Cloudera Flow Management 4.0.0

Planning Your Flow Management cluster

Date published: 2019-06-26
Date modified: 2025-02-27

https://docs.cloudera.com/

Legal Notice

Contents

Data flow design

When planning how to size and configure Cloudera Flow Management clusters, it is important to keep in mind the flow design factors that might impact your cluster sizing needs and the performance of your data flow.

Note:

Flow design is the most important factor influencing the expected data flow performance, as well as the type and amount of resources used by NiFi.

You can use NiFi for a wide array of use cases, and the resource requirements are greatly determined by data flow design. Depending on the actions it performs in a data flow, each processor may or may not be required to read or to write the processed data from or on disks.

For example, a flow ingesting 100 MB of data per second with its first processor, may need to read and write this data on disks multiple times before the result is sent to the final destination. If the data flow has four processors writing the content on disks before being sent to the final destination, the disks used for the content repositories in the NiFi cluster should be able to handle 400 MB per second at the cluster level.

NiFi design principles

NiFi is designed to use all the available resources of the nodes where it is running. It takes advantage of:

- All available cores
- All network capacity
- All disk speed and capacities

Note:

It is crucial to understand that NiFi data flow behavior depends on three major factors:

- Data flow source
- Flow operations during data transfer
- Data flow target

For more information about principles involved in the NiFi design, see the *Apache NiFi Overview*.

Related Information

Apache NiFi Overview

Cluster layout

When planning how to size and configure Cloudera Flow Management clusters, it is important to keep in mind the flow design factors that might impact your cluster sizing needs and the performance of your data flow.

At least three nodes are recommended for a Flow Management cluster used in production environments. NiFi must be running on dedicated nodes.

Starting with Cloudera Flow Management 2.1.3, security has been improved in NiFi and it changed the internal authentication token handling to use HttpOnly Session Cookies. Cookies are specific to a hostname and path, but the browser does not differentiate between different ports on the same host. In case multiple NiFi instances are running on the same node, but listening on different ports, it is required to configure an HTTPS load balancer and setup different context paths for each NiFi instance.

Important: Running multiple NiFi instances on the same node is not recommended by Cloudera.

When NiFi is clustered, you should use an external ZooKeeper cluster. In production deployments, the ZooKeeper instances should not be co-located with the NiFi instances.

Disk configuration

When planning how to size and configure Cloudera Flow Management clusters, it is important to keep in mind the flow design factors that might impact your cluster sizing needs and the performance of your data flow.

For most modern systems, the disk throughput is lower than the network throughput, so the network is usually not a bottleneck. For most data movement use cases, the CPU usage is much lower than the disk I/O, but it is still important to monitor the CPU and tune the number of threads per processor. See *Tuning your Data Flow* for recommendations about fine tuning threads usage.

NiFi has three repositories on disk and the disk configuration is a very important performance factor:

Content Repository

- Contains the content of each FlowFile
- Sequential disk I/O (ideally leveraging the OS cache)

FlowFile (metadata) Repository

- Contains the FlowFile attributes and current FlowFile state (which queue it is in) for each FlowFile
- Sequential and random disk I/O

Provenance (metadata) Repository

- Contains a provenance log with entries for every action performed on a FlowFile (merge, drop, and so on)
- Sequential and random disk I/O

Every FlowFile that NiFi receives or creates is immediately written to disk in the content repository for fault tolerance. Subsequent FlowFile content modifications (decompression, format conversion, and so on) are also written to the content repository. Processors that do not modify the content, such as `RouteOnAttribute`, do not impact the content repository. Instead, the FlowFile repository keeps a pointer for each FlowFile showing its state, such as which queue it is located in. This optimization eliminates the need for redundant writes to the content repository.

For higher performance, configure multiple disks for both the content and provenance repositories.

For more information, see *Configuration Best Practices*, *File System Content Repository Properties*, and *Write Ahead Provenance Repository Properties* in the NiFi System Properties documentation.

Related Information

File System Content Repository Properties

Write Ahead Provenance Repository Properties

Configuration Best Practices

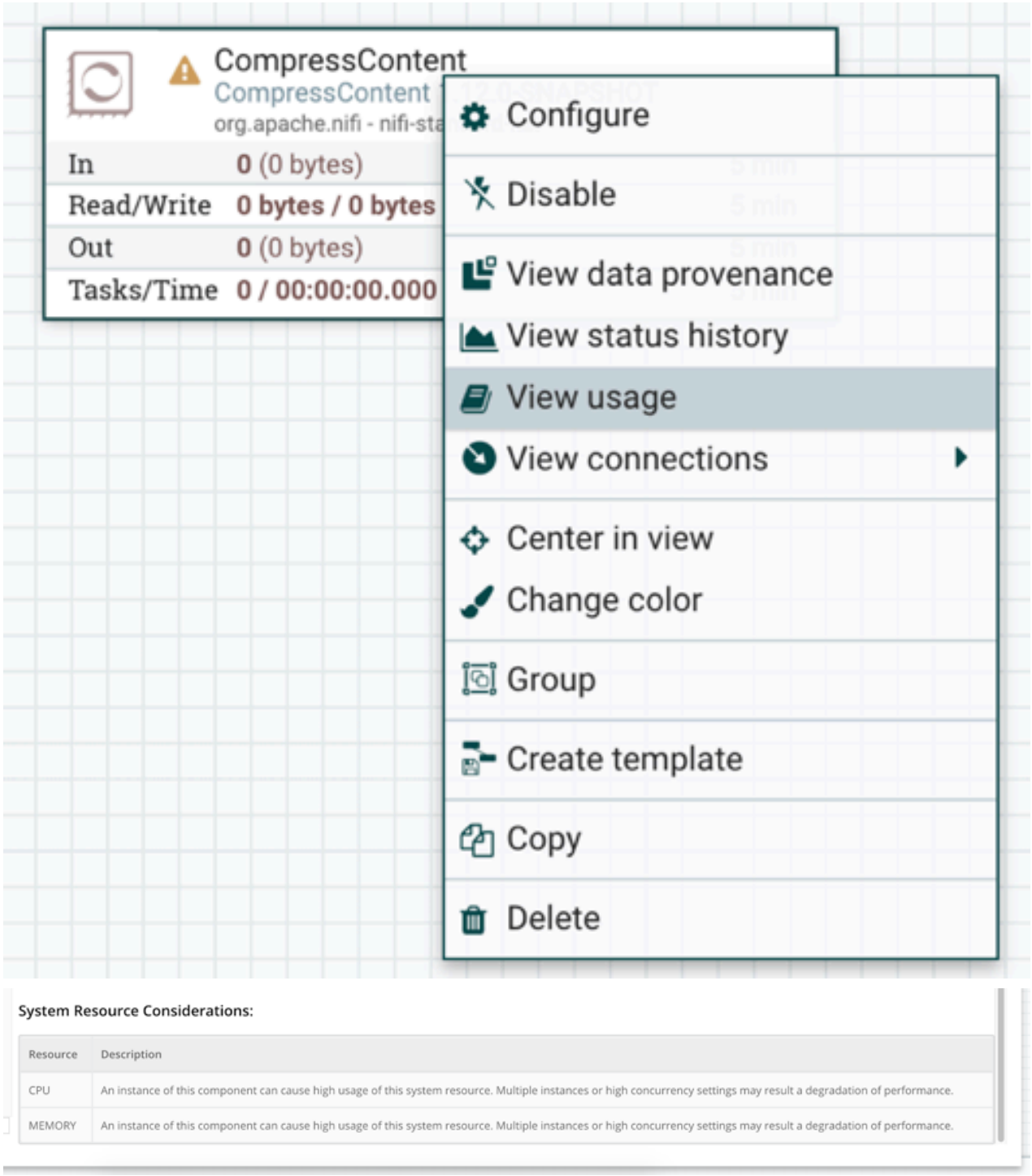
Tuning your Data Flow

Resource-intensive processors

Learn how to identify resource-intensive processors.

You can check in the *Apache NiFi documentation* whether a processor is CPU intensive or memory intensive and use this information when planning your data flows.

For example, if you right-click the `CompressContent` processor to open its documentation, you can see that this processor can be both memory and CPU intensive.



Related Information

Apache NiFi documentation

Sizing recommendations

Properly sizing your Cloudera Flow Management clusters is crucial for optimal performance. Follow these recommendations to configure your environment effectively.

On-premises (bare metal) installations

Cloudera recommends the following disk setup for bare metal environments:

- 1 RAID 1 or 10 array for the operating system (OS)
- 1 RAID 1 or 10 array for the FlowFile repository
- 1 or many RAID 1 or 10 array(s) for the content repository
- 1 or many RAID 1 or 10 array(s) for the provenance repository

For high performance setup, Cloudera recommends SSDs over spinning disks.

Cloud environments

For cloud deployments, larger disks typically offer better throughput. Review your cloud provider's documentation for specific details and best practices.

Memory considerations

NiFi efficiently processes FlowFiles of any size by avoiding direct memory materialization. Instead, NiFi uses input and output streams to process events (there are a few exceptions with some specific processors). This means that NiFi does not require significant memory even if it is processing very large files. Most system memory should remain available for the OS cache, enabling disk read optimizations. By having a large enough OS cache, many of the disk reads are skipped completely. So unless NiFi is used for very specific memory oriented data flows, setting the Java heap to 8 GB or 16 GB is usually sufficient.

Performance and scalability

The performance you can expect directly depends on the hardware and the flow design. For example, when reading compressed data from a cloud object store, decompressing the data, filtering it based on specific values, compressing the filtered data, and sending it to a cloud object store, you can achieve the following results:

Nodes	Data rate per second	Events per second	Data rate per day	Events per day
1	192.5 MB	946,000	16.6 TB	81.7 billion
5	881 MB	4.97 million	76 TB	429.4 billion
25	5.8 GB	26 million	501 TB	2.25 trillion
100	22 GB	90 million	1.9 PB	7.8 trillion
150	32.6 GB	141.3 million	2.75 PB	12.2 trillion

These metrics were collected on Google Kubernetes Engine (GKE) with each node configured with 32 cores, 15 GB RAM, and a 2 GB heap. The content repository used a 1 TB Persistent SSD (400 MB/s write, 1200 MB/s read).

NiFi supports both vertical and horizontal scaling. Depending on the number of data flows running in the NiFi cluster and your operational requirements, you can add nodes to the NiFi cluster over time to meet your needs.

With this information in mind, Cloudera recommends:

- CPU: At least 4 cores per NiFi node (8 cores are preferred for common use cases).
- Disks: At least 6 disks per NiFi node to dedicate separate disks for repositories.
- Memory: At least 4 GB of RAM for the NiFi heap.

For more details and a use case example, see *Processing one billion events per second with NiFi*.

Related Information

Processing one billion events per second with Apache NiFi