1. Delphix Masking Quick Start Guide - An Overview ................................................................. 4
  1.1 Masking Engine Terms Overview ...................................................................................... 7
  1.2 Masking Engine Install, System Configuration, and Network Setup ............................... 10
  1.3 Prepare Data for Masking ........................................................................................................ 12
  1.4 Masking Engine Activities .................................................................................................. 13
    1.4.1 Add an Application and Create a New Environment and Connector ....................... 15
    1.4.2 Create Data Masking Rule Sets, Algorithms, and Inventories .............................. 18
    1.4.3 Mask Data ...................................................................................................................... 24
  1.5 Provisioning Masked VDBs ................................................................................................. 26
    1.5.1 Troubleshoot Provisioning Errors for Masked VDBs .................................................. 33
  1.6 Masking API Calls to Run a Masking Job ........................................................................... 34
  1.7 Advanced Integrated Delphix Masking Workflows .............................................................. 36
Delphix Masking Quick Start Guide - An Overview

- Purpose of this Quick Guide
- What Is Delphix Masking?
- Common Use Cases
  - Masking Production Data for Non-Production Environments
- Before You Begin
- Delphix Masking Engine Workflows
  - Prepare Data for Masking
  - Create and Run a Simple Masking Job
  - Provision a Masked VDB
  - Advanced Integrated Masking Workflows
- Related Links

Purpose of this Quick Guide

The Masking Engine Quick Start Guide is aimed at all users interested in using the Masking Engine in conjunction with Data as a Service Engine. This guide serves as a brief overview and refresher of important Masking Engine concepts as well as a guide for the common masking workflows when using either or both the Masking Engine and Data as a Service Engine. The workflows below describe different options and configurations available for masking virtualized data using both the Masking Engine along with the Data as a Service Engine.

What Is Delphix Masking?

Delphix Masking combines the Data as a Service Engine with the Masking Engine to provide the ability to provision, mask, and manage data in a standardized and automated way. Masking technology works behind the scenes to replace sensitive data with fictitious data, which you can then move out of your production environment and into non-prod environments.

Common Use Cases

Masking Production Data for Non-Production Environments

Developers commonly need access to data from the production environment when they are:

- developing new applications
- maintaining or enhancing existing applications
- working in test environments to test the functionality of applications

Production data commonly includes sensitive information that by law needs to be protected. However, it is not uncommon in the app/dev life cycle for the same sensitive data that is found in production to appear in non-production environments used for application development and testing.

The masking features in the Masking Engine can be used to ensure that application/test environments do not transmit the sensitive information contained in the parent source. The Masking Engine removes or alters sensitive information without changing the character of the production data, allowing developers and testers to work with the highest quality test data for software development.

Before You Begin

Before getting started, familiarize yourself with the terminology in the Masking Engine Terms Overview. The Masking Engine introduces new terms and concepts that may not be familiar to a Delphix administrator or user.

You will also need a fully installed Masking Engine prior to performing the workflows on this page. If your Masking Engine is not yet configured, see the guide at: Masking Engine Install, System Configuration, and Network Setup.
<table>
<thead>
<tr>
<th>Who is using this deployment type?</th>
<th>Masking installations on Delphix 4.x</th>
<th>Masking installations on Delphix 5.0 and above</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installation and deployment details</td>
<td>Masking Engine - see <a href="#">Masking Engine Install, System Configuration, and Network Setup</a></td>
<td>Data as a Service Engine - Includes the Masking Engine. See <a href="#">Installing the Delphix Engine</a>.</td>
</tr>
<tr>
<td>Delphix Engine - see <a href="#">Installing the Delphix Engine</a> for your release version.</td>
<td>Once the Delphix Engine is installed, a ticket may be filed with Delphix Support to enable the Masking Engine.</td>
<td></td>
</tr>
<tr>
<td>Where to Log In</td>
<td>http://&lt;Masking Engine IP or DNS name&gt;:8282/dmsuite</td>
<td>http://&lt;Delphix Engine IP or DNS name&gt;:8282/dmsuite</td>
</tr>
<tr>
<td>Initial Requirements for all Masking Activities</td>
<td>In the Masking Engine interface: 1. Prepare data for masking. 2. Create a masking job.</td>
<td></td>
</tr>
<tr>
<td>For detailed instructions, see <a href="#">Prepare Data for Masking</a> and <a href="#">Mask Data</a>.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>To mask data</td>
<td>Once a masking job has been created, you can run it either:  - Via the GUI  - Via the API</td>
<td>Masking jobs can be created in the Masking Engine GUI and run via the GUI or API just like Standalone Masking Engines. You can also use the Data as a Service Engine GUI to provision masked virtual databases (VDBs). See <a href="#">Provisioning Masked VDBs</a> for more details.</td>
</tr>
</tbody>
</table>
| Post-Masking Features |  | • Secure replication with [Selective Data Distribution](#).  
• Integrated management of Masked VDBs in Jet Stream |

## Delphix Masking Engine Workflows

Ultimately, the way you use the Delphix Masking Engine will depend on your business needs and security practices. However, there are several common workflows that you will need to understand in order to perform various masking tasks.

### Prepare Data for Masking

1. Link a dSource.  
2. Provision a normal VDB in the Delphix Engine.

To learn more, visit [Prepare Data for Masking](#)

### Create and Run a Simple Masking Job

To create and run a simple masking job in the Masking Engine:

1. Identify all sensitive data.  
2. Select methods for securing data.  
3. Validate that the methods will work for your needs.  
4. Implement and create new masking jobs for your target environments.

To learn more, visit [Masking Engine Activities](#). If you are interested in running jobs through the API for automation, you can learn more at [Masking API Calls to Run a Masking Job](#)

### Provision a Masked VDB

1. Associate masking jobs with a dSource in the Delphix Engine Admin console.
2. Use the dSource provision wizard to provision a VDB with a masking job.
3. Provision from a masked VDB to create a new masked VDB in the Delphix Engine.

To learn more, visit Provisioning Masked VDBs.

Advanced Integrated Masking Workflows

Other advanced workflows include using masked VDBs for selective data distribution and Jet Stream.
To learn more, visit Advanced Integrated Delphix Masking Workflows.

Related Links

- Masking Engine Terms Overview
- Masking Engine Install, System Configuration, and Network Setup
- Prepare Data for Masking
- Masking Engine Activities
- Provisioning Masked VDBs
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflows
Masking Engine Terms Overview

Before getting started with the Masking Engine, an overview of universal terms and concepts will build and unify how different masking components come together. The following provides a brief overview of eight key concepts for masking: provisioning, and working with applications, environments, connections, rule sets, profiling, inventory, and algorithms while masking data in place.

- Masking Engine Types
- Provision Data
- Understanding Environments and Applications
- Understanding Connections
- Understanding Rule Sets with Domains
- Understanding Profiling
- Understanding Inventory
- Understanding Algorithms
- Understanding Masked Data
  - Mask Data in Place
- Related Links

Masking Engine Types

There are two primary Masking Engine types.

- **Standalone Masking Engine** - This Engine is deployed as an OVA in a compatible hypervisor and contains the Masking Engine GUI. From here you can create masking jobs, mask data, and administer your Masking Engine. This Engine type is suitable for Delphix installations below Delphix 5.0.

- **Combined Delphix Engine and Masking Engine** - This Engine is built into your Delphix 5.0 and above installation. It contains both the Delphix Engine GUI and Masking Engine GUI, and allows tighter integration between Delphix's Data as a Service and Masking features.

For more information about these types of Masking Engine deployments, read the Before You Begin section in Delphix Masking Quick Start Guide - An Overview.

Provision Data

Delphix allows you to provision data from a linked source to the target you choose. This flexibility empowers development and testing teams to procure fresh, secure data from a source environment and move it to a non-production environment whenever they need it.

Understanding Environments and Applications

Environments define the scope of work in the Masking Engine. The masking environment is a collection of masking constructs (connectors, rule sets / inventories, and jobs) that support masking for a given application environment. In order to mask databases and files within the Delphix Engine, you first need to create an environment in which the Delphix Engine will store the connection information and masking rules for those data stores. An environment can contain multiple database connections and multiple file connections. Environments are connected to applications for informational purposes. For example, an integrated test environment can have multiple applications.

An application refers to the IT assets (programs, data, processes) that support a business function. For example, if a bank offers payroll services to its clients, there would be an application in its IT division to support that business. If the bank develops code to support new functions for its payroll application, the IT division would have environments where code is developed and tested. These environments contain test data used to test the new code. The test data is masked to support data privacy requirements.

Understanding Connections

Delphix stores JDBC database connection information in an object called a “connector.” You can discover a list of connectors within an environment by going to Environment Overview and then clicking the Connector tab. The connection includes fields such as database name, host, user id and password, and port. It is specific to the DBMS type you select. This builds a connector between the source database and the masking interface.

Understanding Rule Sets with Domains
A "rule set" points to a collection of tables or flat files that the Masking Engine uses for masking data. The rule set allows you to identify, select, and configure which tables you need to mask. For those tables that do not have a primary key defined, you can define a logical key with a combination of columns (or ROWID for Oracle database).

Understanding Profiling

Profiling is a major component of the Masking Engine. The objective of profiling is to identify the location of Non-Public Information (NPI) or sensitive data if you are unsure of what data needs to be masked in the first place. Profiling data is not necessary when you have already identified the sensitive data you need to mask.

The Delphix profiler uses two different methods to identify the location of sensitive data:

- Searching through the column names in the target database by querying the database catalog (metadata)
- Looking at the data itself, using a sampling algorithm, to see whether there is any sensitive data. This is especially useful for files and comment and notes fields in a database.

Understanding Inventory

The Delphix Engine automatically stores the masking rules for each sensitive column in the Delphix repository database in the environment's "inventory." When you select a table to mask, its columns will appear, and you can select them for masking. Afterwards, you can edit the columns with an appropriate algorithm required for masking.

Understanding Algorithms

Algorithms are how the Masking Engine masks sensitive data. From the Settings tab, click Algorithm on the left-hand side, and the list of algorithms appears for you to select. The following algorithms are the most commonly used methods for masking:

- **Secure Lookup Algorithm** – Uses a lookup file to assign masked values in a consistent manner
- **Segmented Mapping Algorithm** – Replaces data values based on segment definitions. For example, an ACCOUNT NUMBER algorithm might keep the first segment of an account number but replace the remaining segments with a random number.
- **Secure Shuffle algorithm** – A user-defined algorithm assigned to a specific column. Secure shuffle automates the creation of a secure lookup algorithm by building a list of replacement values based on the existing unique values in the target column and creating a secure lookup using those values. In that respect, it is simply shuffling the values.

Understanding Masked Data

After you create a masking environment, connection, rule set, and inventory, you can mask data.

The Delphix Engine will maintain Referential Integrity (RI) by masking each field with the same algorithm. This repeatable masking automatically maintains RI (for verbatim matches), even if it is between applications or platforms.

As a practical example, assume you have an Social Security Number (SSN) column in a Microsoft SQL Server database, an SSN column in a DB2 database, and an SSN field in a tab-delimited file. If the SSN value was 111111111 across the two databases and the file, and you use the same SSN algorithm for all three, the masked value (for example, 801-01-0838) will be the same for all three.

Note: When defining a masking job, select mask data in place.

Mask Data in Place

"Mask data in place" refers to updating a database with masked data. This includes reading data from the table defined in the rule set, masking the data in the Masking Engine, and updating the tables with the masked data.

Related Links

- [Quick Start Masking Engine Overview](#)
- [Masking Engine Install, System Configuration, and Network Setup](#)
- [Prepare Data for Masking](#)
- [Masking Engine Activities](#)
• Provisioning Masked VDBs
• Masking API Calls to Run a Masking Job
• Advanced Integrated Delphix Masking Workflow
Masking Engine Install, System Configuration, and Network Setup

- Installation Overview
- Delphix and Masking Engine Combined Installation
- Standalone Masking Engine Installation
- Next Steps
- Related Links

Installation Overview

Installations of Delphix 5.0 and above include the Delphix Masking Engine. This combination of the Delphix Data as a Service Engine and Masking Engine provides tight integration and enables additional features such as Selective Data Distribution.

Both installation types require:

- A Delphix Support Account
- The appropriate installation file for your supported hypervisor (e.g. VMWare) and installation type from Delphix Downloads

If you are installing Delphix 5.0 or above, start here: Delphix and Masking Engine Installation
If you are not running Delphix 5.0 or above, and want to install the latest Masking Engine, start here: Standalone Masking Engine Installation
If you are unsure of which Masking Engine is right for you, please contact your Professional Services team or Delphix Support.

Delphix and Masking Engine Combined Installation

Every Delphix 5.0 and above engine includes the Masking Engine. For this installation type:

1. Download Delphix 5.0 or above from Delphix Downloads using your Support credentials.
2. Install the Delphix Engine 5.0 or above. For requirements and details, see Installing the Delphix Engine.
3. Create a ticket through Delphix Support to enable masking functionality on your Delphix Engine.
4. Connect to the Masking Engine at: http://<Delphix Engine IP or DNS name>:8282/dmsuite as the user delphix_admin and password Delphix_123.

Click here to proceed to next steps once your Masking Engine is installed and enabled.

Standalone Masking Engine Installation

If the Masking Engine is required but not included in your Delphix Engine (i.e. your Delphix Engine is below version 5.0), it is possible to install a Standalone Masking Engine.

1. Download the Standalone Masking Engine 5.0 or above from Delphix Downloads using your Support credentials.
2. Deploy the downloaded OVA on a supported hypervisor using the following minimum requirements:
   a. 16 GB RAM
   b. 8 vCPU Cores
   c. 50GB of disk will be automatically provisioned for Masking Engine metadata.
   d. IP binding to a persistent address if DHCP will be used for the Masking Engine (the Masking Engine IP address should not change).
3. If you are unable to use a persistent IP address for your Masking Engine guest, the Masking Engine must be configured to use a static IP after deployment.
   a. Contact Delphix Support or your Professional Services team. This operation requires root access to the Masking Engine, and a member of these teams will need to enter the password via Webex to log in. Once they have logged you in, you can proceed with the following steps.
   b. Edit /etc/sysconfig/network-scripts/ifcfg-eth0.
   c. Change BOOTPROTO= dhcp to BOOTPROTO= static.
   d. Add a line IPADDR=<your IP address>.
   e. Add a line NETMASK=<your netmask>.
f. Edit /etc/sysconfig/network.

g. Add a line GATEWAY=<your gateway IP>

h. /etc/init.d/network restart

i. Edit /etc/resolv.conf.

j. This may have been picked up correctly by dhcp, otherwise modify the nameserver to map to the correct DNS server(s)

k. If necessary, edit the firewall.

l. By default, the firewall is disabled, meaning that /etc/init.d/iptables status shows not running. If this needs to be enabled, open the port 8282 for access to the UI as well as any ports needed to connect to their database servers.

m. Stop and start the Masking Engine from the root prompt with the following commands:

   i. cd /opt/dmsuite

      ii. ./stop_all.sh

      iii. ./start_all.sh

4. Connect to the Masking Engine at: http://<IP or DNS name>:8282/dmsuite as the user delphix_admin and password Delphix_123.

   Click here to proceed to next steps once your Masking Engine is installed and enabled.

Next Steps

- Prepare Data for Masking
- Masking Engine Activities

Related Links

- Quick Start Masking Engine Overview
- Prepare Data for Masking
- Masking Engine Activities
- Provisioning Masked VDBs
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflow
- Installing the Delphix Engine
Prepare Data for Masking

**Link a dSource**

Follow the detailed steps found in the documentation: [Link an Oracle Data Source](#)

**Provision a VDB to Prepare to Configure a Masking Job**

In order to prepare data for masking, you must first provision a virtual databases (VDBs) in the Delphix Engine. If you would like, a second may be provisioned to validate against. To do so, repeat these steps.

1. Login to the Delphix Admin application.
2. Click Manage.
3. Select Databases.
4. Select My Databases.
5. Select a dSource and a dSource snapshot.
6. Click Provision.
7. Review the information for Installation Home, Database Unique Name, SID, and Database Name. Edit as necessary.
8. Review the Mount Base and Environment User. Edit as necessary.
9. If you want to use login credentials on the target environment that are different from the login credentials associated with the Environment User, select Specify Privileged Credentials.
10. Click Next.
11. Select a Target Group for the VDB and a Snapshot Policy for the VDB.
12. Click Next.
13. Specify any Pre or Post Scripts that should be used during the provisioning process.
14. Click Advanced to select Oracle Node Listeners or enter any VDB configuration settings or file mappings.
15. Click Next.
16. Click Finish.

**Next Steps**

Masking Engine Activities

Create Data Masking Rule Sets, Algorithms, and Inventories

Mask Data

**Related Links**

- Quick Start Masking Engine Overview
- Prepare Data for Masking
- Masking Engine Activities
- Provisioning Masked VDBs
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflow
- Link an Oracle Data Source
Masking Engine Activities

- **Login to the Masking Engine**
- **User Roles**
- **Next Steps**
- **Previous Steps**
- **Related Links**

Once you have provisioned a virtual database (VDB) for masking use in the Delphix Engine, you will need to complete the following activities in the Masking Engine. The five primary tasks to be completed are:

- adding a masking application
- adding a masking environment
- adding a connector to the newly provisioned VDB
- defining the rules and profiles for masking
- creating the masking job

Below is a visualization of this data masking user workflow:

1. **Login** to the masking engine.
2. **Add an Application and Environment** to store the connection information and masking rules for data stores.
3. **Create a connector** to store database connection information.
4. **Create Masking Rule Sets** to identify, select, and configure which tables you want to mask.
5. **Mask Data** by running a masking job with data in place

**Login to the Masking Engine**

1. Login to a web browser that points to http://<server_or_IPAdress>:8282/dmsuite
2. Enter default username: **delphix_admin**.
3. Enter default user password: **Delphix_123**.
4. The auto default user role is the **Administrator** role.

**User Roles**

The Masking Engine has a built-in Administrator role, which gives you complete access to masking functions. As an administrator, you can access, update, and delete all environments, and all objects within those environments. You can also add roles in the roles settings.

**Note:** Defining new environments and connections requires different privileges than building masking jobs.

Once logged into the Masking Engine, you can complete the activities needed for masking under the **Environments** tab, seen below:
Next Steps

- Add an Application and Create a New Environment and Connector
- Create Data Masking Rule Sets, Algorithms, and Inventories
- Mask Data

Previous Steps

- Prepare Data for Masking

Related Links

- Quick Start Masking Engine Overview
- Provisioning Masked VDBs
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflow
Add an Application and Create a New Environment and Connector

- Add an Application
- Select an Environment and Create a Connector
- Next
- Related Links

Add an Application

In order to mask, you first need to add an application and create an environment to store the connection information and the masking rules for the data store.

1. Click Add Application.
2. Enter an Application Name.

3. In the upper right-hand side of the screen, click Add Environment. The screen prompts you for the following items:
   a. From the Application Name drop-down menu, select the name of the application associated with this environment, for informational purposes. An integrated test environment can have multiple applications.
   b. Enter an Environment Name.
      This will be the display name of the new environment.
   c. From the Purpose drop-down menu, select Mask.
4. Either:
   - Click **Save** to return to the **Environments List/Summary** screen,
   - or
   - Click **Save & View** to display the **Environment Overview** screen.

**Select an Environment and Create a Connector**

After you create a new environment, it will appear in the user interface (UI). You can then create connections by doing the following:

1. Click an **Environment** name.
2. From the **Environment Overview** screen, click the **Connector** tab.
   The **Connector List** screen appears.
3. In the upper right-hand region of the Connectors Editor, click **Create Connection**.
   The **Create Connection** window appears, prompting you for connection information for the database.
4. For **Type**, select the appropriate database type – for example, **Database – Oracle**.
5. Enter the required information such as the connection name, database name, host name or IP, port, and login credentials.
6. Click **Save**. You will be returned to the **Connector List** screen, where you can add additional connectors if necessary.
After you create an environment and connectors, you need to define a rule set. See the following activity for how to do this.

Next

Create Data Masking Rule Sets, Algorithms, and Inventories

Related Links

- Masking Engine Activities
- Create Data Masking Rule Sets, Algorithms, and Inventories
- Mask Data
Create Data Masking Rule Sets, Algorithms, and Inventories

- Create a Rule Set
- Edit a Rule Set
- Review Masking Inventory, Configure Columns with Domains and Algorithms
- Optional Steps (Not required)
  - Create a New Profile of Data Using the Masking Inventory
  - Profiling Data
- Next
- Mask Data
- Related Links

All rule sets need a name as well as the connector information you previously created. After you provide this information, you will see a list of tables that is owned by the schema owner defined in the connector. This enables you to select the tables you want to mask. Below is a step-by-step description for how to create a rule set.

Create a Rule Set

1. In the Environments tab, click the Rule Set tab.
2. In the upper right-hand corner of the Rule Set tab, click Create Rule Set.
   
   The Create Rule Set screen appears. This screen lets you specify which tables belong in the rule set.

   Note: A similar screen appears when you edit a rule set.

   ![Create Rule Set Screen](image)

Edit a Rule Set

1. Click the name of an environment.
2. Click the Rule Set tab.
3. In the upper right-hand corner of the Rule Set tab, click the Edit (pencil) icon for the rule set you want to edit.
   The Create Rule Set screen appears, allowing you to specify which tables belong in the Rule Set.
   a. Enter a Name for your rule set.
   b. Select a Connector name from the drop-down menu.
      The list of tables for that connector appears.
   c. To select individual tables, click their names in the list to the right. Alternatively, click Select All in the bottom left to select all the tables.
   d. Click Save.
      You are returned to the Rule Set screen.

4. To see the list of tables that you selected, click the name of the newly-created rule set.

5. Optionally, for each table, if there is no primary key for that table, click Edit Table and define the logical key, as seen in the screenshot below:
The following section describes how to define the columns to mask for each table in the rule set.

**Review Masking Inventory, Configure Columns with Domains and Algorithms**

After selecting tables (connectors) and defining a rule set, which you did in the preceding section, you are ready to configure the columns in the masking inventory. When you defined a rule set, an empty inventory was automatically created. Manually edit the columns with sensitive data to assign a domain and algorithm by clicking the pencil icon at the end of the inventory row.

1. Click **Inventory**.
2. Select the **rule set** you want to mask.
3. Click **Contents** to list all the tables or files defined for the rule set.
4. Select a **table**. All the contents in the table will appear.
   - If a column is a primary key (PK), a Foreign Key (FK), or an index (IDX), an icon indicating this will appear to the left of the column name. If there is a note for the column, a Note icon will appear. To read the note, click the icon.
   - If a table, metadata for the column appears: **Data Type** and **Length** (in parentheses). This information is read-only.
5. To mask or unmask a column, click **Edit** and choose the appropriate domain for the column.
6. The selected domain will open a corresponding algorithm. If needed, you can select a different algorithm from the drop-down list. Based on the column you are masking, you may decide to choose one of the following three algorithms:
   - Secure Lookup Algorithm
   - Segmented Mapping Algorithm
   - Secure Shuffle Algorithm
7. Complete the presented form to the right that corresponds to your selected algorithm.
8. Click **Save**.

If you have already selected a column to be masked and you want to undo this:

1. Click the **Edit** (pencil) icon for that column.
2. In the **Edit Properties** dialogue box, reset the **DOMAIN** element to **Choose Domain**. You will exclude that column from being masked. Please refer to the screenshot below:
Optional Steps (Not required)

Create a New Profile of Data Using the Masking Inventory

If you are unsure what data you want to mask, a good practice is to profile data by updating an inventory of your data with sensitive data elements identified. You can then review and edit that inventory.

1. From your environment's Overview tab, click the Profile button. The Create Profile Job screen will appear.
2. Enter a Job Name.
3. Select your rule set.
4. Select a profile set, which will be created when you set up your masking security policy or one of the provided profile sets.
5. Click Save. The job will appear on your overview screen.
Profiling Data

1. Create a **profiling job** using the steps above.
2. Run the profiling job you just created. When you run this profiling job, it updates/populates an inventory.
3. Click the **Profile Job** name to see the results.
4. Click the **Results** sub-tab. You should see the name columns appear as sensitive.

5. If you want to download a PDF report of the sensitive fields that have been profiled, click the profiling report.

6. To view the inventory, click the **Inventory** tab while in an **Environment Overview**.

7. Examine the inventory to ensure that the profiling job has included everything you want to mask. For example, if you selected a First Name field, you probably want the Last Name field as well. You can see which columns were selected for masking by selecting the associated rule set. Make sure that you have included all sensitive data elements, such as personal identifying information, from the table that you want to mask.

8. Modify the inventory, if necessary.

When a profiling job runs, it automatically updates the inventory for the given rule set. If you do not want the Profiler to automatically update the inventory, change the ID Method to **User**.

**Next**

**Mask Data**

**Related Links**

- Masking Activities
- Add an Application and Create a New Environment and Connector
- Mask Data
Mask Data

- Create a New Masking Job
- Run a New Masking Job
- Validate a New Masking Job
- Next Steps
- Related Links

Create a New Masking Job

Now that you have identified and prepared your data for masking, you can create a new masking job and confirm it.

1. Select Mask.
   The Create Masking Job window appears.
2. Choose appropriate rule settings. You will be prompted for the following information:
   - **Job Name** — A free-form name for the job you are creating. Must be unique across the entire application.
   - **Masking Method** — Select In-Place.
   - **Multi Tenant** — Check this box to allow this job to be used with multiple databases.
   - **Rule Set** — Select a rule set against which this job will execute.

   ![Provision Masked VDBs](image)

   - **Rule Set** — Select a rule set against which this job will execute.

2. When you are finished, click **Save**.

Run a New Masking Job

1. Under **Action**, click **Run** to run the masking job.
2. Click the **job** to view the progress of the masking job.

Validate a New Masking Job

If you want to validate the masking job, compare the VDB you have masked to another copy of the data, such as another VDB. Check the results of the masking job by using a SQL tool such as SQL Developer to check that the fields were masked.

You can use the **Certify** button to create a masking certification job that will check for ‘polluted’ columns – for example, columns that contain a value that is not in the secure lookup table. You can also use sqlplus on the target VDB machine to check the masked fields.

Next Steps
Provisioning Masked VDBs

Related Links

- Quick Start Masking Engine Overview
- Masking Engine Activities
- Add an Application and Create a New Environment and Connector
- Create Data Masking Rule Sets, Algorithms, and Inventories
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflow
Provisioning Masked VDBs

- Prerequisites
- Ways to Run a Masking Job on a VDB
- Identifying and Navigating to Masked VDBs in the Delphix Engine
- Restrictions
- Workflow Overview to Provision Masked VDBs in the Delphix Admin Console
  - Associate a Masking Job with the dSource
  - Provisioning a Masked VDB using the dSource Provisioning Wizard
- Refresh a Masked VDB
- Disassociate a Masking Operation on a dSource
  - Masked VDB Data Operations
  - Continue to Next Steps
- Related Links

Masked virtual databases (VDBs) function just like normal VDBs. The only distinction is that the data they contain has been masked by a masking job. Masked VDBs can be selectively distributed to a separate Delphix Engine (in non-prod) without sending the original data that was obfuscated during masking. This topic describes how to work with masked VDBs.

Prerequisites

<table>
<thead>
<tr>
<th>Delphix Masking Pre-Configuration Activities</th>
</tr>
</thead>
<tbody>
<tr>
<td>If you are configuring Delphix Masking for the first time, you must complete all of the activities below in order.</td>
</tr>
</tbody>
</table>

1. Install the Combined OVA.
2. Prepare Your Data.
3. Configure, Create, and Test a Simple Masking Job.
   a. Add an Application.
   b. Create Data Masking Rule Sets.

VDB Snapshot Required
Take a VDB snapshot before masking data. This is required to bring the changes into Delphix if you are going to be provisioning masked VDBs.

c. Mask Data.

Mask Data for Provisioned Masked VDBs
A masking job must be Multi Tenant to use it when creating a masked virtual database (VDB).
Ways to Run a Masking Job on a VDB

You will always need to configure and test new masking jobs in the Delphix Masking Engine as described in the Prerequisites. Once a it has been completely defined, you can run the masking job in the following ways:

- Working in the Delphix Engine Admin Console using the dSource Provisioning Wizard
- Working in the Delphix Masking Engine
- Using the Masking API to Run a Masking Job Command

Identifying and Navigating to Masked VDBs in the Delphix Engine

Masked VDBs appear on the left sidebar, just like regular VDBs. They are most obviously identified by the different icon used to represent them on the left sidebar. A masked VDB will also contain information about the masking job that was applied to it on the database card. Generally, everything that can be done with an unmasked VDB is also possible with a masked VDB.

<table>
<thead>
<tr>
<th>Masked VDB Icon on Left Sidebar</th>
<th>Masked VDB info in Database Card</th>
</tr>
</thead>
<tbody>
<tr>
<td>Untitled</td>
<td><img src="image1.png" alt="Masked VDB Icon on Left Sidebar" /></td>
</tr>
<tr>
<td>dbdhcp1--N277-1451914512260</td>
<td><img src="image2.png" alt="Masked VDB info in Database Card" /></td>
</tr>
<tr>
<td>dbdhcp1--ZRN-1451914716107</td>
<td></td>
</tr>
</tbody>
</table>

Restrictions

- Provisioning masked VDBs through the Delphix Engine does not currently work with DB2. In order to mask DB2, you should use the Masking Engine interface.
- You cannot apply additional masking jobs to a masked VDB or its children.
- If a masking job has been applied to a VDB, you cannot create an unmasked snapshot of that VDB.
- If an existing VDB has not had a masking job applied to it, then you cannot mask that particular VDB at any point in the future. All the data within the VDB and its parents will be accessible if it is replicated or distributed.

Workflow Overview to Provision Masked VDBs in the Delphix Admin Console

1. Associate masking jobs with a dSource in the Admin console.
2. Use dSource provision wizard to provision a VDB with a masking job.

Associate a Masking Job with the dSource

To provision a masked VDB, you must first indicate that the masking job you are using is complete and applicable to a particular database. You do this by associating the masking job with a dSource.
1. Open the database card for the dSource to which the masking job is applicable and with which it will be associated.

2. Click the **Masking** tab.

3. Click the **pencil** icon to edit. All masking jobs on this Delphix Engine that have not been associated with another dSource will be listed on the right-hand side.

4. Select the **job** you want to associate with this dSource.

5. Click on the left-facing **arrow**.

6. Repeat for any other jobs that you want to associate with this dSource at this time.

7. Click the yellow **checkmark** to confirm.

The Delphix Engine now considers this masking job to be applicable to this dSource and ready for use. When provisioning from snapshots of this dSource, this masking job will now be available.

Note: Masking jobs can also be associated with virtual sources in addition to dSources.
Provisioning a Masked VDB using the dSource Provisioning Wizard

The steps required to provision a masked VDB are almost identical to the steps required to provision an unmasked VDB. Once you have created a masked VDB, you cannot un-mask it, nor can you alter which masking job it uses. All snapshots in the VDB’s timeflow will always be masked using the masking job that you selected when you provisioned the masked VDB.

1. Select the dSource.
2. Click Provision.
3. Review the information for Installation Home, Database Unique Name, SID, and Database Name. Edit as necessary.
4. Review the Mount Base and Environment User. Edit as necessary.
   a. If you want to use login credentials on the target environment that are different from the login credentials associated with the Environment User, select Specify Privileged Credentials.
5. Click Next.
6. If necessary, edit the Target Group for the VDB and Snapshot Policy settings.
7. Select the masking job you want to use from the available drop down menu. Only masking jobs that have been associated with the parent dSource will be available.
8. Click Next.
9. Specify any Pre or Post Scripts that should be used during the provisioning process. If the VDB was configured before running the masking job using scripts that impact either user access or the database schema, they should also be used here.
10. Click Next.
11. Click Finish.

In the Actions sidebar on the right-hand side of the window, there will be an action indicating that masking is running. You can verify this and monitor progress by going to the Masking Engine page and clicking the Monitor tab.

Once you have created a masked VDB, you can provision its masked data to create additional VDBs, in the same way that you can provision normal VDBs. Since the parent masked VDB contains masked data, descendent VDBs will only have masked data. This is a great way to distribute multiple independent copies of masked data that is both time and space efficient.

Refresh a Masked VDB

You refresh a masked VDB in exactly the same way as you refresh a normal VDB. As with provisioning a masked VDB, the masking job will be run during the refresh process.

1. Login to the Delphix Admin application.
2. Under Databases, select the VDB you want to refresh.
3. From the database card, click the Refresh icon in the lower right-hand corner.
4. You will then be brought to the timeflow of the dSource from which the VDB was provisioned.
5. Select the snapshot or point in time to which you want to refresh.
6. Click Refresh.

Delphix will now update the masked VDB with the new data and mask it using the masking job with which this Masked VDB was provisioned.

**Disassociate a Masking Operation on a dSource**

If a masking job is found to be unsuitable or should be retired, you can disassociate it though the same database card that was used to associate it.

1. Select the job on the left-hand side to disassociate.
2. Click the right-facing arrow, as seen in the screenshot above.

It is important to note that this will only prevent the creation of new Masked VDBs with this job. It will not alter existing Masked VDBs in any way. When disassociating a job, review the existing masked VDBs and consider whether you need to delete or disable any of them.

Data Operations available to Masked VDBs are listed below:

<table>
<thead>
<tr>
<th>Masked VDB Data Operations</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Rewind</strong></td>
<td>Alter the database to contain masked data from a previous point in time.</td>
</tr>
<tr>
<td><strong>Refresh</strong></td>
<td>Get new data from the parent dSource and mask it.</td>
</tr>
<tr>
<td><strong>Disable</strong></td>
<td>Turn off the database and remove it from the host system.</td>
</tr>
<tr>
<td><strong>Enable</strong></td>
<td>Turn on the database and make it available on the host system.</td>
</tr>
</tbody>
</table>

**Procedure to Rewind an Oracle Masked VDB**

**Procedure to Refresh an Oracle Masked VDB**

**Procedure to Disable an Oracle Masked VDB**

**Procedure to Enable an Oracle Masked VDB**

**Continue to Next Steps**

Troubleshoot Provisioning Errors for Masked VDBs

Advanced Integrated Delphix Masking Workflows

**Related Links**

- Troubleshoot Provisioning Errors for Masked VDBs
- Quick Start Masking Engine Overview
- Masking Engine Activities
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflow
• Masking Engine Install, System Configuration, and Network Setup
• Prepare Data for Masking
• Add an Application and Create a New Environment and Connector
• Create Data Masking Rule Sets, Algorithms, and Inventories
Troubleshoot Provisioning Errors for Masked VDBs

Avoiding and Handling Errors

It is possible that an error might occur when a masking job is run. This will cause the refresh or provision to fail, as can be seen by a failure in the actions sidebar.

If the error message is insufficient to diagnose the problem, you can view the full Masking Engine logs.

1. From the Masking Engine page, click Admin.
2. On the left-hand side of the screen, click Logs.

Related Links

- Provisioning Masked VDBs
- Masking API Calls to Run a Masking Job
Masking API Calls to Run a Masking Job

A limited REST API exists only to implement integration with the Delphix Engine and is not officially supported. The following procedure details the single API to launch existing masking jobs.

Procedure

1. Login User – GET dmsuite/apiV4/login?user={userID}&password={encrypted passwd}
   a. Returns authorization token in HTTP header that should be used in subsequent operations

2. Get Application – GET /dmsuite/apiV4/applications
   a. Returns applications and environments associated to each application in response body – for example:

   ```xml
   <ApplicationsResponse>
   <ResponseStatus>
   <Status>SUCCESS</Status>
   </ResponseStatus>
   <Applications>
   <Application>
   <Name>demo</Name>
   <Link href="applications/demo" rel="details"/>
   <Environments>
   <Link href="environments/1" rel="SAP"/>
   </Environments>
   <Environments>
   <Link href="environments/37" rel="TEST"/>
   </Environments>
   </Application>
   </Applications>
   </ApplicationsResponse>
   
   For on-the-fly masking, pass the target connector in the request body - environments/{environmentID}/connectors/{connectorId}?DataSource={Database,File,Mainframe}

3. Get Job – GET dmsuite/apiV4/applications/{applicationID}/jobs
   a. Returns jobs in response body – for example:

   ```xml
   <JobsResponse>
   <ResponseStatus>
   <Status>SUCCESS</Status>
   </ResponseStatus>
   <Jobs>
   <Profiles/>
   <Profile>
   <Name>OracleProfile</Name>
   <Link rel="details" href="applications/demo/profilejobs/0"/>
   <Status>Succeeded</Status>
   </Profile>
   <Profiles/>
   <Provisions/>
   <Maskings>
   <Masking>
   <Name>OracleMasking</Name>
   <Link rel="details" href="applications/demo/maskingjobs/1"/>
   <Status>Succeeded</Status>
   </Masking>
   <Maskings/>
   <Certifys/>
   </Jobs>
   </JobsResponse>
   
   For on-the-fly masking, pass the target connector in the request body - environments/{environmentID}/connectors/{connectorId}?DataSource={Database,File,Mainframe}

4. Run Job – POST dmsuite/apiV4/applications/{applicationID}/maskingjobs/{maskingjobID}/run
   a. Returns job launch status in response body - for example:

   ```xml
   <MaskingsResponse>
   <ResponseStatus>
   <Status>SUCCESS</Status>
   </ResponseStatus>
   </MaskingsResponse>
   ```
5. Get Job status - GET dmsuite/apiV4/applications/{applicationID}/maskingjobs/{maskingjobID}/results
   a. Returns job run status in response body - for example:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<MaskingsResponse>
  <ResponseStatus>
    <Status>SUCCESS</Status>
  </ResponseStatus>
  <Maskings>
    <Masking>
      <Name>OracleMasking</Name>
      <Status>SUCCESS</Status>
      <StartTime>16:04:00</StartTime>
      <Duration>00:01:12</Duration>
      <PreviousDuration>00:01:13</PreviousDuration>
      <TablesTotal>1</TablesTotal>
      <TablesProcessed>1</TablesProcessed>
      <TablesRemaining>0</TablesRemaining>
      <RowsProcessed>25</RowsProcessed>
      <RowsRemaining>0</RowsRemaining>
      <UpdatesRunning>4</UpdatesRunning>
      <Repository>POSTGRESQL</Repository>
      <Links>
        <Link href="environments/1/connectors/1?DataSource=database" rel="SourceConnector"/>
        <Link href="applications/demo/inventorys/Oracle?environmentId=1" rel="Inventory"/>
      </Links>
    </Masking>
  </Maskings>
</MaskingsResponse>
```

Related Links

- Quick Start Masking Engine Overview
- Masking API Calls to Run a Masking Job
- Masking Engine Activities
- Provisioning Masked VDBs
- Masking API Calls to Run a Masking Job
- Advanced Integrated Delphix Masking Workflow
Advanced Integrated Delphix Masking Workflows

- Using Selective Data Distribution with Masked Data Sources
- Using Jet Stream with Masked Data Sources
- Related Links

Using Selective Data Distribution with Masked Data Sources

You can now replicate masked data in a VDB directly to a target Delphix engine without transmitting the unmasked data in its parent source. To learn more or to get started, go to:

- Selective Data Distribution Overview
- Selective Data Distribution Use Cases
- Selective Data Distribution User Interface
- Configuring Selective Data Distribution
- Selective Data Distribution and Failover

Using Jet Stream with Masked Data Sources

The workflows below detail both best practices and troubleshooting for how an admin user can select and use masked data sources within Jet Stream. The figure below shows a dSource which has two masking jobs associated with it, two masked VDBs provisioned from the dSource each masked with a different masking job, and two VDBs provisioned from the masked VDBs. With this set of objects, an admin user would place the parent masked VDBs in a data template and use the child masked VDBs in data containers.

In order to create a data container with masked data as an admin user:

1. Provision a child masked VDB from the masked parent VDB.
2. Select the masked child VDB as a source for the container.
To learn more or to get started, go to the Jet Stream Admin Guide and follow the procedures for the following activities:

- Selecting Masked Data Sources in Data Templates
- Selecting Masked Data Sources for Data Containers

Related Links

- Selective Data Distribution Overview
- Selective Data Distribution Use Cases
- Selective Data Distribution User Interface
- Configuring Selective Data Distribution
- Selective Data Distribution and Failover
- Jet Stream Admin Guide
- Selecting Masked Data Sources in Data Templates
- Selecting Masked Data Sources for Data Containers