Deployment Guide

Installing Genesys Data Processing Server

12/18/2025

Installing Genesys Data Processing Server

Genesys Data Processing Server processes the complex, high-volume data produced by select Genesys products for a variety of uses. When used for reporting, it provides the users of these products with the ability to transform and aggregate this data, and perform sophisticated calculations on it. The resulting metrics are used for reporting and visualization within Genesys Pulse.

The following instructions describe how to install Data Processing Server.

Note: Many of the procedures involved in configuring the Genesys Data Processing Server are similar to the ones that are used to configure other application clusters, such as Web Engagement Server. The description of how to Install the Genesys Web Engagement Server contains information that may be worth consulting while you are configuring Data Processing Server.

Important: Genesys recommends that you use a dedicated Cassandra data center for reporting purposes. This will minimize the risk of Cassandra-related faults in operational environments that are running under a heavy load.

Important: Genesys recommends that you put each Cassandra node for your reporting data center on the same host as a Data Processing Server node. In other words, the number of nodes in Cassandra reporting data center should be equal to the number of nodes in the Data Processing Server cluster. This will speed up calculations on your analytical data.

Before you begin

Because Genesys Data Processing Server must process a lot of data, it needs to run on top of a high-speed—and highly scalable—cluster computing system. Genesys has chosen Apache Spark for this task.

Spark supports several types of clustering. Fortunately, Data Processing Server works well with the simplest one, Spark Standalone Mode. This mode provides high availability by using a dedicated master node. A typical cluster deployment will consist of one master node and several worker nodes and is usually started by Data Processing Server in the background.

Genesys recommends that you configure Data Processing Server in Genesys Administrator, defining a single Application Cluster object and the appropriate number of individual node objects. You can configure these objects and their options using the procedures provided on the rest of this page.

Recommended versions

Genesys recommends that you use specific Data Processing Server versions with your products. For more information, consult the appropriate page for your product:

- Genesys Web Engagement

Data Processing Server nodes

As we just mentioned, the Spark cluster consists of one master node and several worker nodes. Any Data Processing Server node can be the master node in the cluster, as this role is defined by the value of the Spark startMode option in the node's configuration options. Here is more information about the two types of node:

- The **Master** node is represented by a Spark startMode value of `both`, which indicates that both a Spark Master and a Spark Worker instance will be started on the node. Please note that the Spark host option should be set in agreement with startMode so that the hostname used in the Spark host setting belongs to the node that runs the Spark Master instance. To avoid problems with connectivity inside the Spark cluster, this hostname should be the primary one in the network configuration for this host. In other words, Java's **InetAddress.getLocalHost** should return the same value for the Data Processing Server Master node.
- The **Worker** node is represented by a startMode value of `worker`. Only a Spark worker instance will be launched at this node.

There is one additional mode available for the Spark startMode option. A mode of `off` means that no Spark processes will be launched on this host and that the role of the node in the Data Processing Server cluster is undefined. This mode is for use in situations where you want to have an externally managed Spark cluster, and limits you to one Data Processing Server node, which serves as an entry point for the Spark cluster. You cannot deploy Data Processing Server with multiple nodes if you have set **startMode** to `off`. Also, if you use this mode, you must have an advanced understanding of how to work with and manage a Spark cluster.

Configuring Data Processing Server

We have included information about Data Processing Server–related configuration options at the end of this page.

Deploying Data Processing Server

To deploy Data Processing Server, follow these steps:

1. Importing the Data Processing Server cluster template
2. Creating the cluster application
3. Configuring the cluster application
4. Importing the Data Processing Server template
5. Creating a node application
6. Configuring a node application
7. Tuning analytical Cassandra nodes to skip indexing
8. Adding nodes to a cluster

9. Updating your product's cluster application
10. Data Processing Server data storage
11. Installing Data Processing Server
12. Installing Dashboards and Widgets into Pulse
13. Deploying and Scheduling Job Packages
14. Data Processing Server data flow

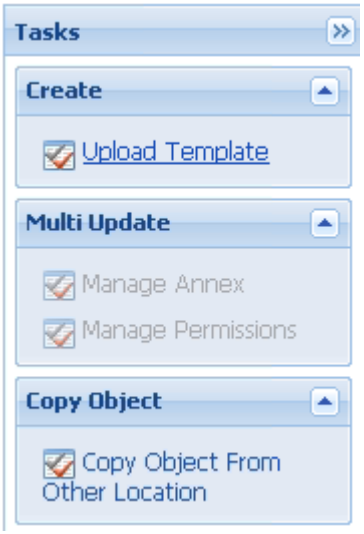
Note: For more information on how to work with templates and application objects in Genesys Administrator, consult Generic Configuration Procedures.

Importing the Data Processing Server cluster template

Note: For more information on how to work with templates in Genesys Administrator, consult Generic Configuration Procedures.

Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Application Templates**.
2. In the **Tasks** panel, click **Upload Template**.



Upload Template link in the Tasks panel

3. In the **Click 'Add' and choose application template (APD) file to import** window, click **Add**.
4. Browse to the **Data_Processing_Cluster.apd** file. The **New Application Template** panel opens.
5. Click **Save & Close**.

End

Creating the cluster application

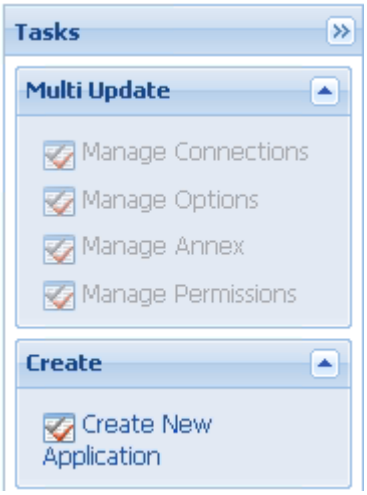
Note: For more information on how to work with application objects in Genesys Administrator, consult Generic Configuration Procedures.

Prerequisites

- You completed Importing the Data Processing Server cluster template.

Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Applications**.
2. In the **Tasks** panel, click **Create New Application**.

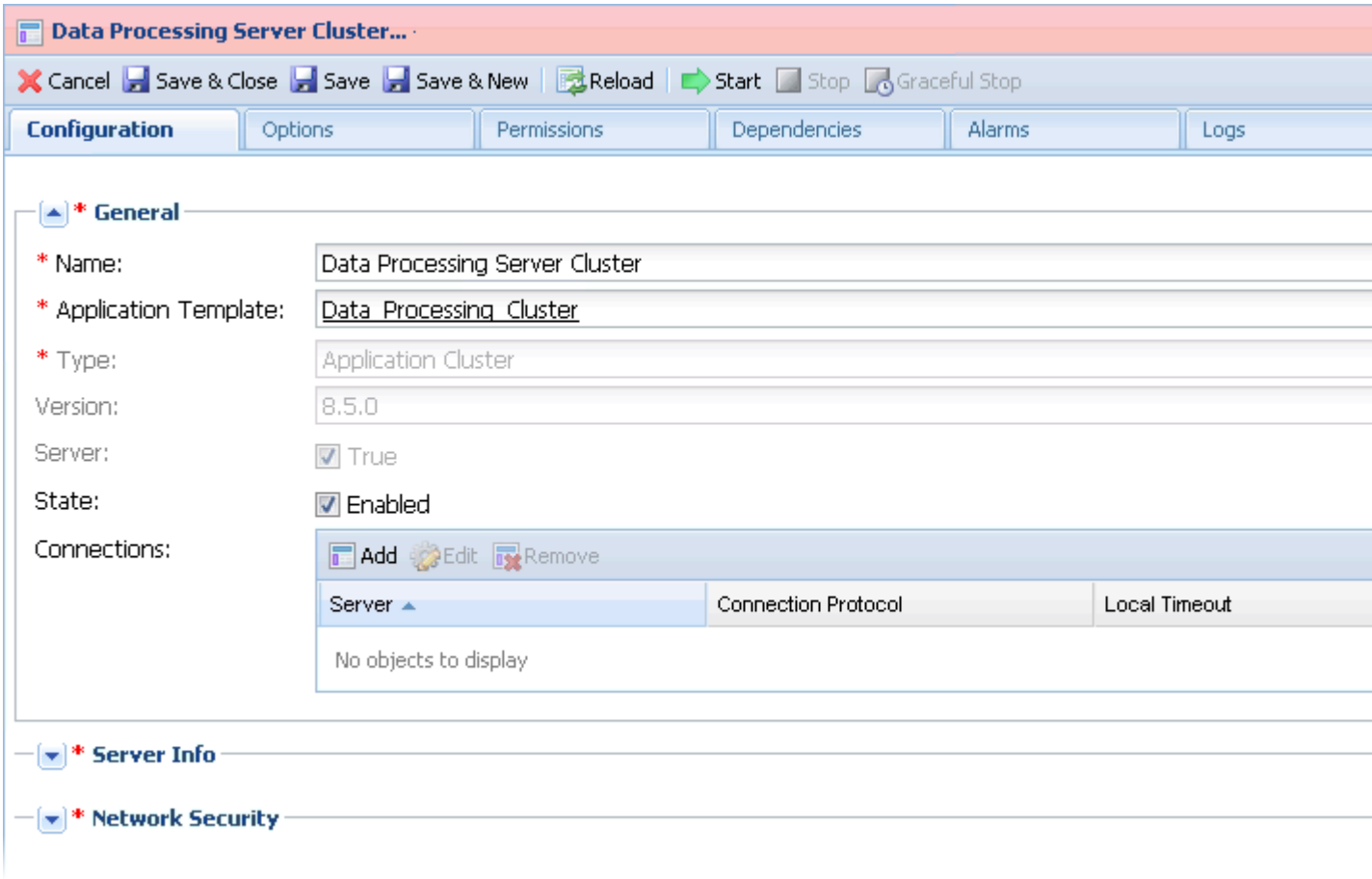


Create New Application link.

3. In the **Select Application Template** panel, click **Browse for Template** and select the Data Processing Server cluster template that you imported in Importing the Data Processing Server cluster template. Click **OK**.
4. The template is added to the **Select Application Template** panel. Click **Next**.
5. In the **Select Metadata file** panel, click **Browse** and select the **Data_Processing_Cluster.xml** file. Click **Open**.
6. The metadata file is added to the **Select Metadata file** panel. Click **Next**.
7. In the **Specify Application parameters** tab:
   - Enter a name for your application. For instance, `Data_Processing_Server_Cluster`.
   - Make sure **State** is enabled.
   - Select the **Host** on which the Data Processing Server cluster will reside.

- Click **Create**.

8. The **Results** panel opens.
9. Enable **Opens the Application details form after clicking 'Finish'** and click **Finish**. The Data Processing Server cluster application form opens and you can start configuring the Data Processing Server cluster application.



Data Processing Server Cluster app opened in Genesys Administrator.

End

Configuring the cluster application

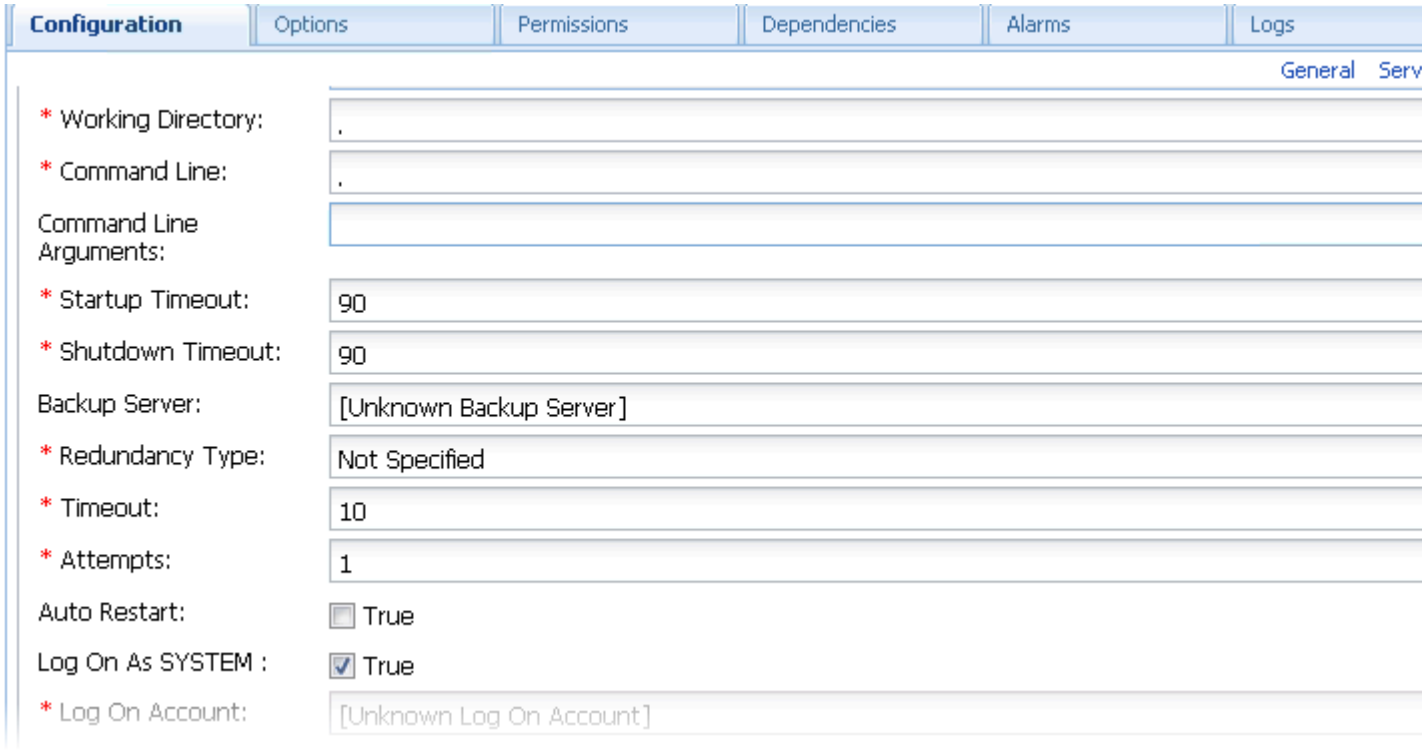
Note: For more information on how to work with application objects in Genesys Administrator, consult Generic Configuration Procedures.

Prerequisites

- You completed Creating the cluster application.

Start

1. If your Cluster application form is not open in Genesys Administrator, navigate to **Provisioning > Environment > Applications**. Select the application defined for the Data Processing Server cluster and click **Edit...**.
2. Expand the **Server Info** pane.
3. If your **Host** is not defined, click the lookup icon to browse to the hostname of your application.
4. Ensure the **Working Directory** and **Command Line** fields contain "." (period).



Commands

5. Click **Save**.
6. In the Listening Ports section, create the default port by clicking **Add**. The **Port Info** dialog opens.
   - Enter the **Port**. For instance, `10081`.
   - Choose `http` for the **Connection Protocol**.
   - Click **OK**. The HTTP port with the default identifier appears in the list of **Listening ports**.
7. Genesys recommends that you use *external* Cassandra and that you configure it by following the guidelines at Working with External Cassandra, applying the steps on that page to the Data Processing Server cluster instead of your product's server cluster.

   Data Processing Server supports the same Cassandra security features as Web Engagement, with the exception of mutual TLS. Refer to the GWE Cassandra security article for instructions on setting up Cassandra security for GDPS.

   However, if you choose to use *embedded* Cassandra, follow these steps:

1. Select the **Options** tab.
2. Make sure that the value of the clusterName option in the [cassandraEmbedded] section of the Data Processing Server cluster application is the same as the value of the clusterName option in your product's cluster application.
3. In the [cassandraEmbedded] section, you can take the default values for all of the options except seedNodes, which requires a comma-separated list of seed nodes, where each seed node is represented by an IP address or fully qualified domain name (FQDN). For example:
   - `192.168.0.1,192.168.0.3`
   - `host1.mydomain.com,host2.mydomain.com`.

   Important: The list of seed nodes should include at least one node from each operational and analytical data center.
4. In the [cassandraKeyspace] section, you may need to tune the replicationStrategyParams option, which by default is set to `'AnalyticalDC':1`. This value indicates a replication factor of **1** for nodes that belong to the `AnalyticalDC` data center. Note that the data center name is important for multi–data center configurations. When using the default endpoint snitch (`GossipingPropertyFileSnitch`), you can specify the data center name in ***Data Processing Server installation folder*/resources/cassandra-rackdc.properties**. For more information, refer to http://docs.datastax.com/en/cassandra/2.2/cassandra/architecture/archsnitchGossipPF.html.
5. Make sure that the value of the name option in the [cassandraKeyspace] section of the Data Processing Server cluster application is **not** the same as the value of the name option in your product's cluster application. This will ensure that your keyspace settings do not churn—for example, that the TTL properties from an operational keyspace will not be applied to an analytical keyspace.

8. In the **[spark]** section, specify the value of the **host** option using the fully-qualified domain name (FQDN) or IP address for your Data Processing Server.
   Important: Spark is very sensitive about host names and sometimes even minor network configuration problems may result in cluster connectivity problems. The best way to ensure correct behavior is to verify that the **spark.host** option uses the same name as the Java **InetAddress.getLocalHost().getHostName()** method would return for this host.
9. Click **Save & Close**. If the **Confirmation** dialog opens, click **Yes**.

End

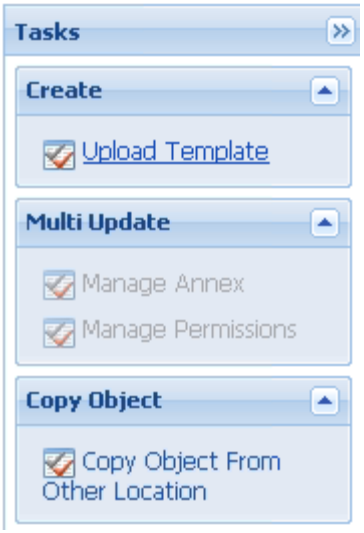
Importing the Data Processing Server template

Prerequisites

- You completed Configuring the cluster application.

Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Application Templates**.
2. In the **Tasks** panel, click **Upload Template**.



Upload Template link in the Tasks panel

3. In the **Click 'Add' and choose application template (APD) file to import** window, click **Add**.
4. Browse to the **Data_Processing_Server.apd** file and select it. The **New Application Template** panel opens.
5. Click **Save & Close**.

End

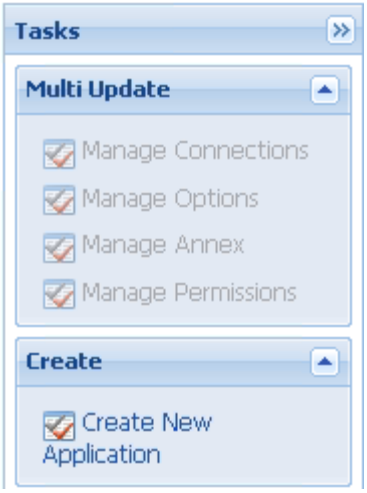
Creating a node application

Prerequisites

- You completed Importing the Data Processing Server template.

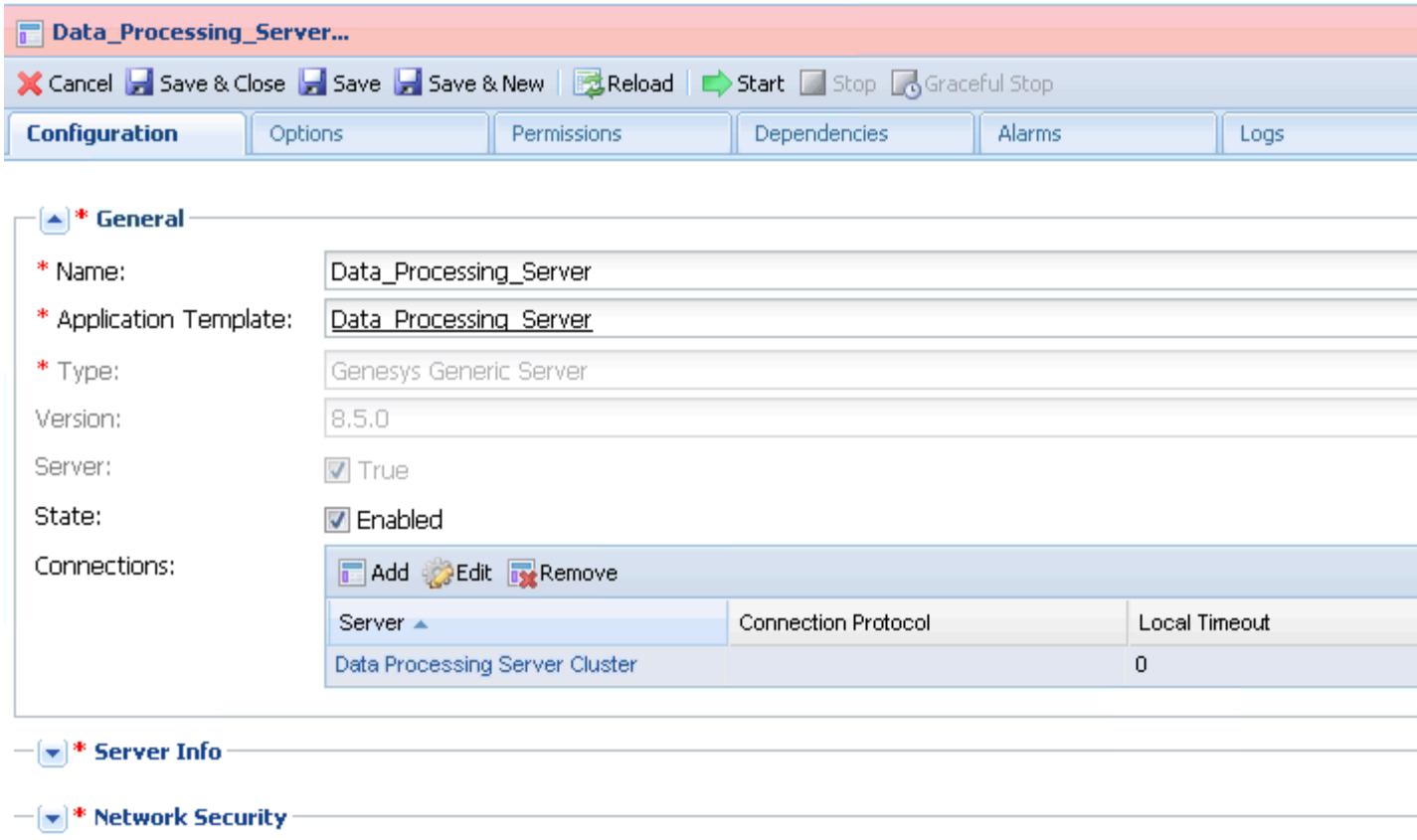
Start

1. Open Genesys Administrator and navigate to **Provisioning > Environment > Applications**.
2. In the **Tasks** panel, click **Create New Application**.



Create New Application link.

3. In the **Select Application Template** panel, click **Browse for Template** and select the Data Processing Server template that you imported in Importing the Data Processing Server template. Click **OK**.
4. The template is added to the **Select Application Template** panel. Click **Next**.
5. In the **Select Metadata file** panel, click **Browse** and select the **Data_Processing_Server.xml** file. Click **Open**.
6. The metadata file is added to the **Select Metadata file** panel. Click **Next**.
7. In **Specify Application parameters**:
   - Enter a name for your application. For instance, `Data_Processing_Server`.
   - Make sure **State** is enabled.
   - Select the **Host** on which the node will reside.
   - Click **Create**.
8. The **Results** panel opens.
9. Click **Save & Close**. If the **Confirmation** dialog opens, click **Yes**.
10. Enable **Opens the Application details form after clicking 'Finish'** and click **Finish**. The `Data_Processing_Server` application form opens and you can start configuring the node application.



Node app opened in Genesys Administrator.

End

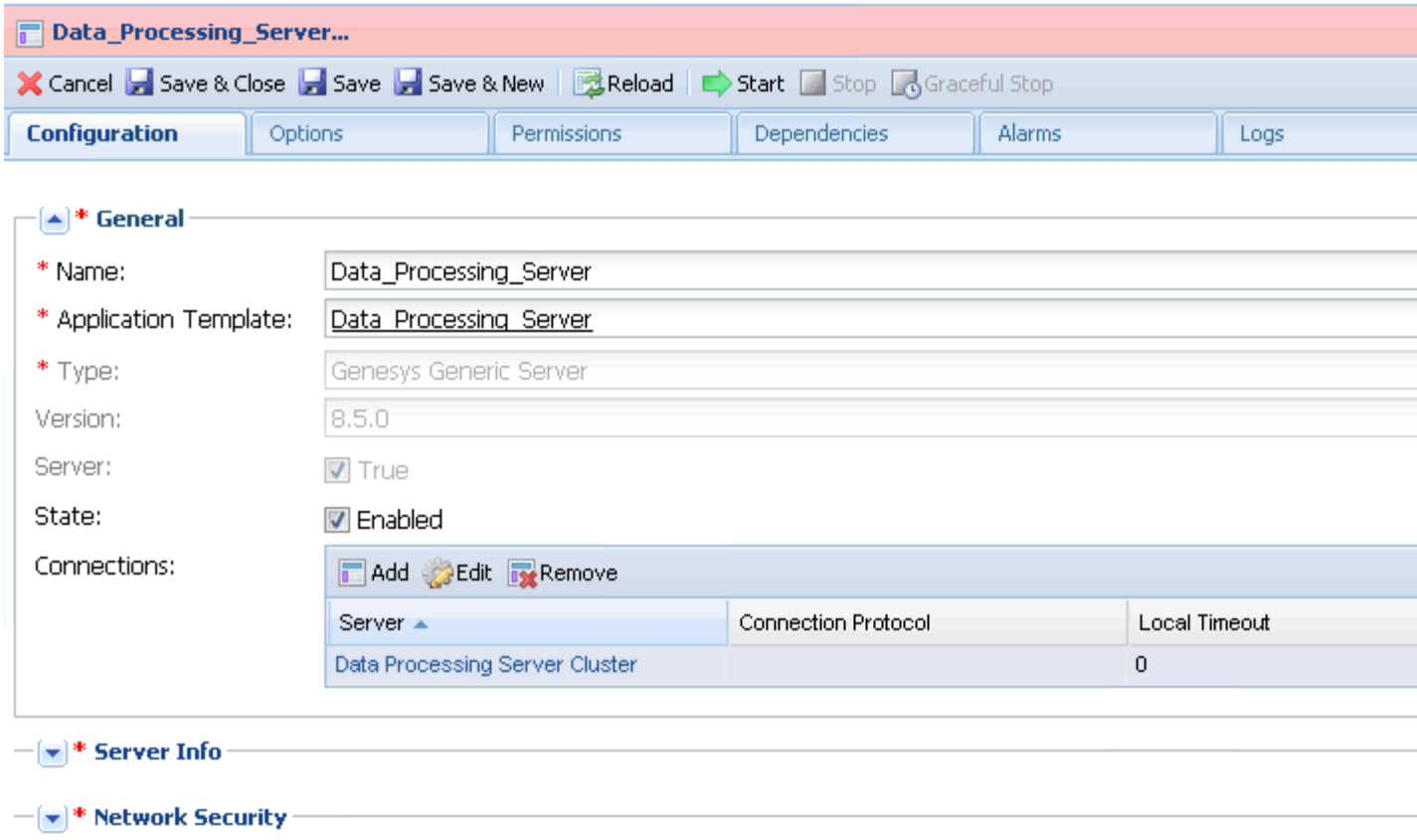
Configuring a node application

Prerequisites

- You completed Creating a node application.

Start

1. If your node application form is not open in Genesys Administrator, navigate to **Provisioning > Environment > Applications**. Select the application defined for the node and click **Edit...**.
2. In the Connections section of the **Configuration** tab, click **Add**. The **Browse for applications** panel opens. Select the Data Processing Server cluster application you defined above, then click **OK**.

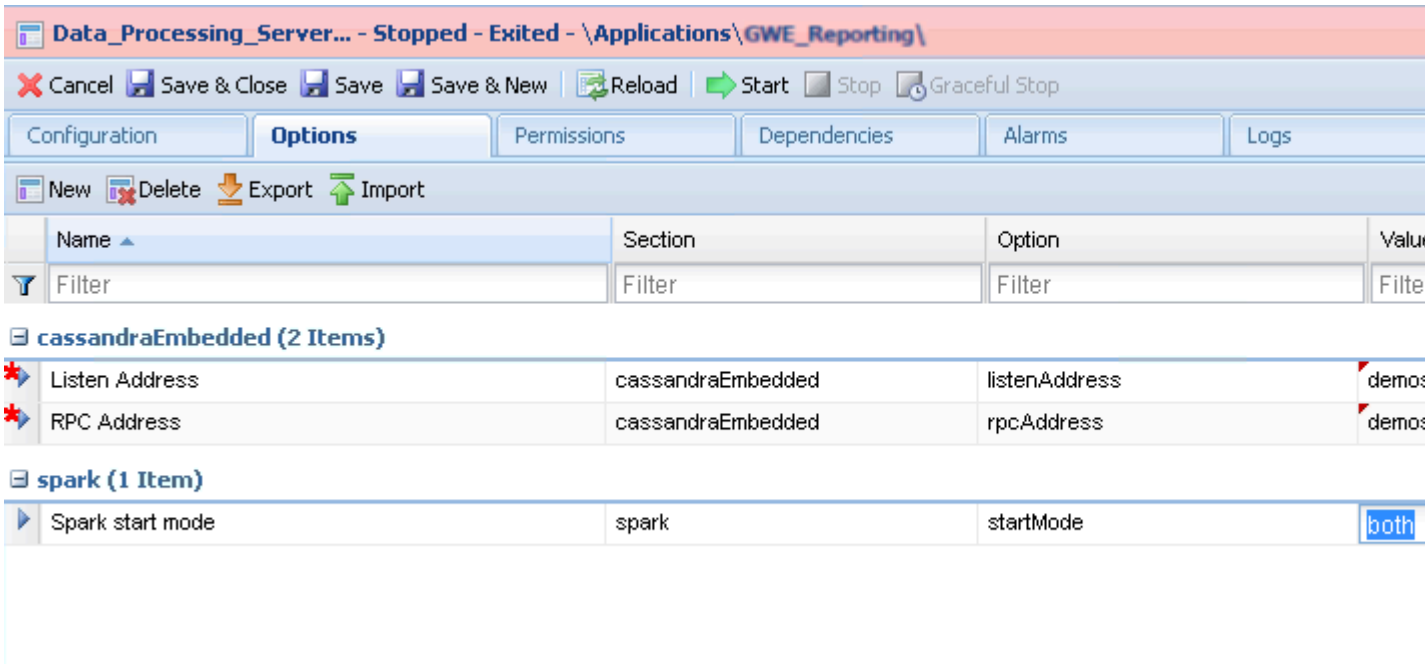


Node connection to Cluster

3. Expand the **Server Info** pane.
4. If your **Host** is not defined, click the lookup icon to browse to the hostname of your application.
5. In the Listening Ports section, create the default port by clicking **Add**. The **Port Info** dialog opens.
   - Enter the **Port**. For instance, `10081`.
   - Choose `http` for the **Connection Protocol**.
   - Click **OK**. The HTTP port with the default identifier appears in the list of **Listening ports**.
6. Click **Save**.
7. Genesys recommends that you use *external* Cassandra and that you configure it by following the guidelines at Working with External Cassandra, applying the steps on that page to the Data Processing Server cluster instead of your product's server cluster.

   However, if you choose to use *embedded* Cassandra, select the **Options** tab.
   - In the **View** field, select **Advanced View (Options)**.
   - In the [cassandraEmbedded] section, set the values for listenAddress and rpcAddress, using a fully qualified domain name or the appropriate IP address.
8. In the **[spark]** section, set the value for **startMode**. If you are configuring a master node, set this value to **both**. For other nodes, set it to **worker**.

Note: You should only have one master node configured for your Data Processing Server cluster. If you have a single-node cluster, then your single node must be configured as a master node.



Data Processing Server node options

End

Tuning analytical Cassandra nodes to skip indexing

For performance reasons, Web Engagement Server's *operational* Cassandra nodes use custom Cassandra indexes that rely on the services of Elasticsearch. However, the *analytical* Cassandra nodes used by Data Processing Server do not require those indexes, as all analytical reads of Cassandra tables are full scan reads. Because of this, Genesys recommends that these indexes be removed from your analytical nodes.

If you are using embedded Cassandra, Web Engagement will automatically avoid creating these indexes. But if you are using external Cassandra, you must carry out the following procedure to get rid of them.

Note: You must do these steps *before* you replicate the Cassandra data from your operational data centers.

Start

1. Copy the required libraries to the Cassandra **lib** folder for each Cassandra node in your analytical data centers.
2. Modify your Cassandra startup scripts to include the **genesys-es-dummy** system property

- On Windows, append the following line to **bin/cassandra.in.bat**:

```
set JAVA_OPTS=%JAVA_OPTS% -Dgenesys-es-dummy=true
```

- On Linux, append the following line to **bin/cassandra.in.sh**:

```
export JVM_OPTS="$JVM_OPTS -Dgenesys-es-dummy=true"
```

End

Adding nodes to a cluster

To create more nodes:

Start

1. Follow the instructions above for Creating a node application, but use a different name for the new node.
2. Configure the new node application, as shown above, but point to a different port.

End

Updating your product's cluster application

Genesys recommends that you use a dedicated data center for reporting. In order to do this, you must do the following to your product's cluster application:

- Modify the seedNodes option in the [cassandraEmbedded] section so that it is in sync with the **seedNodes** option in the Data Processing Server Cluster application.
- Modify the replicationStrategyParams option in the [cassandraKeyspace] section so that it includes replication to the reporting data center. For example:

```
'OperationalDC':1,'AnalyticalDC':1
```

Important: You only need to do this to your product's cluster application. Note also that your Data Processing Server Cluster application should specify replication strategy parameters for the corresponding analytical (reporting) data center only. In other words, all of the data that comes into the analytical data center should be left there, rather than being propagated to an operational data center.

Important: If you have more than one operational data center, then you must replicate **each** of your operational data centers to analytical data centers.

Data Processing Server data storage

Data Processing Server stores several types of information:

- Aggregated data results
- General configuration data
- Tenant-specific configuration data
- Default Pulse dashboards and widgets
- Meta-information

All of this information is stored in a database layer that is indexed by Elasticsearch. By default, you can access the reporting database layer via HTTP, using the URL of the correctly configured Load Balancer:

- For embedded Cassandra, the Load Balancer should redirect requests to **one of your product's hosts** on port `9200` (or the port ID you have specified in the **http.port** option of the [elasticsearch] section of your product's cluster application).
- For external Cassandra, the Load Balancer should redirect requests to one of the **Cassandra hosts** used by your product (Operational DC) on port `9200` (or the port ID you have specified in the **http.port** option of the [elasticsearch] section of your product's cluster application).

We will refer to this URL as *the Reporting Data URL*. When you send your browser or HTTP client requests to the Reporting Data URL, you should receive `HTTP Response 200`.

Installing Data Processing Server

Install the Data Processing Server on Windows or Linux.

Note: For more information on how to install apps that you have configured in Genesys Administrator, consult Generic Installation Procedures.

The Pulse Collector

Data Processing Server uses data gathered by a Pulse Collector. This Pulse Collector must only be installed on one node in the Data Processing Server cluster.

Although the procedures in the next section tell you how to set up your initial Pulse Collector installation, if you decide later that you want to install the Pulse Collector on a different node, you must follow these extra steps:

- Turn the collector off at the node it was originally installed on:

    - Stop the node
    - Set the **PULSE_COLLECTOR_ENABLED** variable in your **setenv.bat** or **setenv.sh** file to `false`
    - Remove the **pulse-collector.war** file from the **webapps** folder
    - Restart the node
- Turn the collector on at another node:
    - Stop the node
    - Set the **PULSE_COLLECTOR_ENABLED** variable in your **setenv.bat** or **setenv.sh** file to `true`
    - Set the value of **DATA_PROCESSING_ES_URL**—also in your **setenv.bat** or **setenv.sh** file—to the value of your Reporting Data URL
    - Copy the **pulse-collector.war** file from the **etc** folder to the **webapps** folder
    - Restart the node

Windows

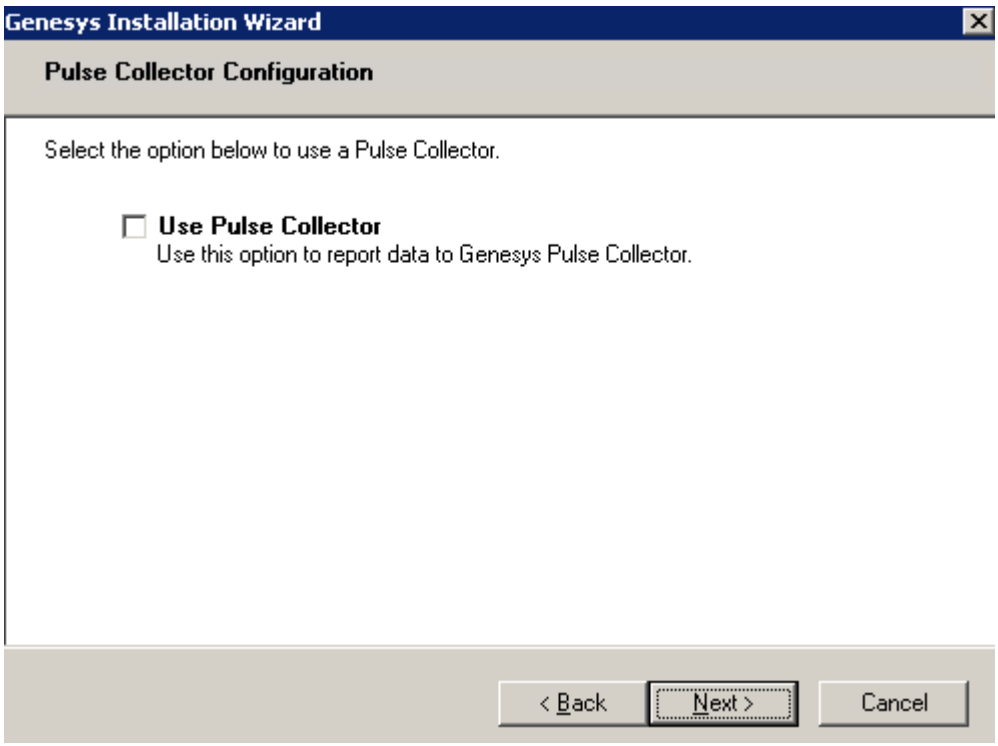
Prerequisites

- Configuring a node application
- A supported JDK is installed. See Java Requirements for details.

Start

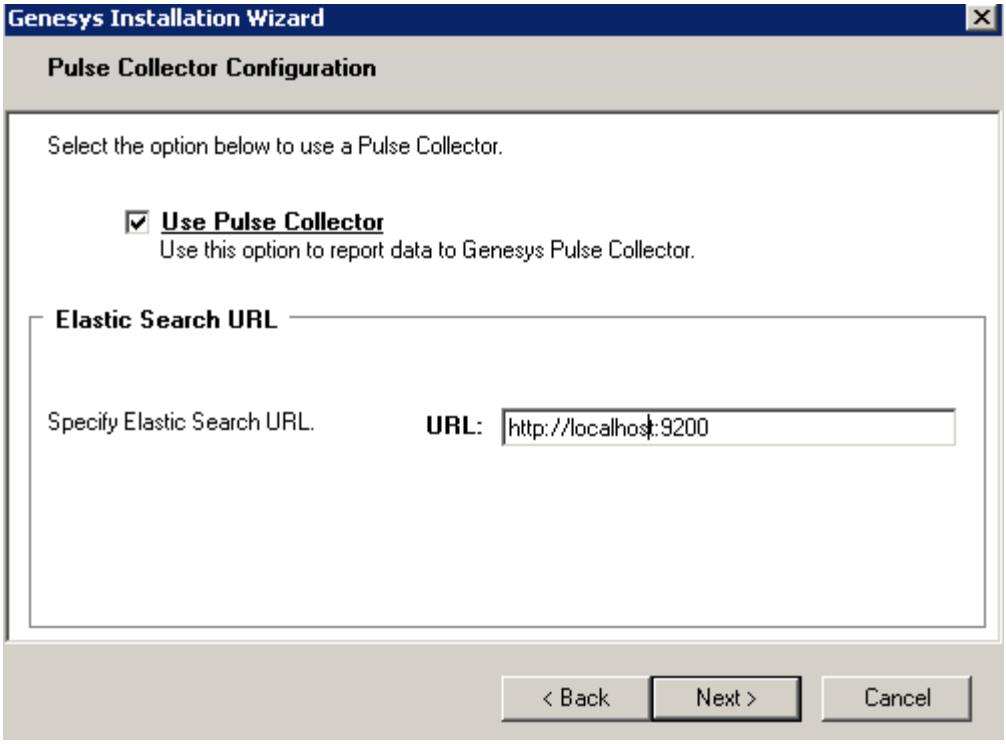
1. In your installation package, locate and double-click the **setup.exe** file. The Install Shield opens the welcome screen.
2. Click **Next**. The **Connection Parameters to the Configuration Server** screen appears.
3. Under **Host**, specify the host name and port number where Configuration Server is running. (This is the main "listening" port entered in the **Server Info** tab for Configuration Server.)
4. Under **User**, enter the user name and password for logging on to Configuration Server.
5. Click **Next**. The **Select Application** screen appears.
6. Select the Data Processing Server Application—that is, the Node app you created above—that you are installing. The **Application Properties** area shows the **Type**, **Host**, **Working Directory**, **Command Line executable**, and **Command Line Argument**s information previously entered in the **Server Info** and **Start Info** tabs of the selected Application object.
   Note: For multi-node clusters, you must install the Data Processing Server Application into exactly the same directory on every node. For example, if the path for Node 1 is `/genesys/gdps/gdps_n1`, it cannot be `/genesys/gdps/gdps_n2` for any of the other nodes. This requires manual intervention, since the installation package offers a default installation path based on the application name, which is therefore different for each node.
7. Click **Next**. The **Choose Destination Location** screen appears.
8. Under **Destination Folder**, keep the default value or browse for the desired installation location. Note that you probably do not want to use the Windows Program Files folder as your destination folder.
9. Click **Next**. The **Backup Configuration Server Parameters** screen appears.

10. If you have a backup Configuration Server, enter the **Host name** and **Port**.
11. In the **Pulse Collector Configuration** window, select **Use Pulse Collector**:



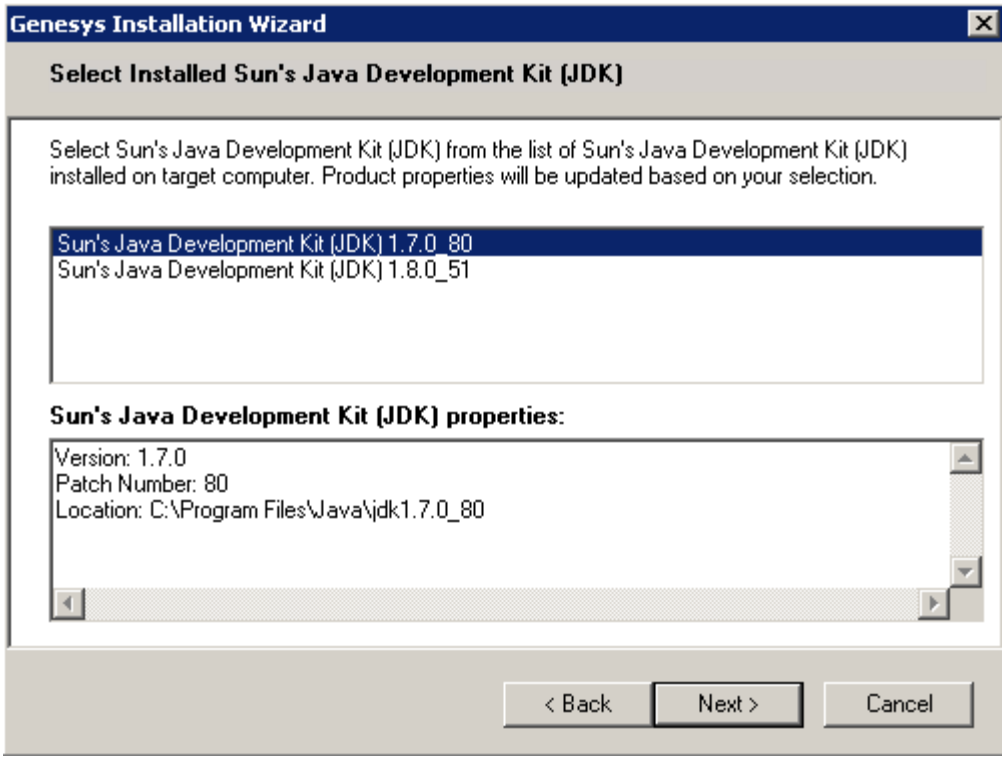
Pulse Collector Configuration

12. In the resulting **Elasticsearch URL** field, specify the host and port of the Data Processing Server cluster node on which you want to install the Pulse Collector. **Note:** You should only install the Pulse Collector on one node at a time, as mentioned above.



Pulse Collector Host and Port

13. Click **Next**. Select the appropriate JDK:



Select JDK

14. Click **Next**. The **Ready to Install** screen appears.
15. Click **Install**. The Genesys Installation Wizard indicates it is performing the requested operation for Data Processing Server. When through, the **Installation Complete** screen appears.
16. Click **Finish** to complete your installation of the Data Processing Server.
17. Inspect the directory tree of your system to make sure that the files have been installed in the location that you intended.
18. Apply the parameters to your Windows service.

End

Note: Genesys recommends that you frequently clear the Spark temporary directory—for example, once a week or before you start Data Processing Server. You can find it in the system temporary directory with a name template of `spark-*`. The default location for this directory is ***system_disk*:\Users*user_name*\AppData\Local\Temp directory**. You can also use the system disk clean-up procedure.

Linux

Prerequisites

- Configuring a node application
- A supported JDK is installed. See Java Requirements for details.

Start

1. Open a terminal in the Genesys Web Engagement CD/DVD or the Genesys Web Engagement IP, and run the **Install.sh** file. The Genesys Installation starts.
2. Enter the hostname of the host on which you are going to install.
3. Enter the connection information to log in to Configuration Server:
   - The hostname. For instance, `demosrv.genesyslab.com`.
   - The listening port. For instance, `2020`.
   - The user name. For instance, `demo`.
   - The password.
4. If you have a backup Configuration Server, enter the **Host name** and **Port**.
   If the connection settings are successful, a list of keys and Web Engagement applications is displayed.
5. Enter the key for the Data Processing Server application—that is, the Node app you created above in Configuration Server.
6. Use the key for Genesys Pulse to indicate whether or not to enable the Pulse Collector
7. If you have enabled the Pulse Collector, enter the Reporting Data Elasticsearch URL.
8. Enter the location where Data Processing Server is to be installed on your web server.
   Note: This location must match the previous settings that you entered in Configuration Server.
   Note: For multi-node clusters, you must install the Data Processing Server Application into exactly the same directory on every node. For example, if the path for Node 1 is `/genesys/gdps/gdps_n1`, it cannot be `/genesys/gdps/gdps_n2` for any of the other nodes. This requires manual intervention, since the installation package offers a default installation path based on the application name, which is therefore different for each node.
9. If the installation is successful, the console displays the following message:
   `Installation of Genesys Data Processing Server, version 8.5.x has completed successfully.`
10. Inspect the directory tree of your system to make sure that the files have been installed in the location that you intended.

End

Note: Genesys recommends that you frequently clear the Spark temporary directory—for example, once a week or before you start Data Processing Server. You can find it in the system temporary directory with a name template of `spark-*`. The default location for this directory is **/tmp**.

Installing dashboards and widgets into Pulse

At this point, you must follow the instructions for installing your product's Pulse dashboards and widgets:

- Genesys Web Engagement

Deploying and scheduling job packages

After you have installed **and started** the Data Processing Server, you need to deploy and schedule jobs for it to process.

To do this, execute the following two scripts that are provided in ***Data Processing Server Installation directory*/deploy/package**.

> ### Important
>
> - You must install curl in order to use these scripts.
> - You must run the scripts on the master node.

Once you have installed curl, make sure it's available from ***Data Processing Server Installation directory*/deploy/package**, and run the following scripts:

1. **deploy-package.bat** or **deploy-package.sh** deploys your jobs and requires the URL (host and port) for your Data Processing Server.

   If the script executes successfully, you will receive the following response: `{“status” : “Ok”}`

   Here is a sample command for Windows:

   ```
   deploy-package.bat reporting-host:10081
   ```

   And one for Linux:

   ```
   GDPS_URL=http://reporting-host:10081 ./deploy-package.sh
   ```

2. **schedule-package.bat** or **schedule-package.sh** schedules your jobs and requires:
   - The URL (host and port) for your Data Processing Server
   - The Reporting Data URL
   - The name of the Cassandra keyspace that is used by your product. In the case of Web Engagement, for example, this is the keyspace name described in the [cassandraKeyspace] configuration option section.

   If the script executes successfully, you will receive the following response: `{"<schedule_guid>" : "Job scheduled"}`

   Here is a sample command for Windows:

```
schedule-package.bat reporting-host:10081 gwe-cluster-lb:9200 gpe
```

And one for Linux:

```
GDPS_URL=http://reporting-host:10081 ES_URL=http://gwe-cluster-lb:9200 KEYSPACE=gpe
 ./schedule-package.sh
```

In this sample, we can see that:

- Data Processing Server is running on `reporting-host` at port `10081`.
- The Load Balancer running on `gwe-cluster-lb` at port `9200` represents a Reporting Data URL.
- The GWE keyspace name is gpe.

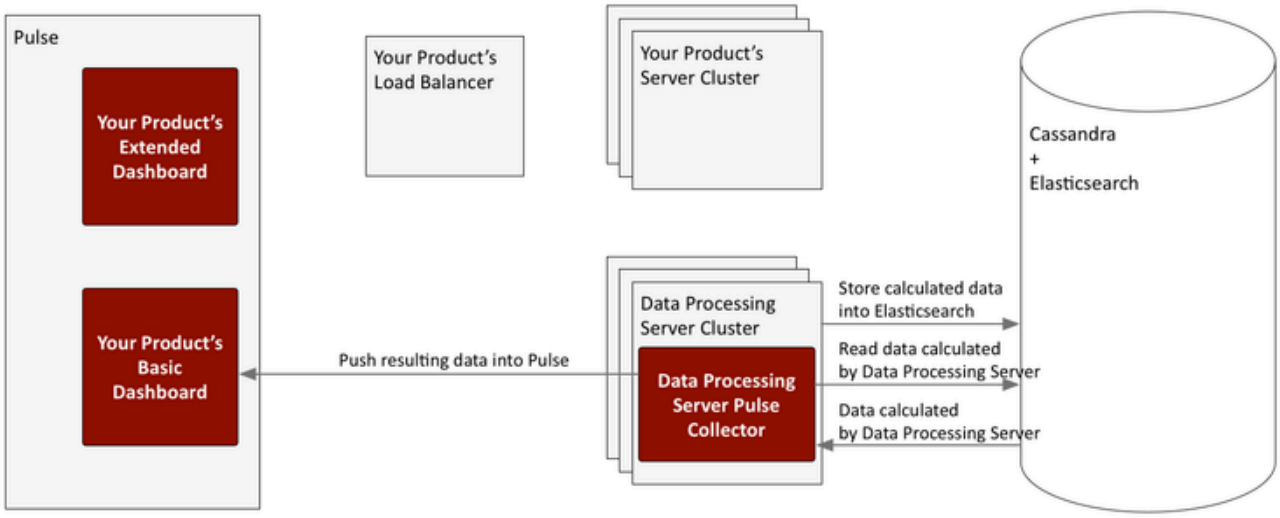
Data Processing Server data flow for reporting

Data Processing Server supports two types of Pulse reporting:

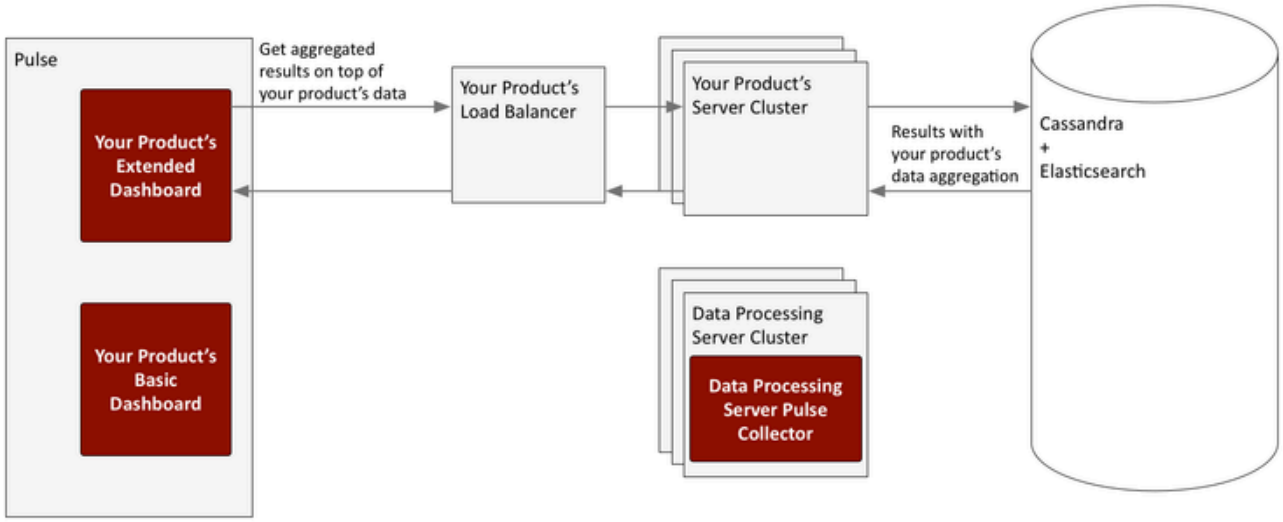
- The basic views, also known as the basic dashboards, are native Pulse widgets that are driven by a Pulse collector which is included with the Data Processing Server. These widgets can be used in other dashboard views created in Pulse.
- The extended views (dashboards) are bespoke metric views of your product's data sources hosted directly within Pulse. These widgets cannot be used with other Pulse dashboards.

The following diagrams show the data flow for each type of reporting.

Basic dashboard



Extended dashboard



Configuration options

These configuration option sections can be useful in setting up Data Processing Server and your Spark cluster.

log

The [log] configuration options are applied to the Data Processing Server environment in a way that is similar to how they are used with Web Engagement Server.

cassandraEmbedded

If you have set up your Data Processing Server with an embedded Cassandra node, the [cassandraEmbedded] section provides its configuration options. Their meanings when applied to the Data Processing Server environment are similar to the way they are used with Web Engagement Server.

cassandraKeyspace

Data Processing Server stores some of its data (packages and schedule) in a dedicated Cassandra keyspace. The [cassandraKeyspace] section provides its configuration options. All of the options mean pretty much the same as they do when used with Web Engagement Server, although some of their values—such as for the **name**—will be different.

spark

Data Processing Server launches a dedicated Spark cluster and all of the Data Processing Server nodes need to share the coordinates of the Spark Master node. In addition to this, each individual node has options that can be used to configure the mode with which Spark starts on its box.

host

Description: The name of the Spark Master host. The value should be the same as what Java's **InetAddress.getLocalHost()** would return for the specified host.
Default Value: None
Valid Values: *hostname of the Spark Master node*
Mandatory: No
Changes Take Effect: After start/restart

port

Description: The port number of the Spark Master host.
Default Value: `7077`
Valid Values: Valid port number
Mandatory: No
Changes Take Effect: After start/restart

startMode

Description: The mode that will be used when starting Spark. If set to `off`, Spark will not be started by Data Processing Server, and will instead have its state managed externally. If set to `worker`, only a worker node will be started. If set to `both`, both a worker node and a master node are started. **Note:** Genesys recommends that you set this option for each node to clearly specify the role. However, you can set the Cluster object to `worker` mode and override that value for the master node by setting that node to `both`.
Default Value: `worker`
Valid Values: `off`, `worker`, or `both`
Mandatory: No
Changes Take Effect: After start/restart

masterWebPort

Description: The number of the TCP port that the Spark Master web UI will listen on. Note that this option is provided for cases when the default port has already been used by another service.
Default Value: `8080`
Valid Values: Valid port number
Mandatory: No
Changes Take Effect: After start/restart

workerWebPort

Description: The number of the TCP port that the Spark Worker web UI will listen on. Note that this

option is provided for cases when the default port has already been used by another service.
Default Value: 8081
Valid Values: Valid port number
Mandatory: No
Changes Take Effect: After start/restart

executorMemory

Description: Use this option to manage the amount of memory used by Spark for executing tasks on each node. Genesys recommends at least two gigabytes per node, but more memory can improve performance if hardware allows. For information about the format, consult the Spark documentation.
Default Value: None
Valid Values: Valid memory limit
Mandatory: No
Changes Take Effect: After start/restart

sparkHeartbeatTimeout

Description: The timeout value in seconds between two heartbeat calls to the Spark metrics API.
Default Value: 60
Valid Values: Positive integer
Mandatory: No
Changes Take Effect: After start/restart

sparkStartTimeout

Description: The timeout value in seconds between a Spark start or restart and the first time its API is checked. On slower machines, it makes sense to increase this value so that Spark has enough time to start successfully (without initiating a restart cycle).
Default Value: 20
Valid Values: Positive integer
Mandatory: No
Changes Take Effect: After start/restart

uri

Description: Advanced. For situations when Spark is running externally, you must set the URI instead of the host and port. The URI must include the protocol, in addition to the host and port.
Default Value: None
Valid Values: Valid Spark URI
Mandatory: No
Changes Take Effect: After start/restart

spark.context

Advanced. This entire section is copied into **SparkContext**, so it can be used to tune the Spark options. You must have an in-depth understanding of Spark configuration if you are going to use this section.