Module 2: Project Requirements

Module 3: Infrastructure Planning: Index Design

Module 4: Infrastructure Planning: Resource Planning

Module 5: Clustering Overview

Module 6: Forwarder and Deployment Best Practices

Module 7: Integration

Module 8: Performance Monitoring and Tuning

Module 9: Use Cases

Module 1: Introduction

Module Objectives

- Define the responsibilities of a Splunk Architect
- Introduce the Splunk deployment planning process and tools
- Review the network topology for Buttercup Games

Splunk Architect Tasks

- Capacity planning
  - What is the current need and how do I plan for future growth?
- Create a deployment strategy
  - Based on capacity planning, what type of hardware is needed?
    - How many search heads, indexers? Do I need clustering? Should I use Splunk Cloud?
  - How many sites need access to Splunk?
- Define backup strategies
- Implement High Availability and Disaster Recovery strategy
- Create test environments
- Document Splunk deployment architecture
- Evangelize Splunk within your organization

Architect and Administrator Responsibilities

Responsibility	Who?	Description
Budget Management	Program Manager	• Facilitate executive and procurement discussions/documentation to obtain funding for additional license and infrastructure as environment grows
Capacity Planning	Architect	• Regular monitoring of the environment in conjunction with projected future growth to identify future capacity needs and recommendations of vertical and horizontal scaling to meet future demands
Splunk Deployment Installation	Architect	• Deploying Splunk to new environments as needed. • Creation of new non-production environments for testing/specific purposes.
Splunk Deployment Management	Architect & Admin Architect Admin	• Scaling existing environments to meet capacity needs by adding new indexers, search heads, etc. • Altering the state of existing environments including switching from non-clustered to clustered system, single site to multi-site, transitioning shared roles to dedicated roles, etc. • Managing configurations effectively including version control processes to ensure bad configurations can be reverted
Splunk Deployment Problem Management	Admin	• Proactively monitor and respond to issues with the system including regular monitoring of the MC. • Respond to system alerts generated by Splunk which indicate system health issues. • Respond to and investigate issues reported by end users. • Manage support tickets with Splunk support as necessary.

Architect and Administrator Responsibilities (cont.)

Responsibility	Who?	Description
Software Upgrades	Admin Architect	• Routine software upgrades to Splunk and underlying layered technologies • Define and execute testing procedures to ensure successful upgrades
Data Onboarding Management	Architect Admin Admin	• Define and manage strategies and processes to ingest new data sources • Work with users to request new data sources to be onboarded • Prioritize new requests
Data Onboarding Execution	Admin	• Document existing and newly ingested data sources • Design and deploy inputs to UFs/HFWs to capture new data • Manage parsing, event breaking, timestamping, etc. • Move configuration through non-production testing as required by the organization • Deploy changes to production • Repeat above for dashboards, reports, alerts, etc.
User Enablement / Onboarding	Admin	• Define and manage strategies and processes for new users • Manage access controls to the system • Provide education and training resources • Define and provide support for end user issues
Backups	Architect & Admin	• Define, implement, and test backup strategies for Splunk including backing up of all Splunk configurations and Splunk Index data

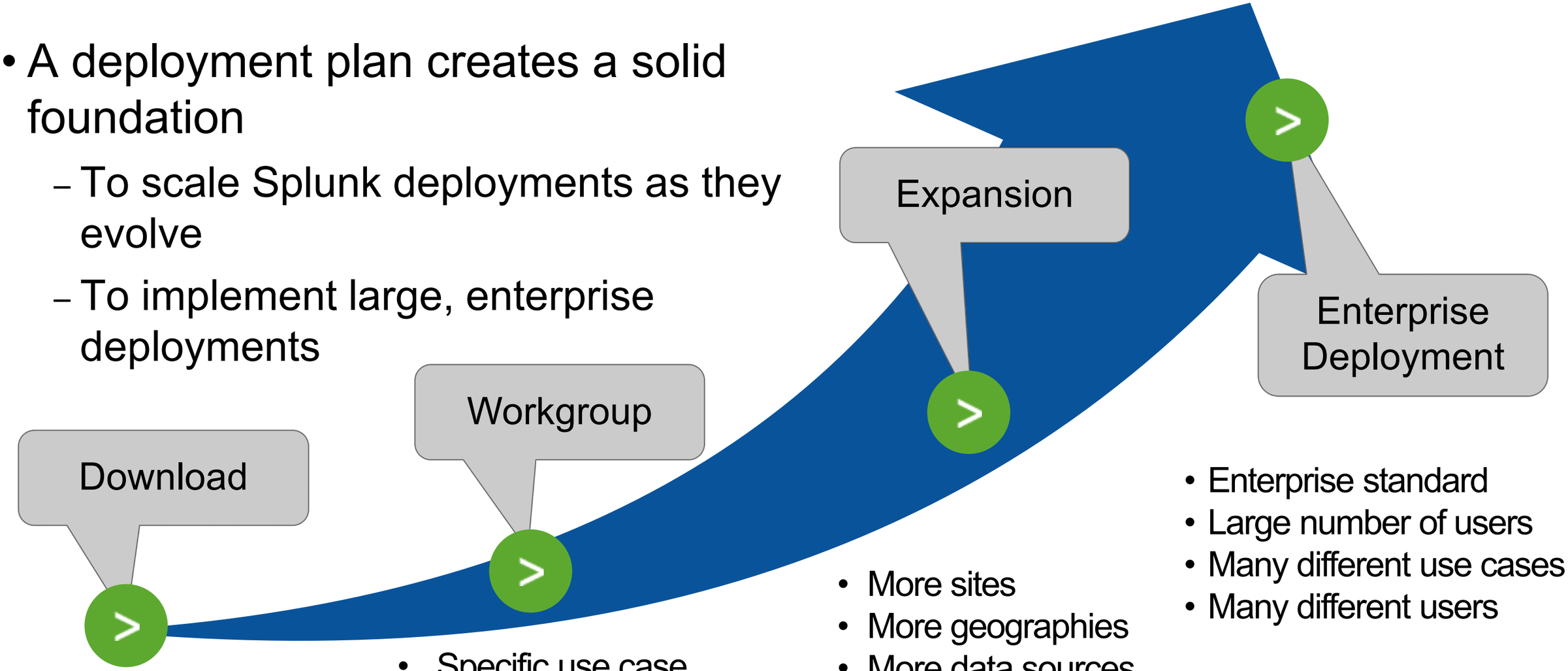
For details about upgrading Splunk, please review the following document:
https://docs.splunk.com/images/d/d3/Splunk_upgrade_order_of_ops.pdf

Architect and Administrator Responsibilities (cont.)

Responsibility	Who?	Description
High Availability / Disaster Recovery	Architect	Define, implement, and test disaster recovery and high availability for Splunk as required by your organization
System Documentation	Architect	Document installation steps, support procedures, backup and recovery, troubleshooting, etc.
Evangelizing Splunk	Architect	Ability to work with prospective Splunk teams or users to do all of the following: • Discuss the problems and business domains of the prospective users to identify opportunities to use Splunk to solve their challenges and pain points • Foster a culture and environment where Splunk can spread/thrive including leading regular office hours and internal collaboration or technology sharing events

Deployment Scaling

- A deployment plan creates a solid foundation
  - To scale Splunk deployments as they evolve
  - To implement large, enterprise deployments

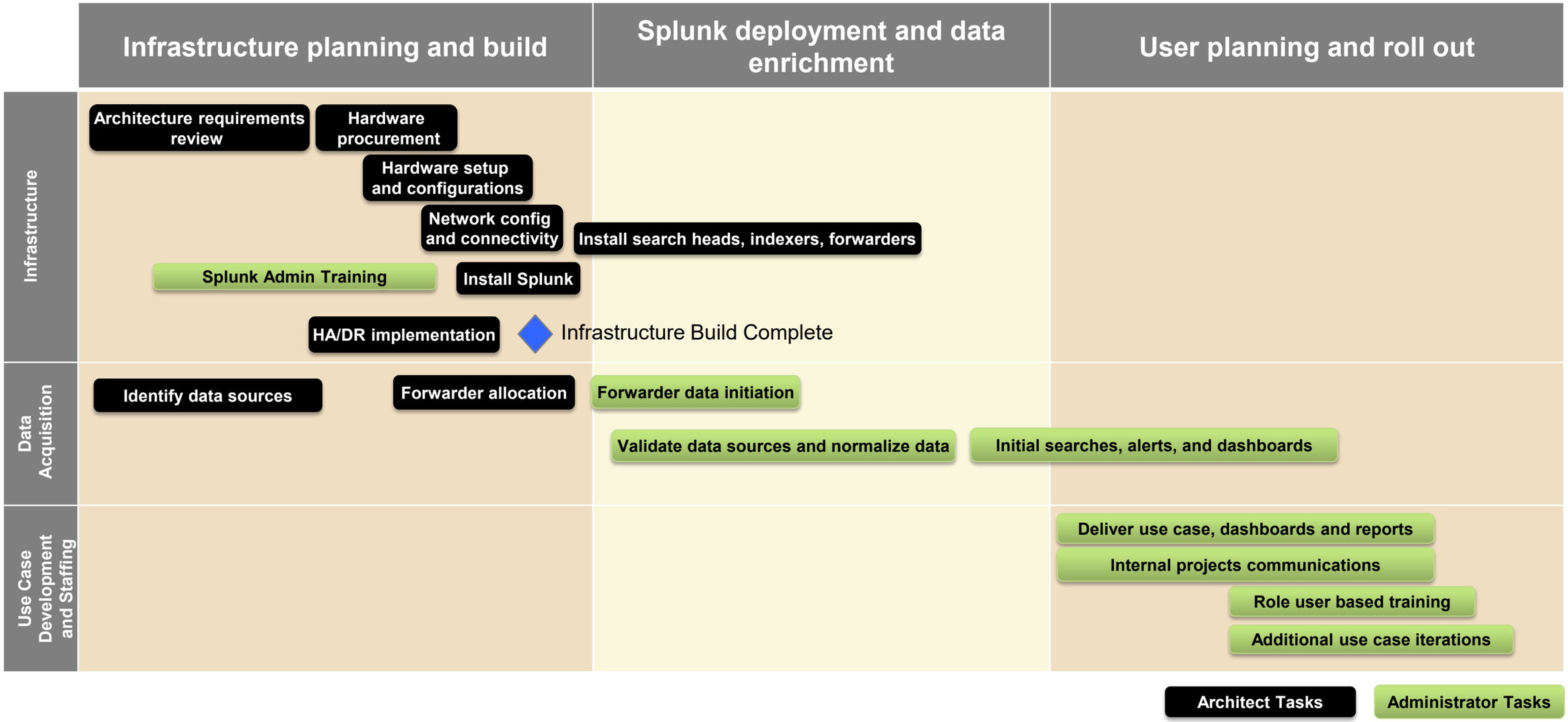


- Initial User
- Specific use case
- Specific users
- More sites
- More geographies
- More data sources
- More data volume
- Enterprise standard
- Large number of users
- Many different use cases
- Many different users

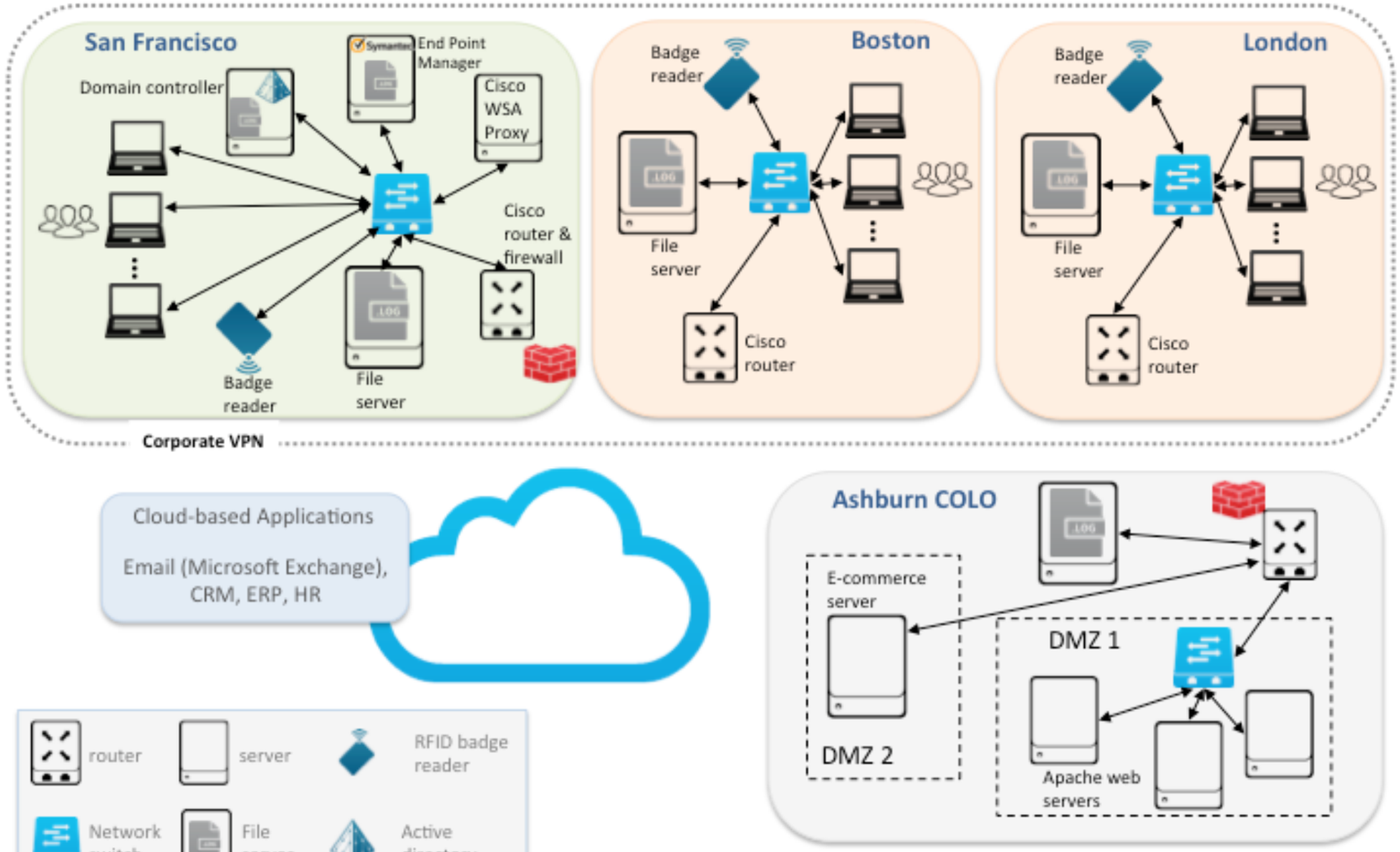
What is in a Deployment Plan?

- A deployment plan should include:
  - Deployment goals
  - User roles / staffing list
  - *Current* topology diagrams
    - Physical environment
    - Logging
  - Splunk deployment topology
  - Data source inventory
  - Data policy definition
  - Suggested Splunk apps
  - Education / training plan
  - Deployment schedule

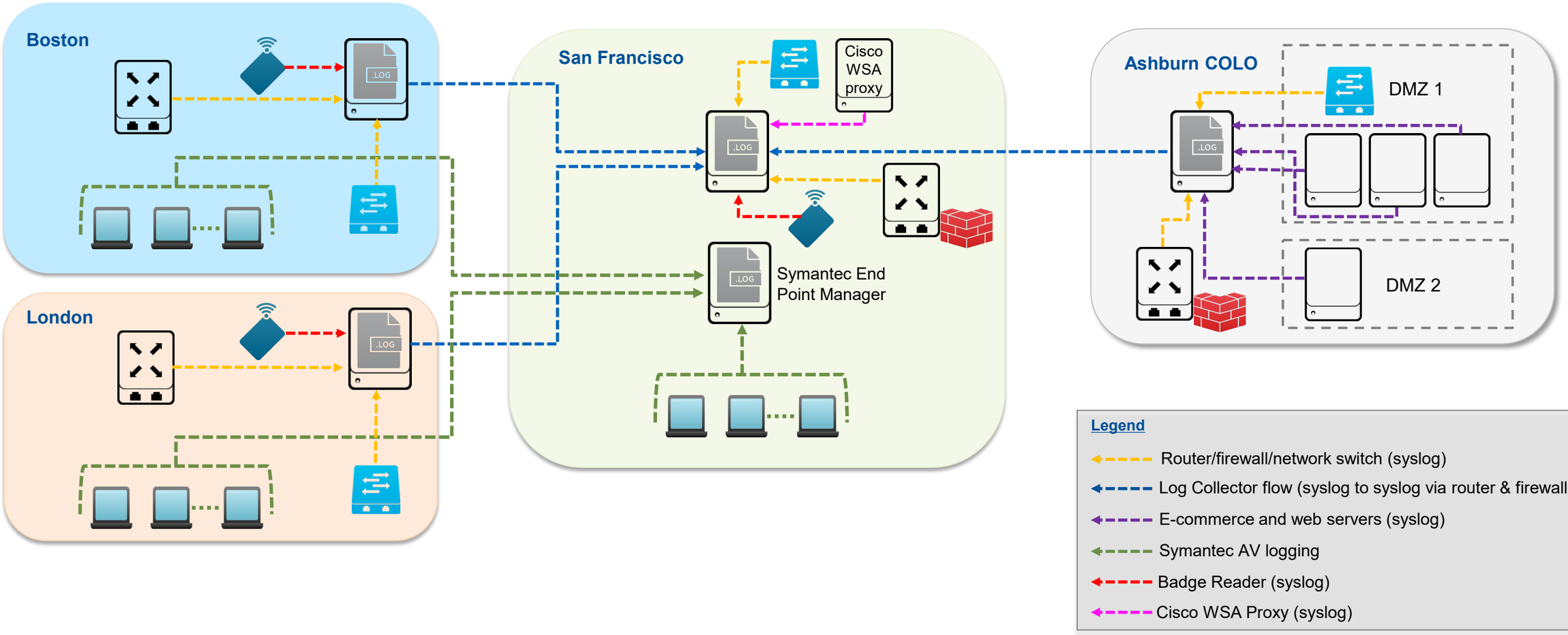
Splunk Deployment Process



Buttercup Games (BCG) - IT Environment



BCG Logging Environment - Corporate



Module 1 Lab Exercise

Time: 10 minutes

Tasks:

- Read the lab document for Module 1 (pages 1- 4)
  - ArchSplunk73_labs.pdf
  - If you have difficulty seeing the graphics in the lab document, they also appear on the previous pages
- Download planning tools (optional)
  - Data Source inventory (`data_inventory.docx`)
  - Index planning and sizing (`data_sizing.xlsx`)
  - Splunk icon library (`SplunkIconLibrary.ppt`)
  - Use case checklist (`use_case.pdf`)

Module 2:
Project Requirements

Module Objectives

- Identify the information that is needed for deployment decisions
- Provide lists and resources to aid in collecting requirements

Key Planning Information

- Identifying requirements is a first step in the deployment
  - More information is uncovered throughout the phases of deployment
- Gather raw material for the deployment plan at the beginning, when possible
  - Overall goals for the deployment
  - Key users, including their goals and use cases
  - Current environment
    - Physical environment
    - Monitoring tools in use
  - Expected daily data ingestion
  - Data sources

Use Cases

- What will the users do with Splunk?
  - What tasks will they perform?
  - What reports will they generate?
- What other use cases may apply?
- For ideas and inspiration:
  - Why Splunk? Customer success stories
    https://www.splunk.com/en_us/customers.html?CustomerSuccessStories
  - Use Case Checklist handout
    use_case.pdf
  - Splunk apps
    http://splunkbase.splunk.com

Current IT Environment

- What is the current overall IT topology?
  - Data centers
  - Network zones
  - Number and type of servers
  - Location of users
- Obtain a network diagram
  - Are there security restrictions among data center and network zones?
  - What is the bandwidth among data center and network zones?
- Authentication
  - What system is currently in place?

Current Logging Environment

- Are any logs or other data sources collected or centralized today?
  - Logged to SAN / NAS devices
  - Centralized via syslog, syslog-ng, rsyslog, Kiwi, Snare, etc.
  - Parsed and stored in a SQL database
- What tools are in use?
  - Are there log parsing / scraping tools or scripts in place?
  - Are there any query tools in place? Who uses each tool?
  - Are logs a source for a monitoring system?
  - What ticketing system is in place?
- Is there an expectation that Splunk will integrate with or replace existing tools?

General Requirements

- Security
  - What security policies may affect the collection, retention, and reporting of data?
  - What approvals will be needed?
- Regulatory
  - Are there laws, regulations, or policies that affect the data collection, reporting, etc.?
- High Availability or Disaster Recovery
  - Is data replication required?
  - Does your data need to be searchable at all times?

Data Sources

- Data Source Inventory
  - What is the superset of data sources needed by all users?
  - How much data is generated per day?
- Data Policy
  - How long should each data source be retained?
  - Who can see particular data elements?
  - What data needs protection against tampering?
  - What proof of integrity needs to be provided?
  - Will Splunk be the primary repository for data?
- Handout
  - data_inventory.docx

Example Data Source Inventory

Data Source:	Cisco ASA Firewall log
Location and Type: file input (name and path), API, network port, device, etc.	UDP:514
Ownership and Access:	Network team owns; Security can also access
Data Volume Average per day: Peak:	 500 MB 6000 EPS at ~100 bytes/event
Retention:	6 months
Format or sourcetype:	syslog?
Collection method:	How will this data get to the indexers? Is there a collection point at each location?

Module 2 Lab Exercise

Time: 15 minutes

Task:

Complete the abbreviated data source inventory

- You may not know enough to supply answers for all data sources
- Part of the inventory has been completed for you
  - Do you agree with the completed portion?
  - Do you have questions for the sponsor?

Module 3:
Index Design

Module Objectives

- Define index implementation
- Design indexes
- Estimate storage requirements for indexes
- Identify relevant apps
  - Document impact on inputs and indexes

Splunk Indexing Review

- Optimized for quickly indexing and persisting unstructured data
- Minimal schema for persisted data
  - An event consists of raw text, timestamp, source, source type and host
    - This is called "rawdata" and is stored in a compressed file
- Once data enters Splunk, it is
  - Parsed for line-breaking, timestamps and other metadata fields
  - Persisted in its raw form
  - Indexed by the metadata fields, along with all the keywords in the raw event text
    - Splunk uses an "inverted index" structure for the `.tsidx` files

Splunk Indexing Review (cont.)

- Additional processing on raw events is deferred until search time
  - This serves 4 important goals:

indexing speed is increased	minimal processing is performed during indexing
bringing new data into the system requires less effort	no schema planning is needed
the original data is persisted only if there is no transformation	
the system is resilient to change	data parsing problems do not require reloading or re-indexing of the data

Index Types

- There are two types of indexes you can create in Splunk:
  - **Event Indexes**
    - Default index type
  - **Metrics Indexes** is a set of measurements that contain:
    - Timestamp
    - Metric names
      - Uses a dotted hierarchy for a namespace such as nginx.upstream.responses.5xx
    - Value
    - Dimensions
      - Metadata about a metric – for example: region, instance type, or technology

For more information about Metrics, please refer to the following documentation:
http://docs.splunk.com/Documentation/Splunk/latest/Metrics/Overview

Disk Storage Details – Files

- As Splunk indexes your event data, it creates two main types of files:
  - **rawdata** contains the original data in a compressed form
    - For "syslog-like" data, this may be 10% to 15% of the raw pre-indexed data
    - Size is dependent on how well data compresses
  - **Index** files (`.tsidx`) which point to the unique terms
    - These files range in size from approximately 10% to 110% of the rawdata size
    - Size is strongly affected by the number of unique terms in the data
    - Adding indexed field extractions will increase the size of the `.tsidx` files
- Searchable buckets contain both rawdata and index files
  - Exception: replicated buckets may have only rawdata

Types of Index Files

- Compressed raw data
- Inverted index (`*.tsidx`)
- Metadata (`*.data`)
  - Information about sources, source types and hosts of the events contained in each bucket
- Bloom filters (bloomfilter)
  - Efficient data structure that authoritatively rules out buckets
    - Provides 100% certainty that a search term is not present in a bucket
  - Consulted by every search
  - Very fast (1-2 IO)
    https://en.wikipedia.org/wiki/Bloom_filter

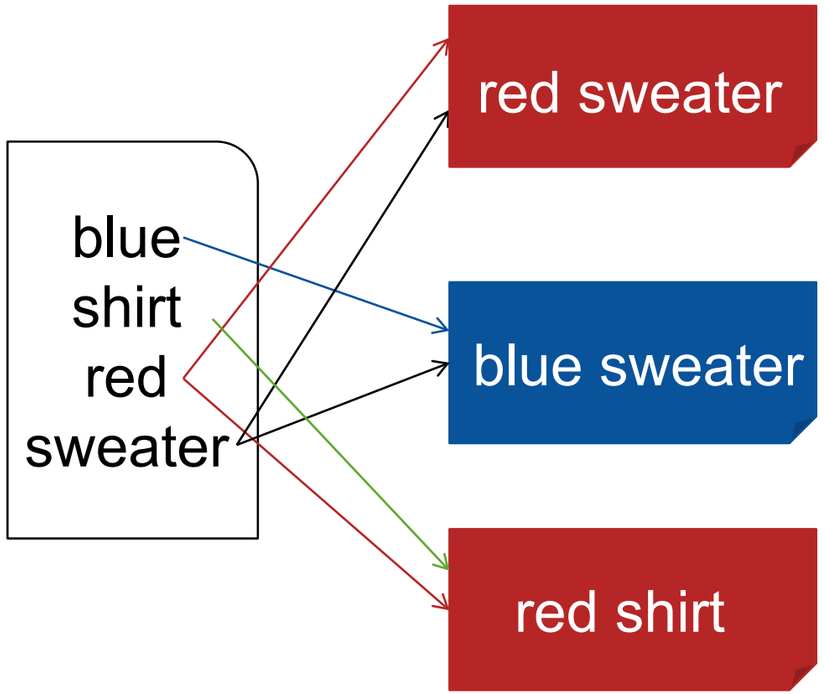
```
drwx--x---  3 root root 4.0K Sep 11 22:10 .
drwx------ 31 root root 4.0K Sep 14 23:15 ..
-rw-------  1 root root  30K Sep 11 22:06 1505167548-1505166495-2478884372982387461.tsidx
-rw-------  1 root root 1.5K Sep 11 22:06 bloomfilter
-rw-------  1 root root   75 Sep 11 22:10 bucket_info.csv
-rw-------  1 root root  249 Sep 11 22:05 Hosts.data
drwx------  2 root root 4.0K Sep 11 22:05 rawdata
-rw-------  1 root root    6 Sep 11 22:05 .rawSize
-rw-------  1 root root  483 Sep 11 22:05 Sources.data
-rw-------  1 root root  175 Sep 11 22:05 SourceTypes.data
-rw-------  1 root root  305 Sep 11 22:05 Strings.data
```

Inverted Index

- Index data structure that maps from content (such as words or numbers) to its locations

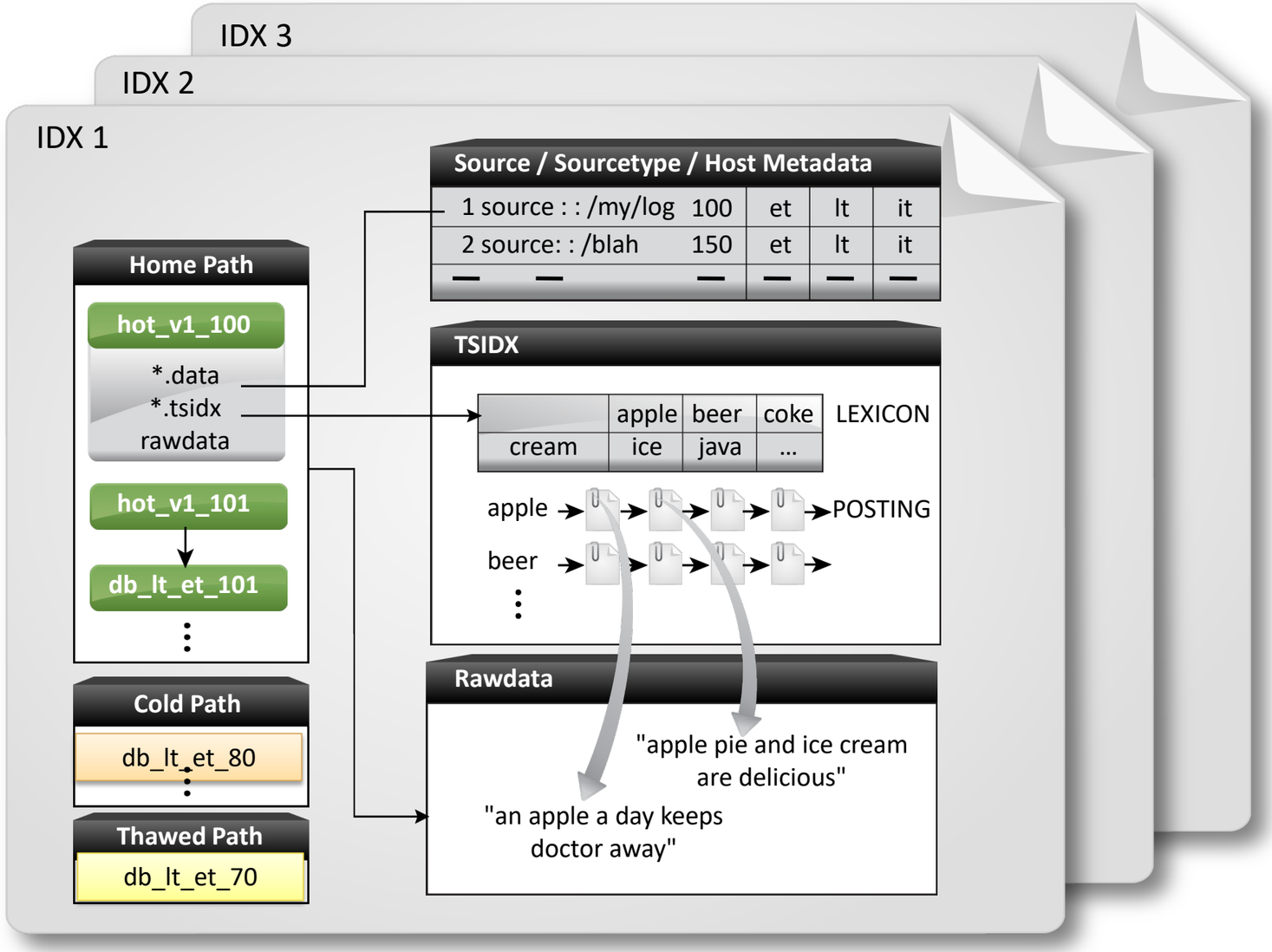
  en.wikipedia.org/wiki/Inverted_index

- An inverted index allows fast full text searches
- Splunk `.tsidx` uses an inverted index structure to map keywords to their locations in the rawdata



Source: Wikipedia

Splunk Index Files



Managing Disk Usage with TSIDX Reduction

- Full index files are normally kept with the rawdata until the bucket is frozen
- To save disk space, you can enable tsidx reduction in `indexes.conf`
  - Based on an age setting
    - **`timePeriodInSecBeforeTsidxReduction = 7776000`** (90 days)
  - Replaces some index files with mini-index files
  - Removes `merged_lexicon.lex`
  - Overall, the bucket size is typically reduced by 1/3 to 2/3
- Searches will take much longer over reduced buckets
    - Searches for rare terms are particularly affected

Note

Change the default search time to a small time range (i.e. 24 hours) to avoid each search hitting the reduced buckets.

http://docs.splunk.com/Documentation/Splunk/latest/Indexer/Reducetsidxdiskusage

Estimating Indexing Volume

- Estimate Input Volume (Data Capacity)
  - Verify raw log sizes
  - Daily, peak, retained, future volume
  - Total number of data sources
  - Total number of hosts
  - Add volume estimates to data source inventory / spreadsheet

- A rule of thumb for syslog-type data is that once it has been compressed and indexed in Splunk, it occupies approximately 50% of its original size:
  - 15% for the rawdata file
  - 35% for associated index files

Testing Indexing Compression

- Confirm estimates with actual data
  - Create a baseline with real or simulated data
  - Find compression rates
  - Leverage `_internal` index metrics to determine actual input volumes
- To test specific types of data
  - Index a sample of data, then check the sizes of the resulting directories in `defaultdb (at least two buckets)`
- Use the Monitoring Console to get a baseline of compression rates for each index

See documentation for step-by-step space estimation method:
http://docs.splunk.com/Documentation/Splunk/latest/Capacity/Estimateyourstoragerequirements

Apps and Indexes

- Many apps define inputs and indexes
- Check apps against your initial index design and data source inventory
  - You may need to acquire new data sources to support the app
    - Add them to the inventory
  - Identify the indexes created by the app
  - Identify which inputs will map to which indexes
    - There should be no overlap or conflicts
    - Each data source goes to a single index
    - Include any sourcetype information that is defined in the app
  - Apps must be integrated with the overall index and inputs

Splunk Apps

- Which Splunk apps are appropriate?
  - Apps are often chosen based on
    - Devices or technologies in the production environment
    - Use cases
    - Inputs
- Apps often have particular infrastructure requirements
- Most apps are free
- While apps can always be added later, plan up front if possible

http://www.splunkbase.com

Key Criteria for Index Design

- When to partition data into different indexes
  - To separate event and metrics indexes
  - Retention
    - How long should the data be kept? Online? Offline?
    - All retention settings apply on a per-index basis
    - All data sources within an index should have the same retention
  - Access
    - Who can search against the data?
    - All data sources within an index will have the same visibility
      - Access to indexes is controlled by role

Additional Design Criteria

- Search Performance
  - Less common to differentiate here, but *may* be important
  - Mixed data = mixed lexicon, larger number of keywords, therefore larger index
  - Things searched together can be indexed together
  - Example:
    - Both a high-volume / high-noise data source and a low-volume data source feed into the same index
    - If you search mostly for events from the low-volume data source, the search speed will be slower than necessary
    - To improve performance, create a separate index for the high-volume data source

Module 3 Lab Exercise

Time: 15 minutes

Tasks:

- Estimate the disk space required
- Index design
  - Identify the indexes
  - Estimate the Splunk license required for indexing

Challenge:

- Identify potential apps
  - Identify apps for the environment
  - Add any index and input information from the apps to your solution

Module 4: Resource Planning

Module Objectives

- Determine sizing based on Splunk usage
- Define reference server requirements
- Describe the impact of acceleration and apps on resource sizing

Basic Sizing Considerations

- Amount of incoming data
- Amount of indexed (stored) data
- Number of concurrent users
- Number of scheduled searches
- Types of searches
- Acceleration
- Specific Splunk apps
  - Can affect incoming and indexed data
  - Can affect number of concurrent searches

Disk Storage Details – About IOPS

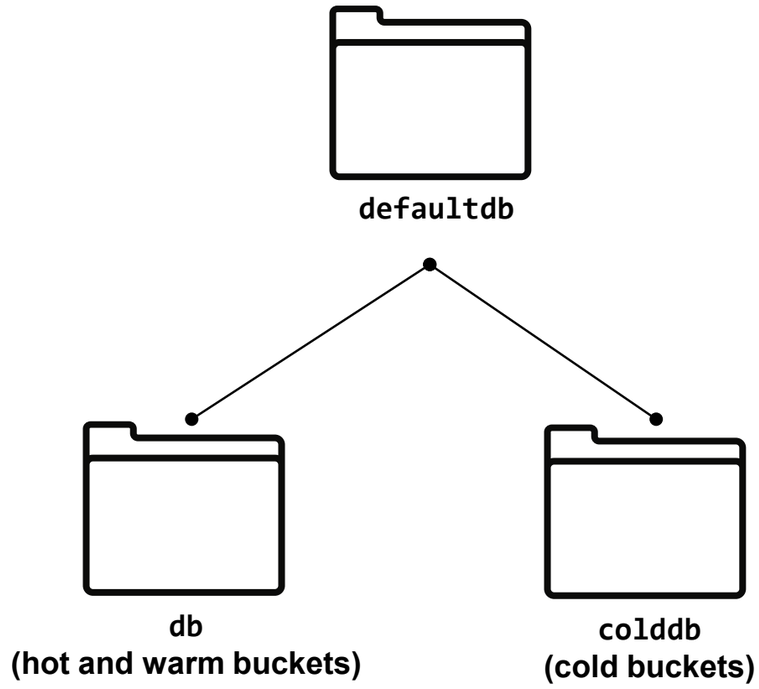
- Input/Output Operations Per Second (IOPS) measures disk throughput
  - Hard drives read and write at different speeds
    - Average IOPS is the blend between read and write speed
  - To get the most IOPS, choose drives with
    - High rotational speeds
    - Low average latency and seek times
- Splunkbase has an app to analyze disk performance using Bonnie++

  https://splunkbase.splunk.com/app/3002/
- Links to additional information on determining IOPS

  www.symantec.com/connect/articles/getting-hang-iops-v13
  www.cmdln.org/2010/04/22/analyzing-io-performance-in-linux

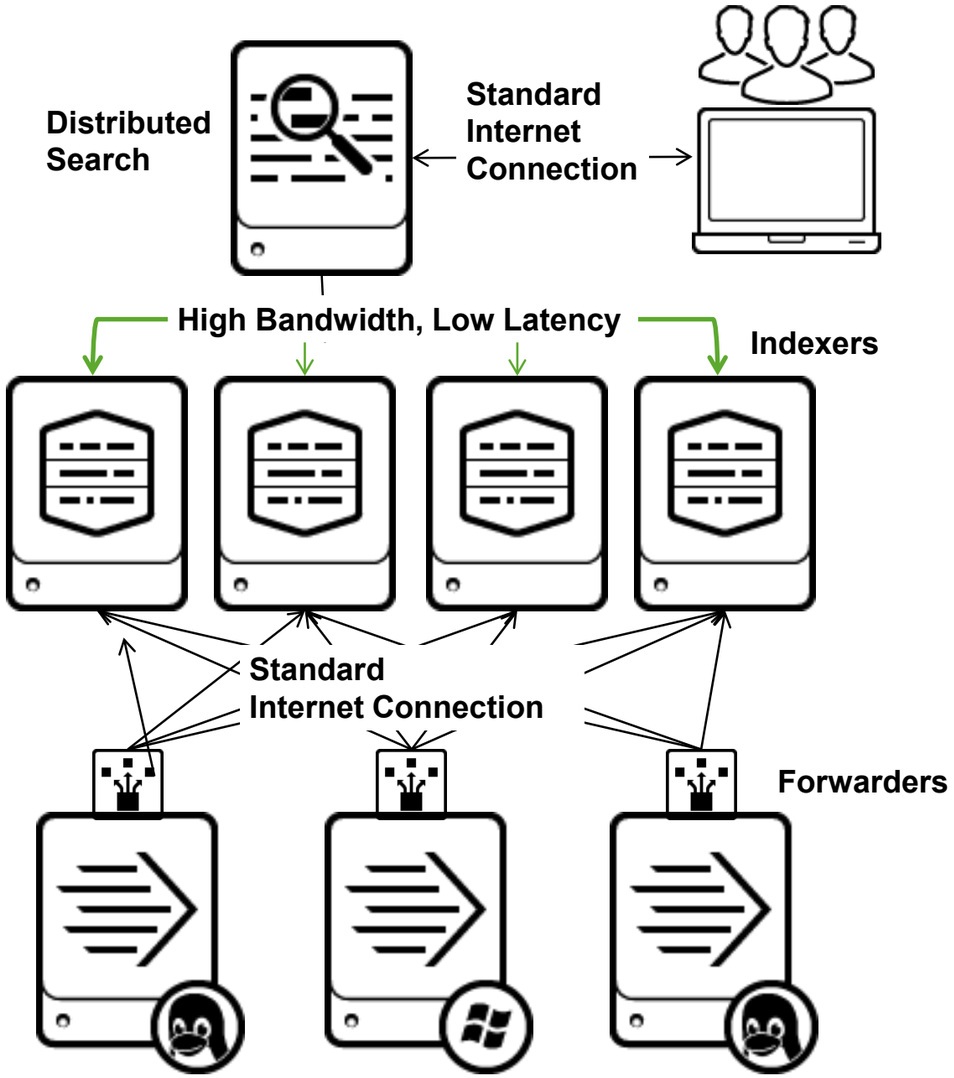
Disk Storage Details – SAN and NAS

- Storage Area Networks (SAN) and Network-attached Storage (NAS)
  - Not a good choice for hot and warm buckets (**`db`**)
  - Suitable for cold buckets (**`colddb`**)
  - Searches over long time periods that utilize **`colddb`** are slower
- High performance SAN can be used in environments with high speed networking and low latency
- **IOPS** and **reliability** are the definitive factors



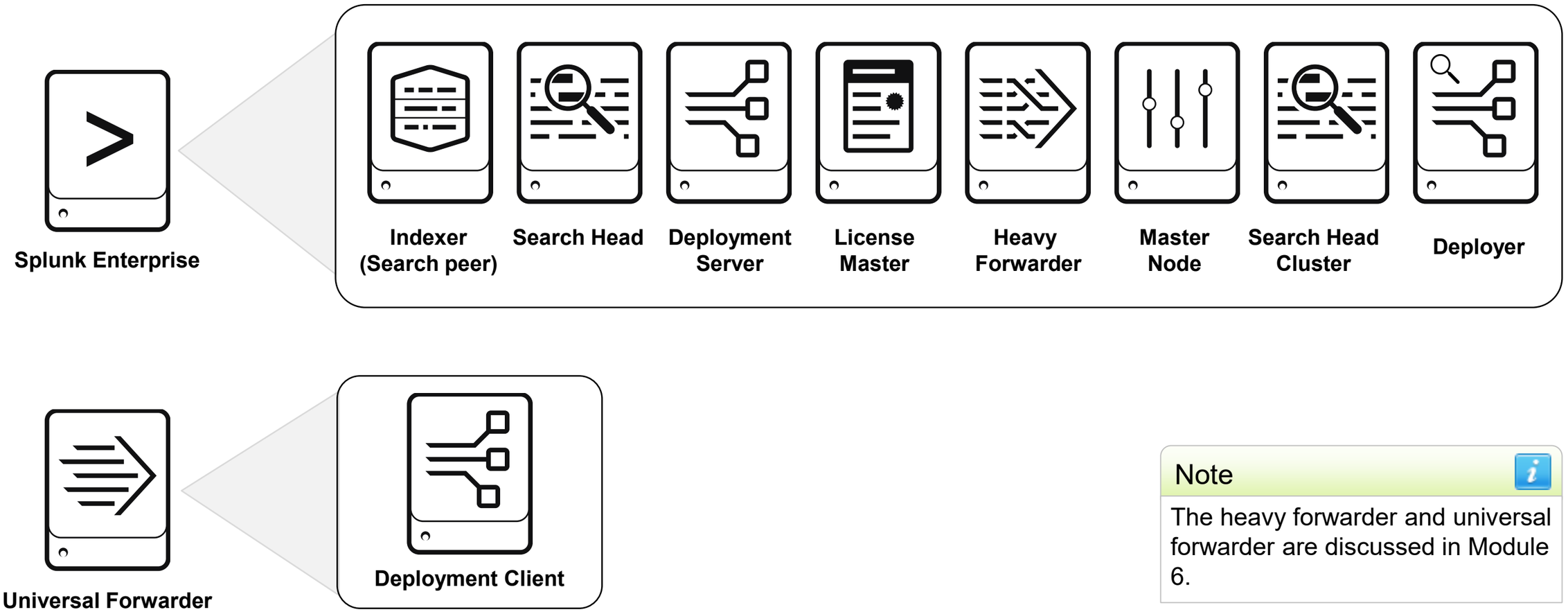
Foundation for Splunk Deployment

- A low latency network
  - 1 Gb is the minimum bandwidth
  - Under 200ms search head to search head in a search head cluster
  - Under 100ms indexer to indexer
- A solid enterprise-wide time infrastructure
  - NTP for time synchronization
- A solid Domain Name Service (DNS)
  - Splunk can significantly increase load on DNS
- Turn off Transparent Huge Pages (THP)
  - THP is a feature on some Linux distros
- Increased linux ulimit settings
  - To accommodate a large number of buckets, forwarders, and users



Splunk Components

What are the requirements for each type of Splunk instance?

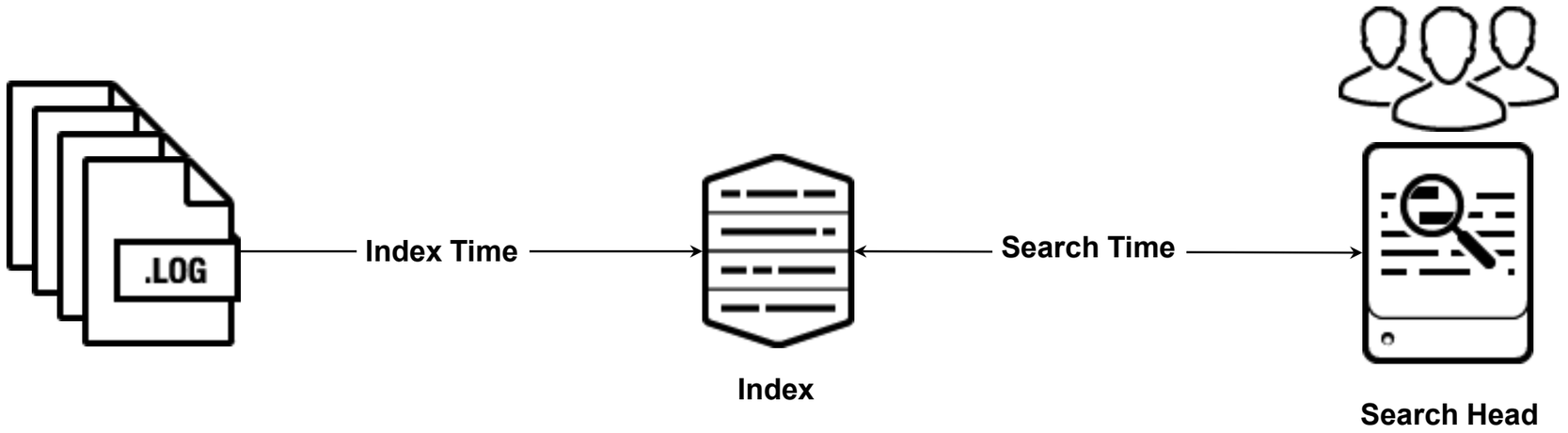


Note

The heavy forwarder and universal forwarder are discussed in Module 6.

Component Types – Indexer

- The Splunk component that indexes data, transforming raw data into **events** and placing the results into an **index**
- It also searches the indexed data in response to search requests



Indexer – (Single Instance) Reference Server

- Need additional servers for:
  - Increased reporting, searching, users
- Indexing alone can use **4** full cores at full load
- Each concurrent search needs a full core
  - More servers reduce search duration and increase search throughput
- Based on minimum hardware requirements, indexer can ingest up to 300GB/day while supporting a search load

Hardware	Intel x86 64-bit chip architecture
CPU	**12 CPU cores** at **2+ GHz** or more per core
Memory	**12 GB RAM**
Disk	Disk subsystem capable of **800 IOPS** (e.g. 8x15K RPM SAS drives in RAID 1+0 configuration)
Network	Standard 1 Gb Ethernet NIC Optional 2nd NIC for management network
OS	Linux or Windows - 64-bit version

Indexer – (Single Instance) Max Performance

- Indexing performance
  - **Up to 20MB per second** (1700GB/day) of raw indexing performance
    - Does not include search or other index related activity
- Search performance
  - Up to 50,000 events/second for dense searches
  - Up to 5,000 events/second for sparse searches
  - Up to 2 seconds/index bucket for super-sparse searches
  - From 10 – 50 buckets/second for rare searches with bloom filters

> **Note**
>
> This information is based on the reference server specifications on the previous page.

For more information about how searches impact performance, read:
http://docs.splunk.com/Documentation/Splunk/latest/Capacity/HowsearchtypesaffectSplunkEnterpriseperformance

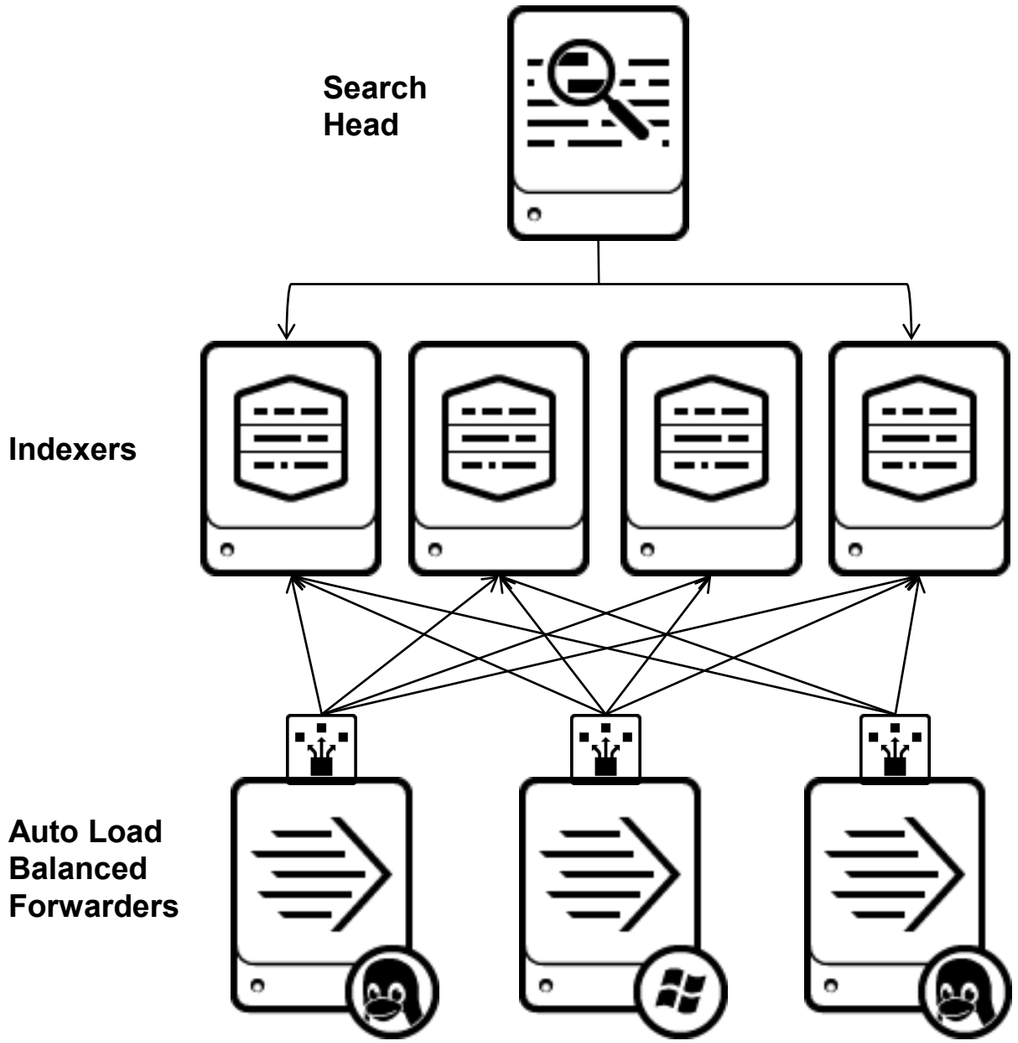
Indexer – Reference Server Specifications

	Minimum Specifications	Mid-Range Specifications	High-Performance Specifications
Hardware	Intel 64-bit chip architecture	Intel 64-bit chip architecture	Intel 64-bit chip architecture
CPU	**12 CPU cores** at **2+ GHz** or more per core	**24 CPU cores** at **2+ GHz** or more per core	**48 CPU cores** at **2+ GHz** or more per core
Memory	**12 GB RAM**	**64 GB RAM**	**128 GB RAM**
Disk	Disk subsystem capable of **800 IOPS** (e.g. 8x15K RPM SAS drives in RAID 1+0 configuration)	Disk subsystem capable of **800 IOPS** (e.g. 8x15K RPM SAS drives in RAID 1+0 configuration)	Solid State Disk (SSD) subsystem capable of **1200 IOPS**
Network	Standard 1 Gb Ethernet NIC Optional 2nd NIC for management network	Standard 1 Gb Ethernet NIC Optional 2nd NIC for management network	Standard 1 Gb Ethernet NIC Optional 2nd NIC for management network
OS	Linux or Windows - 64-bit version	Linux or Windows - 64-bit version	Linux or Windows - 64-bit version

http://docs.splunk.com/Documentation/Splunk/latest/Capacity/Referencehardware

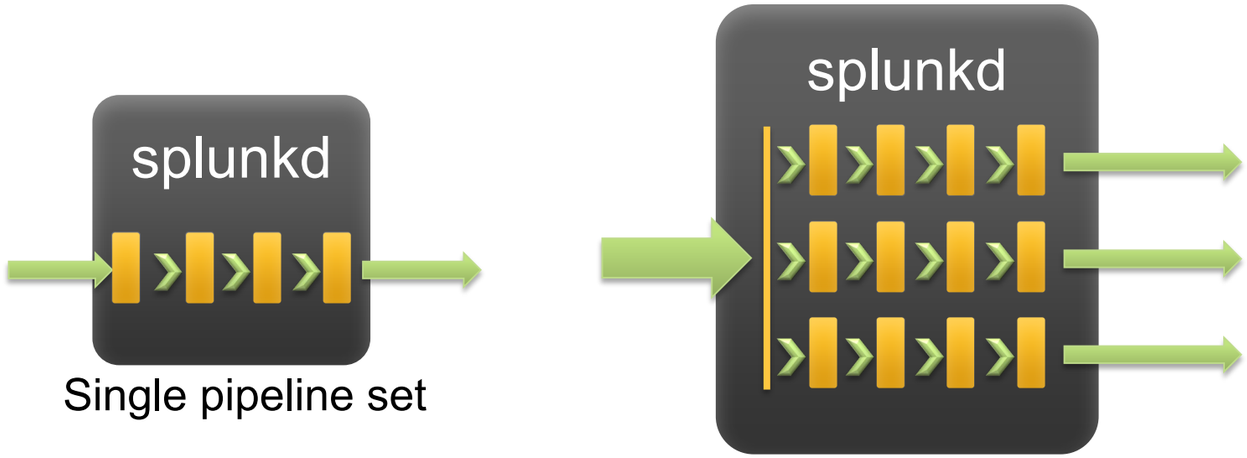
Horizontal Scaling

- Adding indexers scales capacity
  - Allows a greater daily indexing volume
  - Speeds searches
- When using multiple indexers
  - Use Splunk's built-in forwarder load balancing
  - Use distributed search
- When in doubt, the first rule of scaling is to *add another commodity indexer*



Increasing Index Parallelization

- Normally, the indexer uses approximately 4-6 cores for indexing
  - Possibly underutilizing the indexer hardware
- You can configure multiple pipeline sets to increase hardware utilization
- Use on an intermediate forwarders to spread indexer load

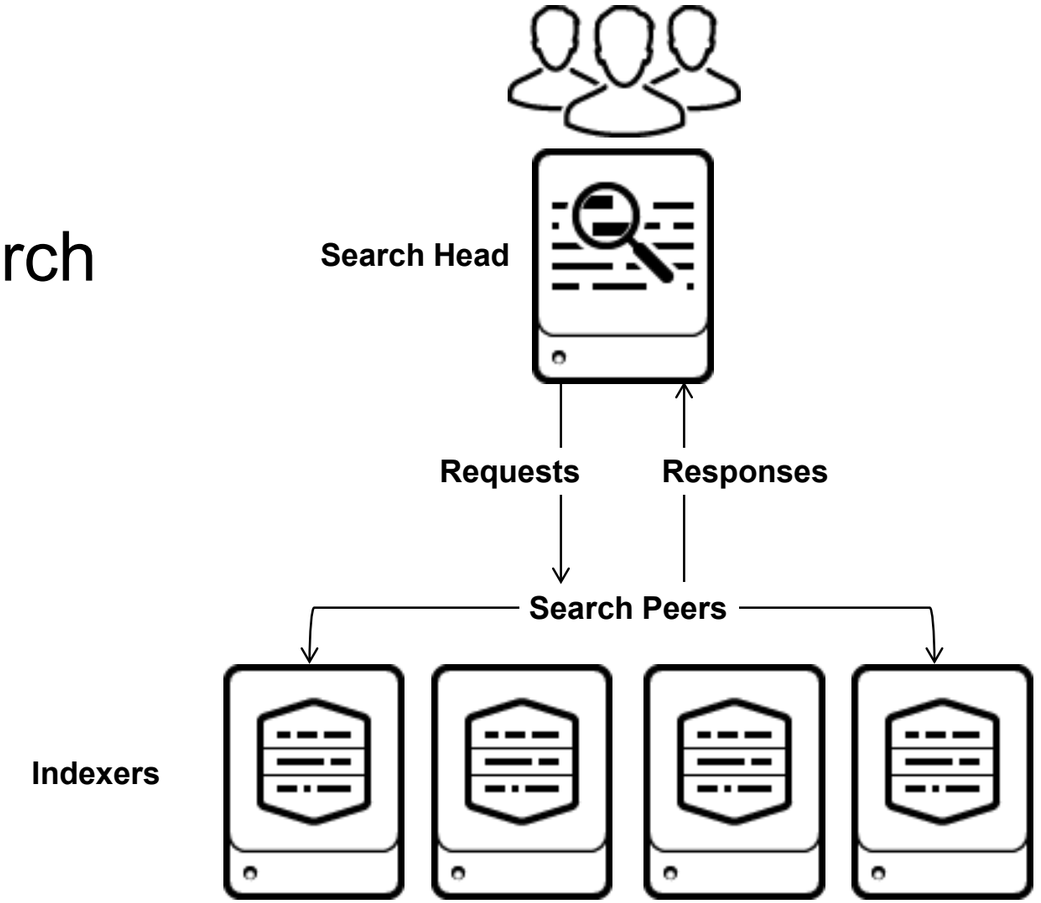


Storage Requirements

- The disk storage needed on an indexer is primarily determined by the size of all the indexes, plus base storage for the OS and configuration files
- Additional disk storage is needed for
  - Indexer Clustering
    - Data replication
    - Discussed in a later module
  - Summarization and Acceleration
    - Data models
    - Report acceleration
    - Summary indexing
    - Refer to appendix B for more information

Component Types – Search Head

- The search head handles search management functions
  - Directs search requests to a set of search peers
  - Federates or merges the results and presents them to the user



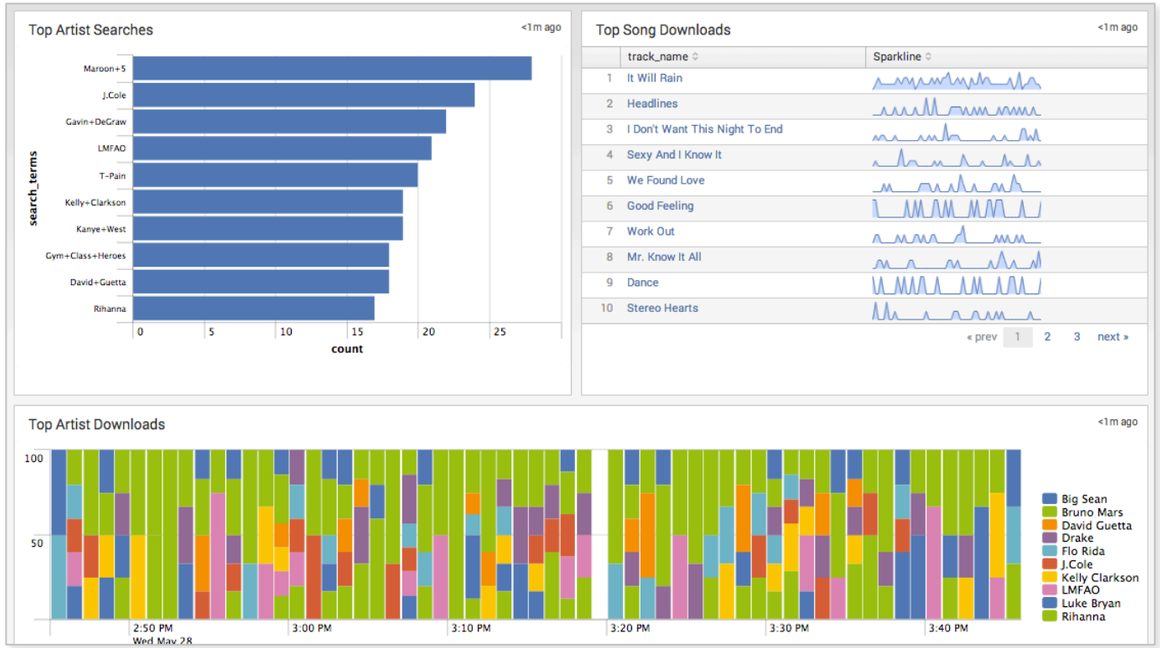
Search Head – Reference Server

- Requires more CPU than an indexer
- A search request uses 1 CPU core while the search is active
- Search heads mostly aggregate results
  - Some types of searches may create bottlenecks
- Account for scheduled searches in addition to ad-hoc searches
- More users and concurrent searches require additional CPU cores

Hardware	Intel 64-bit chip architecture
CPU	**16 CPU cores** at **2+ GHz** or more per core
Memory	**12 GB RAM**
Disk	**2 x 300GB**, **10,000 RPM** SAS hard disks, configured in RAID 1
Network	Standard 1 Gb Ethernet NIC Optional 2nd NIC for management network
OS	Linux or Windows - 64-bit version

Sizing Factors for Searching

- Use cases determine search needs
  - Number of searches (concurrent & total)
  - Number of users (concurrent & total)
- Types of searches
  - Reporting, dashboard, ad-hoc
- Jobs
  - Summarization, Alerting, Reporting
- Acceleration
  - Report, Summary Indexing, Data Model
- Apps may increase the search load
  - Real time and/or scheduled searches may be features of apps
- Plan for expansion as adoption grows



Daily Indexing Volume Guidelines

	< 2GB/day	2 to 300 GB/ day	300 to 600 GB/day	600 GB to 1 TB/day	1 to 2 TB/day	2 to 3 TB/day
Total Users: Less than 4	1 combined instance	1 combined instance	1 Search head, 2 Indexers	1 Search Head, 3 Indexers	1 Search Head, 7 Indexers	1 Search Head, 10 Indexers
Total Users: Up to 8	1 combined instance	1 Search Head, 1 Indexer	1 Search Head, 2 Indexers	1 Search Head, 3 Indexers	1 Search Head, 8 Indexers	1 Search Head, 12 Indexers
Total Users: Up to16	1 Search Head, 1 Indexer	1 Search Head, 1 Indexer	1 Search Head, 3 Indexers	2 Search Heads, 4 Indexers	2 Search Heads, 10 Indexers	2 Search Heads, 15 Indexers
Total Users: Up to 24	1 Search Head, 1 Indexer	1 Search Head, 2 Indexers	2 Search Heads, 3 Indexers	2 Search Heads, 6 Indexers	2 Search Heads, 12 Indexers	3 Search Heads, 18 Indexers
Total Users: Up to 48	1 Search Head, 2 Indexers	1 Search Head, 2 Indexers	2 Search Heads, 4 Indexers	2 Search Heads, 7 Indexers	3 Search Heads, 14 Indexers	3 Search Heads, 21 Indexers

http://docs.splunk.com/Documentation/Splunk/latest/Capacity/Summaryofperformancerecommendations

Additional Component Types

License Master	Allocates licensing capacity and manages licensing usage of all instances • All indexers, search heads, Master Nodes, deployers and deployment servers should be slaves to a license master • Contacted over network frequently by license slaves
Deployment Server	Manages Splunk configuration files for deployment clients • Operates on a "pull" model – clients "phone home" at intervals over network • Can handle 2000 phone-homes/minute on Windows and 10,000 on Linux (by default)
Master Node	Regulates the functioning of an indexer cluster
Deployer	Distributes configurations to search head cluster members

Additional Components - Sizing

	CPU	Memory	Disk	Network	Factors
License Master	low	low	low	1 Gb	number of slaves
Deployment Server	med*	med*	low	1 Gb	number of clients and number of apps + config files *CPU & memory can spike during downloads
Master Node	med	med	low	1 Gb	required for indexer cluster
Deployer	low	low	low	1 Gb	number of SHC members

- Splunk recommends that you dedicate a host for each role
  - You can enable multiple Splunk server roles on a server with caveats
- Master Node and Deployer are discussed in more detail in the Clustering section

Virtualizing Splunk

- You can virtualize any Splunk instance if you meet the minimum resource requirements
- Tips for virtualization
  - Use locally attached volumes; direct access to a dedicated volume is best
  - Separate search head VMs from indexer VMs, as together they can burst CPU
  - Ensure that sufficient resources are reserved – do not overcommit CPU or memory for Splunk VMs
  - Expect virtualization to reduce performance by 10% - 15%

  The following tech brief contains more info about virtualization:
  https://www.splunk.com/pdfs/technical-briefs/splunk-deploying-vmware-tech-brief.pdf

Splunk Enterprise in the Cloud (Self-managed)

- Offers similar performance to bare-metal hardware, but dependent on vendor provisioning the instance
  - On AWS:
    - AWS measures CPU on Elastic Compute Cloud (EC2) instances in virtual CPUs (vCPUs)
    - Each vCPU is a hyper thread of an Intel Xeon core on most AWS instance types
  - Indexing and data storage considerations
    - Elastic Block Storage (EBS) volume determines performance
    - Not all EBS volume types provide necessary IOPS
    - Provisioned IOPS and Magnetic EBS volume types are recommended
    - Verify EC2 instance type offers needed network throughput

Splunk Cloud

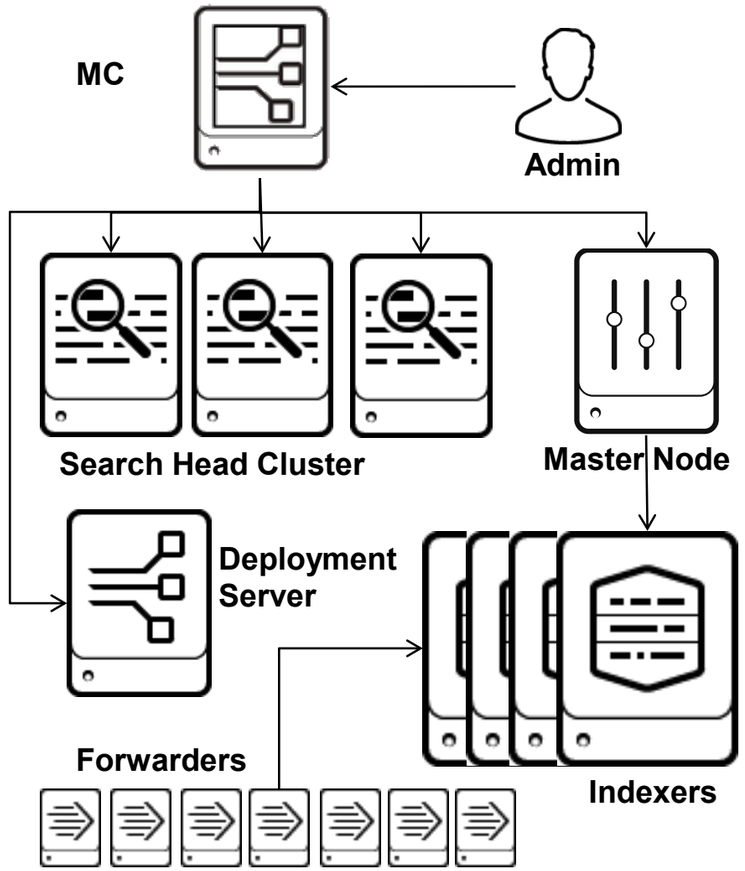
- Subscription service to store, index and search your data
  - Indexing volume is 5GB – 1TB+/day
  - Purchase capacity based on your daily ingestion rate
- Cloud resources are managed and maintained by Splunk
  - Contact Splunk Cloud Support to scale your deployment
- Data is sent to Splunk Cloud via Forwarders and HTTP Event Collector (HEC)

Note

If you are using Splunk Cloud in your environment, attending the *Splunk Cloud Administration* course is recommended.

Monitoring Console App

- Monitoring Console (MC)
  - Monitoring tool for Splunk Enterprise
  - Provides detailed topology and performance information
- A dedicated search head
  - For sizing: follow the reference server guidelines for search heads
  - Should not be a production search head
  - Can run on a shared instance with caveats
  - Only admins should access
  - Forward all internal indexes (including internals and summaries) to the indexing tier



For more information about the MC, go to:

http://docs.splunk.com/Documentation/Splunk/latest/DMC/DMCoverview

Acceleration Review

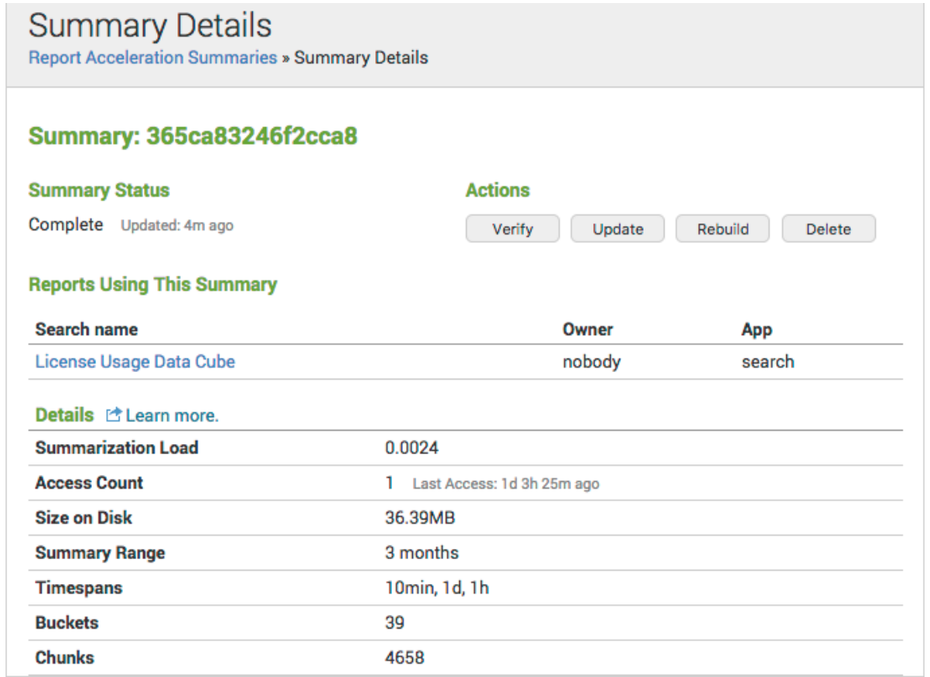
Report Acceleration	• **Accelerates individual reports** • Uses automatically-created summaries to speed completion times for qualified reports • Easier to create than summary indexes and backfills automatically • Data is stored on the indexers, alongside the related buckets • Depending on the defined time span, periodically ages out data • Always ages data when the corresponding bucket ages out • Cannot create a data cube and report on smaller subsets
Summary Indexing	• **Accelerates reports that don't qualify for report acceleration** • Uses manually created summary indexes that exist separate from main indexes • Data is stored on the search head, but can be forwarded to the indexer tier • Can persist after events have been frozen by controlling retention period or index size • Backfill is a manual (scripted) process
Data Model Acceleration	• **Accelerates all of the fields defined in a data model** • Uses automatically-created summaries to speed completion times for pivots • Takes the form of time-series index (.tsidx) files, which are created on the indexer

Disk Sizing: Report Acceleration

- Report Acceleration Summaries
  - Exist on the indexer tier, in parallel with the buckets that contain the events
- By default, accelerations can use an unlimited amount of disk space
- Location and maximum size can be set in `indexes.conf`
  - Default location is `$SPLUNK_HOME/var/lib/splunk/`*`indexName`*`/summary`
  - Location is specified with `summaryHomePath`
- Can be examined, managed and deleted from **Settings > Report Acceleration Summaries**

Note

By default, the user and power user roles have the *accelerate_search* capability to accelerate reports.



Summary Indexing Review

What is summary indexing?

- Efficiently report on large volumes of data
- Spread the cost of a computationally expensive report over time
  - Run reports over long time ranges for large datasets more efficiently
- Retain summarized data after original data sources have been frozen

How does it work?

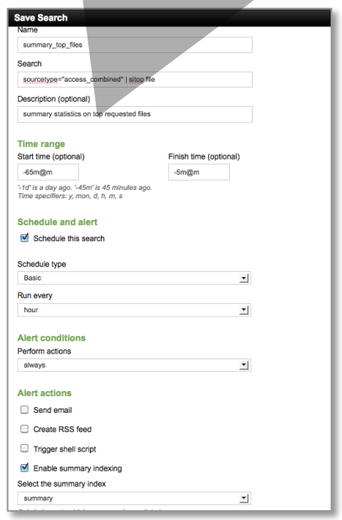
- Data is collected and summarized
  - A populating search is scheduled to run periodically
  - The populating search stores its results in a summary index
- Reports are run against the summary index
  - Searches run faster over the smaller summary index
  - The summary index may be used to report when data ages out of the original index

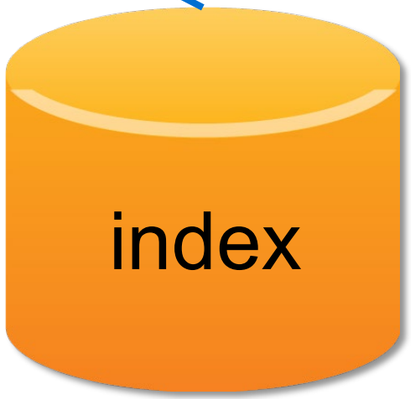
Summary Indexing Flow

purchasedProducts search Populating Search

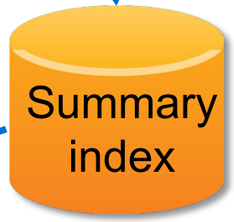
```
sourcetype=access_combined | sistats count by product_name
```

Run every hour, time range 1 hour

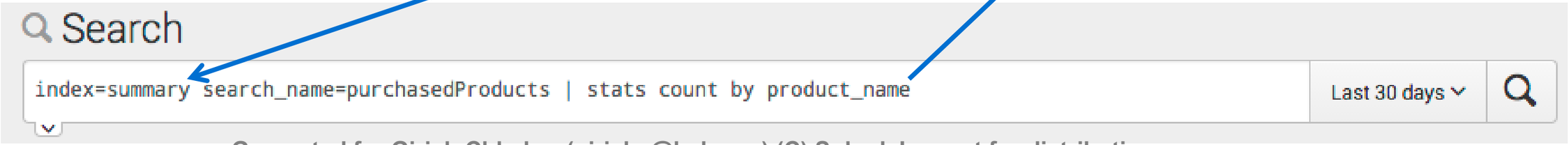




Populating search results are formatted and stored as statistics



	product_name	psrsvd_gc	psrsvd_v
1	Birthday Bouquet	2139	1
2	Cake Serving Set	2185	1
3	Chocolate Dreams Confections	2174	1
4	Day Spa Certificate	2210	1
5	Dozen Red Roses	2096	1
6	Greetings Fruit Basket	2087	1
7	Mixed Rose Bouquet	2196	1
8	Sweet Dreams Bouquet	2117	1
9	Sweet Splendor Bouquet	2143	1
10	Tulip Bouquet	2173	1



Forwarding Summary Indexes

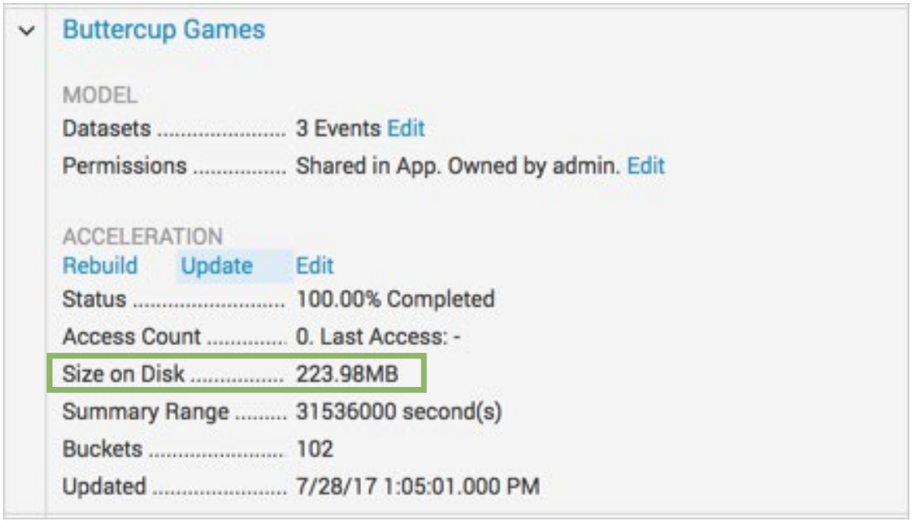
- Summary indexes are created on the search head by default
- If you use summary indexes, forward your summary indexes to the indexing layer
  - Configure `outputs.conf` on the search head to forward to indexers
  - This allows all members to access them
    - Otherwise, they're only available on the search head that generates them
  - This is *required* for search head clustering
- Configure the search head as a forwarder

splunk> Best Practice

http://docs.splunk.com/Documentation/Splunk/latest/DistSearch/Forwardsearchheaddata

Disk Usage: Data Model Acceleration

- Data model acceleration summaries are stored on the indexer in buckets next to raw data
- By default, accelerations can use an unlimited amount of disk space
  - Set up size-based retention for data model acceleration summaries in `indexes.conf`:
    - `maxVolumeDataSizeMB = volume size`



- To see the size on disk of a particular data model acceleration, go to **Data Models** > ***[data model name]***

http://docs.splunk.com/Documentation/Splunk/latest/Knowledge/Acceleratedatamodels

How Apps Affect Infrastructure Sizing

- Review the documentation for each app
  - Identify any impact on the Splunk deployment
- For example
  - Splunk App for Enterprise Security places a heavy load on search heads
  - Splunk IT Service Intelligence requires a dedicated search head or search head cluster

Note

Refer to Appendix B for detailed information about Premium App resource requirements

Staging & Testing Environments

- Part of your Splunk deployment should include a separate "sandbox" or testing / staging environment
  - It should have the same version of Splunk as production
- Sizing the testing environment depends on the nature of your tests

splunk> Best Practice

Test Inputs	A standalone indexer with minimal performance and capacity
Test Configurations	A minimum set of components (one Search Head, one Indexer, one Deployment Server, …)
Test Performance	An accurate duplication of the production environment

Authentication / Authorization

- **LDAP/AD** for authentication and group management

  http://docs.splunk.com/Documentation/Splunk/latest/Security/SetupuserauthenticationwithLDAP

- **SSO** authentication can be leveraged by Splunk using the following methods:
  - SAML

    http://docs.splunk.com/Documentation/Splunk/latest/Security/HowSAMLSSOworks
  - ProxySSO

    http://docs.splunk.com/Documentation/Splunk/latest/Security/AboutProxySSO
  - Reverse proxy

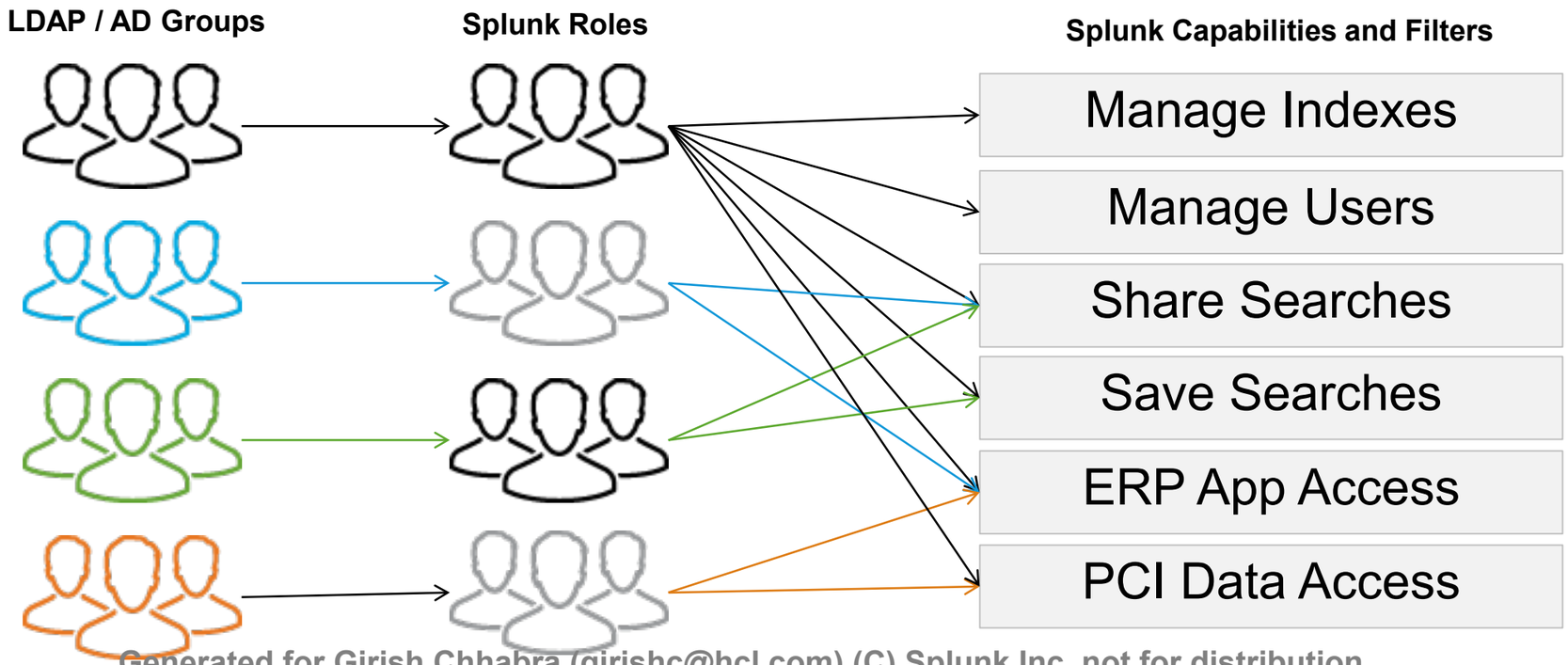
    http://docs.splunk.com/Documentation/Splunk/latest/Security/HowSplunkSSOworks

- In scripted authentication, a user-generated Python script serves as the middleman between the Splunk server and an external authentication system such as PAM or RADIUS

http://docs.splunk.com/Documentation/Splunk/latest/Security/ConfigureSplunkToUsePAMOrRADIUSAuthentication

Access Control

- Integrate authentication / access control with LDAP and Active Directory
- Map LDAP and AD groups to flexible Splunk capability or data-access roles
  - Define any search as a filter



Security, Privacy, Integrity Measures

- HTTPS transport is available end-to-end
  - Create your own certificates or acquire from a valid 3rd party
  - Distributed search (enabled by default between search head and indexer)
  - Forwarder to indexer over TCP
  - Web browser access to Splunk Web
- Indexer Acknowledgement
- Index Splunk's configurations and logs to track changes
- For more information about using SSL, go to:

docs.splunk.com/Documentation/Splunk/latest/Security/AboutsecuringyourSplunkconfigurationwithSSL

- For more information about event auditing, go to:

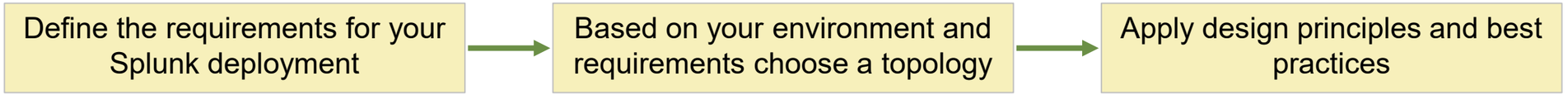
http://docs.splunk.com/Documentation/Splunk/latest/Security/Adityoursystemactivity

Security, Privacy, Integrity Measures (cont.)

- Disable Splunk Web when it is not required
  - For example, indexers in a distributed deployment do not require Splunk Web
- Harden your Splunk servers
  - Follow best practices: http://docs.splunk.com/Documentation/Splunk/latest/Security/Hardenyourserversandsecureyouroperatingsystem

Splunk Validated Architectures

- Splunk Validated Architectures (SVAs) are proven reference architectures that:
  - Are designed by Splunk Architects based on best practices
  - Are repeatable deployments
  - Offer topology options for your environment and requirements



Note

Refer to Appendix A for more information about Splunk Validated Architectures.

Module 4 Lab Exercise

Time: 10 minutes

Task:

- Size the infrastructure
  - How many indexers, search heads and other components?

Module 5: Clustering Overview

Clustering Overview

- Splunk offers two different, independent clustering capabilities
  - Indexer clustering
    - Provides availability and recovery
    - Replicates data (buckets) based on configured policies
  - Search head clustering
    - Provides improved resource scheduling and increased search accessibility
    - Replicates knowledge objects and search artifacts across a set of search heads

Note

These topics are discussed in detail in the Splunk Enterprise Cluster Administration course.

About Indexer Clusters

- **Indexer clusters** – A group of Splunk Enterprise nodes that, working in concert, provide a redundant indexing and searching capability

docs.splunk.com/Documentation/Splunk/latest/Indexer/Aboutclusters

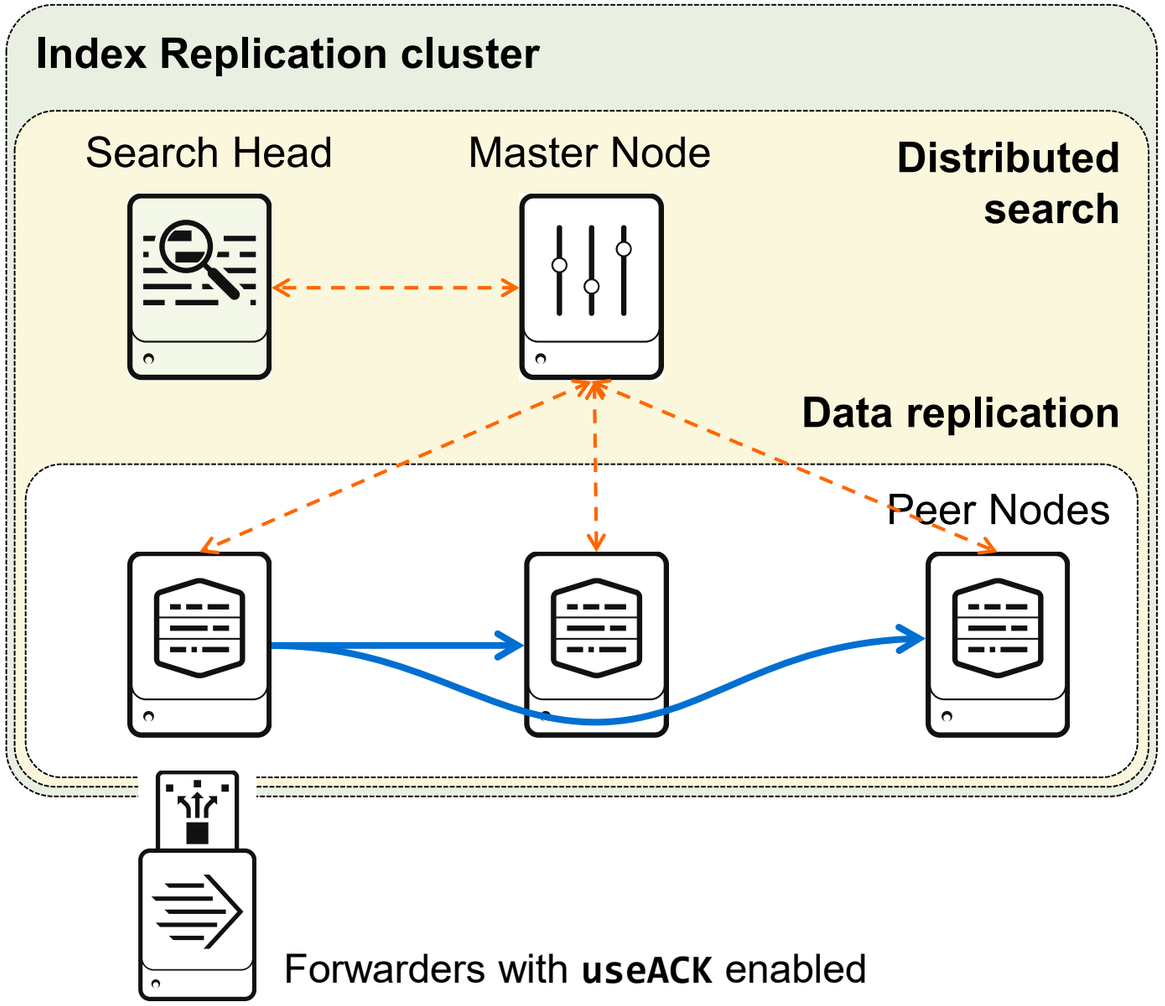
- **Multisite indexer cluster** – An indexer cluster that spans multiple sites, such as data centers
  - Each site has its own set of peer nodes and search heads
  - Each site also obeys site-specific replication and search factor rules
  - The cluster administrator defines the "sites"
    - Can represent a physical location (city, data center, rack) or something else

For more information about multisite clusters, go to:

docs.splunk.com/Documentation/Splunk/latest/Indexer/Multisiteclusters

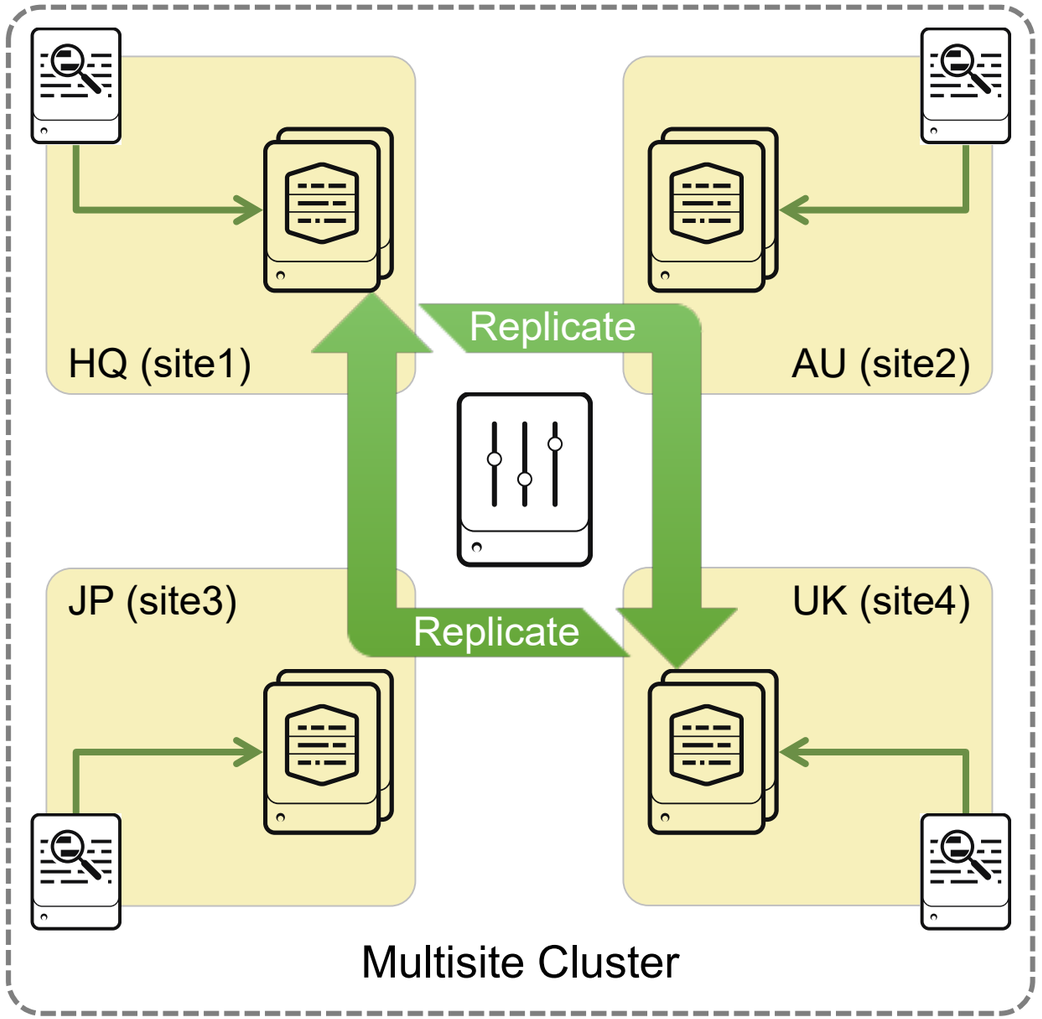
Single-site Indexer Cluster Overview

- **Master node**
  - There can only be one master
  - Controls and manages index replication
  - Distributes apps and configurations to peer nodes
- **Peer nodes**
  - Indexes data from inputs/forwarders
  - Replicates data to other peer nodes as instructed by the master
- **Search head**
  - Works the same as any Splunk search head
  - Required component of indexer cluster
- **Forwarders**
  - Send data to peer nodes



Multisite Indexer Cluster Overview

- Allows for an extra layer of data partitioning
  - Indexers are grouped by “sites”
- Multisite clusters offer two key benefits:
  1. Disaster recovery
     - Stores index copies at multiple sites (i.e. geo-location or rack)
     - Provides automatic site-failover capability
     - In case of a disaster, indexing and searching continue on the surviving sites
  2. Search affinity
     - Preferentially searches assigned site
     - Greatly reduces WAN network traffic



Master Node

- Master Node manages an indexer cluster
  - Coordinates the replicating activities of the peer nodes
  - Tells search heads where to find data
  - Manages the configuration of peer nodes
  - Orchestrates remedial activities if a peer becomes unavailable
- There can be *only one* Master Node, even in a multisite cluster
  - A stand-by Master Node should exist offline for manual Master Node failover
  - Note that the cluster will continue to operate while the Master Node is offline

Replication and Search Factors

- Peer nodes copy buckets to each other (replication)
  - The copied buckets may be complete buckets or contain only rawdata
  - The replication factors apply to the entire cluster
  - Replication must be enabled for each index
    - The default is *not* to replicate indexes
- **Replication factor (RF)**
  - Specifies how many total copies of **rawdata** the cluster should maintain
  - Sets the total failure tolerance level
- **Search factor (SF)**
  - Specifies how many copies will be ***searchable***
    - Searchable buckets have both rawdata and index files
  - Determines how quickly you can recover the search capability

Indexer Clustering Requirements

- A Master Node
  - Plus "stand-by" Master Node indexers
  - Single-site mode
    - The minimum "replication factor" peer nodes in single-site mode
      - That is, for a replication factor (RF) of 3, three peer nodes are required
    - **Best Practice**: at least RF + 1 nodes as a minimum
  - Multisite mode
    - At least 2 peer nodes per site in multisite mode

http://docs.splunk.com/Documentation/Splunk/latest/Indexer/Clustersinscaledoutdeployments

Local Storage Requirements for Clustering

- With index clustering, you must consider the replication factor (RF) and search factor (SF) to arrive at total storage requirements
  - Total rawdata disk usage = rawdata total * RF
  - Total index disk usage = index data total * SF

For more information about storage requirements, go to:
docs.splunk.com/Documentation/Splunk/latest/Indexer/Systemrequirements#Storage_considerations

Note

This does not include disk space needed to rebuild search factor if required.

Estimating Disk Usage

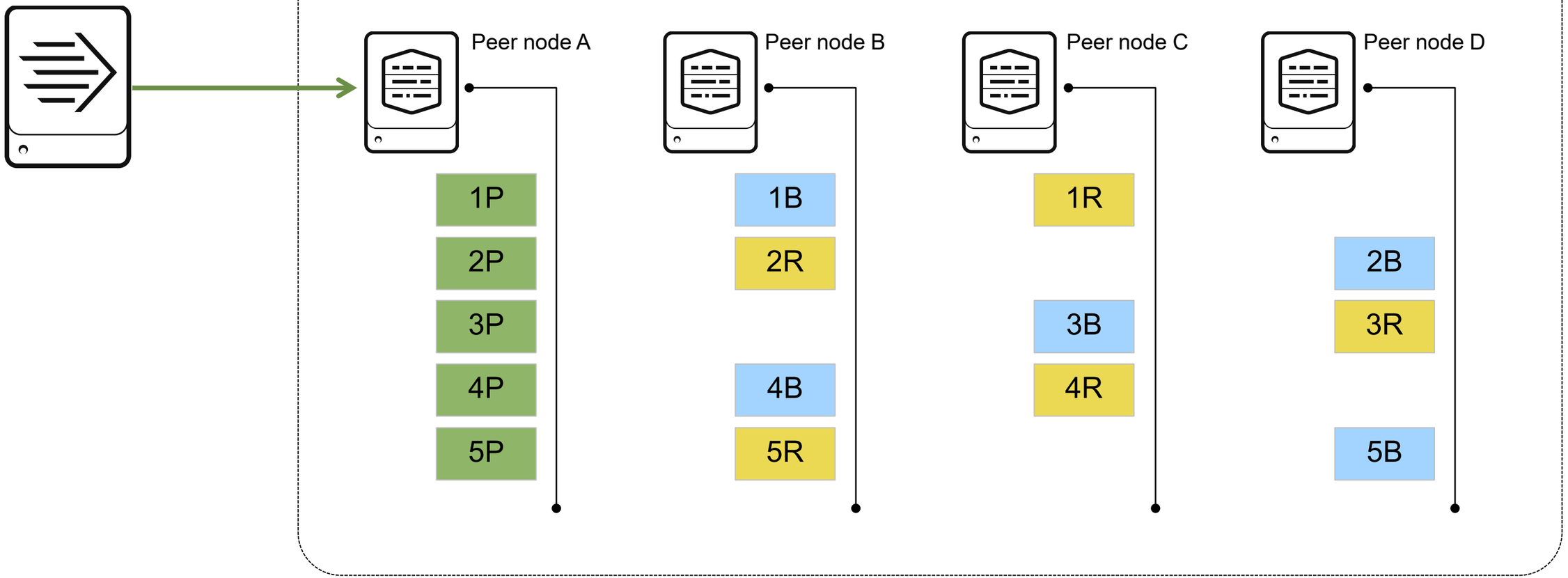
- For this example, assume
  - rawdata on disk = ~15% of daily index data
  - index files on disk = ~35% of daily index data

> **Note**
> Per day storage must still be multiplied by the # days retention!

Daily Index data = 100GB	RF=3 & SF=2 on 3 peer nodes	RF=3 & SF=2 on 6 peer nodes	RF=3 & SF=3 on 6 peer nodes
rawdata (15 GB daily)	15 * 3 = 45 GB	15 * 3 = 45 GB	15 * 3 = 45 GB
index files (35 GB daily)	35 * 2 = 70 GB	35 * 2 = 70 GB	35 * 3 = 105 GB
Total size across cluster	115 GB	115 GB	150 GB
Per Peer storage per day	115 / 3 = 38.3 GB	115 / 6 = 19 GB	150 / 6 = 25 GB

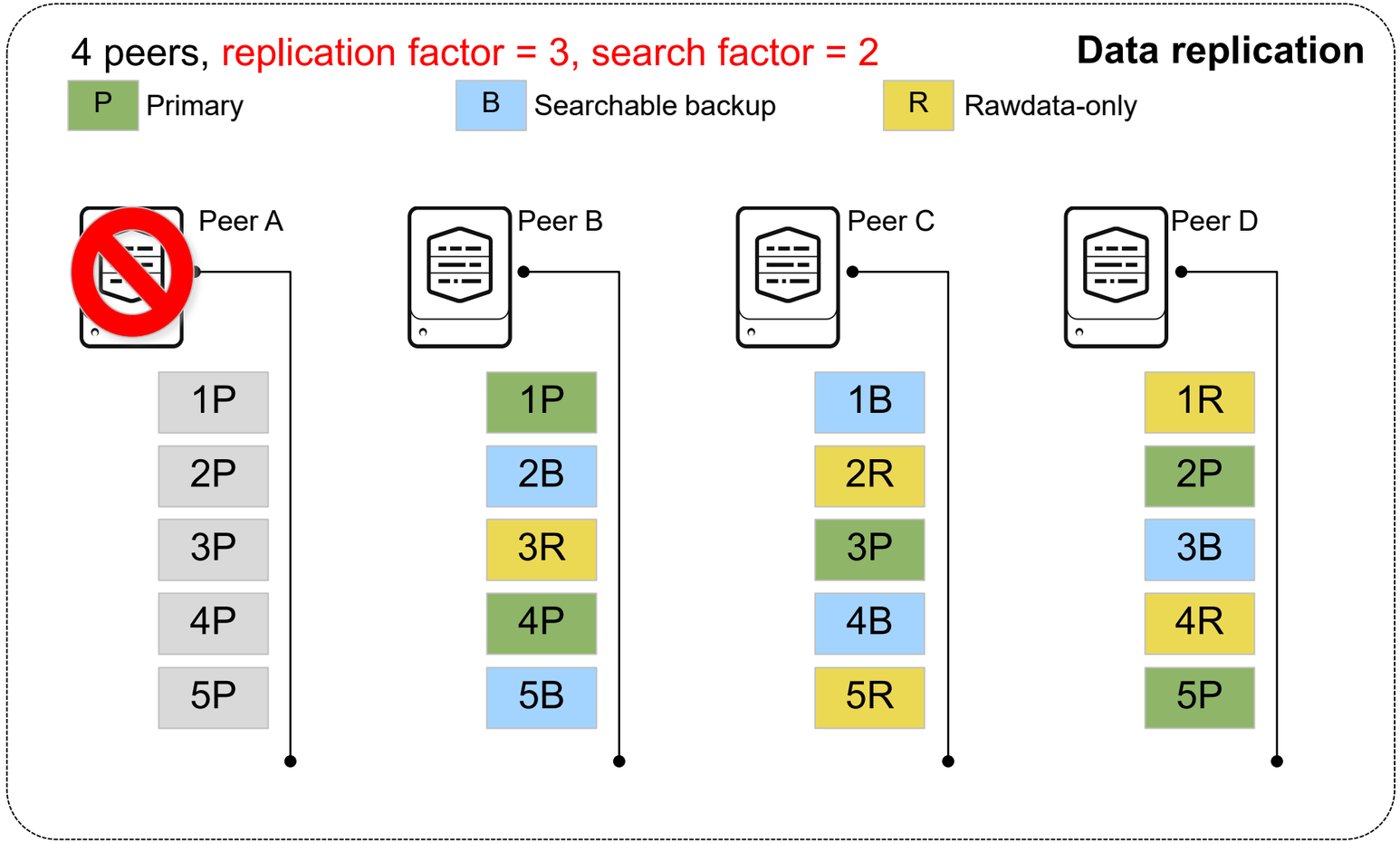
Cluster Replication - Example

Complete & Valid



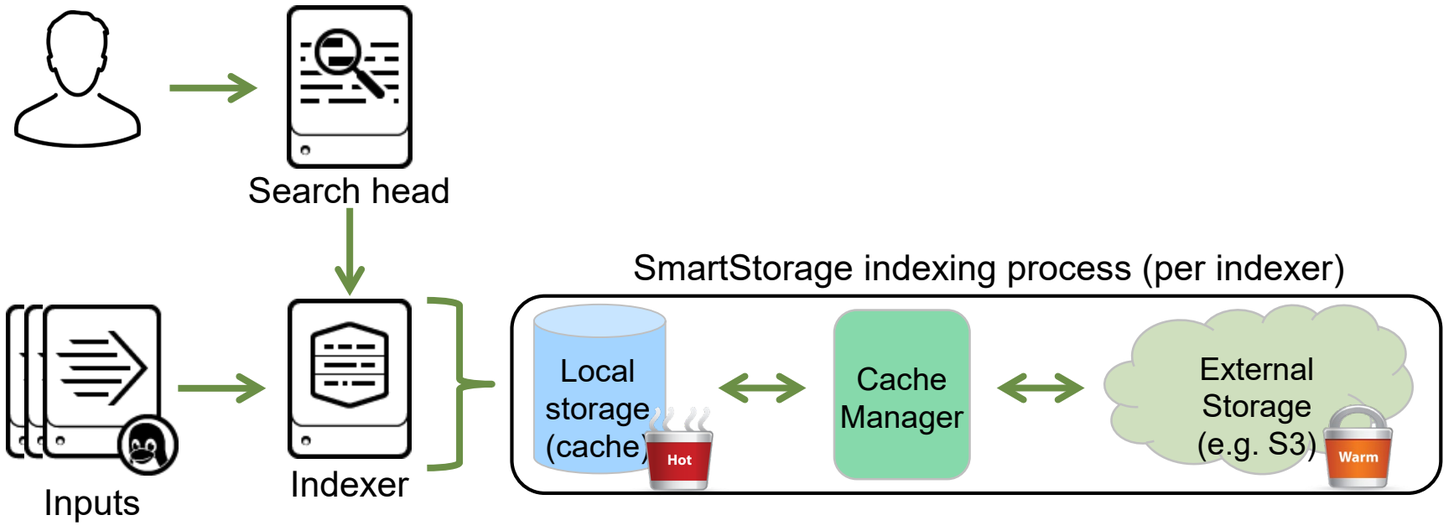
Peer Failure Impact on Disk Sizing

- When a peer is lost, the Master Node initiates additional replication across the surviving peers to return to specified RF and SF
- Consumes additional disk space
  - Plan for additional disk space for sustained outage



Remote Storage – SmartStore Overview

- Reduces local storage requirements for index clusters
- A cost-effective way to store your indexed data
- Hot buckets are still stored in local storage, but warm buckets are stored remotely and retrieved using the cache manager



SmartStore and Index Clustering

- Indexer clusters maintain replication and search factor copies of hot buckets only
- During SmartStore replication, target peer nodes also create an empty directory for each warm bucket that has metadata in their `.bucketManifest` files
  - The file contains metadata for each bucket copy that the peer node maintains
- The indexer cluster can recover all of its warm bucket data even when the number of failed nodes equals or exceeds the replication factor

For detailed information about Index clustering with SmartStore, please refer to:
http://docs.splunk.com/Documentation/Splunk/latest/Indexer/IndexerclusteroperationsandSmartStore

SmartStore Use Cases

- The cost of storing replicated copies in local storage is costly
- Leverages data fidelity guarantees provided by storage or cloud vendors
  - Eliminates the need to store local copies of warm or cold buckets
- Upgrade/bring down clusters by temporarily moving data to remote storage

SmartStore Unsupported Features

- TSIDX reduction
  - Do not set `enableTsidxReduction = true`
- Data integrity
- Disabling bloom filters
  - Do not set `createBloomFilter = false`
- Changing the location of bloom filters
  - Do not change `bloomHomePath`
- Summary replication

For more information about SmartStore, please refer to:
http://docs.splunk.com/Documentation/Splunk/latest/Indexer/AboutSmartStore

Search Head Clusters

- A **search head cluster** is a group of Splunk Enterprise search heads that serves as a resource for searching
- All search heads in a cluster must have matching hardware specs
- You can run the same searches, view the same dashboards, and access the same search results from any search head in the cluster
- To achieve this, the search heads in the cluster share
  - Configurations and apps
  - Search artifacts
  - Job scheduling

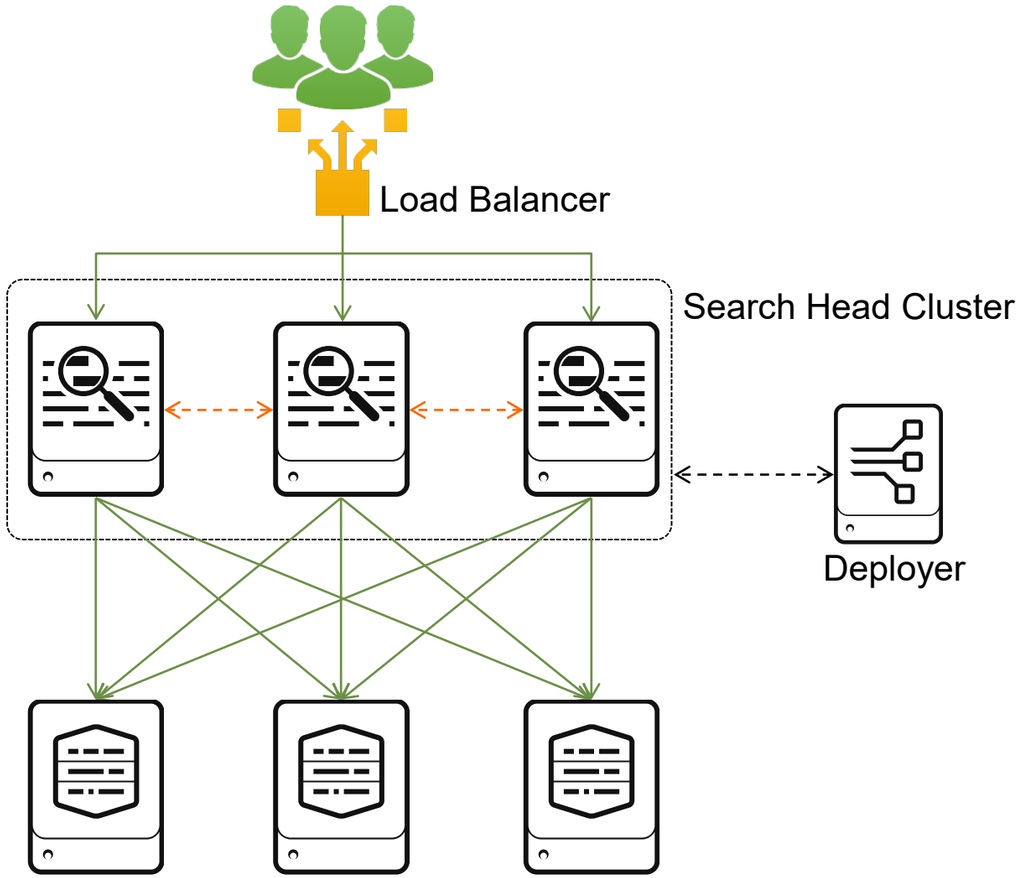
For more information about search head clustering, go to:
http://docs.splunk.com/Documentation/Splunk/latest/DistSearch/AboutSHC

Search Scaling: Do you Need a Cluster?

- To mitigate performance issues as the search load increases
  - Distribute search load
    - Optimize scheduled searches to run on non-overlapping time slots
  - Limit resource usage
    - Limit the time range of end-user searches
    - Configure user roles to limit the number of concurrent real-time searches
  - Increase the number of peer nodes (indexers)
  - Add more search heads
  - **Use a search head cluster**

Search Head Cluster Overview

- Highly available and scalable search service that groups search heads into a cluster
  - Always-on search services
  - Simple horizontal scaling
    - Add more members any time
    - Commodity hardware
    - No need for NFS
- Seamless user experience
  - Easy to on-board users and apps
- Reliable alerts
  - Search job failure aware and reschedule
- Dedicated configuration bundle management



Search Head Clustering Requirements

- Requirements
  - At least 3 search heads
  - A deployer
- Sizing guidelines – Search Heads
  - Search heads should meet the minimum reference server requirements
- Sizing guidelines – Deployer
  - No published minimums
  - Must have sufficient CPU and network resources to service requests and to push configurations

Notes for Search Head Clustering

- Summary indexes must be forwarded to the indexer tier
  - This is a general best practice, but **required** for search head clustering
  - This may increase the disk space required on the indexers
- In search head cluster, add more search heads to horizontally scale the capacity
  - Assumes that the indexer layer has sufficient search capacity
  - If capacity is insufficient, a bottleneck is created at the indexers/peer nodes
  - Remember that the rule of thumb for scaling is "add another indexer"

Final Notes on Clustering

- Planning is essential
  - What are the requirements for high availability and disaster recovery?
  - How quickly must recovery occur?
  - How many servers can be lost before service degrades or data is lost?
- You can mix clustered and non-clustered servers
  - You can have a mix of clustered and standalone indexers
  - You can have a mix of clustered and standalone search heads
    - This may be the best solution for some special-purpose or single app search heads

Module 5 Lab Exercise

Time: 10 minutes

Task:

Using the infrastructure from the previous module, update your data sizing table to use indexer clustering for data replication

Module 6: Forwarder and Deployment Best Practices

Module Objectives

- Review forwarder types
- Describe how to manage forwarder installation in an enterprise environment
- Review configuration of all Splunk components using
  - Deployment Server
  - Master Node
  - Deployer

Types of Forwarders

- Universal forwarder (UF)
  - A streamlined binary package that contains only the components needed to forward data
  - Use the universal forwarder whenever possible
    - Smaller, more efficient
    - Network bandwidth defaults to 256 KBps (configurable)

- Heavy forwarder (HF)
  - Uses the Splunk Enterprise binary with all capabilities
  - Parses data before forwarding, so only use a heavy forwarder when
    - UI is needed
    - Advanced event level routing is needed
    - Filtering more than 80% of incoming events
    - Anonymizing or masking data before forwarding to indexer
    - Predictable version of Python needed
    - Required by an App/Modular Input
      - HEC, DBX, Checkpoint OPSEC LEA

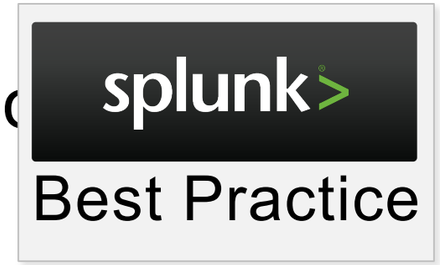
http://www.splunk.com/blog/2016/12/12/universal-or-heavy-that-is-the-question.html

Note

Parsing on a HF can consume more resources than parsing on the indexer.

Forwarding Tier Design

- Use the UF unless there are specific requirements that necessitate an HF
  - Sending cooked data through a HF to indexers impacts overall throughput performance
- Use a syslog server for syslog data
- Avoid intermediate forwarders when possible:
  - Bottlenecks can occur
  - Reduces the distribution of events across indexers
- If intermediate forwarders are required, ensure there are enc of them



Forwarding Tier Design (cont.)

- Forwarders automatically load balance over available indexers
  - Splunk 6.6 and newer - `AutoLB` is enabled by default
  - Splunk 6.5 and older - enable `forceTimebasedAutoLB` or use the following:

    Example:

    ```
    [sourcetype_name]
    EVENT_BREAKER_ENABLE=TRUE
    EVENT_BREAKER = ([\r\n]+)(\d\d\d\d-\d\d-\d\d \d\d?:\d\d:\d\d)
    ```

  - May need to increase UF thruput setting in `limits.conf` (default: 256KBps) for high velocity sources
    - This value should be based on the ratio of forwarders to indexers

      ```
      [thruput]
      maxKBps = 0
      #zero is unlimited
      #default was 256
      ```

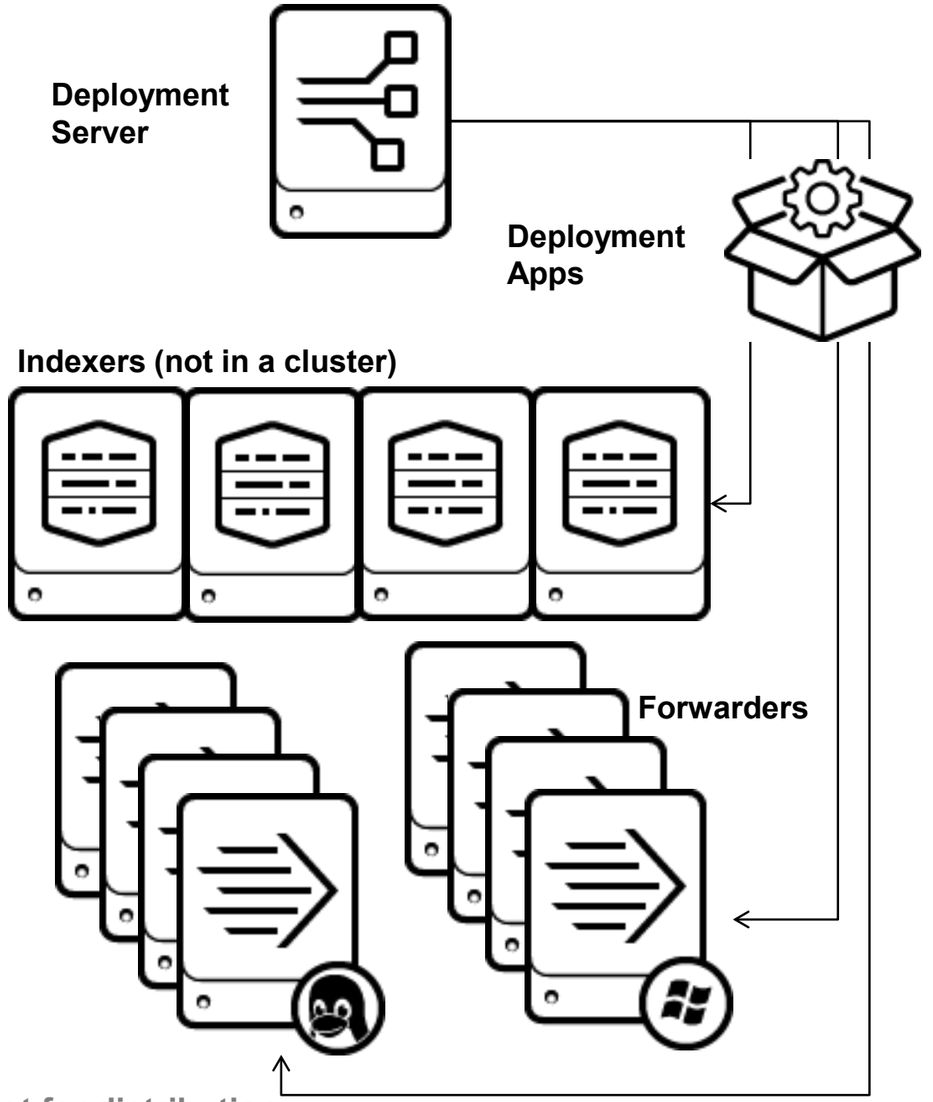
splunk> Best Practice

Deployment Management

Type of Instance	Manage Configurations with
Forwarder	Deployment Server or other configuration management tool
Stand-alone Search Head	
Stand-alone Indexer	
Search Head Cluster Member	Deployer only
Peer Node in Indexer Cluster	Master Node only

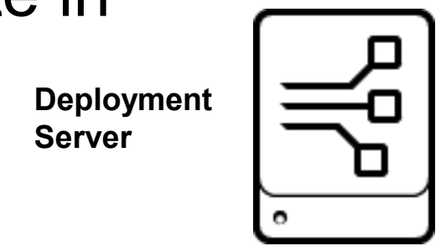
Deployment Server

- A centralized configuration manager that delivers updated content to deployment clients
  - Units of content are known as deployment apps
  - Do **NOT** store configuration in **`$SPLUNK_HOME/etc/system/local`** on clients
    - System-level configurations on clients cannot be over-ridden with Deployment Server



Deployment Server (cont.)

- One reference server safely handles approximately 2000+ polls/minute (Windows) and 10,000 + polls/minute (Linux)
- Be sensitive to the **`phoneHomeIntervalInSecs`** attribute in **`deploymentclient.conf`**
  - Default client poll is 60 seconds
  - Prioritize servers that need more frequent updates
  - In general, adjust client polling interval to scale
- For more than 50 deployment clients
  - The deployment server should be on its own server
    - Otherwise, it can share a server with another component



splunk> Best Practice

docs.splunk.com/Documentation/Splunk/latest/Updating/Calculatedeploymentserverperformance

Config Management – Deployment Server

- When using Forwarder Management or the Deployment Server (DS)
  - Build an install script/package for clients with only the files needed to contact the DS (basic installation + **`deploymentclient.conf`**)
    - Clients will get the rest of the configuration information from the DS
- Use in combination with another configuration management (CM) tool for:
  - Authentication certificates, passwords, **`log-local.cfg`**, **`deploymentclient.conf`**, upgrades of forwarder software, remote boot-start of production system
  - Use other CM tools for rare tasks, deployment server for routine tasks

Deploy Apps to Search Head Cluster

- Use the Deployer to distribute apps to SHC members
  - DO NOT stage any out-of-the-box apps in deployer (search, launcher, etc.)
  - All members of the cluster will receive the same apps
- Deployer supports both push and pull mechanisms
  - Push apps to search head cluster members
  - Polled by new or restarted search head cluster members for updates
- Knowledge objects that are created by users in Splunk Web
  - Are automatically replicated
  - Are not sent to the Deployer

Deploy Apps to Search Head Cluster (cont.)

- You CANNOT use deployment server to directly distribute apps to the peer nodes or SHC members
- You CAN use it to distribute apps to the **master-apps** and **shcluster** directories

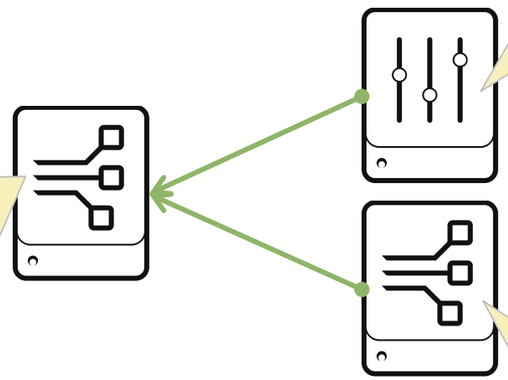
`deploymentclient.conf` on **Master Node**

```
[deployment-client]
serverRepositoryLocationPolicy = rejectAlways
repositoryLocation = SPLUNK_HOME/etc/master-apps
```

`serverclass.conf`
on **Deployment Server**

```
[serverClass:idxc_x]
stateOnClient = noop
restartSplunkd = false

[serverClass:shc_y]
stateOnClient = noop
restartSplunkd = false
```



`deploymentclient.conf` on **Deployer**

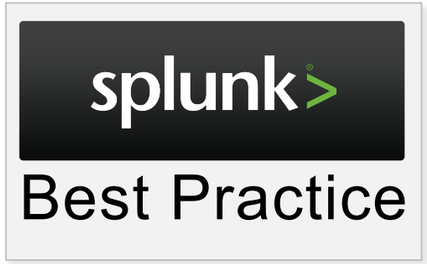
```
[deployment-client]
serverRepositoryLocationPolicy = rejectAlways
repositoryLocation = SPLUNK_HOME/etc/shcluster/apps
```

Deploy Apps to Indexer Cluster

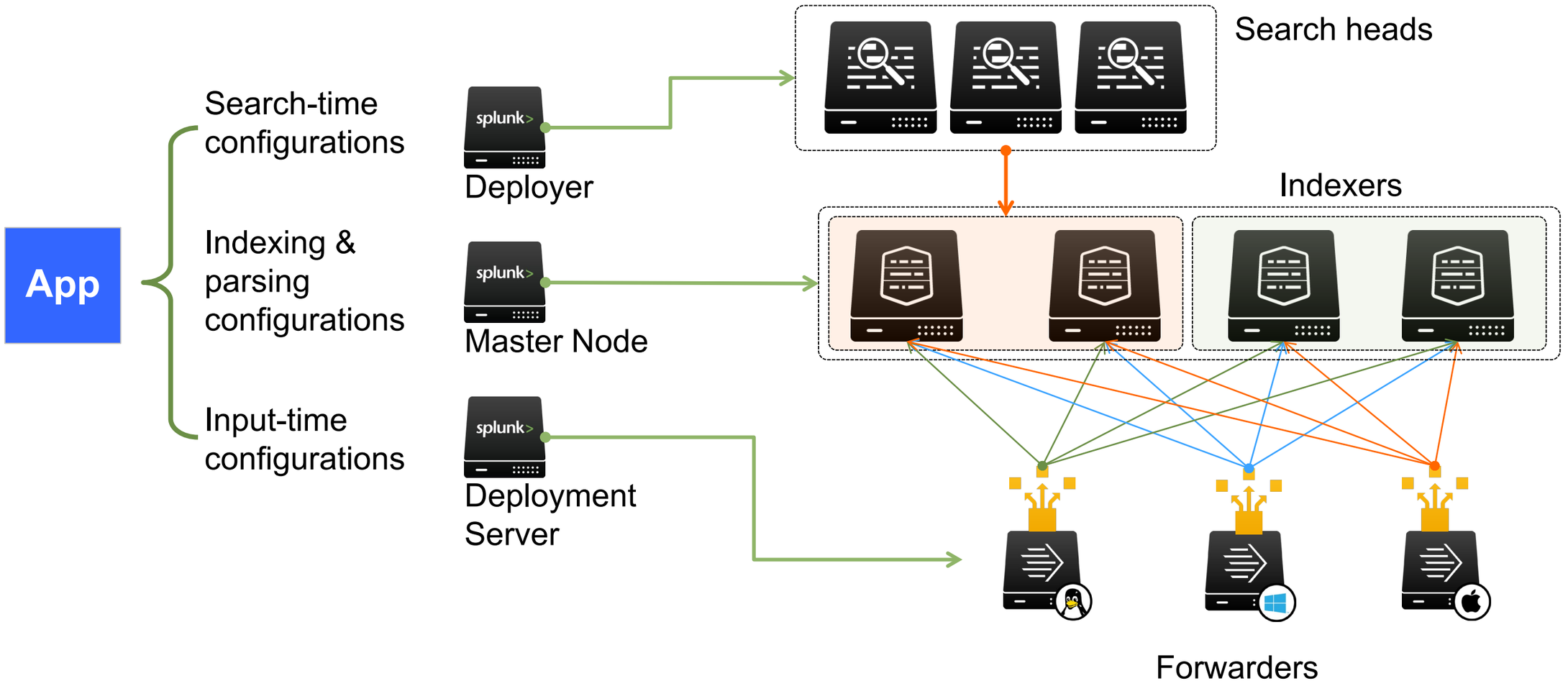
- Use the Master Node to distribute apps to the peer nodes
  - DO NOT stage any out-of-the-box apps in Master Node (search, launcher, etc.)
  - All members of the cluster will receive the same apps
  - Master Node pushes apps and rolling-restarts peer nodes if needed
  - Master Node repository: **`/etc/master-apps`**
  - Destination directory on indexers: **`/etc/slave-apps`**
- You cannot use the Deployment Server to distribute apps to the peer nodes

Deployment Apps

- Design your deployment app
  - An app is a set of deployment content (a configuration bundle)
  - An app is deployed as a unit and should be small
  - Take advantage of Splunk's configuration layering
  - Use a naming convention for the apps
  - Create classes of apps, for example
    - Input apps
    - Index apps
    - Web control apps
- Carefully design apps regardless of your configuration management tool (DS, Master Node, Puppet, etc.)



Splunk Deployment Tools



Module 6 Lab Exercise

Time: 15 minutes

Tasks:

- Create a Phase 1 topology diagram
- Analyze apps and understand deployment issues
  - Remember that some Splunkbase apps may need to be repackaged into deployment apps for your environment
- Design a test environment
  - How many servers are needed for testing?
  - How will you test the deployment, including the apps?

Module 7: Integration

Module Objectives

- Describe integration methods
- Identify common integration points

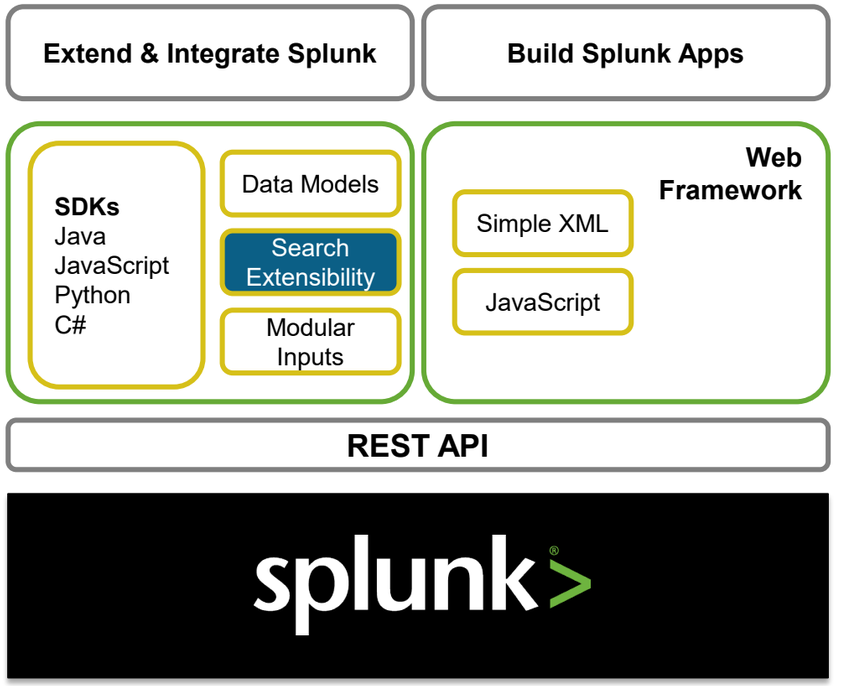
Types of Integration

- Accessing other data or applications from within Splunk
- Accessing Splunk from another application
- Re-forwarding data to other applications after it is indexed in Splunk
- Integration with the Hadoop File System (HDFS)
- Apps may also provide convenient integration points

Search Extensibility Review

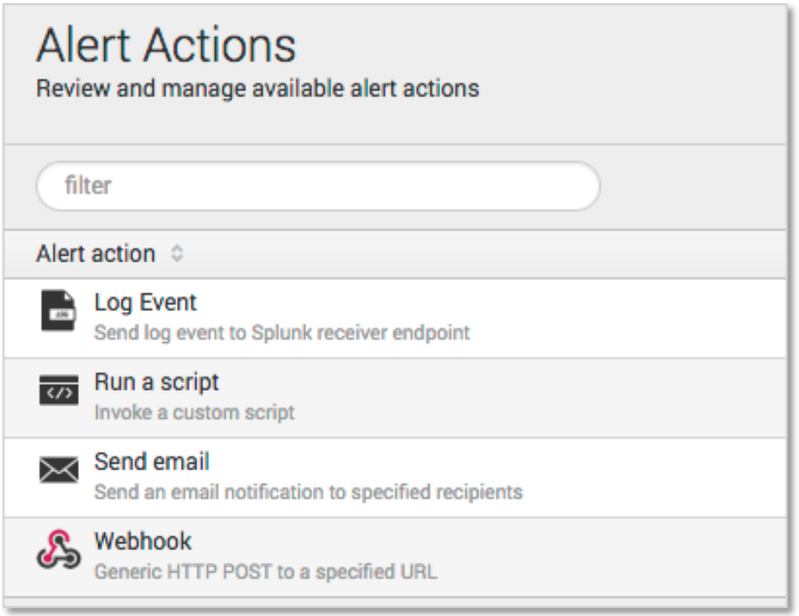
- Custom search commands
  - Write a new search command in Python 2.7

    http://dev.splunk.com/view/python-sdk/SP-CAAAEU2
- Scripted lookups
  - Programmatically script lookups in Python
- Workflow actions and custom navigation
  - Access additional resources outside Splunk with a single click



Integration Using Alert Actions

- Allows developers to extend Splunk to build, package and publish new alert actions for Splunk
- Can be used by any scheduled search
- Several modular alerts are available on Splunkbase



https://splunkbase.splunk.com/apps/#/search/alert/

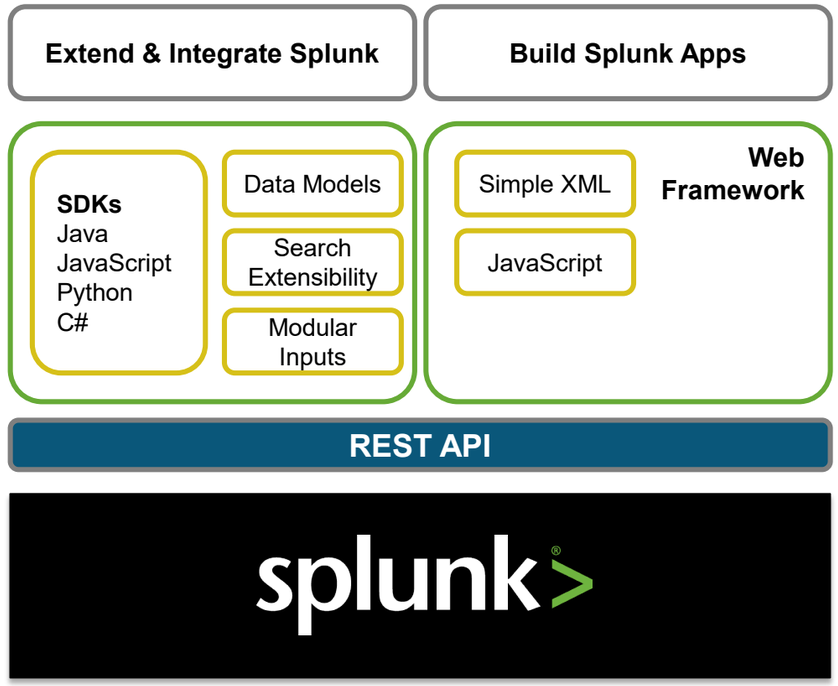
docs.splunk.com/Documentation/Splunk/latest/Alert/CreateCustomAlerts

docs.splunk.com/Documentation/Splunk/latest/AdvancedDev/ModAlertsIntro

Splunk REST API

REpresentational **S**tate **T**ransfer

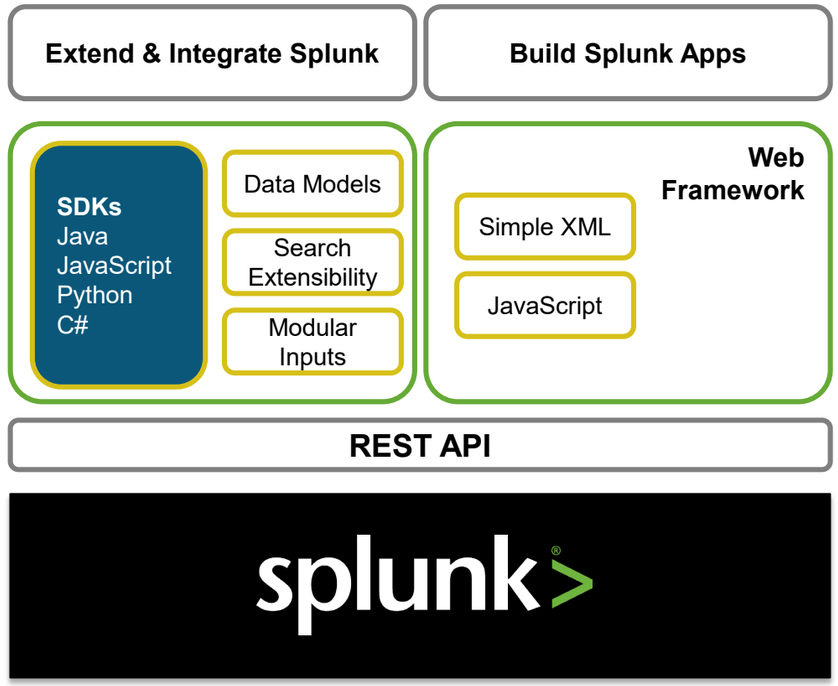
- The most basic way to programmatically access Splunk is to use the REST API via HTTP requests
- With over 200 endpoints, the REST API allows you to interact directly with a Splunk instance
- For more information go to: http://dev.splunk.com/restapi



Software Development Kits (SDKs)

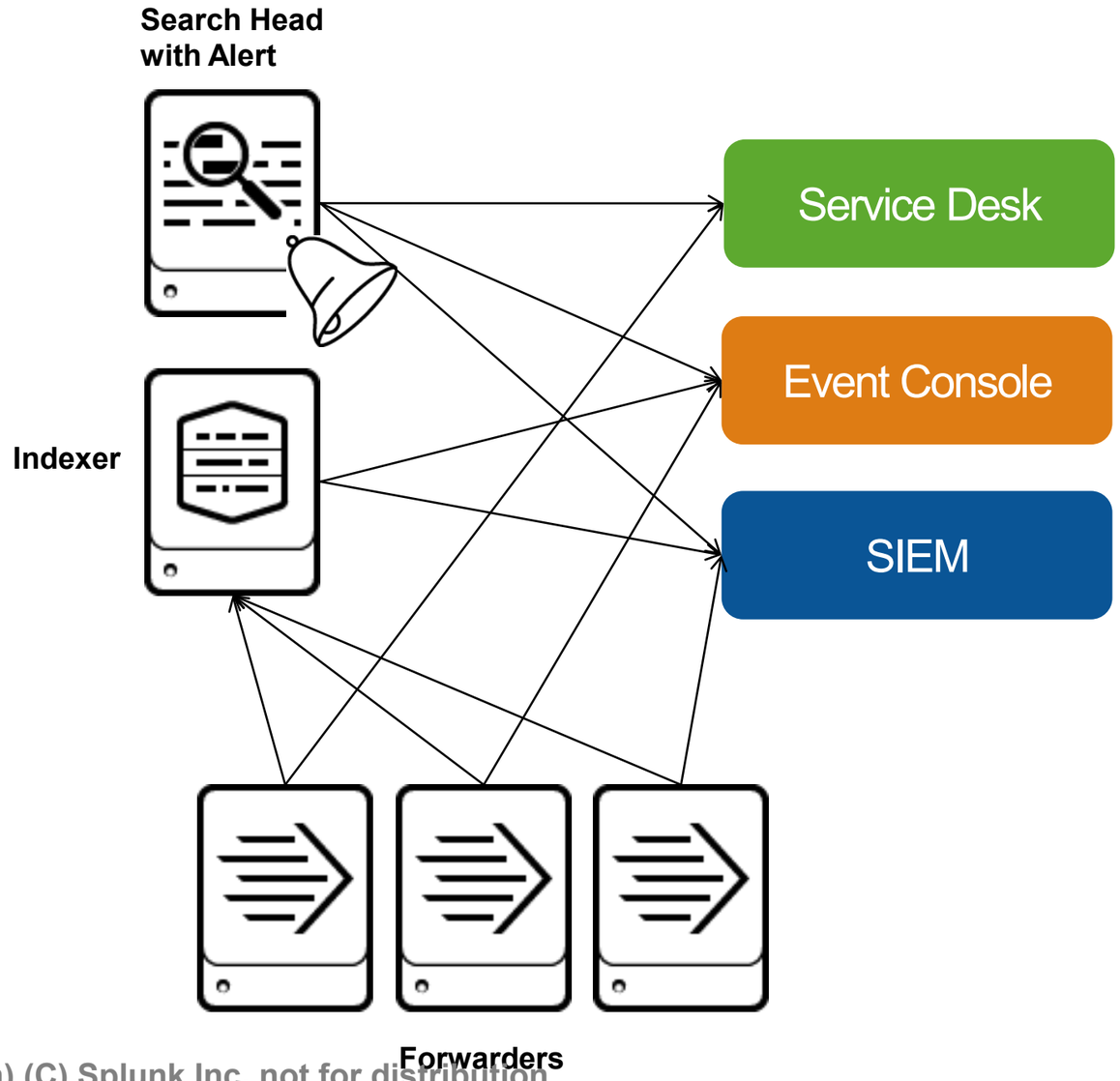
- SDKs provide language-specific libraries for accessing the Splunk REST API
  - Includes documentation, code samples, resources, and tools
  - Simplifies code development for
    - Python
    - JavaScript
    - Java
    - C#

http://dev.splunk.com/sdks



Sending Data to Other Systems

- Forward all or a subset of data via TCP to other systems
  - Forward either raw text or syslog
  - Can be done centrally via the indexer
  - Does not increase license
- Use scheduled searches to insert events into other systems
  - Leverages alert functionality
  - For example, use correlation searches and configure the alerts to open tickets or insert events into a SIEM



Splunk Analytics for Hadoop

- Add a Splunk Analytics for Hadoop license to Splunk Enterprise
  - Add-on can be downloaded from Splunkbase https://splunkbase.splunk.com/app/3311/
- Uses the MapReduce framework to HDFS when performing searches
  - Hadoop searches only work in Linux installs
- Accesses both Splunk indexes and HDFS from a single Splunk search head
- License is based on Hadoop Nodes

For more information, go to:
https://www.splunk.com/en_us/products/apps-and-add-ons/splunk-analytics-for-hadoop.html
http://docs.splunk.com/Documentation/Splunk/latest/HadoopAnalytics/MeetSplunkAnalyticsforHadoop
https://docs.splunk.com/Documentation/Splunk/latest/HadoopAnalytics/Importantinformationaboutinstallationandusers

Module 7 Lab Exercise

Time: 10 minutes

Task:

- Using the topology from Lab 6, review the forwarder configuration and create a Phase 2 topology diagram

Module 8: Performance Monitoring and Tuning

Module Objectives

- Use the Monitoring Console (MC) to track performance of your test environment before going into production
- Identify options for enhancing the performance for your production environment

Monitoring your Test Environment

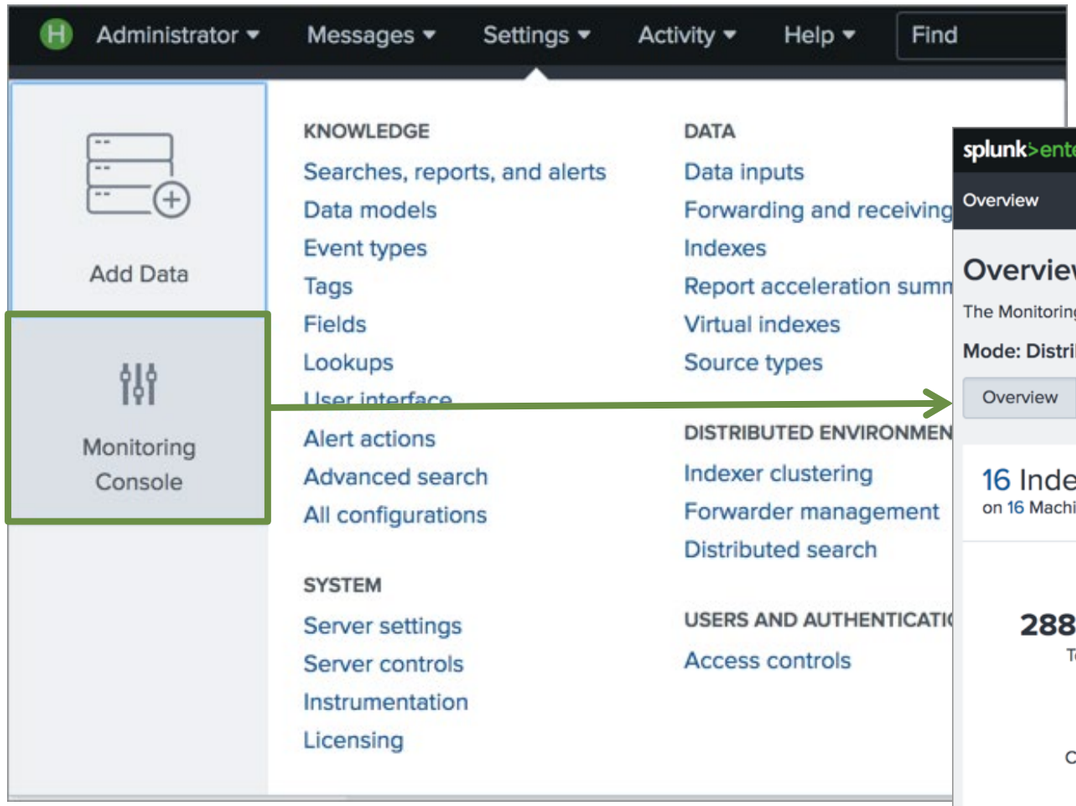
- Perform searches expected in production environment
  - Are specific searches taking a long time?
  - Are specific searches resource heavy?
- Run reports expected in production environment
  - Ad-hoc searches
  - Scheduled searches
- Test load and performance on system with max concurrent users
  - Should user roles be modified based on searches?
    - Restrict access to certain indexes

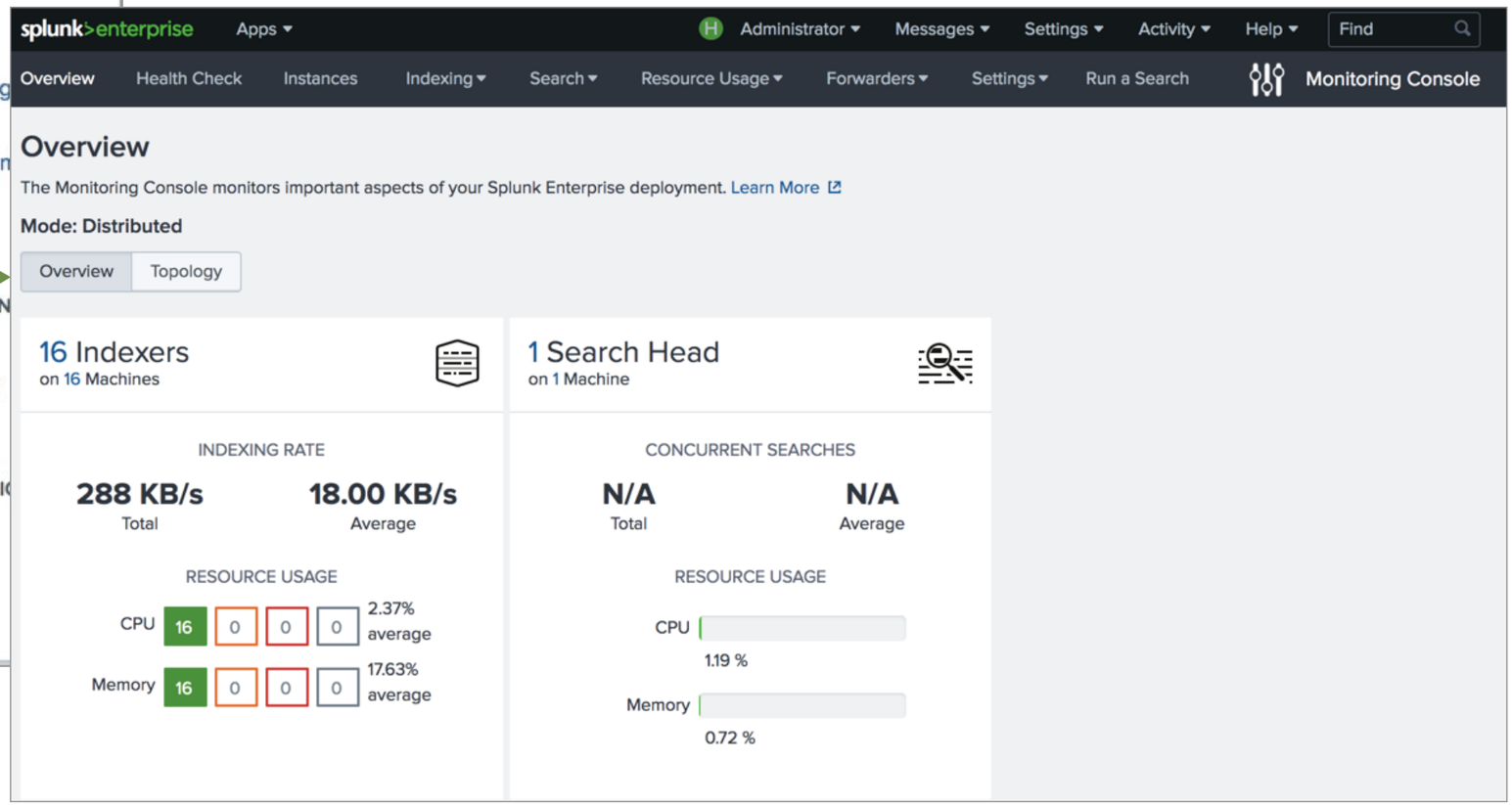
Note

Work with the Splunk Administrator to monitor your test environment.

Using the Monitoring Console

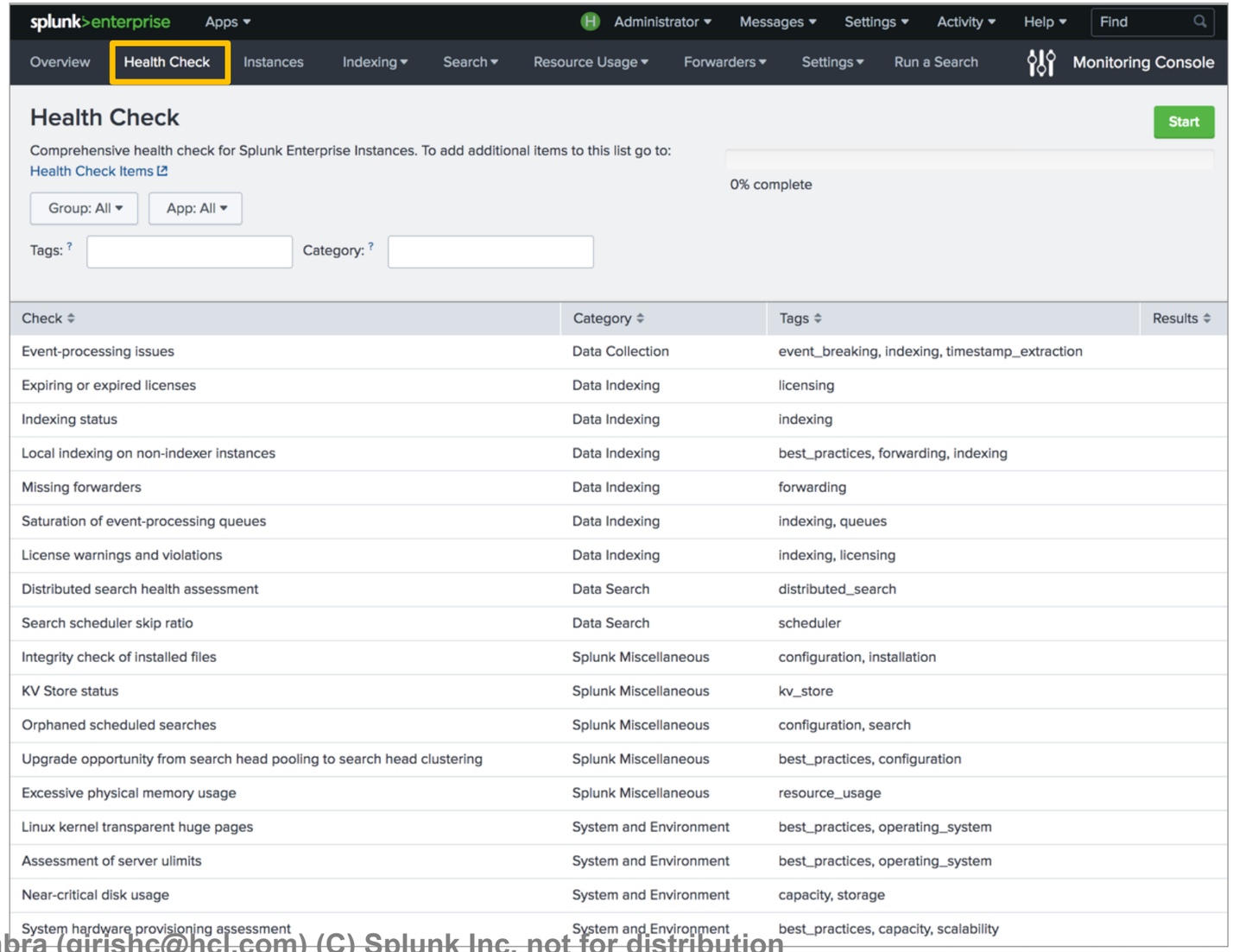
Create a base line for performance





Perform Health Checks

Use **Health Check** for a high-level summary of your system's performance

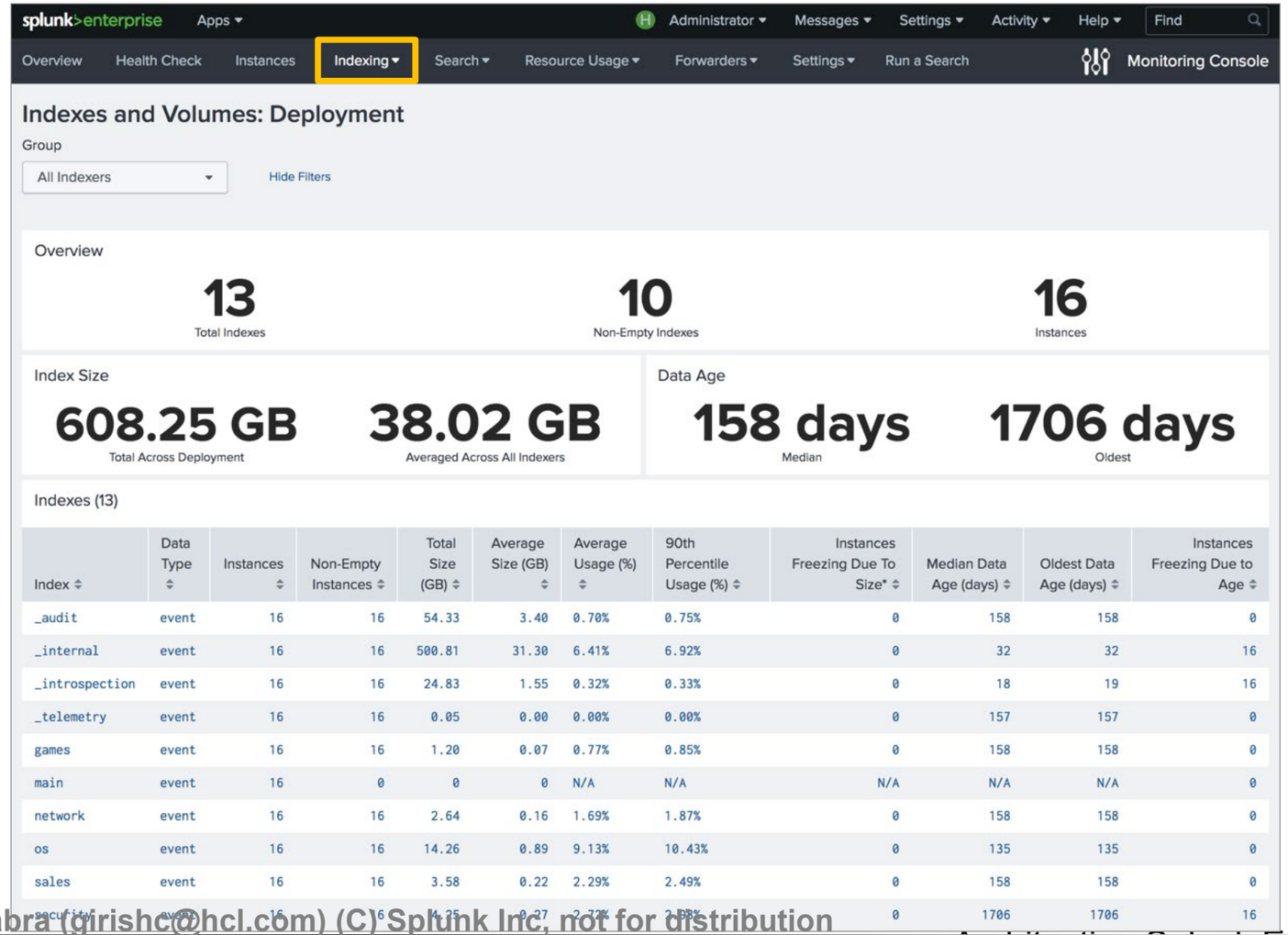


Check	Category	Tags	Results
Event-processing issues	Data Collection	event_breaking, indexing, timestamp_extraction	
Expiring or expired licenses	Data Indexing	licensing	
Indexing status	Data Indexing	indexing	
Local indexing on non-indexer instances	Data Indexing	best_practices, forwarding, indexing	
Missing forwarders	Data Indexing	forwarding	
Saturation of event-processing queues	Data Indexing	indexing, queues	
License warnings and violations	Data Indexing	indexing, licensing	
Distributed search health assessment	Data Search	distributed_search	
Search scheduler skip ratio	Data Search	scheduler	
Integrity check of installed files	Splunk Miscellaneous	configuration, installation	
KV Store status	Splunk Miscellaneous	kv_store	
Orphaned scheduled searches	Splunk Miscellaneous	configuration, search	
Upgrade opportunity from search head pooling to search head clustering	Splunk Miscellaneous	best_practices, configuration	
Excessive physical memory usage	Splunk Miscellaneous	resource_usage	
Linux kernel transparent huge pages	System and Environment	best_practices, operating_system	
Assessment of server ulimits	System and Environment	best_practices, operating_system	
Near-critical disk usage	System and Environment	capacity, storage	
System hardware provisioning assessment	System and Environment	best_practices, capacity, scalability	

Indexing Activity

Indexing provides detailed information about:

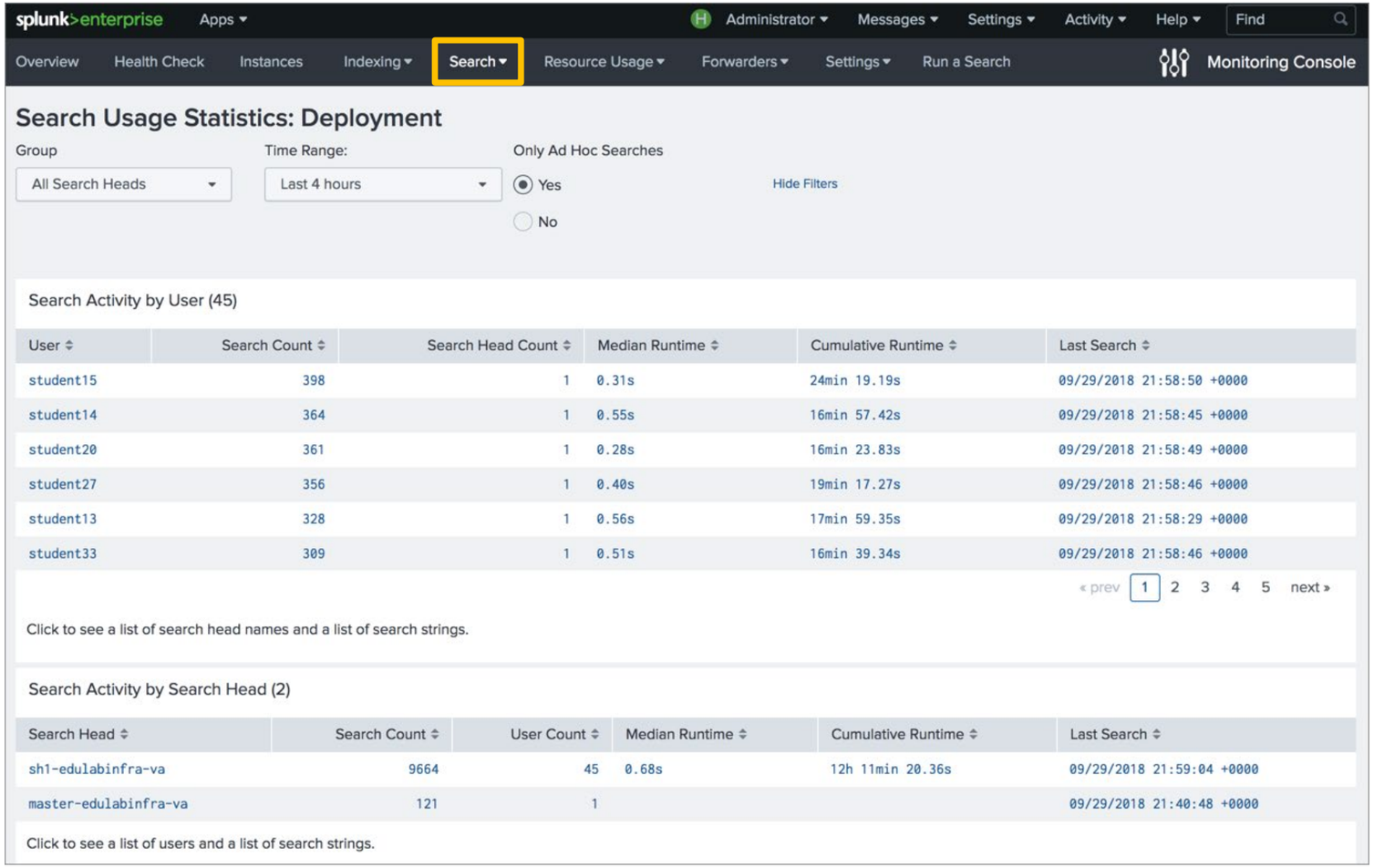
- Performance
- Indexer Clustering
- Indexes and volumes
- Inputs
- License usage



Search Statistics

Search provides stats for search activity such as:

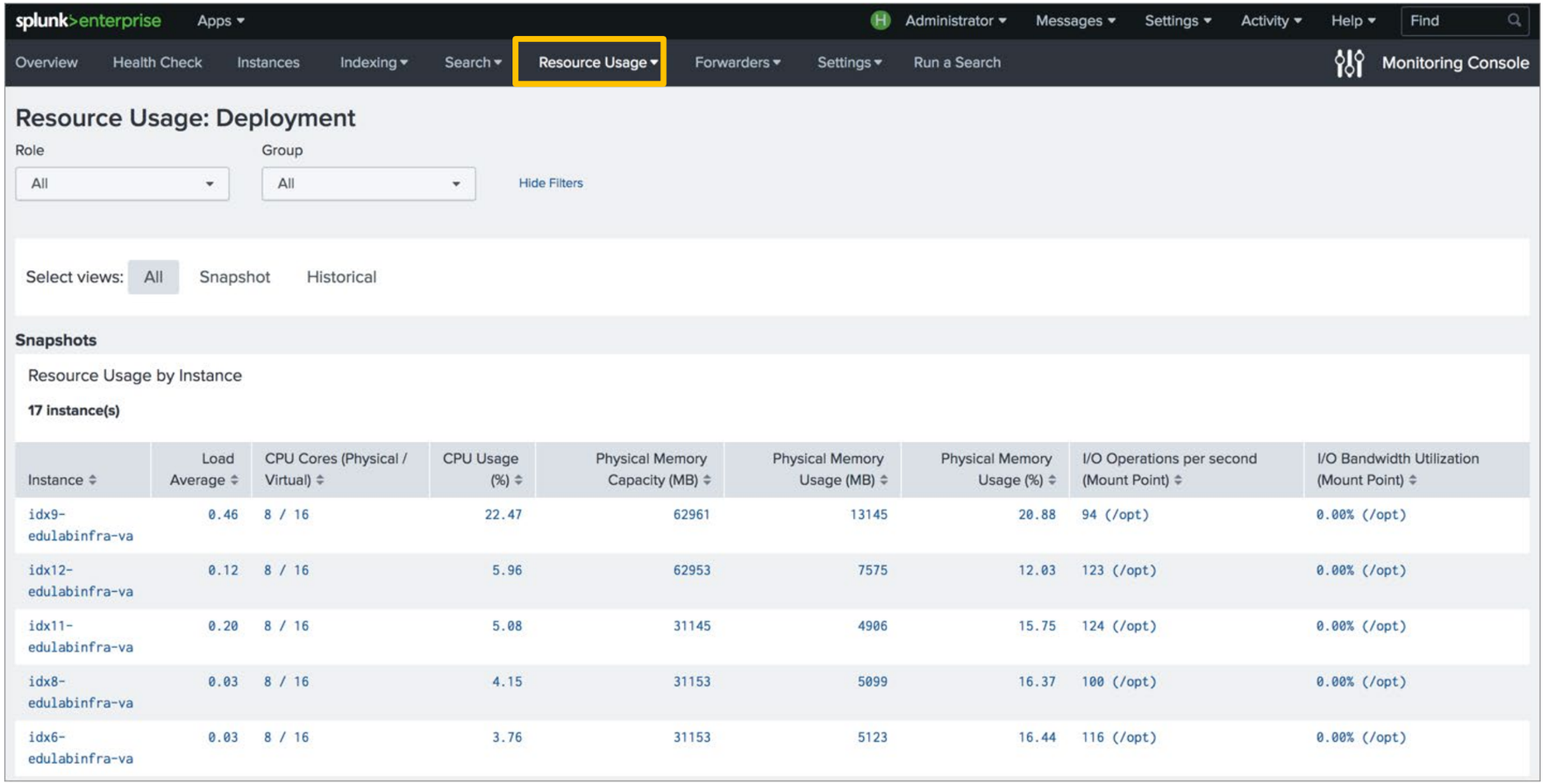
- Searches run by user
- Search duration
- Most run searches



Resource Usage

Resource Usage provides snapshots and detailed information about:

- Memory
- CPU
- Disk usage (by deployment or machine)



Instance	Load Average	CPU Cores (Physical / Virtual)	CPU Usage (%)	Physical Memory Capacity (MB)	Physical Memory Usage (MB)	Physical Memory Usage (%)	I/O Operations per second (Mount Point)	I/O Bandwidth Utilization (Mount Point)
idx9-edulabinfra-va	0.46	8 / 16	22.47	62961	13145	20.88	94 (/opt)	0.00% (/opt)
idx12-edulabinfra-va	0.12	8 / 16	5.96	62953	7575	12.03	123 (/opt)	0.00% (/opt)
idx11-edulabinfra-va	0.20	8 / 16	5.08	31145	4906	15.75	124 (/opt)	0.00% (/opt)
idx8-edulabinfra-va	0.03	8 / 16	4.15	31153	5099	16.37	100 (/opt)	0.00% (/opt)
idx6-edulabinfra-va	0.03	8 / 16	3.76	31153	5123	16.44	116 (/opt)	0.00% (/opt)

Improve Performance with `limits.conf`

- If you have unused CPU/memory resources, you can set multiple search pipelines
- Indexing real-time significantly improves performance if there are many real-time searches
  - Normally, real-time searches read the indexing queue
  - Indexing real-time causes the indexer to read the indexes on disk and collect events as they arrive in the hot buckets

Indexer

```
[search]
batch_search_max_pipeline = 2

[realtime]
indexed_realtime_use_by_default = true
indexed_realtime_disk_sync_delay = 60
indexed_realtime_default_span = 1
indexed_realtime_maximum_span = 0
```

Increase Bucket Limit Size

- While many small buckets can be better for search, having a large number of buckets presents a challenge with Index Clustering
  - A high number of buckets in an index can cause stability issues
- Prevent high-volume indexes from bucket explosion
- Increase bucket size from 750MB to 10GB
- Create fewer buckets within index cluster
- Keep the total # of buckets (source and replicas) under tested limits when using IDX clustering
  - 6.3, 6.4: 1M buckets
  - 6.5 and later: 1.5M - 10Mbuckets

$SPLUNK_HOME/etc/system/local/indexes.conf

```
maxDataSize = auto_high_volume
```

Tune `props.conf`

- Indexing time improves significantly by including the following parameters in all `props.conf` files at the indexer-level
- These parameters can also be included in `props.conf` files on search heads, however they apply at index-time and will have no effect if they are only on the search heads

`$SPLUNK_HOME/etc/system/local/props.conf`

```
    line_breaker =
(LM) should_linemerge = false
(TP) time_prefix =
(MLA)max_timestamp_lookahead =
(TF) time_format =
     truncate =
(AP) annotate_punct = false
     tz=
```

Note

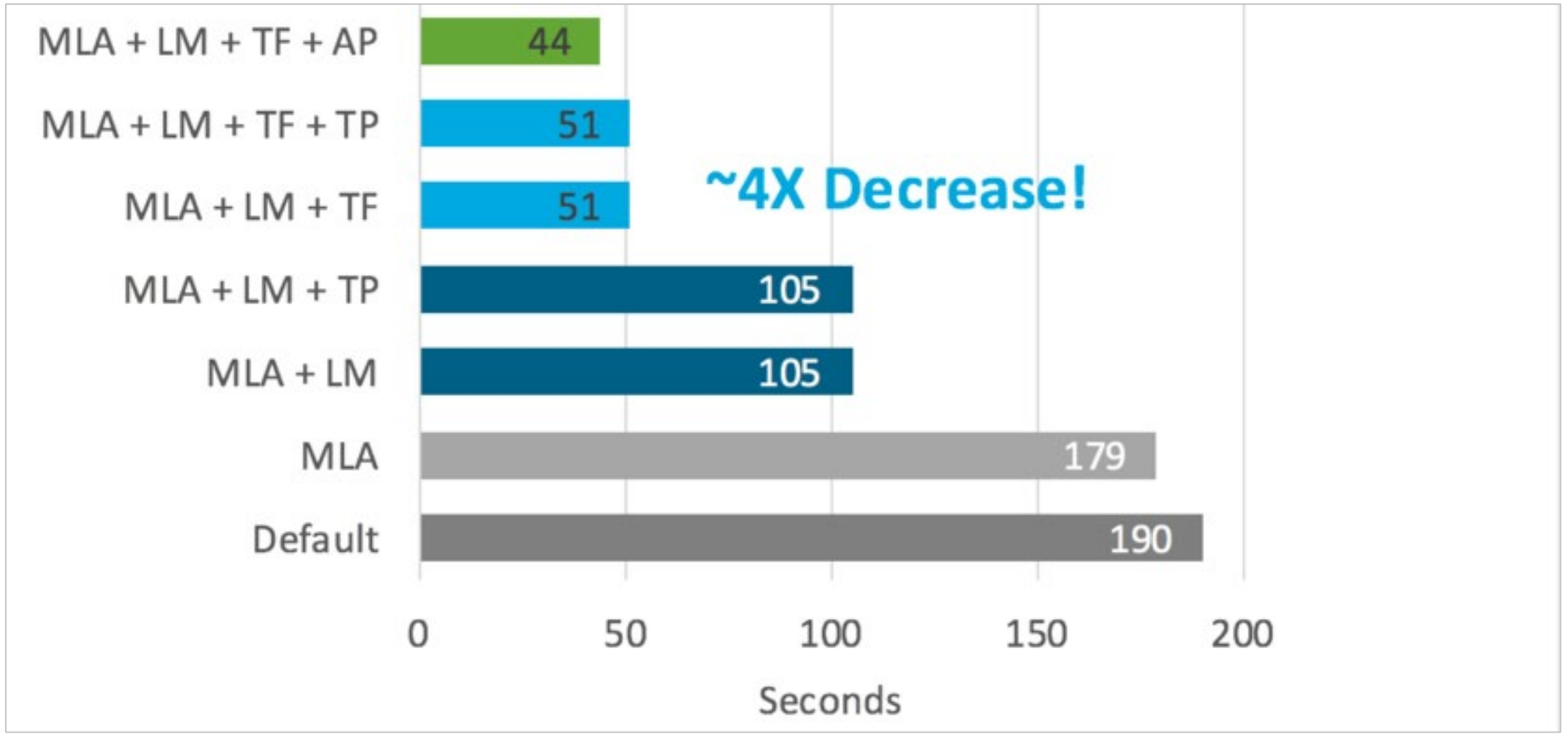
If the `punct` field is being used, DO NOT set `annotate_punct = false`

Tune `props.conf` (cont.)

- `line_breaker=` goes hand-in-hand with `should_linemerge=`
- Understand use cases before turning off `annotate_punct=`
- If this is disabled, the `punct` field will no longer be available
- `tz=` The Universal Forwarder will automatically include the time zone for source system
- If you are not using the Universal Forwarder, it is important to include `TZ` in your configs so that time is displayed properly

Tune `props.conf` (cont.)

Indexing Pipeline Test Results



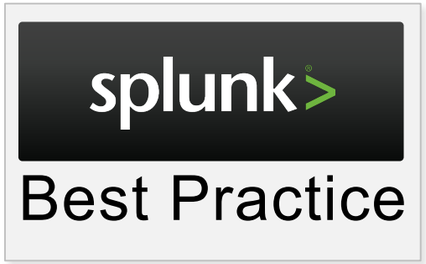
http://conf.splunk.com/session/2015/conf2015_DBitincka_Splunk_Deploying_NotesonOptimizingSplunk.pdf

Search Performance – Types of Searches

Type of search	Description	Indexer throughput	Impact
Dense	A large percentage of the data matches the search	Up to 50K matching events per second	CPU bound
Sparse	A small percentage of data matches the search	Up to 5K matching events per second	CPU bound
Super-sparse	A "needle in a haystack" search Indexer must check all buckets to find results Time consuming for large numbers of buckets	Up to 2 seconds per bucket	Primarily I/O bound
Rare	Similar to super-sparse searches, but bloom filters are able to eliminate buckets that don't include search results	10 – 15 buckets per second	Primarily I/O bound

Improving Search Performance

- Make sure disk I/O is as good as you can get
  - Increase CPU hardware only if needed
- Most search performance issues can be addressed by adding additional search peers (indexers)
- Look at resource consumption on both the indexer tier and search head tier to diagnose slow searches
- Rebalance buckets (only available in indexer clustering)



Splunk Workload Management

A mechanism in Splunk that allows you to:

- Reserve system resources for search and indexing processes
- Prioritize critical search workloads
- Prevent over-usage of system resources
- Avoid data ingestion latency due to heavy search load
- Create rules to reserve and assign system resources based on apps and roles

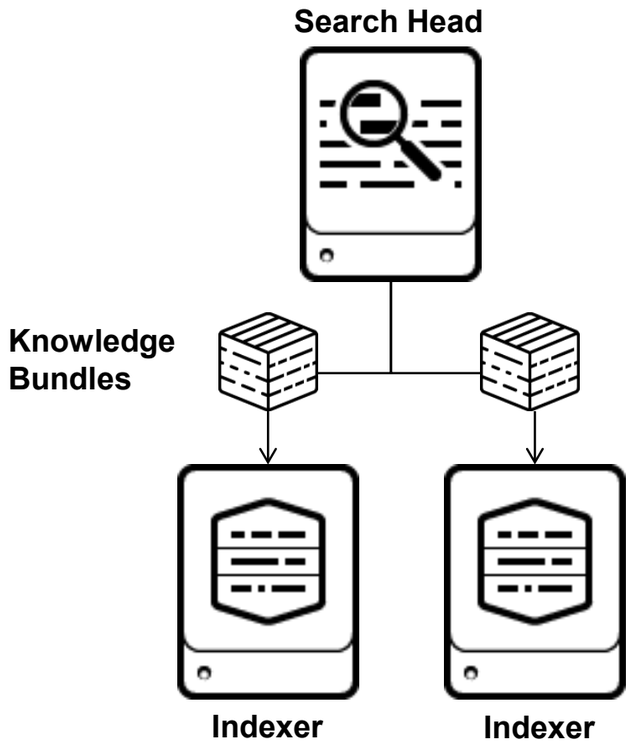
Workload Management – Use Cases

- Workaround solutions to managing search loads are ineffective when there is excessive usage by a single incoming ingest data stream or by a single end user
- Critical searches may be skipped or queued and indexing is slow resulting in data lag
- Onboarding new users/data disrupts existing ingestion and search performance

For detailed information on the configuration of Workload management, please refer to the following documentation:
http://docs.splunk.com/Documentation/Splunk/latest/Workloads/PrerequisiteLinuxconfiguration
http://docs.splunk.com/Documentation/Splunk/latest/Workloads/Configureworkloadmanagement

Knowledge Bundles

- When initiating a distributed search, the search head replicates its knowledge objects (KOs) in the form of a knowledge bundle to its search peers
  - Therefore, indexers may receive nearly the entire contents of all the search head's apps
- If an app contains large files that do not need to be shared with the indexers, blacklist large lookup files
- Be careful not to eliminate needed KOs



http://docs.splunk.com/Documentation/Splunk/latest/DistSearch/Limittheknowledgebundlesize

Module 8 Lab Exercise

Time: 15 minutes

Task:

- Use the MC to determine the following:
  - The resource usage for the test environment
  - The index performance for the test environment
  - The search performance for the test environment
  - Based on your findings, what changes should be made to maximize performance?

Module 9:
Use Cases

Module Objectives

- Review different use cases
- Answer questions and discuss final architecture solution

Energy Service Provider

Use Case

Energy Service provider needs to detect potential breaches and increase the scalability of their infrastructure. The Ops and SCO teams need access to the data.

Assets

- Cisco Routers
- Cisco Firewalls
- IBM iSeries
- Windows Server Logs
- Cisco Switches
- MSSQL/Oracle Servers
- F5 Load Balancers
- App Servers
- Syslog Forwarders

Energy Service Provider (cont.)

Data Centers

1

Data Indexed Daily

- 100GB

Data Retention Period

- Hot/Warm 30 – 90 Days
- Cold - 12 months

Data Inventory

The customer decided on ES and wants a single site index cluster with a RF=2 and a SF=2

Data Source	Max Daily Usage	Retention	Days in Hot/Warm	Access	Collection Method
Cisco general logs	15	60	30	IT Ops/Sec	UF on Syslog server
Cisco SEC logs	2	60	30	Sec	UF on Syslog server
IBM iSeries	20	365	30	IT Ops/Sec	
Windows server logs	6	60	30	sec	UF on Windows servers
MSSQL /Oracle servers	15	365	60	IT Ops/Sec	UF on server
F5 load balancers	1	365	30	IT Ops/Sec	UF on Syslog server
Windows app servers	30	60	30	IT Ops/Sec	UF on Windows servers
Syslog server	1	90	30	IT Ops/Sec	UF on Syslog server
total	**90**				

Energy Service Provider - Discussion

- What is the minimum number of indexes required?
- How many indexers do you recommend?
- Would you have the master node and license master on the same Splunk instance?
- In the long term, would you look to replace the syslog servers and just listen on TCP ports for the data?
- How would data get from iSeries into Splunk?

Financial Service Provider

Use Case

This company needs to monitor the health of its servers on a global deployment level. They also need to monitor the SAP, Informix and dev-ops environment for an app development project. The customer is also interested in ES and ITSI.

Assets

- Solaris Servers (10,000)
- Windows Servers (25,000)
- Linux Servers (15,000)
- Cisco Switches and Routers (2,000)
- IBM Mainframe (1)
- Legacy Gateways (8)

Data Indexed Daily

3TB – 4TB

Financial Service Provider (cont.)

Data Centers

2 (New York, Zurich)

Data Indexed Daily

3TB – 4TB

Financial Service Provider - Discussion

- Would you treat this as one Splunk deployment or two?
- How would you look to collect data from the IBM mainframe?
- The customer ends up using 50 indexers and 5 search heads. Assuming an equal split of users/data at each site, what does your network topology look like?

Retail Company

Use Case

This department store needs to monitor its rewards program cash flow and the amount of time a credit card takes to process a sale from the bluebird banking servers. There are two dual-homed data centers connected via WAN and all 1000 stores are connected via corporate VPN.

They want to roll out their deployment in two phases:

- Start with four users to monitor the rewards program
- Add 10 more users over the next 6 months

Retail Company (cont.)

Assets

- RF Scanners
- Bluebird Banking Servers
- HP Printers

Data Centers

2

Data Indexed Daily

2.4TB

Data Retention Period

- Hot/Warm - 3 months (186 days)
- Cold - 12 months

- Windows Desktop Servers
- Windows Store Domain Servers (two at each store to collect data from devices)

Retail - Discussion

- How many indexers and search heads do you suggest for this customer?
- Would you use index or search head clustering for this deployment?
- Is a two stage or one stage approach recommended to the customer?
- The customer is on a virtualization campaign. Which Splunk instances do you recommend for virtualization?

Wrap-up Slides

Resources and Additional Study

- Splunk documentation for architects
  - Installation Manual (for hardware and capacity planning information)
  - Distributed Deployment Manual
  - Capacity Planning Manual
  - Other manuals for specific topics
- Splunk Answers

  answers.splunk.com
- Splunk's public wiki (watch dates for outdated topics)

  www.splunk.com/wiki/Deploy:Deployment_topics

  Deployment scenarios, performance info and other useful topics

Support Programs

- **Community**
  - **Answers:** answers.splunk.com
    Post specific questions and get them answered by Splunk community experts.
  - **Splunk Docs:** docs.splunk.com
    These are constantly updated. Be sure to select the version of Splunk you are using.
  - **Wiki:** wiki.splunk.com
    A community space where you can share what you know with other Splunk users.
  - **IRC Channel:** #splunk on the EFNet IRC server Many well-informed Splunk users “hang out” here.
- **Global Support**
  Support for critical issues, a dedicated resource to manage your account – 24 x 7 x 365.
  - **Phone: (855) SPLUNK-S or (855) 775-8657**
  - **Web:** http://www.splunk.com/index.php/submit_issue
- **Enterprise Support**
  Access your customer support team by phone and manage your cases online 24 x 7 (depending on support contract).

What's Next?

- Splunk Certification program

  https://www.splunk.com/en_us/training/faq-training.html

  - Splunk Core Certified User
  - Splunk Core Certified Power User
  - Splunk Enterprise Certified Admin
  - Splunk Enterprise Certified Architect
  - Splunk Certified Developer
- Program information
  - https://www.splunk.com/pdfs/training/Splunk-Certification-Handbook-v.8.31.2018.pdf
- Exam registration
  - https://www.splunk.com/pdfs/training/Exam-Registration-Tutorial.pdf
- If you have further questions, send an email to: certification@splunk.com

splunk> .conf19

.conf19
October 21-24, 2019

Splunk University
October 19-21, 2019

Las Vegas, NV
The Venetian Sands Expo

4 Days of Innovation

350 Education Sessions

20 Hours of Networking

"Hands down the most beneficial and attendee focused conference I have attended!"

– Michael Mills, Senior Consultant, Booz Allen Hamilton

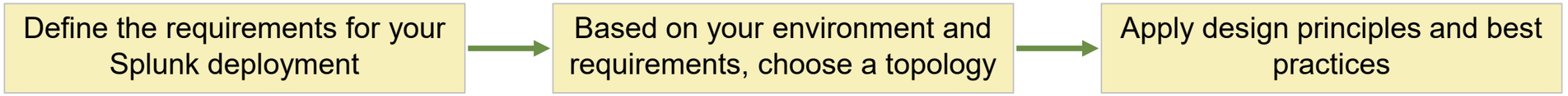
sign up for notifications @ conf.splunk.com

Appendix

Appendix A: Splunk Validated Architectures

Splunk Validated Architectures - Overview

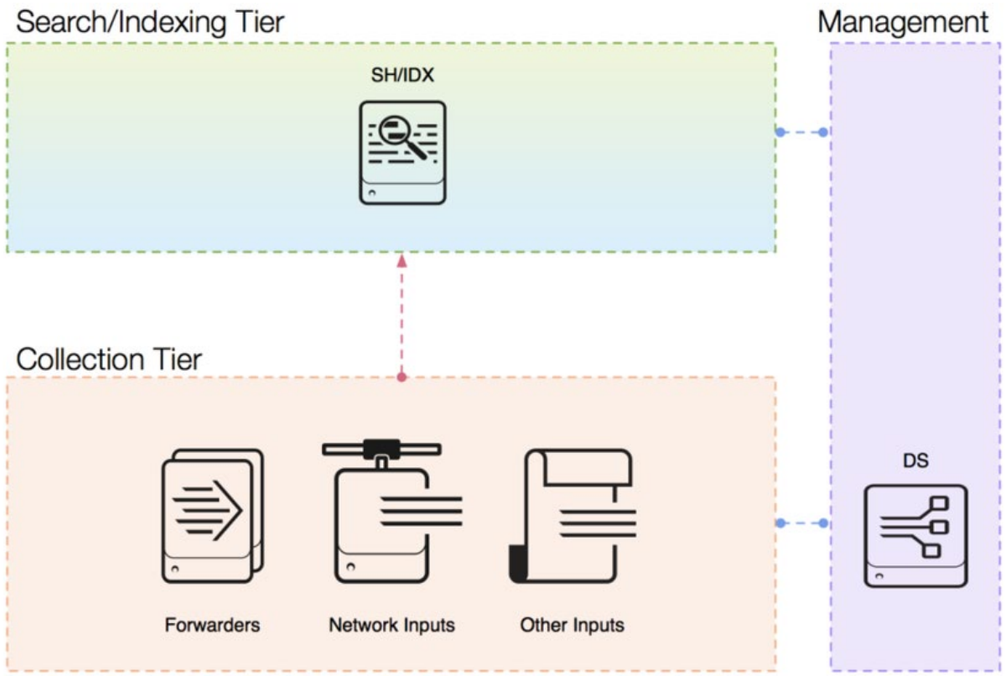
- Splunk Validated Architectures (SVAs) are proven reference architectures that:
  - Are designed by Splunk Architects based on best practices
  - Are repeatable deployments
  - Offer topology options for your environment and requirements
  - Are recommended for new Splunk customers



For detailed information about SVAs read the following white paper:
https://www.splunk.com/pdfs/white-papers/splunk-validated-architectures.pdf

SVA Examples – S1

Single server deployment



Indexing Tier Categories

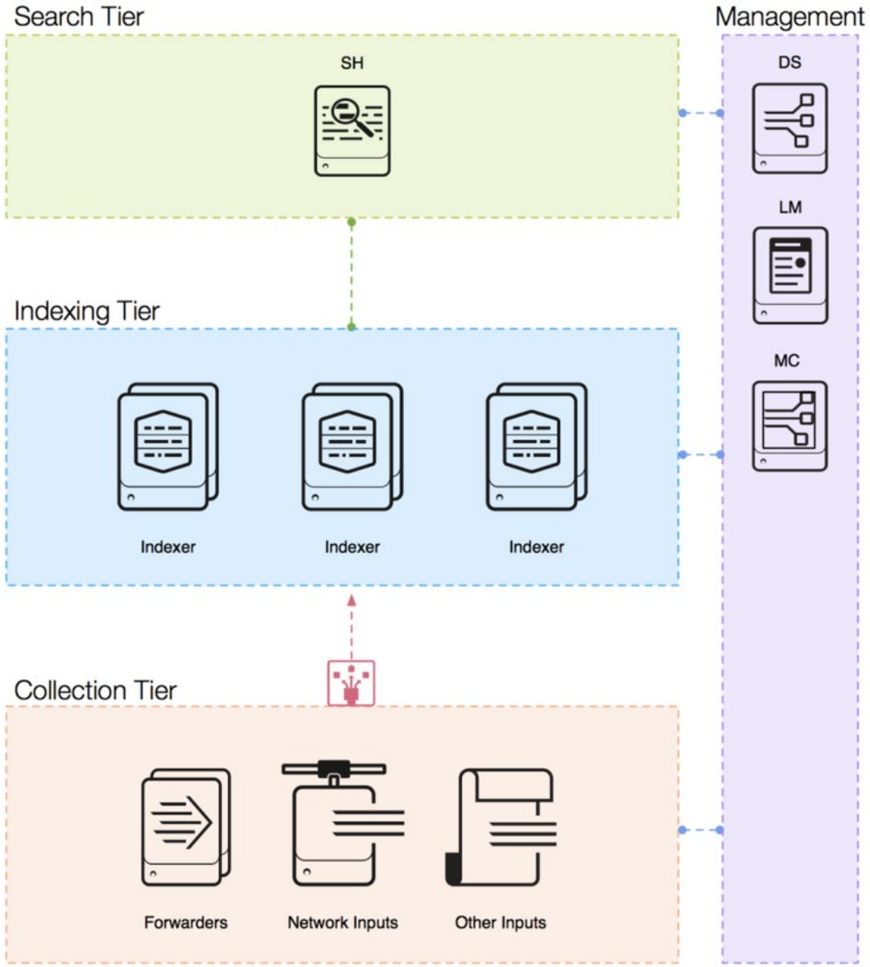
Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

SVA Examples - D1/D11

Distributed Clustered Deployment – Single Site



Indexing Tier Categories

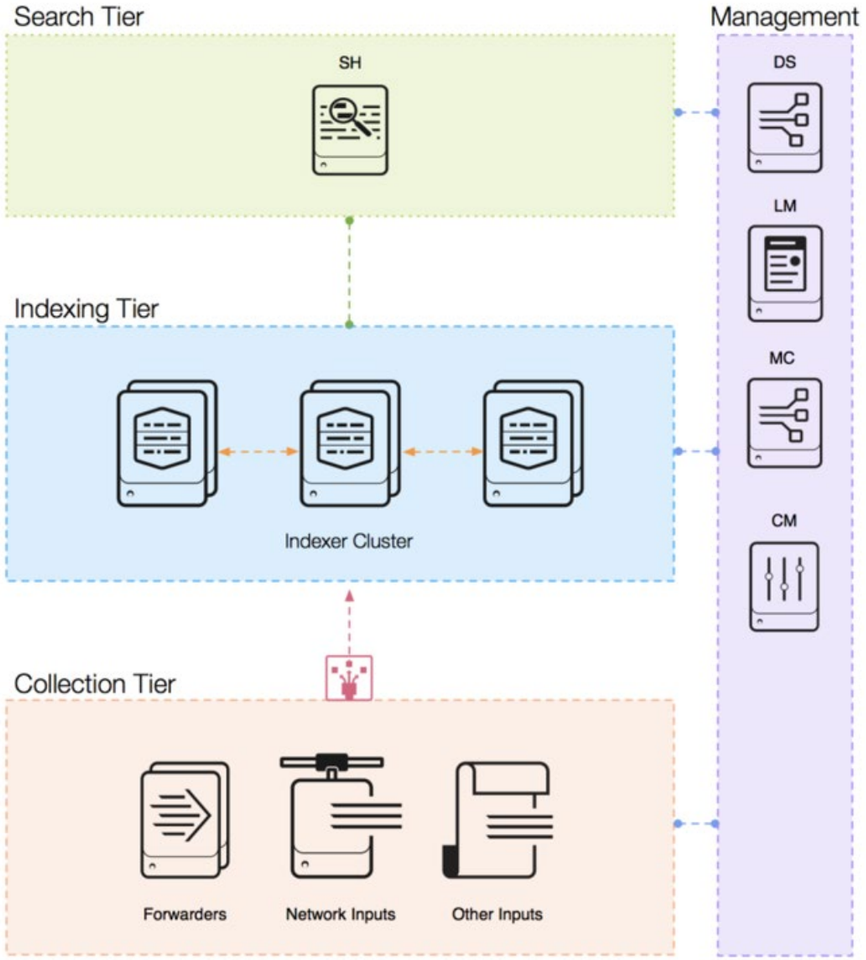
Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

SVA Examples - C1/C11

Distributed Clustered Deployment – Single Site



Indexing Tier Categories

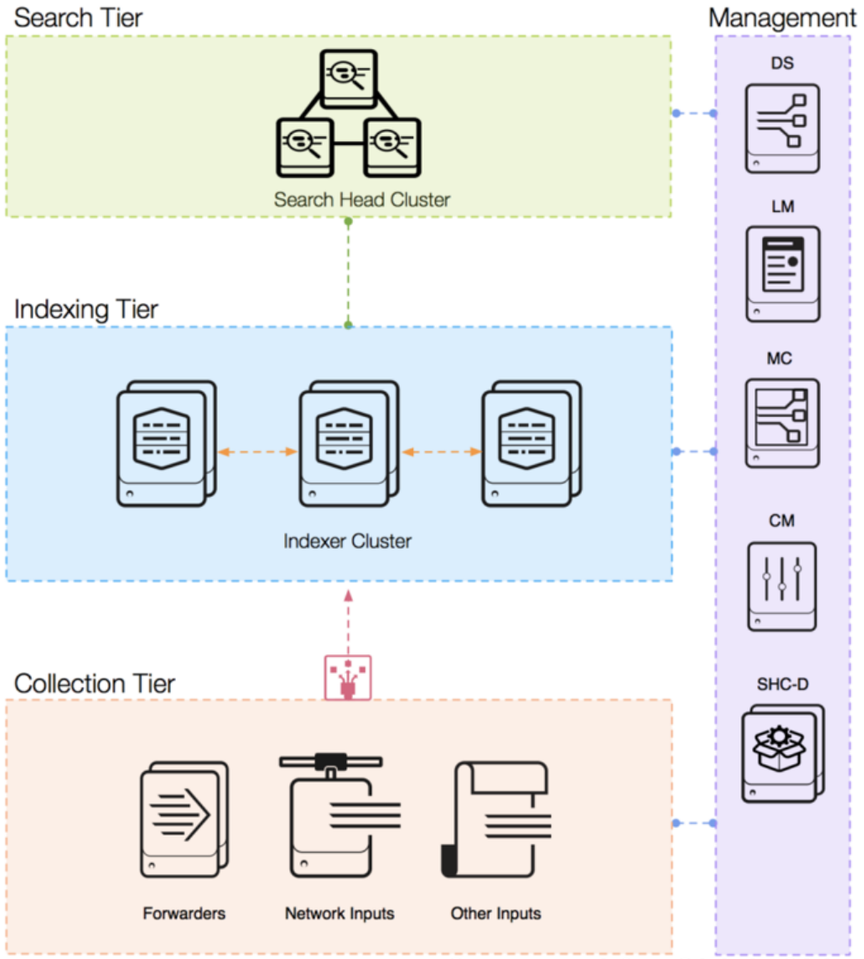
Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

SVA Examples – C3/C13

Distributed Clustered Deployment + SHC – Single Site



Indexing Tier Categories

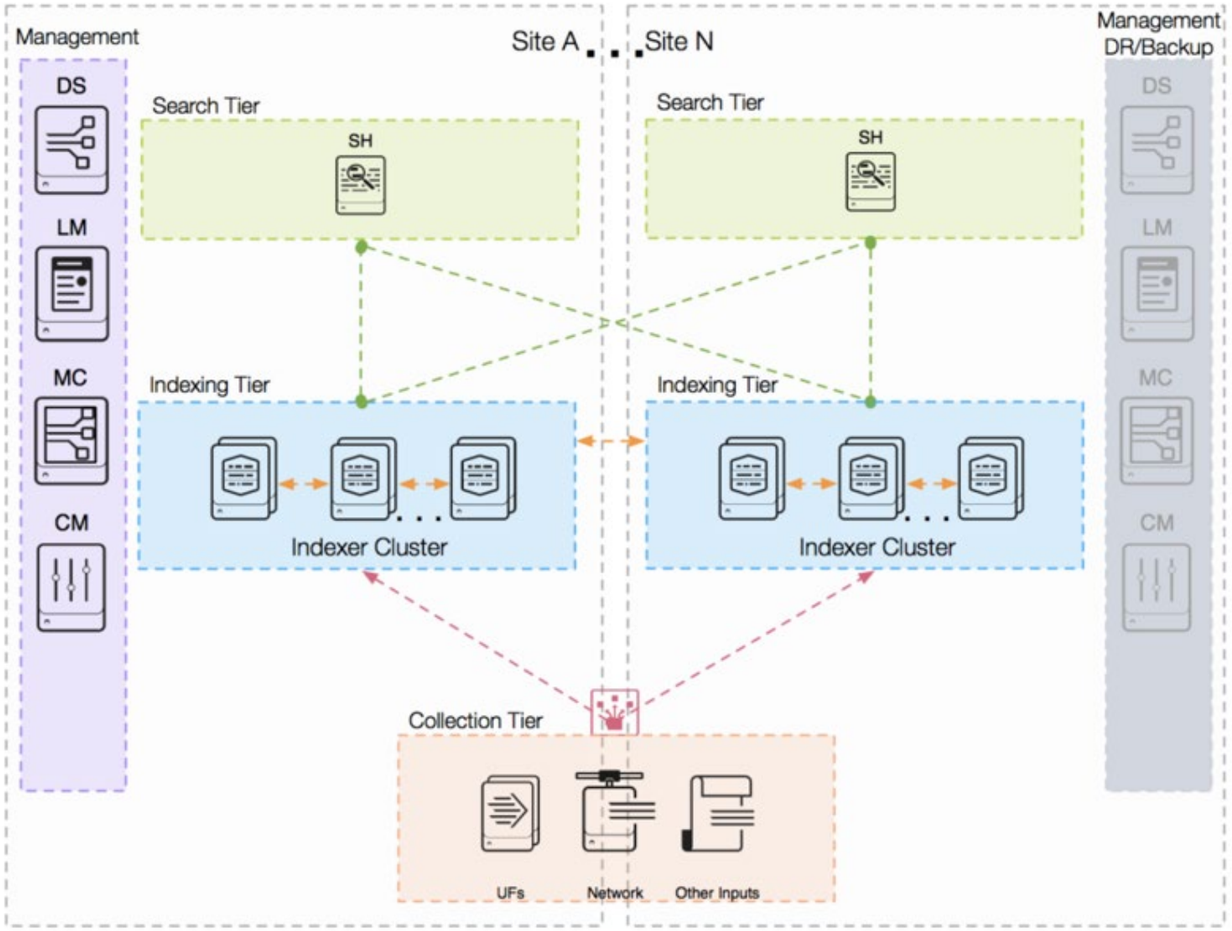
Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

SVA Examples – M2/M12

Distributed Clustered Deployment – Multi-Site



Indexing Tier Categories

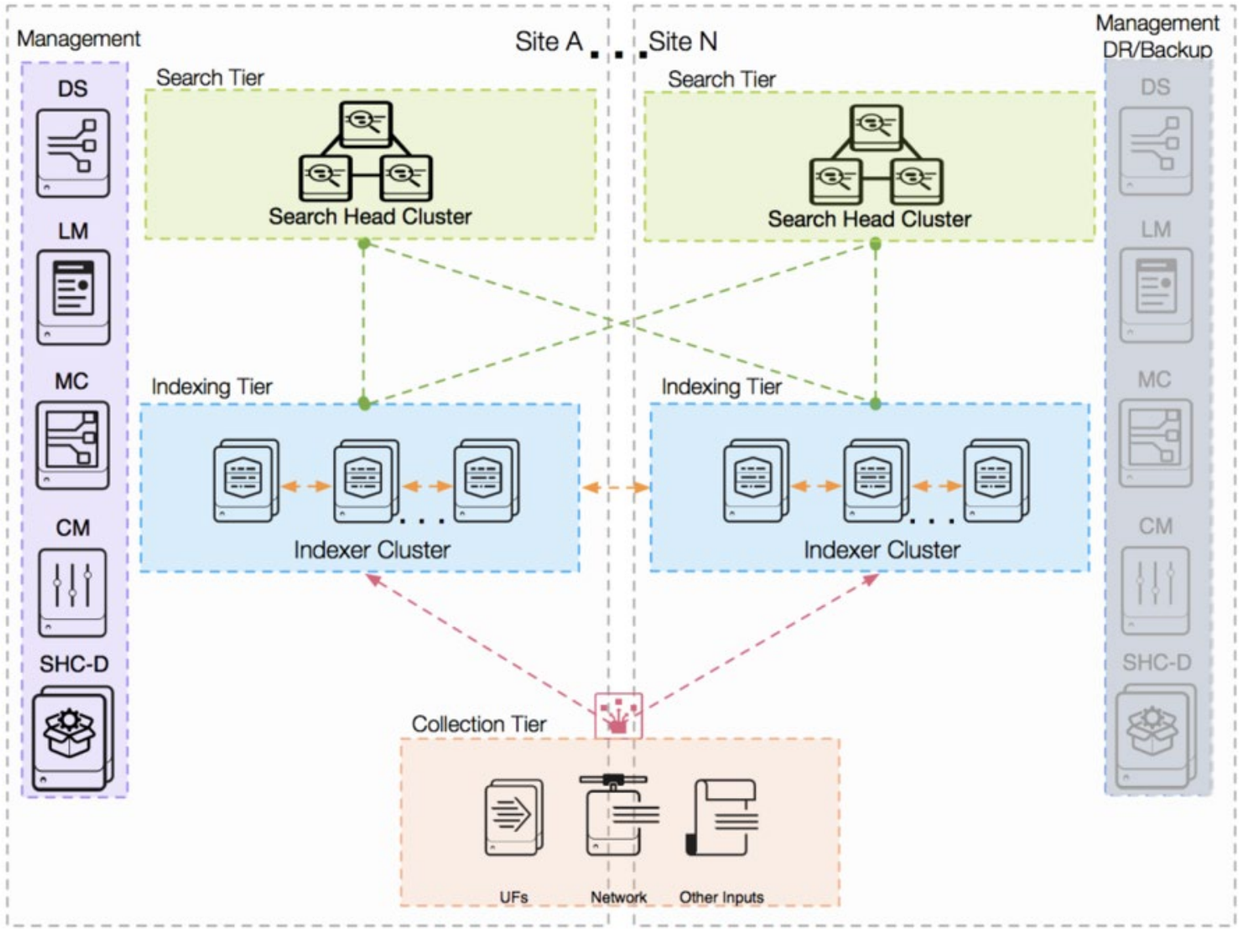
Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

SVA Examples – M2/M12

Distributed Clustered Deployment + SHC – Multi-Site



Indexing Tier Categories

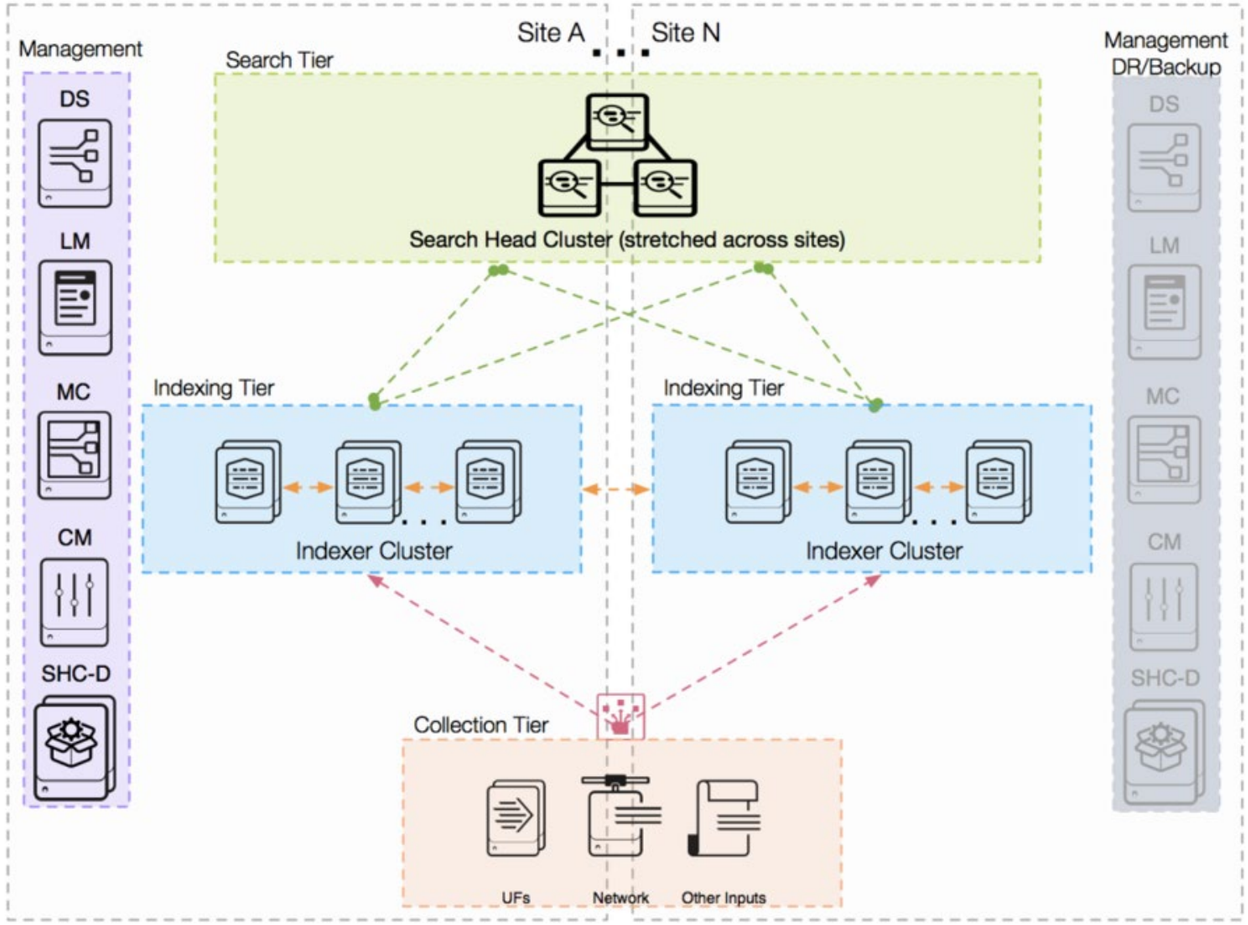
Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

SVA Examples – M4/14

Distributed Clustered Deployment + SHC – Multi-Site



Indexing Tier Categories

Category Code	Explanation
S	Indexer of a single-server Splunk deployment
D	Distributed indexer tier with at least 2 indexers
C	Clustered indexer tier (data replication is required)
M	Cluster indexer tier with multiple sites

Search Tier Categories

Category Code	Explanation
1	Single search head
2	Multiple search heads are required
3	Search head cluster is required
4	Search head cluster that spans multiple sites
+10	Dedicated search head is required for ES app. Add 10 to the search tier and read specific requirements for ES

Appendix B: Premium App Requirements

Splunk App for Enterprise Security (ES)

- Infrastructure impacts
  - A dedicated search head or search head cluster
  - 16 CPU / 32 GB RAM (Indexers and SH) ***minimum***
  - Requires extensive amount of data onboarding and configuration to work *properly*
  - One indexer per 100GB data indexed per day maximum
    - Assumes 15 correlation searches running
    - For more information about ES, read the following documentation

http://docs.splunk.com/Documentation/ES/latest/Install/Overview

Proper ES Sizing

- Sizing for ES is based on:
  - Data mix
  - Concurrent searches and users
  - Data Model acceleration
  - Assets & Identities (lookups)
- Example: 330GB Authentication Data + 70GB Network Traffic + 20GB Web + 130GB Other Data == 550GB/day total @ 20 concurrent searches
  - Sizing == 8 indexers with 24 cores

For more information about ES requirements, refer to:
docs.splunk.com/Documentation/ES/latest/Install/DeploymentPlanning

Note

Detailed information about ES deployments is discussed in the *Administering Splunk Enterprise Security* course.

Splunk App for IT Service Intelligence (ITSI)

- Infrastructure Impacts
  - Additional infrastructure may be needed, depending on the number of Key Performance Indicators (KPIs) that are tracked
    - The documentation has recommendations

For more information about ITSI, go to:

http://docs.splunk.com/Documentation/ITSI/latest/Configure/DeploymentPlanning

Splunk User Behavior Analytics (UBA)

- Hardware requirements
  - 50GB disk space for Splunk UBA installation
  - 500GB additional disk space for metadata
  - 16 CPU cores
  - 64 GB RAM
  - 800 IOPS
- Network Requirements
  - Static IP addresses for UBA servers
  - Firewalls and proxies must support inbound and outbound ports

> **Note**
> Contact Splunk Professional Services for installation assistance.

For more information go to:
http://docs.splunk.com/Documentation/UBA/latest/Install/Requirements

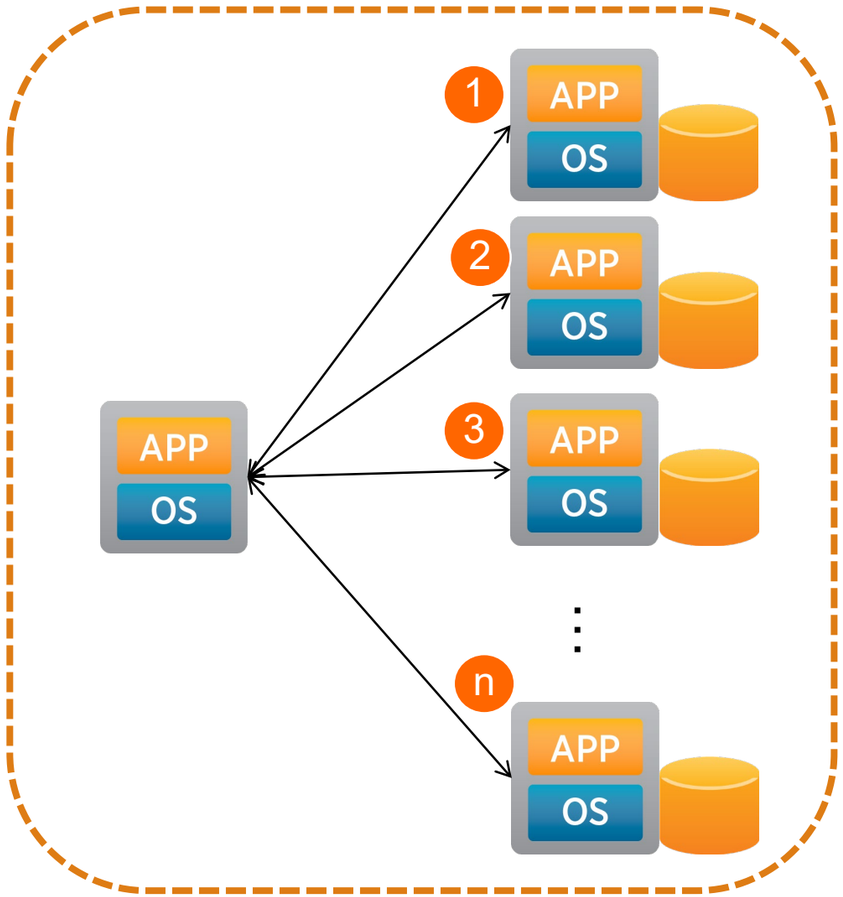
Appendix C: Search Considerations

Module Objectives

- Define MapReduce
- Discuss search performance
- Review how scheduled reports are dispatched
- Discuss differences between data summary methods

MapReduce Introduction

- A framework for processing parallelizable problems across huge datasets using a large number of computers (nodes) in a cluster
  en.wikipedia.org/wiki/MapReduce
- Splunk search uses this technology

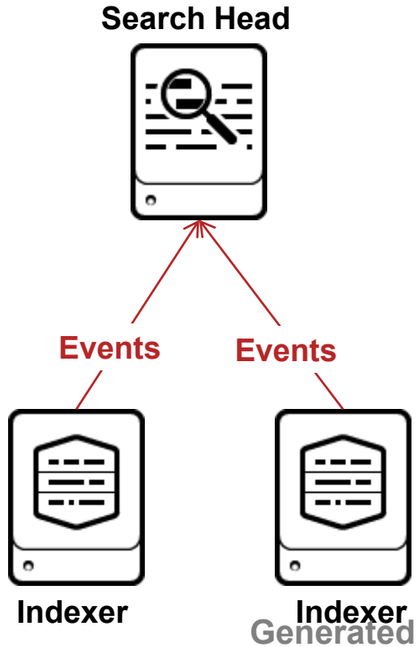


Source: Wikipedia

Performance Considerations for MapReduce

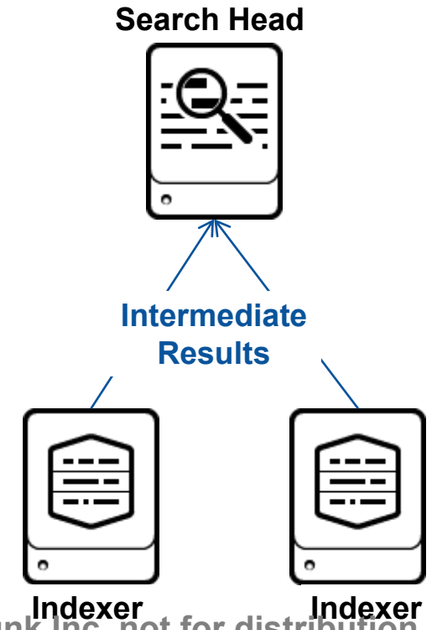
- Which searches MapReduce well? Which do not?
  - This is determined by how much of the work can be done on the indexers vs. how much manipulation of the search results must be done by the search head

Event searches and non-distributable commands



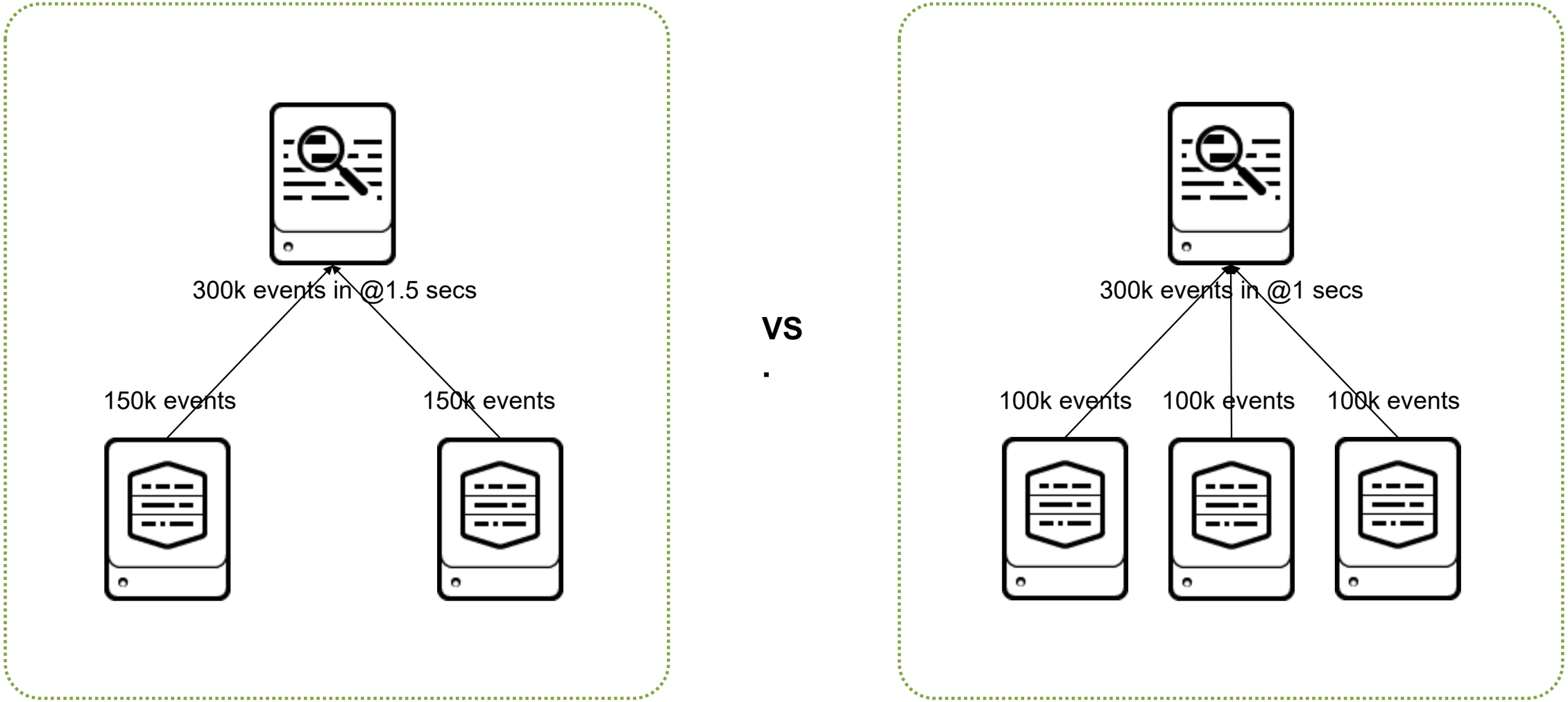
- Indexers retrieve events in parallel
- Efficient for event searches
- For other searches
  - Additional steps, if needed, must be done on the search head
  - CPU & memory bottlenecks can occur on the search head

Reporting searches



- Indexers can perform event retrieval and additional steps in parallel
- Intermediate results are returned
- The search head combines and presents the results

MapReduce – Why Add More Indexers?

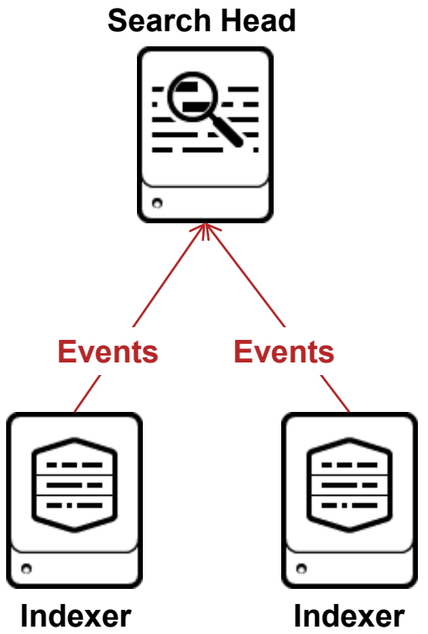


Questions for Search Types and Volume

What types of searches do you expect?	• Free-text search • Search by fields • Complex correlations and transactional searches
What types of alerts do you expect?	• Alerts and notifications of specific events • Alerts and notifications of thresholds reached or changed • Alerts and notifications of complex thresholds
Reporting	• How frequently will the reports run? • How current must the report data be? Near real time, to an hour, to the previous day? • Will historical or retroactive reports be required, or only current reports? • How will the reports be delivered to users? Via email, web interface, or other method? • Are there users who do not have access to the raw data who must see the reports?

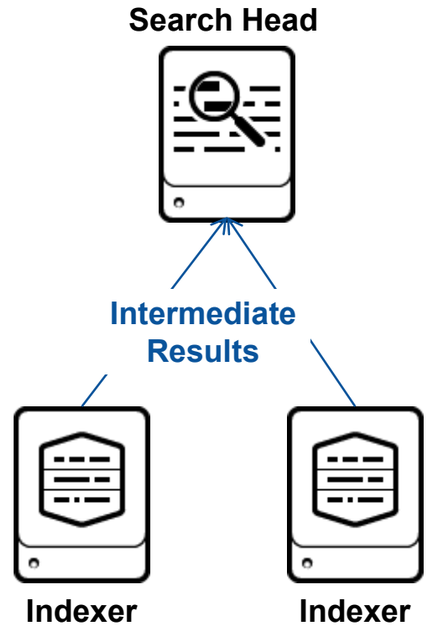
Raw Event Searches

- These search queries contain only the search command
  - The results are usually a list of raw events
  - Usually categorized as *super-sparse* or *rare* type searches
- A typical use case would be the deep analysis of a specific problem or incident
- Examples include:
  - Checking error codes
  - Correlating events
  - Investigating security issues
  - Analyzing failures



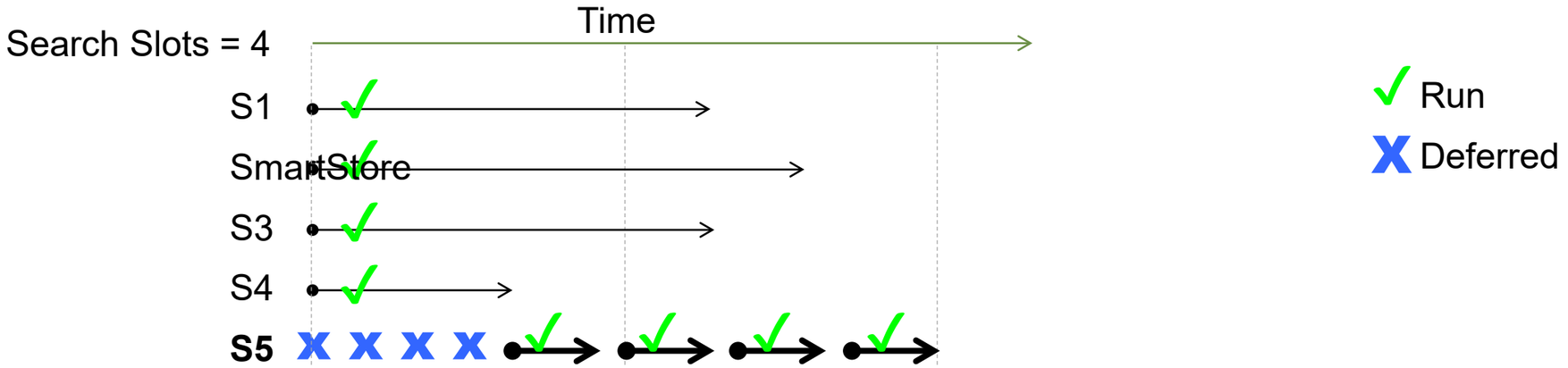
Report Generating Searches

- These search queries consist of initial search followed by some statistical calculation
  - The result is usually a table, graph or a single value
  - Often categorized as *dense* or *sparse* type searches
- They always require fields and at least one statistical command
- Examples include:
  - Compiling a daily count of error events
  - Counting the number of times a specific user has logged in
  - Calculating the 95th percentile of field values

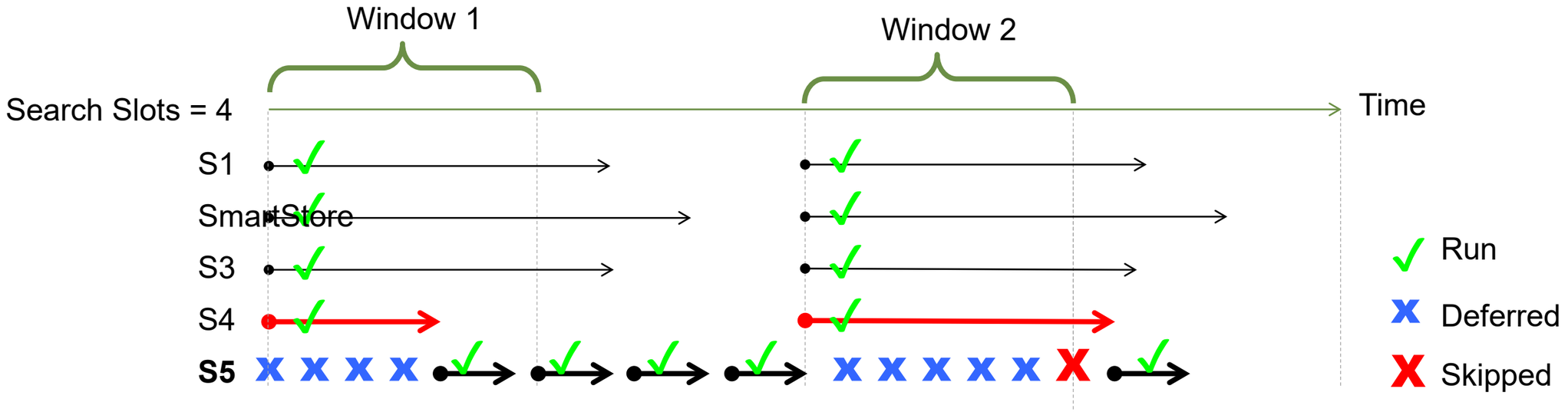


Splunk Report Scheduler Behavior

- Splunk runs as many concurrent jobs as it can, based on `limits.conf`
  - On a 16 CPU search head, the default is 11 concurrent scheduled searches
- In the example, 4 concurrent scheduled searches are used
  - Jobs S1, SmartStore, S3, and S4 take the first 4 slots
  - Job S5 must wait until another job finishes (S4)



Skipped Scheduled Jobs



- Every scheduled job has a **window** – a time period when it *should* run
- A job is deferred if it can't start when scheduled
  - A deferred job is retried (repeatedly for the duration of its window)
  - Example: Job S5 is deferred 4 times, but then runs during Window 1
- A job is skipped if it cannot start in its window
  - Example: Job S5 never runs during Window 2, and so is skipped

Report Scheduler Priorities

- The scheduler calculates priorities so that a skipped job has a higher priority in its next scheduled window
- To identify if Splunk has skipped searches, run the following

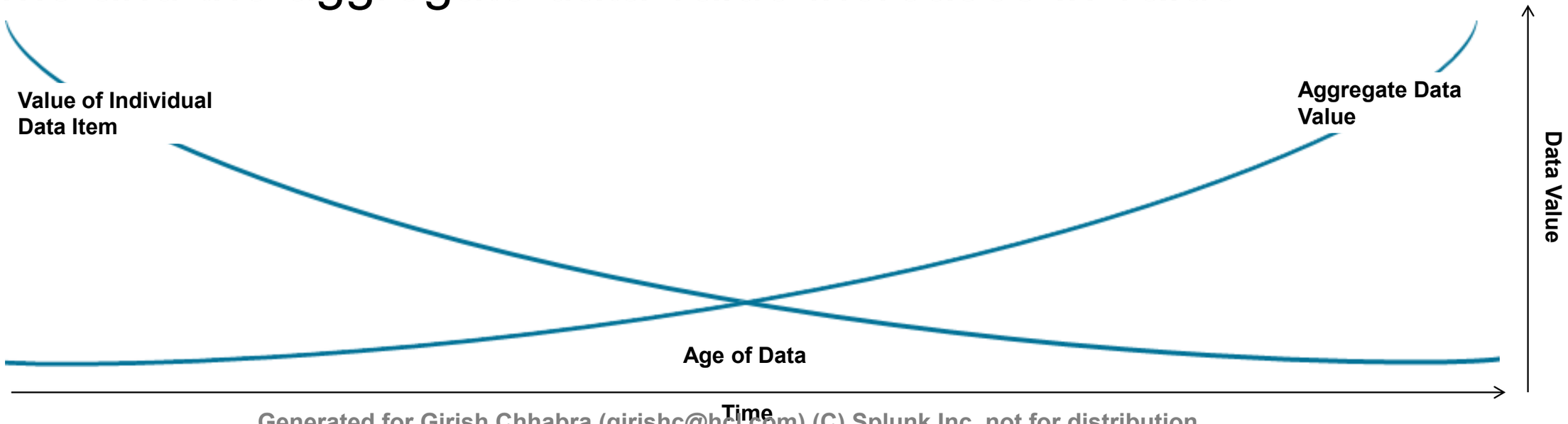
```
index=_internal source=*scheduler.log status!=success
| stats count by user app savedsearch_name status
```

```
Priority  = Next_runtime
          + Avg_runtime x priority_runtime_factor
          - skipped_count x period x priority_skipped_factor
          + window_adjustment
```

https://wiki.splunk.com/Community:TroubleshootingSearchQuotas

Summarization and Time Value of Data

- The value of data can be broken into two curves
  - Value of an individual data item
  - Aggregate data value
- In most cases, the value of an individual data item declines over time and the aggregate data value increases in value



How Acceleration Works

- Report acceleration and summary indexing speed up individual searches on a report-by-report basis
  - This is accomplished by building collections of pre-computed search result aggregates
- Data model acceleration speeds up reporting for the specific set of attributes (fields) that you define in a data model
  - Creates summaries for the specific set of fields, accelerating the dataset represented by that collection of fields rather than a particular search

Data Model Acceleration

- Used to speed up retrieval of the events that underlie a data model
- Speeds up reporting for the *entire set* of attributes (fields)
- Updated every five minutes
- Affects only *event* object hierarchies
  - Hierarchies based on search or transaction objects *cannot* be accelerated
- Most efficient if the root event objects include the index(es) in their initial constraint search

Data Model Acceleration (cont.)

- The High Performance Analytics Store
  - Consists of time-series index files with the `.tsidx` file extension
  - Exists on the indexer tier, parallel to the buckets that contain the events referenced in the `.tsidx` files
  - Spans a "summary range," which is the time range selected for acceleration
- Location and size can be set in **`indexes.conf`**
  - Default location is

    `$SPLUNK_HOME/var/lib/splunk/`*`indexName`*`/data_model_summary`

Report Acceleration

- Report Acceleration
  - Report results are "pre-computed" at regular intervals and stored on the indexer tier
  - All reports that use a similar set of data and computations automatically use the same report acceleration summaries if they can
- Report Acceleration Summaries
  - Span a time range selected for acceleration
  - Update every 10 minutes by default
  - Are automatically rebuilt if the data in the underlying bucket changes

docs.splunk.com/Documentation/Splunk/latest/Knowledge/Manageacceleratedsearchsummaries

Report Acceleration (cont.)

- Requirements for acceleration
  - The report was not created using Pivot
  - The underlying search qualifies for acceleration
- Splunk typically won't generate a summary if:
  - There are fewer than 100K events in the summary range
    - Faster to execute the search without a summary
  - Summary size is projected to be too large
    - Faster to execute the search using the normal index because it is smaller
- If a summary is defined but not created for the above reasons
  - Splunk continues to check periodically
  - Automatically creates a summary when/if the search meets the requirements

Thank You