Getting Started

Date of Publish: 2018-07-03

http://docs.hortonworks.com

Contents

DLM Concepts

Data Lifecycle Manager terminology

Data Lifecycle Manager (DLM) is a UI service that is enabled through DPS Platform. From the DLM UI you can create and manage replication and disaster recovery policies and jobs.

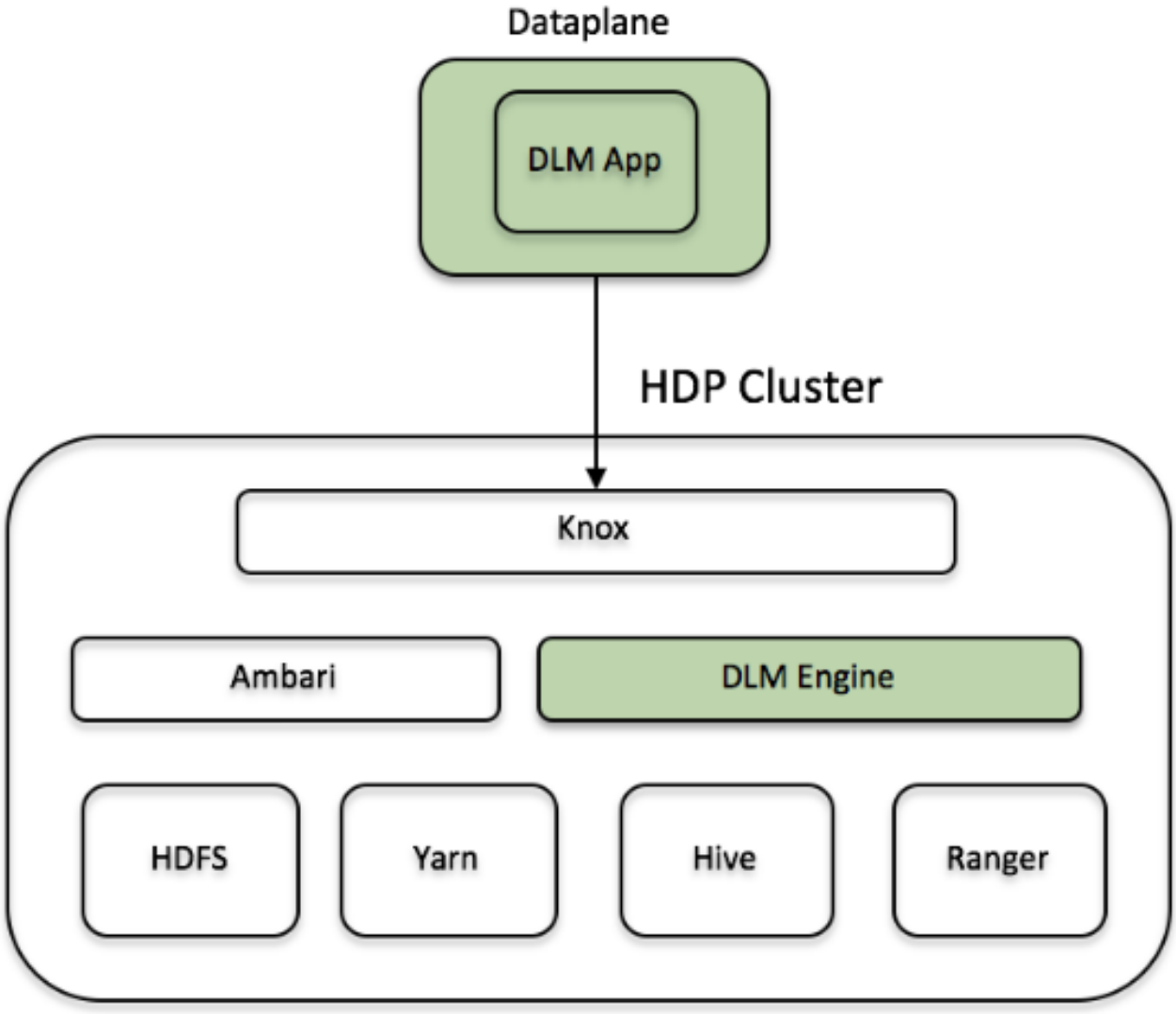
DLM App or Service	The web UI that runs on the DPS platform host. The corresponding agent needs to be installed on the clusters.
DLM Engine	The agent required for DLM. Also referred to as the Beacon engine, this replication engine must be installed as a management pack on each cluster to be used in data replication jobs. The engine maintains, in a configured database, information about clusters and policies that are involved in replication.
data center	The facility that contains the computer, server, and storage systems and associated infrastructure, such as routers, switches, and so forth. Corporate data is stored, managed, and distributed from the data center. In an on-premise environment, a data center is often composed of a single HDP cluster. However, a single data center can contain multiple HDP clusters.
IaaS cluster	A full HDP cluster on cloud VMs with Apache services running, such as HDFS, YARN, Ambari, Hiveserver2, Ranger, Atlas, and DLM Engine. Replication behavior is similar to on-premise cluster replication. The data is on local HDFS.
cloud data lake or data lake	An HDP cluster on the cloud, using VMs, with data retained on cloud storage. A cloud data lake requires minimal services for metadata and governance, such as Hive metastore, Ranger, Atlas, and DLM Engine. The data is on the cloud.
cloud storage	Any storage retained in a cloud account, such as Amazon S3 web service.
on-premise cluster	A full HDP cluster in a data center, with Apache services running, such as HDFS, Yarn, HMS, hiveserver2, Ranger, Atlas and Beacon. Replication behavior is similar to IaaS cluster replication. The data is on local HDFS.
policy	A set of rules applied to a replication relationship. The rules include which clusters serve as source and destination, the type of data to replicate, the schedule for replicating data, and so on.
job	An instance of a policy that is running or has run.

source cluster	The cluster that contains the source data that will be replicated to a destination cluster. Source data could be an HDFS dataset or a Hive database.
destination cluster	The cluster to which an HDFS dataset or Hive database is replicated.
target	The path on the destination cluster to which the HDFS dataset or Hive database is replicated.

Communication with HDP clusters

DPS Platform and the DLM App communicate with the HDP cluster through Knox. Knox SSO is a required configuration for the DPS Platform host.

DLM replication also requires HDFS, YARN, Hive, and Ranger on the cluster. Knox Gateway is recommended to protect data being transferred between clusters.



How pairing works in Data Lifecycle Manager

Pairing verifies compatibility and establishes communication between the clusters that you want to use as source or destination clusters in a replication relationship. Pairings are bi-directional, so either cluster in a pair can serve as the source or as the destination in a replication policy.

After pairing is complete, you can create a replication policy between the paired clusters, and you can establish through the policy which cluster is the source and which is the destination.

Pairings can only be performed on clusters that have been registered with Data Lifecycle Manager. If a cluster you want to use is not visible, you need to register it from the DPS Platform UI, logged in as DataPlane Admin.

How policies work in Data Lifecycle Manager

In Data Lifecycle Manager, you create policies to establish the rules you want applied to your replication and disaster recovery jobs. The policy rules you set can include which cluster is the source and which the destination, what data is replicated, what day and time the replication job occurs, the frequency of job execution, bandwidth restrictions, etc.

When scheduling how often you want a replication job to run, you should consider the recovery point objective (RPO) of the data being replicated; that is, what is the acceptable lag time between the active site and the replicated data on the destination. Data Lifecycle Manager supports a one-hour RPO: data is preserved up to one hour prior to the point of data recovery. To meet a one-hour RPO, you must consider how long it takes to replicate the selected data, how often the data is replicated, and network bandwidth capabilities.

As an example, if you have a set of data that you expect to take 15 minutes to replicate, then to meet a one-hour RPO, you would schedule the replication job to occur no more often than every 45 minutes, depending on network bandwidth.

Snapshot replication between HDP clusters

You can optionally enable HDFS snapshots for replication in Data Lifecycle Manager. Understanding how snapshots work, and some of the benefits and costs involved, can help you to decide whether or not to enable snapshot replication.

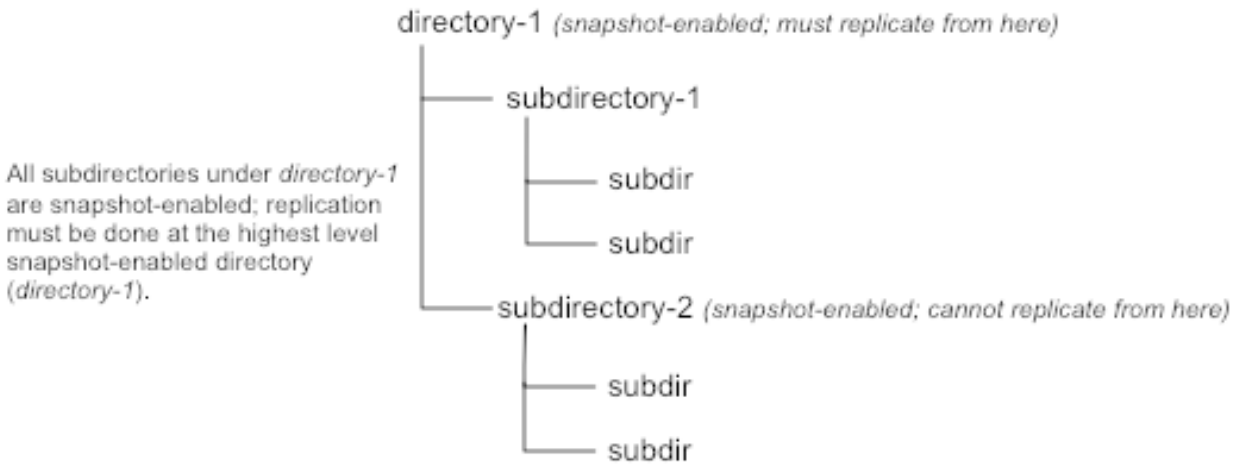
Understanding HDFS Snapshots

HDFS snapshots are read-only point-in-time copies of the filesystem. You can enable snapshots on the entire filesystem, or on a subtree of the filesystem. For DLM, you enable snapshots at a dataset level.

Enabling snapshots on a folder requires HDFS admin permissions, because it impacts the NameNode. When you enable snapshots, all subdirectories are automatically enabled for snapshots as well. So when you create a snapshot copy of a directory, all content in that directory, including subdirectories, is included as part of the copy. If a directory contains snapshots but the directory is no longer snapshot-enabled, you must delete the snapshots prior to enabling the snapshot capability on the directory.

Snapshots must be taken on the highest-level parent directory that is snapshot-enabled. Snapshot operations are not allowed on a directory if one of its parent directories is already snapshot-enabled (snapshottable) or if descendants already contain snapshots. If a directory contains snapshots but the directory is no longer snapshot-enabled, you must delete the snapshots prior to enabling the snapshot capability on the directory.

For example, in the directory tree image below, if directory-1 is snapshot-enabled but you want to replicate subdirectory-2, you cannot select only subdirectory-2 for replication. You must select directory-1 for your replication policy.



There is no limit to the number of snapshot-enabled directories you can have. A snapshot-enabled directory can accommodate 65,536 simultaneous snapshots.

Blocks in datanodes are not copied during snapshot replication. The snapshot files record the block list and the file size. There is no data copying.

When snapshots are initially created, a directory named .snapshot is created on the source and destination clusters, under the directory being copied. All snapshots are retained within .snapshot directories. By default, the last three snapshots of a file or directory are retained. Snapshots older than the last three are automatically deleted.

Benefits of snapshots

Snapshot-based replication helps you to avoid unnecessarily copying renamed files and directories. If a large directory is renamed on the source side, a regular DistCp update operation sees the renamed directory as a new one and copies the entire directory.

Generating copy lists during incremental synchronization is more efficient with snapshots than using a regular DistCp update, which can take a long time to scan the whole directory and detect identical files. And because snapshots are read-only point-in-time copies between the source and destination, modification of source files during replication is not an issue, as it can be using other replication methods.

A snapshot cannot be modified. This protects the data against accidental or intentional modification, which is helpful in governance and in meeting disaster recovery (DR) requirements.

Considerations for using snapshots

There is a memory cost to enabling and maintaining snapshots. Tracking the modifications that are made relative to a snapshot increases the memory footprint on the NameNode and can therefore stress NameNode memory.

Because of the additional memory requirements, snapshot replication is recommended for situations in which it is most useful. Such circumstance might include: if you expect to do a lot of directory renaming, if the directory tree is very large, or if you expect changes to be made to source files while replication jobs execute.

Requirements for snapshot-based replication

You must have HDFS superuser privilege to enable or disable snapshot operations.

Replication using snapshots requires that the target filesystem data being replicated is identical to the source data for a given snapshot. There must not be any modification to the data on the target. Otherwise, the integrity of the snapshot cannot be guaranteed on the target and replication can fail in various ways.

You must delete any existing snapshot copies in a directory before you can re-enable the snapshot capability on the directory.

Understanding the UI

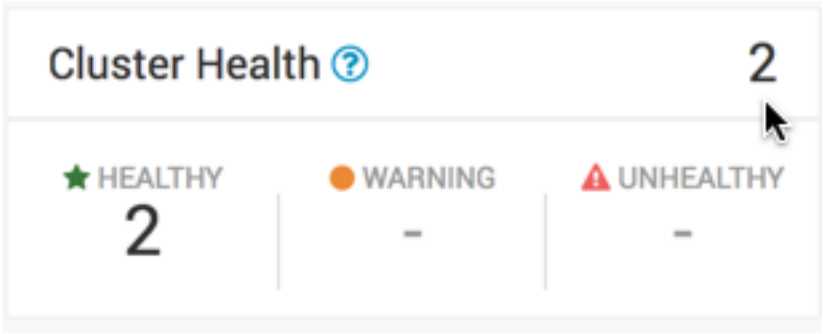
Overview Page

From the Overview page you can quickly identify any issues with, or verify the health of, the clusters, policies, or jobs in Data Lifecycle Manager (DLM).

Cluster Health panel

You can use the Cluster Health panel of the Overview page to view the total number of clusters enabled for Data Lifecycle Manager, the number that are healthy, the number for which a warning is issued, and the number that are unhealthy.

You can investigate the issues associated with clusters that have a warning or unhealthy status by navigating to the Ambari web UI.



Healthy	Specifies the total number of clusters currently available to run replication jobs. The DLM Engine can be reached and all services are running.
Warning	Specifies the total number of clusters for which remaining disk capacity is less than 10%. If this value is greater than zero, you can click the number to open a table that displays the cluster name and remaining capacity.
Unhealthy	Specifies the total number of clusters for which at least one Apache Ambari service required for DLM (DLM Engine, HDFS, Apache Hive, or Apache Knox) is not started. If this value is greater than zero, you can click the number to open a table that displays the cluster name and the names of any Ambari services that have stopped.

Policies panel

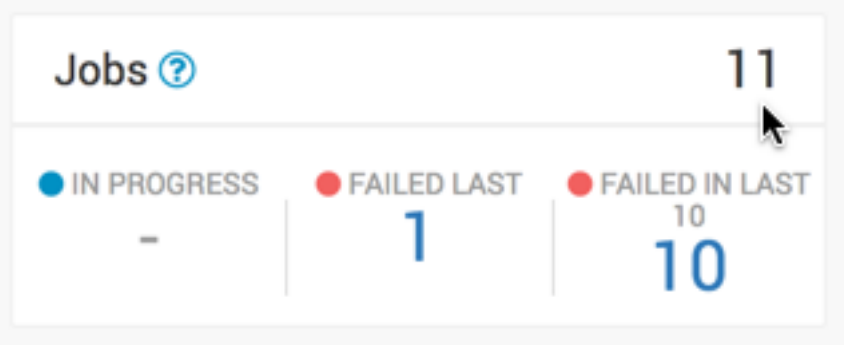
You can use the Policies panel of the Overview page to view the total number of policies in use and their status.

Active	Specifies policies with status of Submitted or Running. This item is not actionable
Suspended	Specifies policies that have been suspended by an administrator. This item is not actionable.
Unhealthy	Specifies policies associated with any cluster designated as Unhealthy in the Cluster Health panel. If the value is greater than zero, the number becomes clickable. You can click the number to display a table that contains the policy name, the names of the source and destination

clusters, and which services are stopped on the source or destination cluster.

Jobs panel

You can use the Jobs panel of the Overview page to view the total number of running and failed jobs and their status.



In Progress	Specifies the number of jobs with the status Running. If the value is greater than zero, the number becomes clickable. You can click the number to apply a filter to the Issues and Updates table, so that the table displays only in-progress jobs. The filter label Jobs: In Progress appears above the table. Running jobs display as a blue dot in the Policy History column of the table.
Failed Last	Specifies the last job that completed with status Failed. If the value is greater than zero, the number becomes clickable. You can click the number to apply a filter to the Issues and Updates table, so that the table displays only policies for which the last job had a status of Failed. The filter label Jobs: Failed Last appears above the table. Failed jobs display as a red dot in the Policy History column of the table.
Failed in Last 10	Indicates the number of policies for which at least one of the last 10 jobs completed with status Failed. If the value is greater than zero, the number becomes clickable. You can click the number to apply a filter to the Issues and Updates table, so that the table displays only policies for which at least one job failed out of the last 10 jobs. The filter label Jobs: Failed Last appears above the table. Failed jobs display as a red dot in the Policy History column of the table.

Recent Issues panel

This panel shows the last four events with severity of warning, critical, or error. For each event, the panel shows the severity, the type, a message that includes the policy name and file icon, and the age of the event.

Severity icon	Displays in orange for warning and red for critical or error.
Event type	Displays in bold text above the event message. The type can be succeeded, deleted, or suspended.
Event message	Displays as text under the event name. When an event is associated with a policy or policy instance (job), then the message text contains two items:

	• Policy name: You can click this term to navigate to the Policies page with a preset filter that displays information about only the selected policy. • File icon: You can move the cursor over the icon to display the text “View Log” and click the text to display log content for the associated policy or job.
Event age	Displays in numeric form how long ago from the current time the event occurred.

You can click View All at the bottom of the event list to navigate the browser to the Notifications page.

Clusters map

The Clusters map indicates the geolocation of each cluster, using red, orange, and green markers on the map.

The colored markers indicate the following:

- Red: At least one required service has stopped on the cluster.
- Orange: All required services are running but the remaining capacity on a cluster is less than 10%.
- Green: All required services are running and remaining disk capacity is greater than 10%.

You can move the cursor over a marker on the map displays a tooltip specifying the data center associated with the cluster, the cluster name, and the number of DLM policies that are associated with that cluster.

You can click a marker to open a panel showing the same information as in the tooltip, plus a Launch Ambari link. Clicking the link opens a new browser tab with the login page for the Ambari host for that cluster.

If the dot is red, the panel also displays a list of services that are in a Stopped state in Ambari.

Issues & Updates table

The Issues & Updates table shows policies that have running jobs but at least one failed out of the most recent 10 jobs. You do not see any policy if its last 10 jobs were all successful.

Table columns include the following:

Job Status	When the status of a job is Running, a status circle icon and progress bar display. For jobs that are not running, a status circle icon displays along with the text Success, Failed, or Ignored. You can move the cursor over a Failed status to see a “View Log” tooltip, which you can click to see the job log.
Source & Destination	The names of the source and destination clusters associated with the policy.
Service	Indicates whether the data being replicated is from HDFS or Hive.
Policy	The name assigned to the policy.
Policy History	Shows up to 10 job statuses as colored dots.

Color	Status	Description
Green	Succeeded	Job completed with no issues.
Red	Failed	Job did not complete.
Gray	Ignored	Job did not start because a previous ins Only one run of a job can be in progres ignored, you might need to modify its f

	Clicking the colored dots navigates the browser to the Policies page, with the filter preset to show information only about the specified policy.
Transferred/Files	The amount of data transferred, in gigabytes, and the number of objects transferred, if available. When a job is running, the column displays In Progress.
Runtime	How long it took to run the most recent job.
Started	When the most recent job started.
Ended	When the most recent job ended.
Actions icon	• Abort Job: Aborts a running job. Enabled only when the job status is Running. • Re-run Job: Starts another instance of the policy. Disabled when a job is executing. • Edit Policy: Allows editing of some policy settings. Disabled if a policy is expired. • Delete Policy: Removes a policy from Data Lifecycle Manager. Delete cannot be undone. Always enabled. • Suspend Policy: Suspends the policy and any job that is executing. Disabled when the policy status is Suspended. • Activate Policy: Resumes a suspended policy. Disabled when the policy status is Running.

Planning for a DLM Installation

General DLM requirements and recommendations

Understanding the requirements and recommendations indicated below can help to avoid common issues during and after installation of the DLM service.

- Be sure to review the Platform Support matrix to confirm you meet the environment and system requirements including Docker and networking.
- You need to have root access to the nodes on which the DLM App and DLM Engine will be installed.
- Apache Hive should be installed during initial installation, unless you are certain you will not use Hive replication in the future.

  If you decide to install Hive after creating HDFS replication policies in Data Lifecycle Manager, all HDFS replication policies must be deleted and then recreated after adding Hive.
- Clusters used in DLM replication must have symmetrical configurations.

  That is, each cluster in a replication relationship must be configured exactly the same for Kerberos, LDAP, High Availability (HA), Apache Ranger, and so forth.

Cluster security requirements for DLM-enabled clusters

You must configure a minimum set of security actions on each HDP cluster as part of configuring security for DLM-enabled clusters. You can perform any additional security-related tasks as appropriate for your environment and company policies. You must also have completed the security configuration requirements for clusters used with DPS.

If you will be performing Hive replication with the Data Lifecycle Manager (DLM) service, the following tasks must be completed during cluster installation, prior to configuring Hive.

Table 1: Minimum Security Requirements Checklist for DLM

Task	Comments	Instructions
Configure LDAP for Ambari		Configuring Ranger Authentication with UNIX, LDAP, or AD
Configure clusters to point Knox to LDAP		Configuring Ranger Authentication with UNIX, LDAP, or AD
Configure LDAP with Ranger		Configuring Ranger Authentication with UNIX, LDAP, or AD
Configure user synchronization for policy administration		Configure Ranger User Sync
Configure Ranger plugins for Knox		Enabling Ranger Plugins: HDFS, YARN, Hive, Knox
Configure Ranger plugins for Kerberos		Ranger Plugins--Kerberos: HDFS, Hive, Knox
Configure Knox SSO for Ambari		HDP Security Guide, Setting up Knox SSO for Ambari
Configure Knox SSO for Ranger		Setting up Knox SSO for Ranger
Configure Knox SSO for DLM Engine	Perform this task only after installing DLM Engine	See the DLM installation instructions
Configure Knox Gateway for proxying	Only required if using Knox proxying; proxying required for wire encryption	Perimeter Security with Apache Knox