Hortonworks DataFlow

Planning Your Deployment

(November 9, 2017)

docs.cloudera.com

Hortonworks DataFlow: Planning Your Deployment

Table of Contents

1. Deployment Scenarios

1.1. Possible Deployment Scenarios

Your particular deployment scenario for installing and configuring HDF components depends on your particular use case:

Use Case	Deployment Scenario	Steps
Installing HDF Services on a New HDP Cluster	This scenario applies to you if you are both an HDP and HDF customer and you want to install a fresh cluster of HDP and add HDF services. The stream processing components include the new Hortonworks Streaming Analytics Manager (SAM) and **all** of its modules. This includes installing the technical preview version of the SAM Stream Insight module, which is powered by Druid and Apache Superset. This requires that you install both an HDF cluster and an HDP cluster.	1. Install Ambari 2. Install Databases 3. Install HDP Cluster using Ambari 4. Install HDF Management Pack 5. Update HDF Base URL 6. Add HDF Services to HDP cluster
Installing an HDF Cluster	This scenario applies if you want to install the entire HDF platform, consisting of all flow management and stream processing components on a **new** cluster. The stream processing components include the new Streaming Analytics Manager (SAM) modules that are in **GA** (General Availability). This includes the the SAM Stream Builder and Stream Operations modules but **does not** include installing the technical preview version of SAM Stream Insight, which is powered by Druid and Superset. This requires that you install an HDF cluster.	1. Install Ambari 2. Install Databases 3. Install HDF Management Pack 4. Install HDF cluster using Ambari
Installing HDF Services on an Existing HDP Cluster	You have an **existing** HDP cluster with Apache Storm and or Apache Kafka services and want to install Apache NiFi or SAM modules on that cluster. This requires that you upgrade to the latest version of Apache Ambari and HDP, and then use Ambari to add HDF services to the upgraded HDP cluster.	1. Upgrade Ambari 2. Upgrade HDP 3. Install Databases 4. Install HDF Management Pack 5. Update HDF Base URL 6. Add HDF Services to HDP cluster
Performing any of the previous deployments by using a local repository See *Using Local Repositories* in the instructions appropriate for your scenario.	Local repositories are frequently used in enterprise clusters that have limited outbound internet access. In these scenarios, having packages available locally provides more governance and better installation performance. This requires that you perform several steps to create a local repository and prepare the Ambari repository configuration file.	1. Obtain the Public Repositories 2. Set Up the Local Repository 3. Prepare the Ambari Repository Configuration File

1.2. HDF Cluster Types and Recommendations

Cluster Type	Description	Number of VMs or Nodes	Node Specification	Network
Single VM HDF Sandbox	Evaluate HDF on local machine. Not recommended to deploy anything but simple applications.	1 VM	At least 4 GB RAM	
Evaluation Cluster	Evaluate HDF in a clustered environment. Used to evaluate HDF for simple data flows and streaming applications.	3 VMs or nodes	• 16 GB of RAM • 8 cores/vCores	
Small Development Cluster	Use this cluster in development environments.	6 VMs/Nodes	• 16 GB of RAM • 8 cores or vCores	
Medium QE Cluster	Use this cluster in QE environments.	8 VMs/Nodes	• 32 GB of RAM • 8 to16 cores or vCores	
Small Production Cluster	Use this cluster in small production environments.	15 VMs/Nodes	• 64 - 128 GB of RAM • 8 - 16 cores of RAM	1 GB Bonded Nic
Medium Production Cluster	Use this cluster in a medium production environment.	24 VMs/Nodes	• 64 - 128 GB of RAM • 8 - 16 cores of RAM	10 GB bonded network interface card (NIC)
Large Production Cluster	Use this cluster in a large production environment.	32 VMs/Nodes	• 64 - 128 GB of RAM • 16 cores of RAM	10 GB Bonded Nic

More Information

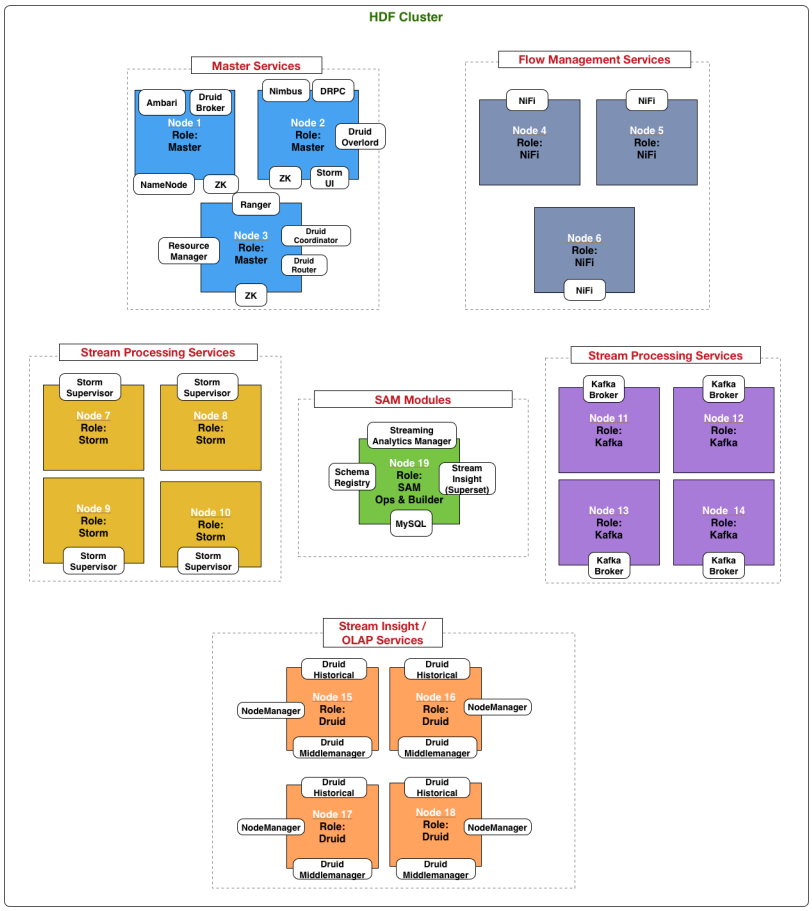
Download the Sandbox

1.3. Production Cluster Guidelines

General guidelines for production guidelines for service distribution:

- NiFi, Storm, and Kafka should not be located on the same node or virtual machine.
- NiFi, Storm, and Kafka must have a dedicated ZK cluster with at least three nodes.
- If the HDF SAM is being used in an HDP cluster, the SAM should not be installed on the same node as the Storm worker node.

The following diagram illustrates how services could be distributed for a small production cluster across 15 nodes:



1.4. Hardware Sizing Recommendations

Recommendations for Kafka

- Kafka Broker Node: eight cores, 64 GB to128 GB of RAM, two or more 8-TB SAS/SSD disks, and a 10-Gige Nic.
- Minimum of three Kafka broker nodes
- Hardware Profile: More RAM and faster speed disks are better; 10 Gige Nic is ideal.
- 75 MB/sec per node is a conservative estimate (can go much higher if more RAM and reduced lag between writing/reading and therefore 10GB Nic is required).

With a minimum of three nodes in your cluster, you can expect 225 MB/sec data transfer.

You can perform additional urther sizing by using the following formula:

num_brokers = desired_throughput (MB/sec) / 75

Recommendations for Storm

- Storm Worker Node: 8 core, 64 GB RAM, 1 Gige Nic
- Minimum of 3 Storm worker nodes

- Nimbus Node: Minimum 2 nimbus nodes, 4 core, 8 GB RAM
- Hardware profile: disk I/O is not that important; more cores are better.
- 50 MB/sec per node with low to moderate complexity topology reading from Kafka and no external lookups. Medium-complexity and high-complexity topologies might have reduced throughput.

With a minimum 2 nimbus, 2 worker cluster, you can expect to run 100 MB/sec of low to medium complexity topology.

Further sizing can be done as follows. Formula: num_worker_nodes = desired_throughput(MB/sec) / 50

Recommendations for NiFi

NiFi is designed to take advantage of:

- all the cores on a machine
- all the network capacity
- all the disk speed
- many gigabytes of RAM (although usually not all) on a system

Hence, it is important that NiFi be running on dedicated nodes. Following are the recommended server and sizing specifications for NiFi:

- Minimum of 3 nodes
- 8+ cores per node (more is better)
- 6+ disks per node (SSD or Spinning)
- At least 8 GB

If you want this sustained throughput...	Then provide this minimum hardware ...
50 MB and thousands of events per second	• 1 or 2 nodes • 8 or more cores per node, although more is better • 6 or more disks per node (solid state or spinning) • 2 GB memory per node • 1 GB bonded NICs
100 MB and tens of thousands of events per second	• 3 or 4 nodes • 8 or more cores per node, although more is better • 6 or more disks per node (solid state or spinning) • 2 GB of memory per node • 1 GB bonded NICs
200 MB and hundreds of thousands of events per second	• 5 to 7 nodes • 24 or more cores per node (effective CPUs)

If you want this sustained throughput...	Then provide this minimum hardware ...
	• 12 or more disks per node (solid state or spinning) • 4 GB of memory per node • 10 GB bonded NICs
400 to 500 MB/sec and hundreds of thousands of events per second	• 7 - 10 nodes • 24 or more cores per node (effective CPUs) • 12 or more disks per node (solid state or spinning) • 6 GB of memory per node • 10 GB bonded NICs

2. Navigating the HDF Library

To navigate the Hortonworks DataFlow (HDF) documentation library, begin by deciding your current goal.

If you want to...	See this document...
Install or upgrade an HDF cluster using Apache Ambari	• Release Notes • Support Matrices • Planning Your Deployment • Ambari Upgrade
Manually install or upgrade HDF components  **Note** This option is not available for Streaming Analytics Manager or Schema Registry.	• Command Line Installation • MiNiFi Java Agent Quick Start • Manual Upgrade
Get started with HDF	• Getting Started with Apache NFi • Getting Started with Stream Analytics
Use and administer HDF Flow Management capabilities	• Apache NiFi User Guide • Apache NiFi Administration Guide • Apache NiFi Developer Guide • Apache NiFi Expression Language Guide • MiNiFi Java Agent Administration Guide
Use and administer HDF Stream Analytics capabilities	• Streaming Analytics Manager User Guide • Schema Registry User Guide • Apache Storm Component Guide • Apache Kafka Component Guide