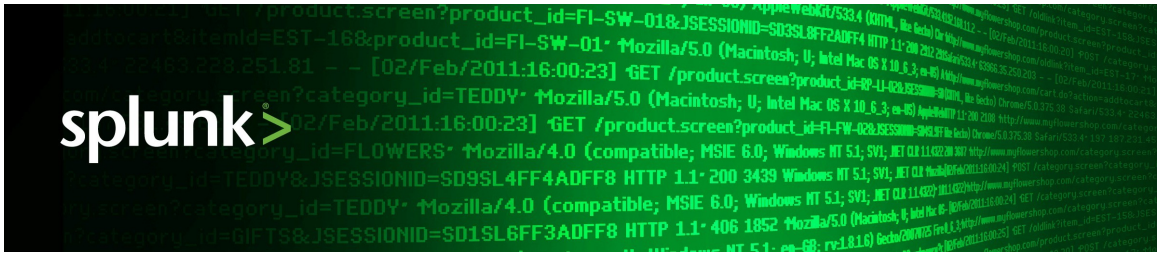


Splunk® Enterprise Distributed Deployment Manual 6.5.0

Generated: 10/30/2016 5:27 am

Table of Contents

Overview of Splunk Enterprise distributed deployments

Scale your deployment with Splunk Enterprise components

In single-instance deployments, one instance of Splunk Enterprise handles all aspects of processing data, from input through indexing to search. A single-instance deployment can be useful for testing and evaluation purposes and might serve the needs of department-sized environments.

To support larger environments, however, where data originates on many machines and where many users need to search the data, you can scale your deployment by distributing Splunk Enterprise instances across multiple machines. When you do this, you configure the instances so that each instance performs a specialized task. For example, one or more instances might index the data, while another instance manages searches across the data.

This manual describes how to distribute Splunk Enterprise across multiple machines. Distributed deployment provides the ability to:

- Scale Splunk Enterprise functionality to handle the data needs for enterprises of any size and complexity.
- Access diverse or dispersed data sources.
- Achieve high availability and ensure disaster recovery with data replication and multisite deployment.

How Splunk Enterprise scales

Splunk Enterprise performs three key functions as it processes data:

1. It ingests data from files, the network, or other sources.

2. It parses and indexes the data.

3. It runs searches on the indexed data.

To scale your system, you can split this functionality across multiple specialized instances of Splunk Enterprise. These instances can range in number from just a

few to many thousands, depending on the quantity of data that you are dealing with and other variables in your environment.

In a typical distributed deployment, each instance occupies one of three tiers that correspond to the key processing functions:

- Data input
- Indexing
- Search management

You might, for example, create a deployment with many instances that only ingest data, several other instances that index the data, and one instance that manages searches.

It is possible to combine some of these tiers or configure processing in other ways, but these three tiers are typical of most distributed deployments.

Splunk Enterprise components

Specialized instances of Splunk Enterprise are known collectively as **components**. With one exception, components are full Splunk Enterprise instances that have been configured to focus on one or more specific functions, such as indexing or search. The exception is the **universal forwarder**, which is a lightweight version of Splunk Enterprise with a separate executable.

There are several types of Splunk Enterprise components. They fall into two broad categories:

- Processing components. These components handle the data.
- Management components. These components support the activities of the processing components.

This topic discusses the processing components and their role in a Splunk Enterprise deployment. For information on the management components, see "Components that help to manage your deployment."

Types of processing components

There are three main types of processing components:

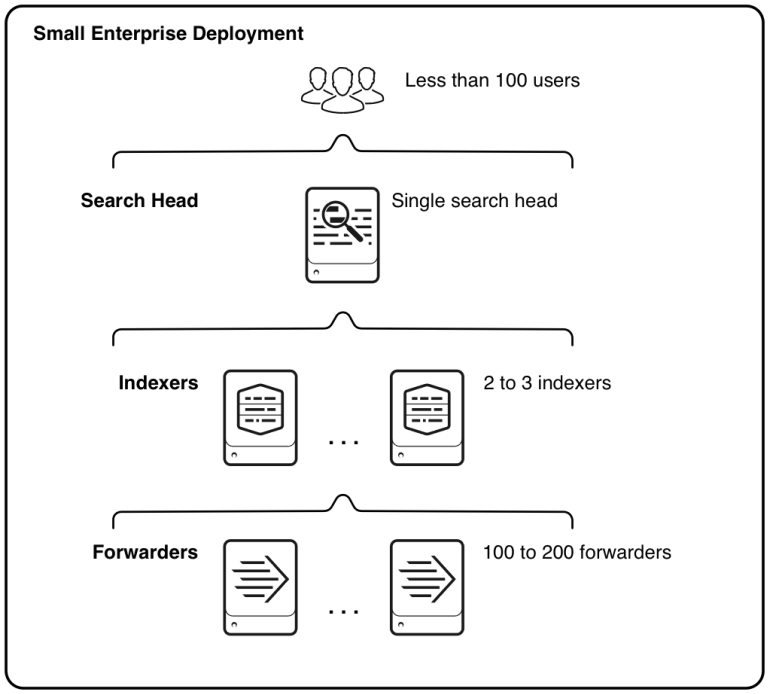
- Forwarders
- Indexers
- Search heads

Forwarders ingest data. There are a few types of forwarders, but the universal forwarder is the right choice for most purposes. It uses a lightweight version of Splunk Enterprise that simply inputs data, performs minimal processing on the data, and then forwards the data to an indexer. Because its resource needs are minimal, you can co-locate it on the machines that produce the data, such as web servers.

Indexers and **search heads** are built from Splunk Enterprise instances that you configure to perform the specialized function of indexing or search management, respectively. Each indexer and search head is a separate instance that usually resides on its own machine.

Processing components in action

This diagram provides a simple example of how the processing components can reside on the various processing tiers. It illustrates the type of deployment that might support the needs of a small enterprise.



Starting from the bottom, the diagram illustrates the three tiers of processing, in the context of a small enterprise deployment:

- **Data input.** Data enters the system through forwarders, which consume external data, perform a small amount of preprocessing on it, and then

forward the data to the indexers. The forwarders are typically co-located on the machines that are generating data. Depending on your data sources, you could have hundreds of forwarders ingesting data.

- **Indexing.** Two or three indexers receive, index, and store incoming data from the forwarders. The indexers also search that data, in response to requests from the search head. The indexers reside on dedicated machines.

- **Search management.** A single search head performs the search management function. It handles search requests from users and distributes the requests across the set of indexers, which perform the actual searches on their local data. The search head then consolidates the results from all the indexers and serves them to the users. The search head provides the user with various tools, such as dashboards, to assist the search experience. The search head resides on a dedicated machine.

To scale your system, you add more components to each tier. For ease of management, or to meet high availability requirements, you can group components into **indexer clusters** or **search head clusters.** See "Use clusters for high availability and ease of management."

This manual describes how to scale a deployment to fit your exact needs, whether you are managing data for a single department or a global enterprise, or for anything in between.

What comes next

The rest of this chapter focuses primarily on the **data pipeline**, from the point that the data enters the system to when it becomes available for users to search. It then correlates the Splunk Enterprise processing components with their roles in facilitating the data pipeline.

Other topics discuss indexer and search head clusters, the management components, and the manuals that provide configuration details for each type of component.

The remaining chapters in this manual offer practical guidance for implementing a distributed deployment. First, they discuss representative deployment types. Next, they provide end-to-end frameworks for implementing each of those deployments. Finally, they describe the post-deployment activities that an administrator needs to perform.

Use clusters for high availability and ease of management

You can group certain Splunk Enterprise **components** into clusters, so that they closely coordinate their activities. This serves two key purposes:

- High availability
- Ease of management

Indexer clusters

An **indexer cluster** is a group of Splunk Enterprise indexers that are configured to replicate each others' data, so that the system keeps multiple copies of all data. This process is known as **index replication**. By maintaining multiple, identical copies of Splunk Enterprise data, the cluster prevents data loss while promoting data availability for searching.

Splunk Enterprise clusters feature automatic failover from one indexer to the next. This means that, if one or more indexers fail, incoming data continues to get indexed and indexed data continues to be searchable.

Besides enhancing high availability, clusters have other features that help to simplify the management of a distributed deployment. For example:

- They include a capability to coordinate configuration updates easily across all indexers in the cluster.
- They include a built-in distributed search capability.
- They feature indexer discovery, which enables the set of forwarders to automatically load-balance across all indexers in the cluster.

Even if high availability is not a concern in your environment, you can still take advantage of the simplified management features by deploying an indexer cluster without index replication.

For guidance on implementing an indexer cluster, see "High availability deployment: Indexer cluster."

Search head clusters

A **search head cluster** is a group of search heads that serves as a central resource for searching. The search heads share knowledge objects, apps, and all other configurations. You can run the same searches, view the same

dashboards, and access the same search results from any search head in the cluster.

Search head clusters provide several important benefits:

- **Horizontal scaling.** As the number of users and the search load increases, you can add new search heads to the cluster. By combining a search head cluster with a third-party load balancer placed between users and the cluster, the topology can be transparent to the users.
- **High availability.** If a search head goes down, you can run the same set of searches and access the same set of search results from any other search head in the cluster.
- **No single point of failure.** The search head cluster uses a dynamic **captain** to manage the cluster. If the captain goes down, another search head automatically takes over management of the cluster.

For guidance on implementing a search head cluster, see "Medium to large enterprise deployment: Search head cluster with multiple indexers."

How data moves through Splunk deployments: The data pipeline

The processing tiers in a Splunk deployment correspond to the **data pipeline**, which is the route that data takes through Splunk software.

The processing tiers and the data pipeline

A Splunk deployment typically has three processing tiers:

- Data input
- Indexing
- Search management

See "Scale your deployment with Splunk Enterprise components."

Each Splunk processing **component** resides on one of the tiers. Together, the tiers support the processes occurring in the data pipeline.

As data moves along the data pipeline, Splunk components transform the data from its origin in external sources, such as log files and network feeds, into searchable events that encapsulate valuable knowledge.

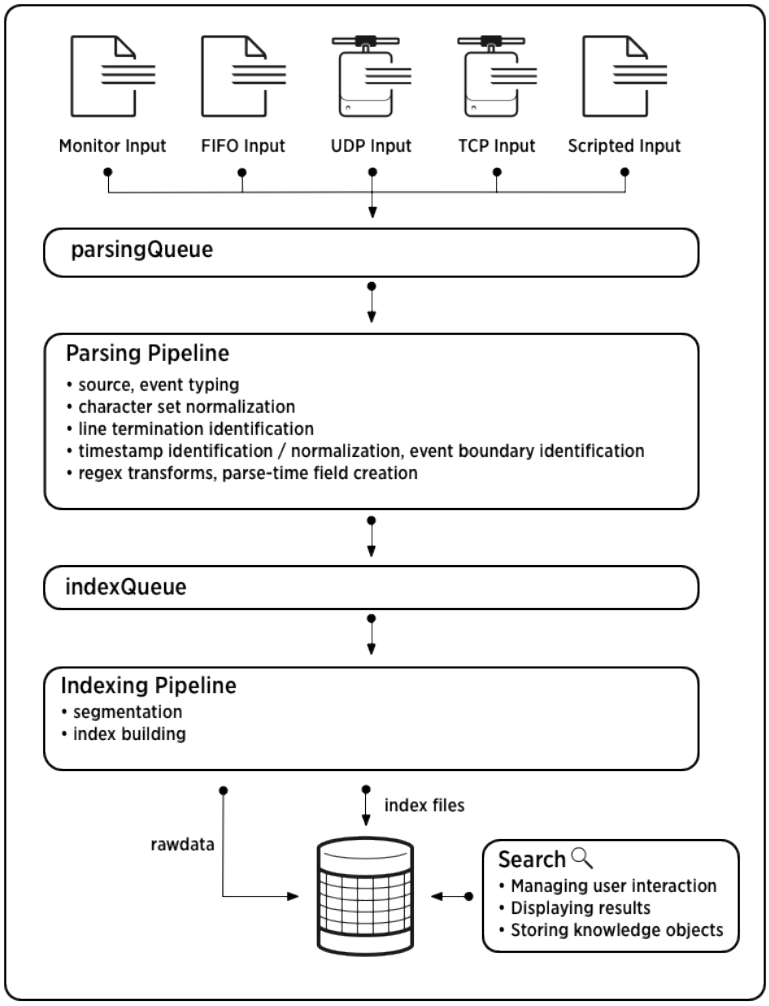
The data pipeline has these segments:

- **Input**
- **Parsing**
- **Indexing**
- **Search**

The correspondence between the three typical processing tiers and the four data pipeline segments is this:

- The data input tier handles the input segment.
- The indexing tier handles the parsing and indexing segments.
- The search management tier handles the search segment.

This diagram outlines the data pipeline:



Splunk components participate in one or more segments of the data pipeline. See "Components and the data pipeline."

Note: The diagram represents a simplified view of the indexing architecture. It provides a functional view of the architecture and does not fully describe Splunk software internals. In particular, the parsing pipeline actually consists of three pipelines: **parsing**, **merging**, and **typing**, which together handle the parsing function. The distinction can matter during troubleshooting, but does not ordinarily affect how you configure or deploy Splunk Enterprise components. For a more detailed diagram of the data pipeline, see "How Indexing Works" in the Community Wiki.

The data pipeline segments in depth

This section provides more detail about the segments of the data pipeline. For more information on the parsing and indexing segments, see also "How indexing works" in the *Managing Indexers and Clusters of Indexers* manual.

Input

In the input segment, Splunk software consumes data. It acquires the raw data stream from its source, breaks it into 64K blocks, and annotates each block with some metadata keys. The keys apply to the entire input source overall. They include the **host**, **source**, and **source type** of the data. The keys can also include values that are used internally, such as the character encoding of the data stream, and values that control later processing of the data, such as the index into which the events should be stored.

During this phase, Splunk software does not look at the contents of the data stream, so the keys apply to the entire source, not to individual events. In fact, at this point, Splunk software has no notion of individual events at all, only of a stream of data with certain global properties.

Parsing

During the parsing segment, Splunk software examines, analyzes, and transforms the data. This is also known as **event processing**. It is during this phase that Splunk software breaks the data stream into individual events.The parsing phase has many sub-phases:

- Breaking the stream of data into individual lines.
- Identifying, parsing, and setting timestamps.

- Annotating individual events with metadata copied from the source-wide keys.
- Transforming event data and metadata according to regex transform rules.

Indexing

During indexing, Splunk software takes the parsed events and writes them to the index on disk. It writes both compressed raw data and the corresponding index files.

For brevity, parsing and indexing are often referred together as the indexing process. At a high level, that makes sense. But when you need to examine the actual processing of data more closely or decide how to allocate your components, it can be important to consider the two segments individually.

Search

The search segment manages all aspects of how the user accesses, views, and uses the indexed data. As part of the search function, Splunk software stores user-created **knowledge objects**, such as reports, event types, dashboards, alerts, and field extractions. The search function also manages the search process itself.

Where to go next

While the data pipeline processes always function in approximately the same way, no matter the size and nature of your deployment, it is important to take the pipeline into account when designing your deployment. For that, you must understand how Splunk components map to the data pipeline segments. See "Components and the data pipeline."

Components and the data pipeline

Each segment of the **data pipeline** corresponds to one or more Splunk Enterprise processing **components**. For example, data input is a pipeline segment. You can use either an indexer or a forwarder to input data.

Most segments of the data pipeline can be handled by multiple component types. The component that you employ for a segment depends on how you structure your deployment.

For example, although you can input data directly into an indexer, you would typically do so only in small deployments consisting of a single instance. In larger deployments, and frequently also in single-instance deployments, you would instead input data through a forwarder. Delegating input tasks to a forwarder can offer greater flexibility for your deployment.

For information on the data pipeline, see How data moves through Splunk Enterprise: The data pipeline.

For information on the processing components, see Scale your deployment with Splunk Enterprise components.

How components support the data pipeline

This table correlates the data pipeline segments with the components that can handle the segment:

Data pipeline segment	Components
Data input	**indexer** **universal forwarder** **heavy forwarder**
Parsing	indexer heavy forwarder
Indexing	indexer
Search	indexer **search head**

Some typical interactions between components

These are examples of some of the ways that you can distribute and manage Splunk Enterprise functionality.

Forward data to an indexer

In most deployments, **forwarders** handle data input only, collecting data and sending it on to a Splunk Enterprise indexer. The indexer then performs both parsing and indexing. In some deployments, however, the forwarders also parse the data before sending it to the indexer, which then only indexes.

See How data moves through Splunk Enterprise: The data pipeline for the distinction between parsing and indexing.

Forwarders come in two flavors:

- **Universal forwarders.** These maintain a small footprint on their host machine. They perform minimal processing on the incoming data streams before forwarding them on to an indexer. The indexer then parses and indexes the data.
- **Heavy forwarders.** These retain much of the functionality of a full Splunk Enterprise instance. They can parse data before forwarding it to the receiving indexer. When a heavy forwarder parses the data, the indexer handles only the indexing segment.

Both types of forwarders tag data with metadata, such as host, source, and source type, before forwarding it on to the indexer.

Forwarders allow you to use resources efficiently while processing large quantities or disparate types of data. They also enable a number of interesting deployment topologies, by offering capabilities for **load balancing**, data **filtering**, and **routing**.

For an extended discussion of forwarders, including configuration and detailed use cases, see About forwarding and receiving in *Forwarding Data*.

Search across multiple indexers

In **distributed search**, you separate the indexing/parsing and search segments. Search heads send search requests to indexers and merge the results back to the user. This topology is particularly useful for horizontal scaling. To expand your deployment beyond the departmental level, you will likely employ distributed search.

For an extended discussion of distributed search, including configuration and detailed use cases, see About distributed search in *Distributed Search*.

Configurations and the data pipeline

For guidance on where to configure Splunk Enterprise settings, see Configuration parameters and the data pipeline in the *Admin Manual*. The topic lists configuration settings and the data pipeline segments that they act upon.

If you know which components in your Splunk Enterprise topology handle which segments of the data pipeline, you can use that topic to determine where to configure any particular setting. For example, if you use a search head to handle the search segment, you need to configure all search-related settings on the search head.

For more information

In summary, these are the fundamental components of a Splunk Enterprise distributed environment:

- **Indexers.** See Indexes, indexers, and indexer clusters in *Managing Indexers and Clusters of Indexers*.
- **Forwarders.** See About forwarding and receiving in *Forwarding Data*.
- **Search heads.** See About distributed search in *Distributed Search*.

Components that help to manage your deployment

Management components support the activities of the processing components. A deployment usually includes one or more of these management components:

- The **monitoring console** performs centralized monitoring of the entire deployment.
- The **deployment server** updates configurations and distributes apps to processing components, primarily forwarders.
- The **license master** handles Splunk Enterprise licensing.
- The **cluster master**, or "master node," coordinates the activities of an **indexer cluster**. It also handles updates for indexer clusters.
- The **deployer** handles updates for **search head clusters**.

As with processing components, management components are specially configured versions of Splunk Enterprise instances.

Depending on the component and its workload, you can frequently combine management components on a single Splunk Enterprise instance. In some cases, you can locate the management component on the same instance as a processing component.

See the documentation for the specific component to learn of any co-location possibilities:

- For the monitoring console, see "Which instance should host the console?" in *Monitoring Splunk Enterprise.*
- For the deployment server, see "Plan a deployment" in the *Updating Splunk Enterprise Instances* manual.
- For the license master, see "Configure a license master" in the *Admin Manual.*
- For the cluster master, see "Additional roles for the master node" in the *Managing Indexers and Clusters of Indexers* manual.
- For the deployer, see "Deployer requirements" in the *Distributed Search* manual.

Key manuals for a distributed deployment

All Splunk Enterprise features are available in a distributed deployment, so all Splunk Enterprise manuals are important to realizing the full potential of your deployment. However, there is a small subset of manuals of particular importance when you are setting up and configuring the deployment. These manuals are mostly organized by component type:

- *Installation Manual.* Describes how to install a Splunk Enterprise instance.
- *Forwarding Data.* Describes how to install, deploy, and configure any type of forwarder.
- *Managing Indexers and Clusters of Indexers.* Describes how to configure indexers and how to deploy and configure indexer clusters.
- *Distributed Search.* Describes distributed search: Setting up search heads, connecting indexers to the search heads, and deploying and configuring search head clusters.
- *Getting Data In.* Provides guidance for configuring the range of data inputs.* *Admin Manual.* Covers a wide range of administration-related activity. The chapters on the license master are of particular importance when setting up your deployment.
- *Monitoring Splunk Enterprise.* Covers the set up of the monitoring console, and how to use the console to monitor your deployment.
- *Updating Splunk Enterprise Instances.* Describes how to use the deployment server to update the components in the deployment.

Implement a distributed deployment

Start implementing your distributed deployment

Splunk Enterprise deployments range from single-instance departmental deployments, indexing a few gigabytes of data a day and servicing just a few users searching the data, to large enterprise deployments distributed across multiple data centers, with indexing requirements in the terabyte range and searches performed by hundreds of people.

A production deployment of Splunk Enterprise typically requires that you install and configure a variety of components, such as forwarders, search heads, and indexers. This manual includes a series of frameworks for implementing common distributed deployment scenarios, ranging in size from departmental to large enterprise deployments.

The frameworks serve as high-level roadmaps for navigating the implementation process. Each framework describes a common deployment scenario. It then provides an overview of the process for implementing that scenario, with links to detailed documentation for every step of the process.

Choose the scenario that most closely reflects your need, and follow its framework. The framework will take you to the point where you have a running deployment. At that point, you are ready to focus on the range of administration tasks, like setting up users, dealing with security concerns, and, finally, creating knowledge objects like dashboards and searches for your end users.

Splunk Enterprise components and deployment scenarios

To implement a Splunk Enterprise production deployment, you must install a variety of Splunk Enterprise components. The specific components that you install depend on the deployment type. Even to implement a single-instance deployment, in which a single Splunk Enterprise instance serves as both indexer and search head, you need to install forwarders on the data-generating hosts, to feed data to the instance. To scale beyond a single instance, you must install and configure several types of Splunk Enterprise components.

The components that you configure vary according to the size and the specific requirements of your deployment. For example, a deployment that ensures high availability of data requires a different configuration from a deployment where high availability is not a strong concern.

The different components are, for the most part, built from the same Splunk Enterprise software package, with different configurations to meet the different roles. The exception is the universal forwarder, which uses a lightweight package of Splunk Enterprise.

For a thorough discussion of components, see "Scale your deployment with Splunk Enterprise components."

How to get started

The process of implementing your deployment requires that you make a series of decisions based on your goals. It also requires that you follow procedures described in numerous topics that are spread across a large body of documentation. The procedures that you implement vary according to the needs of your deployment.

It can be difficult to determine the right set of procedures for your particular deployment needs, and then to locate all the procedures in the documentation. The intent of this chapter is to simplify this process. The topics in this chapter provide you with information to help you understand your deployment needs and make the right decisions from the start.

The chapter "Typical deployment scenarios, with implementation frameworks", which follows next, provides separate topics for each of several representative deployment types, or scenarios. These topics contain end-to-end deployment frameworks for each scenario. Each framework includes a set of high-level steps that you can follow to deploy the scenario, with links to topics that contain the detailed procedures for each step.

What to do next

Follow this path:

1. See "Types of deployments" to understand the choices you that you must make and the characteristics of various types of deployments.

2. In "Types of deployments", read through the high-level descriptions for the deployment types to find the one that best correlates to your need.

3. To proceed with your deployment, turn to the topic for the scenario that you want to implement, and follow its implementation framework. For example, "Small enterprise deployment: Single search head with multiple indexers."

The scenario topics mostly assume that you are implementing the deployments from scratch. The issues are similar for expansions of existing deployments, however. The topics discuss any issues that you particularly need to be aware of when migrating from a smaller, or otherwise different, deployment type.

4. See "Post-deployment activities" for guidance on the activities that you need to perform after you complete the initial deployment.

Types of distributed deployments

You can customize your Splunk Enterprise deployment in a wide variety of ways. There are, however, some fundamental groupings into which most deployments fall. This topic discusses some key characteristics and considerations for various types of deployments.

Key factors that determine the type of deployment

These are the main issues that determine the type and scale of your deployment:

- **Indexing volume.** How much data are planning to index on a daily basis? To handle increased indexing loads, you might need multiple **indexers**.
- **Number and type of searches.** How frequently will you be running searches, either scheduled or ad hoc? What type of searches will you be running? Large numbers of searches, or frequent process-intensive searches, can tax both **search head** and indexer resources.
- **Number of concurrent users.** How many users will be viewing dashboards or running searches concurrently? To handle increased numbers of users, you might need to add search heads, usually through a **search head cluster**.
- **Data fidelity requirements.** If you must ensure that the system never loses data, an **indexer cluster** is a necessity.
- **Availability requirements.** What requirements do you have for data availability? If you must always have access to the full set of data, you might need to deploy both an indexer cluster and a search head cluster.
- **Disaster recovery requirements.** How important is fast disaster recovery? A **multisite indexer cluster** can ensure fast failover to identical

sets of data across geographically dispersed data centers.

Other considerations can also enter into your overall deployment plans, such as security requirements and the location of the data.

Representative deployment types

These are some of the main types of deployments, based on size:

- **Departmental.** A single instance that combines indexing and search management functions.
- **Small enterprise.** One search head with two or three indexers.
- **Medium enterprise.** A small search head cluster, with several indexers.
- **Large enterprise.** A large search head cluster, with large numbers of indexers.

These deployment types are just points on a continuous scale, ranging from single-instance deployments to deployments that provide enterprise-wide coverage for a vast number of use cases.

In addition, you can deploy an indexer cluster in an enterprise deployment of any size. An indexer cluster offers advantages such as high availability, disaster recovery, and simplified scaling.

It is also possible to combine topologies in various ways. For example, you can deploy a search head that searches across both an indexer cluster and a set of independent indexers.

Note: The terms "small enterprise," "medium enterprise," and so on, do not specifically address the size of the enterprise using the Splunk platform. Instead, they are indicators of the breadth and depth of the functions that the Splunk platform supports in the enterprise. As awareness of the value of the Splunk platform for handling a wide range of use cases grows with continued success, the size of a deployment also typically grows. So, for example, a Fortune 500 company might start with a departmental-level, single-instance Splunk Enterprise installation for a very specific use case, and then, over time, transition through small enterprise and medium enterprise deployments, to eventually adopt a large enterprise deployment that provides key value to organizations and use cases distributed throughout the company.

Get started with your deployment

Read the rest of this topic to get a clear sense of the type of deployment you want to implement. Then turn to one of the following topics, accordingly:

- "Departmental deployment: Single indexer"
- "Small enterprise deployment: Single search head with multiple indexers"
- "Medium to large enterprise deployment: Search head cluster with multiple indexers"
- "High availability deployment: Indexer cluster"

These topics provide further details on each deployment type, including a diagram of the basic architecture. Most importantly, each includes a high-level, end-to-end guide to the implementation process, with links to the specific procedures to follow to implement the deployment.

Primary characteristics of deployments at representative scaling levels

The characteristics of a deployment change as it grows in size. This table gives you some idea of what to expect, with information on the Splunk components that you need to deploy to meet your needs.

	Departmental	Small enterprise	Medium enterprise	Large enterprise
Indexing volume (daily)	0-20GB	20-100GB	100-300GB	300GB-1TB+
# of forwarders	Median < 10; maximum 100	Median in the 10's; maximum in the 100's	Median in the 10's; maximum in the low 1000's	Median in the 10's; maximum in the 1000's
# of users	Median < 10	Median in the 10's	Median in the 10's; maximum in the low 100's	Median in the 10's; maximum 500+
# of apps (pre-packaged and customer-developed,	1-10	1-10	1-20+	10-50

combined)				
Indexing tier	1 indexer	2-3 indexers, possibly in a cluster	4-9 indexers, possibly in a cluster	10+ indexers, possibly in a cluster
Search management tier	Combined with indexer	1 standalone search head	3 search heads in a cluster	3+ search heads in a cluster
Configuration management function	Manual configuration or deployment server	Manual configuration or deployment server	Deployment server or 3rd party tool for forwarders and indexers. Deployer for search head cluster.	Deployment server or 3rd party tool for forwarders and indexers. Deployer for search head cluster.

Design considerations

Design considerations also change as the deployment scales. This table summarizes some of the issues you need to consider when designing your deployment.

	Departmental	Small enterprise	Medium enterprise
Forwarder issues	Management, monitoring	Load balancing, management, monitoring	Load balancing, management, monitorin intermediate forwarders
Search issues	User counts, alerts, apps	Search head/indexer knowledge management, user counts	Search head/indexer knowledge manageme user counts, search he clustering, job servers
Scheduled search workload	Alerts, app/dashboard dependent, summary searches	Alerts, app/dashboard dependent, summary searches	Alerts, app/dashboard dependent, summary searches, job server
Input types	Network, scripted	Network, scripted, batch, integrations	Network, scripted, batc integrations
Availability	Platform-dependent (RAID, power supplies)	Data fabric (forwarder load balancing, storage, index	User interface (search head clustering, load

		replication)	balancers); data fabric (forwarder load balanci storage, index replicati
Recoverability	Backup, retention	Backup, index replication, bucket/index restoration	Backup, index replicati bucket/index restoratio
Accessibility	Local vs. enterprise authentication	Authentication method	Authentication method
Staffing	Admin: 0.5-1 person; search/dashboard/appdev/ knowledge manager: 0.25-1 person	Admin: 0.5-1 person; search/dashboard/appdev/ knowledge manager: 0.5-1.5 persons	Admin/architect: 1-2 persons; knowledge manager: 0.5-2 person search/dashboard/appd 1-3 persons; program/project manag 1 person

For information regarding training opportunities and Professional Services offerings appropriate to your deployment scale, contact your Splunk sales representative.

Further reading

For more guidance in determining the size and type of your deployment:

- For details on hardware capacity planning and deployment scaling, see the *Capacity Planning* manual.
- For a discussion of the benefits and trade-offs of implementing a high availability deployment, see "About indexer clusters and index replication" in the *Managing Indexers and Clusters of Indexers* manual.

Typical deployment scenarios, with implementation frameworks

Departmental deployment: Single indexer

A single Splunk Enterprise instance, serving as both indexer and search head, usually meets the indexing and search needs of a single department within a larger organization. You typically also install forwarders on the data-generating hosts. The forwarders feed data from the hosts to the indexer.

This topic describes how to implement a deployment consisting of:

- One combined indexer/search head
- Multiple forwarders

Use case

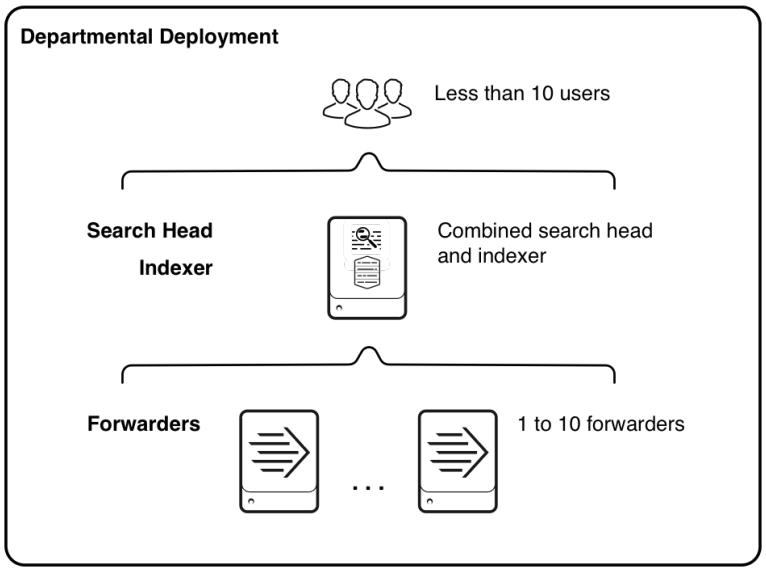
Characteristics of this type of deployment include:

- Indexing volume of under 20GB/day.
- A few users, typically less than 10.
- A relatively small number of forwarders sending data to the instance, typically less than 10 and rarely exceeding 100.

For details on the characteristics of a departmental deployment, see "Types of deployments."

Architecture

This diagram shows a high-level view of the architecture for this type of deployment:



Starting from the bottom, the diagram illustrates the tiers of processing:

- **Data input.** Data enters the system through **forwarders**, which consume external data, perform a small amount of preprocessing on it, and then forward the data to the indexer. In a departmental deployment, you typically have fewer than ten forwarders, although, for some use cases, you might have as many as 100.

- **Indexing and searching (combined).** A single **indexer** receives, indexes, and stores incoming data from the forwarders. The indexer also doubles as a **search head.** In that capacity, it handles search requests, such as ad hoc requests from users and saved search requests. The search head provides the user with various tools, such as dashboards, to assist the search experience.

Implementation framework

To implement this type of scenario:

1. Install one Splunk Enterprise instance to serve as the combined indexer/search head. For installation instructions, see "Installation overview" in the *Installation Manual.*

2. Configure the Splunk Enterprise license. See "How Splunk Enterprise licensing works" in the *Admin Manual*.

3. Configure the instance's **receiving port.** The forwarders send data to the indexer through this port. See "Enable a receiver" in the *Forwarding Data* manual.

4. Install **universal forwarders** on the machines hosting your data sources, and configure the forwarders to send data to the Splunk Enterprise instance. How you do this depends on your needs and preferences, as well as on how many forwarders you are deploying. For example:

- You can install forwarders one-by-one or simultaneously across many machines.
- You can install forwarders and then configure them later, or install and configure forwarders at the same time.
- You can configure forwarders either manually or by means of the deployment server or third-party software.

For information on installing and configuring universal forwarders, see "Install the universal forwarder software" in the *Splunk Universal Forwarder Manual*.

5. Configure the inputs to the forwarders, so that data begins to enter the system. See "Configure your inputs" in the *Getting Data In* manual.

Next steps

Once you have your Splunk Enterprise instance up and running, you can further refine your system and prepare the data and its presentation for the benefit of your end users. For a summary of the types of activities you need to perform now, see "Post-deployment activities."

To scale further

To increase your indexing and search capacity, the first step is to separate the indexing function from the search management function. To do this, add a second Splunk Enterprise instance to serve as a dedicated search head. Once you have a dedicated search head, you can boost indexing capacity by adding more indexers. See "Small enterprise deployment: single search head with multiple indexers."

For guidance on determining when to add more instances, and whether to add only indexers, or both indexers and search heads, see:

- "Types of deployments" in this manual.

- Summary of performance recommendations" in the *Capacity Planning* manual.

Small enterprise deployment: Single search head with multiple indexers

To increase indexing and search capacity beyond a certain point, you must transition from a single Splunk Enterprise instance to a multi-instance deployment. You split the search management and indexing functions, allocating them to separate instances running on separate machines. At this first level of scaling, you typically deploy a single search head that communicates with two or three indexers. This type of deployment is characteristic of a multi-departmental, or small enterprise, solution.

This topic describes how to implement a deployment consisting of:

- One search head
- Multiple indexers
- Multiple forwarders

Note: Instead of deploying multiple individual indexers, you can deploy an **indexer cluster**. This topology can be useful even if you do not want high data availability and the accompanying storage overhead. It still entails a small amount of overhead, but it offers the benefit of simplified indexer management. See "High availability deployment: Indexer cluster."

Use case

This **distributed search** scenario provides the first level of horizontal scaling. It allows users to run searches across a set of indexers. As your needs increase further, you can add more indexers.

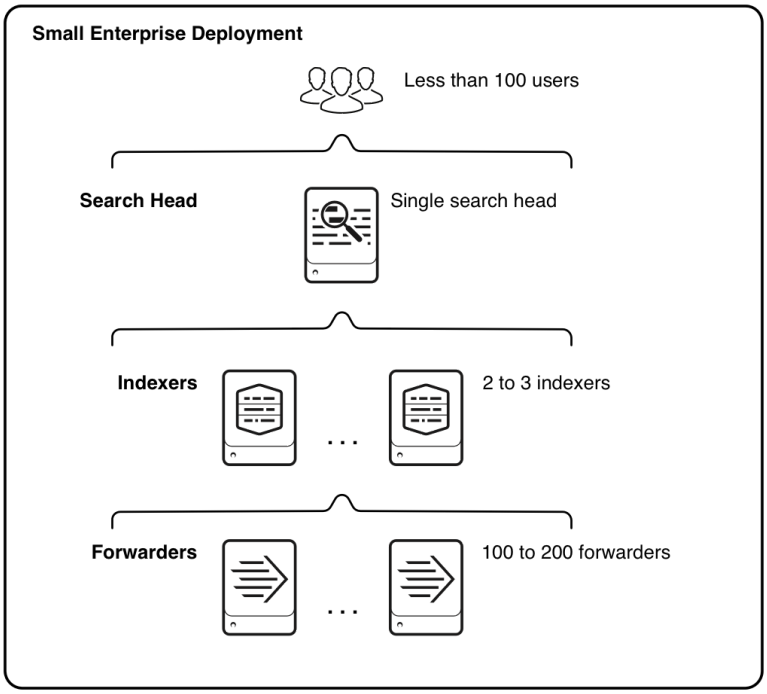
Characteristics of this type of deployment include:

- Indexing volume between 20 and 100GB/day.
- Between 10 and 100 users.
- Up to several hundred forwarders feeding data to the indexers. The forwarders typically make use of load balancing to distribute the data across the set of indexers.

For details on the characteristics of a small enterprise deployment, see "Types of deployments."

Architecture

This diagram shows a high-level view of the architecture for this type of deployment:



Starting from the bottom, the diagram illustrates three tiers of processing:

- **Data input.** Data enters the system through **forwarders**, which consume external data, perform a small amount of preprocessing on it, and then forward the data to the indexers. You usually configure the forwarders to use their built-in load-balancing capabilities to spread the data across the set of indexers.

- **Indexing.** Two or three **indexers** receive, index, and store incoming data from a set of forwarders. The indexers also search that data, in response to requests from the search head.

- **Search management.** A single **search head** performs the search management function. It handles search requests, such as ad hoc requests from users and saved search requests, and distributes the

requests across the set of indexers, which perform the actual searches on their local data. The search head then consolidates the results from the indexers and serves them to the users. The search head provides the user with various tools, such as dashboards, to assist the search experience.

Migration issues

This topic assumes that you are implementing this deployment from scratch. If, instead, you are expanding an existing single-instance deployment, the process is similar but there are a few additional issues to be aware of:

- Make your current single-instance deployment an indexer, not a search head, in the new deployment. This instance is likely to be provisioned more appropriately for the needs of an indexer. The data already on the instance will continue to be searchable in the new deployment.
- Transfer the standalone instance's apps and knowledge objects to the new search head instance.
- Configure your licensing to accommodate the additional instances.

The implementation framework procedure includes steps to handle these issues.

Implementation framework

To implement this type of scenario:

1. Install Splunk Enterprise instances for the indexers and search head. For example, to deploy a single search head with two indexers, install three instances. For installation instructions, see "Installation overview" in the *Installation Manual.*

Note: Search heads and indexers have different hardware requirements. For information on provisioning the hardware for your instances, see "Reference hardware" in the *Capacity Planning* manual.

2. (Migration only) If you are migrating from a standalone instance, use that instance as one of the indexers. The data already on the instance continues to be available in the new environment.

3. (Migration only) If you are migrating from a standalone instance, transfer the existing set of knowledge objects and apps to the new search head instance:

a. Update the former standalone instance, if necessary, so that it is running the same version as the new search head instance.

b. Copy the contents of the standalone instance's `$SPLUNK_HOME/etc/apps` and `$SPLUNK_HOME/etc/users` directories to the same locations on the search head.

c. Restart the search head.

4. On the instance that you have chosen as your search head, configure the indexer instances as **search peers** to the instance. This step formally declares the respective roles (indexers, search head) for the instances. See "Add search peers to the search head" in the *Distributed Search* manual.

5. Configure the Splunk Enterprise license. If you have an existing license, ensure that it covers the new instances. See "How Splunk Enterprise licensing works" in the *Admin Manual*.

6. On each indexer, configure the **receiving port.** The forwarders send data to the indexer through this port. See Enable a receiver in the *Splunk Universal Forwarder* manual.

7. Install **universal forwarders** on the machines hosting your data sources, and configure the forwarders to send data to the set of indexers. How you do this depends on your needs and preferences, as well as on how many forwarders you are deploying. For example:

- You can install forwarders one-by-one or simultaneously across many machines.
- You can install forwarders and then configure them later, or install and configure forwarders at the same time.
- You can configure forwarders either manually or by means of the deployment server or third-party software.

For information on installing and configuring universal forwarders, see How to forward data to Splunk Enterprise in the *Universal Forwarder' manual.*

It is recommended that you configure the forwarders to load-balance their data across the set of indexers, rather than sending data to just a single indexer. See "Set up load balancing" in the *Forwarding Data* manual.

(Migration only) If you already have some forwarders from a previous deployment, reconfigure them to load-balance across all the indexers.

8. Configure the inputs to the forwarders, so that data begins to enter the system. See "Configure your inputs" in the *Getting Data In* manual.

Next steps

Once you have your Splunk Enterprise instances up and running and talking to each other, you can further refine your system and prepare the data and its presentation for the benefit of your end users. For a summary of the types of activities you need to perform now, see "Post-deployment activities."

To scale further

To increase your indexing and search capacity, the first step is to add more indexers. To do this, install another Splunk Enterprise instance, and configure the search head to treat it as a search peer.

To service more users and increase search capacity beyond a certain level, you must eventually add more search heads. Ordinarily, the best way to deploy multiple search heads is to deploy a search head cluster. See "Medium to large enterprise deployment: Search head cluster with multiple indexers."

For detailed information on determining when to add more instances, and whether to add only indexers or both indexers and search heads, see:

- "Types of deployments" in this manual.
- Summary of performance recommendations" in the *Capacity Planning* manual.

Medium to large enterprise deployment: Search head cluster with multiple indexers

In the transition from a small enterprise to a medium enterprise deployment, you need to boost both indexing and search capacity. For indexing, you can continue to add indexers. For search, you can add search heads to service more users and more search activity.

The recommended approach for deploying multiple search heads is to combine the search heads in a **search head cluster**. Search head clusters allow users and searches to share resources across the set of search heads. They are also easier to manage than groups of individual search heads. Search head clusters require a minimum of three search heads.

The differences between a medium and a large enterprise deployment are mainly issues of scale and management. The fundamental deployment topologies are similar. They both employ a search head cluster with multiple indexers.

This topic describes how to implement a deployment consisting of:

- One search head cluster, containing multiple search heads
- Multiple indexers
- Multiple forwarders

Note: Instead of deploying multiple individual indexers, you can deploy an **indexer cluster**. This topology can be useful even if you do not want high data availability and the accompanying storage overhead. It still entails a small amount of overhead, but it offers the benefit of simplified indexer management. See "High availability deployment: Indexer cluster."

Medium enterprise deployment use case

A medium enterprise deployment provides greater horizontal scaling than a small enterprise deployment. It services larger numbers of users and searches. As your needs continue to increase, you can continue to add indexers and search heads.

Characteristics of this type of deployment include:

- Indexing volume between 100-300GB/day.
- Users numbering possibly a hundred or more.
- Up to a few thousand forwarders feeding load-balanced data to the indexers.

For details on the characteristics of a medium enterprise deployment, see "Types of deployments."

Large enterprise deployment use case

A large enterprise deployment provides even greater horizontal scaling.

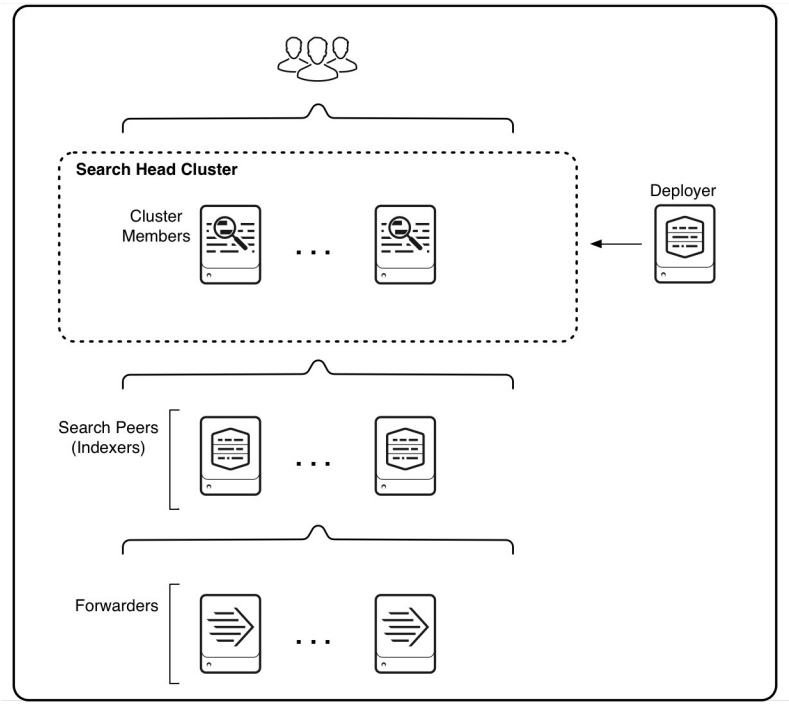
Characteristics of this type of deployment include:

- Indexing volume ranging from 300GB to many TBs per day.
- A large number of users, potentially numbering in the several hundreds.
- Many thousands of forwarders.

For details on the characteristics of a large enterprise deployment, see "Types of deployments."

Architecture

This diagram shows a high-level view of the architecture for a medium or large enterprise deployment:



Starting from the bottom, the diagram illustrates three tiers of processing:

- **Data input.** Data enters the system through **forwarders**, which consume external data and forward the data to the indexers. You configure the forwarders to use their built-in load-balancing capabilities to spread the data across the set of indexers.

- **Indexing. Indexers** receive, index, and store incoming data from the forwarders.

- **Search management.** A search head cluster, consisting of three or more search head members, performs the search management function. The search heads in the cluster coordinate their activities to handle search requests, such as ad hoc requests from users and saved search requests, and to distribute the requests across the set of indexers. A **deployer**

distributes apps to the members of the search head cluster.

Migration issues

This topic assumes that you are implementing a search head cluster deployment from scratch. If, instead, you are expanding from an existing, non-clustered deployment, the process is similar but there are a few additional issues to be aware of:

- You must use a new Splunk Enterprise instance for each search head in a cluster. You can migrate your current search head's settings and apps to a search head cluster, but you cannot reuse the search head itself. See "Migrate from a standalone search head to a search head cluster" in the *Distributed Search* manual.
- You must configure your licensing to accommodate the additional instances.

The implementation framework procedure covers these issues.

Implementation framework

To implement a search head cluster with indexers and forwarders:

1. Install and deploy the search head cluster. See "Deploy a search head cluster" in the *Distributed Search* manual.

2. (Migration only) If you are migrating from a smaller deployment consisting of a standalone search head, you can migrate the settings from the old search head. See "Migrate from a standalone search head to a search head cluster" in the *Distributed Search* manual.

3. Install the instances that you plan to use as indexers. For installation instructions, see "Installation overview" in the *Installation Manual.*

(Migration only) If you are migrating from a smaller deployment, you can continue to use the existing indexers. You can also add new indexer instances, as needed.

4. Connect the search heads to the indexers, also known as **search peers.** See "Connect the search heads in clusters to search peers" in the *Distributed Search* manual.

5. Configure the Splunk Enterprise license. If you have an existing license, you must ensure that it covers any new instances. See "How Splunk Enterprise licensing works" in the *Admin Manual*.

6. On each new indexer, configure the **receiving port.** The forwarders send data to the indexer through this port. See "Enable a receiver" in the *Forwarding Data* manual.

7. Install **universal forwarders** on the machines hosting your data sources, and configure the forwarders to send data to the set of indexers. For information on installing and configuring universal forwarders, see Install the universal forwarder software in the *Universal Forwarder* manual.

It is recommended that you configure the forwarders to load-balance their data across the set of indexers. See "Set up load balancing" in the *Forwarding Data* manual.

(Migration only) If you already have forwarders from a previous deployment, and you added indexers in step 3, reconfigure the existing forwarders to load-balance across the entire set of indexers, including the new indexers.

8. Configure the inputs to the forwarders, so that data begins to enter the system. See "Configure your inputs" in the *Getting Data In* manual.

Next steps

Once you have your Splunk Enterprise instances up and running and talking to each other, you can further refine your system and prepare the data and its presentation for the benefit of your end users. For a summary of the types of activities you need to perform now, see "Post-deployment activities."

To scale further

You can continue to scale your deployment as needed. To increase your indexing and search capacity, the first step is to add more indexers. To do this, install another Splunk Enterprise instance and configure the search heads and forwarders to connect with it.

To service more users and continue to increase search capacity beyond a certain level, you must add another search head to the cluster. See "Add a cluster member" in the *Distributed Search* manual.

For detailed information on determining when to add more instances, and whether to add only indexers or both indexers and search heads, see:

- "Types of deployments" in this manual.
- Summary of performance recommendations" in the *Capacity Planning* manual.

High availability deployment: Indexer cluster

To ensure high availability for your data, you can deploy an **indexer cluster**. Indexer clusters are groups of indexers configured to replicate each others' data, so that the system keeps multiple copies of all data. This process is known as **index replication**. Indexer clusters prevent data loss while promoting data availability for searching. Indexer clusters can also be simpler to manage than groups of individual indexers.

The trade-off with indexer clusters is that you need additional storage, to handle the replicated copies. You can control the degree of index replication, and thus the storage requirements, to match the availability needs of your enterprise.

For an introduction to indexer clustering, along with more details on the benefits and trade-offs, see "About indexer clusters and index replication" in the *Managing Indexers and Clusters of Indexers* manual.

You can use indexer clustering with an enterprise deployment of any size.

As your search needs grow, you can combine an indexer cluster with a **search head cluster**.

This topic describes how to implement a deployment consisting of:

- One or more individual search heads, or a search head cluster
- One indexer cluster
- Multiple forwarders

High availability use case

The main use case for an indexer cluster is an enterprise deployment that requires high data availability and is willing to allocate the additional disk space necessary to store multiple copies of the data.

Simplified management use case

You can also implement an indexer cluster without replication. When you eliminate replication, you lose some of the key benefits of an indexer cluster, such as data availability and data recovery, but you still gain the benefits of simplified management of multiple indexers. See "Use indexer clusters to scale indexing" in the *Managing Indexers and Clusters of Indexers* manual.

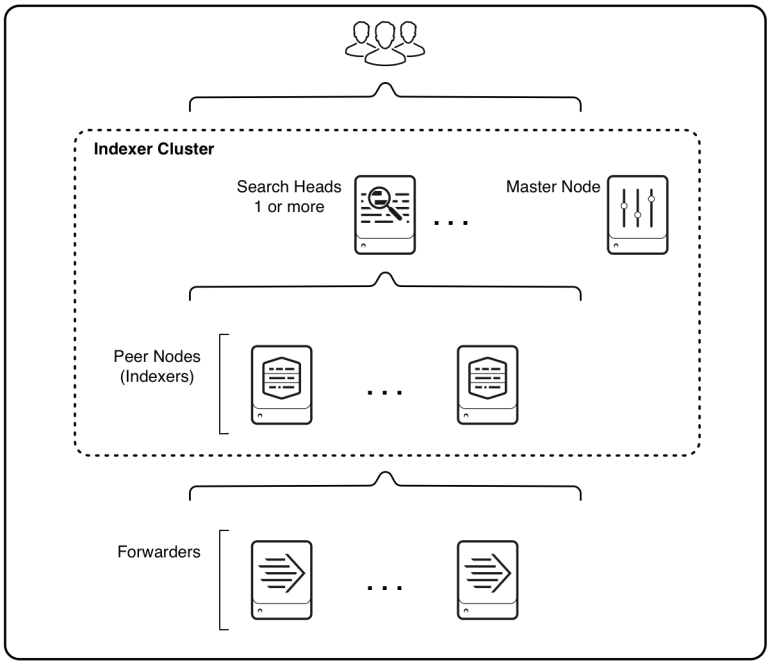
Architecture

The main types of architecture for an indexer cluster are:

- Indexer cluster with one or more individual search heads
- Indexer cluster with search head cluster

Indexer cluster with individual search heads

This diagram shows a high-level view of the architecture for an indexer cluster with one or more individual search heads:



Starting from the bottom, the diagram illustrates three tiers of processing:

- **Data input.** Data enters the system through forwarders, which consume external data and forward the data to the indexers. You configure the

forwarders to use their built-in load-balancing capabilities to spread the data across the set of indexers.

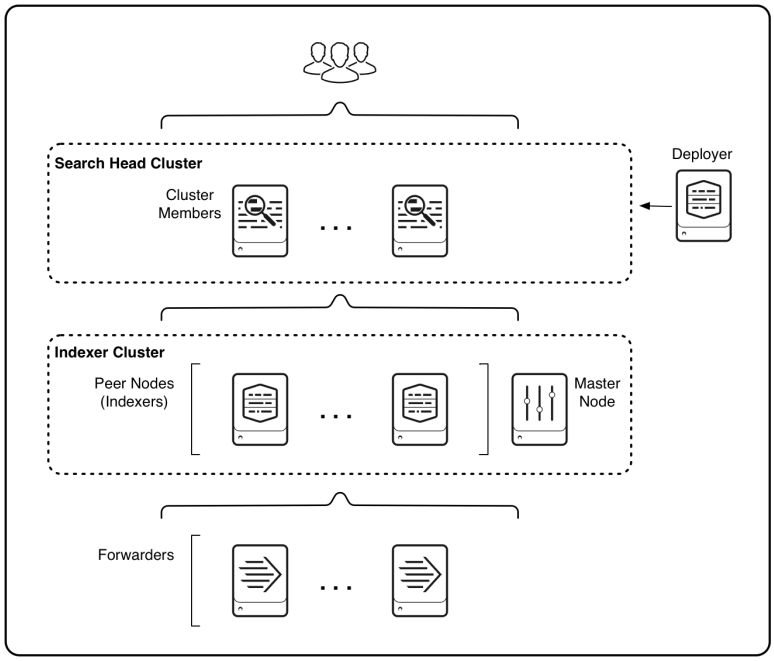
- **Indexing.** Indexers, or cluster **peer nodes**, receive, index, and store incoming data from the forwarders.

- **Search management.** One or more individual search heads perform the search management function.

A **master node** regulates the functioning of the cluster nodes, but does not index, store, or search data.

Indexer cluster with search head cluster

You can combine an indexer cluster with a search head cluster. This is the recommended approach if you need to deploy multiple search heads:



The main difference compared to the previous architecture is that a search head cluster replaces the individual search heads. The indexer cluster interacts with the search head cluster in essentially the same way that it would interact with individual search heads.

Migration issues

This topic assumes that you are implementing an indexer cluster deployment from scratch. If, instead, you are expanding from a set of non-clustered indexers, you can incorporate the existing indexers into your cluster, but there are a few issues to be aware of:

- Data already on the individual indexers remains available for searching, but it does not get replicated.
- You can no longer use the deployment server to manage your apps. Instead, you must use the **configuration bundle** method built into the indexer cluster. This requires that you migrate your apps to the cluster.

For details on these and other issues, see:

- "Key differences between clustered and non-clustered deployments of indexers" in the *Managing Indexers and Clusters of Indexers* manual.
- "Migrate non-clustered indexers to a clustered environment" in the *Managing Indexers and Clusters of Indexers* manual.

The implementation framework procedure mentions migration issues briefly.

Implementation framework

To implement an indexer cluster:

1. Install the necessary Splunk Enterprise instances, and deploy the indexer cluster. See "Indexer cluster deployment overview" in the *Managing Indexers and Clusters of Indexers* manual.

2. (Migration only) If you are migrating from a deployment consisting of one or more standalone indexers, also read "Migrate non-clustered indexers to a clustered environment" in the *Managing Indexers and Clusters of Indexers* manual.

3. If you want to implement a search head cluster for the search tier, see "Deploy a search head cluster" and "Integrate the search head cluster with an indexer cluster" in the *Distributed Search* manual.

4. Configure the Splunk Enterprise license. If you have an existing license, you must ensure that it covers any new instances. See "How Splunk Enterprise licensing works" in the *Admin Manual*. Also, see the section on licensing information in "System requirements and other deployment considerations for

indexer clusters" in the *Managing Indexers and Clusters of Indexers* manual.

5. Install **universal forwarders** on the machines hosting your data sources, if you have not already done so. See Install the universal forwarder software in the *Universal Forwarder* manual.

6. Connect the peer nodes to the forwarders:

> **a.** Decide on the method to use. See "Use forwarders to get data into the indexer cluster" in the *Managing Indexers and Clusters of Indexers* manual.
>
> **b.** Depending on your decision, follow either "Use indexer discovery to connect forwarders to peer nodes" or "Connect forwarders directly to peer nodes" in the *Managing Indexers and Clusters of Indexers* manual.

7. Configure the inputs to the forwarders, so that data begins to enter the system. See "Configure your inputs" in the *Getting Data In* manual.

Next steps

Once you have your Splunk Enterprise instances up and running and talking to each other, you can further refine your system and prepare the data and its presentation for the benefit of your end users. For a summary of the types of activities you need to perform now, see "Post-deployment activities."

To scale further

You can scale the cluster as needed. To increase indexing and search capacity, the first step is to add another indexer. To do this, enable a new Splunk Enterprise instance as a peer node.

To service more users and continue to increase search capacity beyond a certain level, you must add more search heads. The recommended way to include multiple search heads in an indexer cluster is to deploy them in a search head cluster. See "Migrate from a standalone search head to a search head cluster" in the *Distributed Search* manual.

To add more search heads to an existing search head cluster, see "Add a cluster member" in the *Distributed Search* manual.

For detailed information on determining when to add more instances, and whether to add only indexers or both indexers and search heads, see:

- "Types of deployments" in this manual.
- Summary of performance recommendations" in the *Capacity Planning* manual.

If your enterprise spans multiple sites, you can implement a **multisite indexer cluster** instead of a single-site cluster. See "Multisite indexer cluster deployment overview" in the *Managing Indexers and Clusters of Indexers* manual. If you already have a single-site cluster and want to convert it to a multisite cluster, see "Migrate an indexer cluster from single-site to multisite" in the *Managing Indexers and Clusters of Indexers* manual.

Administer your deployment

Post-deployment activities

Deployment implementation is the first step in a series of admininstration-related tasks that you must perform to take full advantage of Splunk Enterprise. This topic provides a broad outline of the typical post-deployment tasks, with links to the topics that cover these issues in detail.

Key manuals for a distributed deployment lists the manuals directly related to deployment. You have already encountered sections of these manuals during the deployment process. These same manuals cover post-deployment configuration and management issues. They will serve as an ongoing resource as you fine-tune your system, and you should familiarize yourself with their contents. In addition, other manuals provide guidance on improving and extending your system and fitting the system to the knowledge needs of your end users.

Do these next

These are some of the tasks that you should perform soon after you complete the initial deployment:

- **Set up users and roles.** See the chapter Users and role-based access control in *Securing Splunk Enterprise*.
- **Read about Splunk Enterprise security.** Look closely at the manual *Securing Splunk Enterprise*.
- **Forward the search heads' internal data to their search peers.** See Best practice: Forward search head data to the indexer layer in *Distributed Search*.

Increase the value of your deployment

Once your deployment is up and running and you have dealt with the basics, like security, you are ready to focus on your data: What data to ingest, how to ingest the data, and how to present the data so that your users can use it effectively.

Splunk Enterprise can handle virtually any kind of data. There is a lot to learn about the different types of data and how to configure them, including the important matters of source typing and event processing. For details on all matters related to data input, read *Getting Data In.* Be sure to study the material

on source typing, beginning with Why source types matter.

Next, you need to develop the searches, reports, dashboards, and so on, that make the data valuable and accessible to your users. These objects are collectively known as **knowledge objects**. The *Knowledge Manager Manual* is your primary resource for this.

Splunk offers a wide range of pre-built apps that can do most of this work for you. They define data inputs, source types, knowledge objects, and other configurations. They offer you and your users ready-made solutions to many common and uncommon needs. For example, there are apps that monitor the security of your system and other apps for IT operations management. To learn more about, and to download, pre-built apps, see Splunkbase.

You can also create your own apps. See dev.splunk.com for guidance on developing apps.

Resources for administering your deployment

The *Admin Manual* provides guidance on other important tasks. In particular, see Splunk administration: The big picture. It provides links to topics, across a variety of manuals, that describe key administration tasks.

The monitoring console provides a variety of dashboards that you can use to monitor most aspects of the deployment. See Monitor your distributed deployment in this manual. In addition, see *Monitoring Splunk Enterprise.*

For information on internal log files and other tools for troubleshooting your deployment, see the *Troubleshooting Manual.*

Distribute apps and other configurations to groups of instances

Splunk Enterprise provides the **deployment server** to distribute apps and other sets of configurations to groups of Splunk Enterprise instances. This tool is of particular value for managing configurations on forwarders, but it can distribute updates to any Splunk Enterprise instance, including indexers and search heads. See *Updating Splunk Enterprise Instances.*

To update the nodes on clusters, you do not use the deployment server. Instead, clusters use their own tools:

- In an indexer cluster, the cluster master distributes updates to peer nodes. See Update common peer configurations and apps in *Managing Indexers and Clusters of Indexers*.
- In a search head cluster, the deployer distributes updates to cluster members. See Use the deployer to distribute apps and configuration updates in *Distributed Search*.

You can also use third-party tools to distribute updates.

The rest of the Splunk universe

Splunk Enterprise is only one world in the Splunk universe. Other products include:

- **Splunk Cloud** for cloud-based access to the features of Splunk Enterprise.
- **Splunk Analytics for Hadoop** for data exploration, analysis and visualizations for Hadoop, NoSQL, and other data stores.
- **A variety of apps and add-ons** for extending the capabilities of Splunk Enterprise.

For more information, visit the Splunk documentation portal and the Splunk product overview.

Monitor your distributed deployment

You can use the monitoring console to monitor most aspects of your deployment. This topic describes the console dashboards that provide an overview of the entire deployment.

The primary documentation for the monitoring console is *Monitoring Splunk Enterprise*.

Deployment-wide dashboards

The monitoring console includes dashboards that provide an overview of the entire deployment, as well as others that drill down deeply into your deployment, focusing on specific features of the deployment, such as indexing or search head clustering. This topic describes the overview dashboards. The manuals that describe specific features cover the dashboards relevant to those features.

For example, for search head clusters alone, the monitoring console provides five dashboards that cover activities such as artifact replication, configuration replication, and app deployment. These dashboards are discussed in the documentation on search head clustering, in *Distributed Search*. Similarly, dashboards pertinent to indexer clusters or indexing performance are described in *Managing Indexers and Clusters of Indexers*.

There are three types of dashboards or pages that provide a deployment-wide view:

- Overview
- Instances
- Resource Usage

Overview dashboards

The dashboards that provide an overview of the deployment are located under the **Overview** menu. They are also the dashboards that appear when you initially start up the console. There are two dashboards:

- Overview
- Topology

You toggle between these dashboards by clicking on the **Overview** or **Topology** button.

The Overview dashboard specifies the number of indexers, search heads, cluster masters, and license masters. It also includes information on usage and alerts.

The Topology dashboard shows the instances for each component type, and the connections between indexers and search heads. It also provides some high-level information about each instance, such as the indexing rate for indexers and whether an instance is up or down.

Instances page

The Instances dashboard lists all Splunk Enterprise instances in your deployment. For each instance, it also provides information about its basic characteristics and status. You can access it through the **Instances** menu.

Resource Usage dashboards

There are several Resource Usage dashboards, which you access through the **Resource Usage** menu. The Resource Usage: Deployment dashboard provides deployment-wide resource information, such as CPU usage, physical memory usage, and disk usage. The other dashboards provide usage information by instance or machine.

See Resource usage: Deployment in *Monitoring Splunk Enterprise.*