

# Quick Start for HVR - HANA

Contents
<ul style="list-style-type: none"><li>• <a href="#">Create Demo Schemas and Tables</a></li><li>• <a href="#">Create Hub Database</a></li><li>• <a href="#">Grants/Access Privileges</a></li><li>• <a href="#">Download, Install, and Configure HVR</a></li><li>• <a href="#">Launch HVR GUI</a></li><li>• <a href="#">Register Hub</a></li><li>• <a href="#">Create Locations</a></li><li>• <a href="#">Create Channel</a></li><li>• <a href="#">Create Location Groups</a></li><li>• <a href="#">Define Actions</a></li><li>• <a href="#">Select Table(s)</a></li><li>• <a href="#">Initialize</a></li><li>• <a href="#">Start Scheduler</a></li><li>• <a href="#">Start Capture Job</a></li><li>• <a href="#">Refresh</a></li><li>• <a href="#">Start Integrate Job</a></li><li>• <a href="#">Verify Replication</a></li></ul>

This quick start guide helps you to get started with HVR for replicating data between SAP HANA databases.

To proceed with this replication you must have basic understanding about HVR's [architecture](#) and [terminology](#) like Hub, Location, Channel, Location Groups, and Actions.

The example here demonstrates how to replicate tables from a HANA database on a source location to a HANA database on a target location. The HVR hub database will be located on an Oracle server because HVR does not support Hub database on HANA. In real-life scenarios, the source location(s) and the target location(s) reside on different machines and the HVR hub can reside on source or target or a separate machine. In this demonstration, for simplicity we have the source and target databases on the same machine and the HVR hub will be located on a remote Oracle machine.

For HVR hub to connect to the remote source and target locations, an HVR remote agent should be installed on the HANA server.

Before proceeding with this demonstration, ensure that the requirements for using HVR with HANA and Oracle are met.

For information about access privileges and advanced configuration changes required for performing replication using HANA and Oracle, see:

- [Requirements for HANA](#)
- [Requirements for Oracle](#)

## Create Demo Schemas and Tables

The initial steps of this demonstration is to create:

- a schema in the source location
- column-storage tables in the source schema (and insert values into these tables)

## Source Location

Skip this section if you already have a database with tables which you plan to use for this replication.

For this demonstration, in the source location database (e.g. **testdb**), create a schema (e.g. **sourcedb**) and two tables (e.g. **product** and **orders**) and insert values into these tables.

Sample SQL statements to create a schema and tables in source location, and also to insert values into these tables

## Create Schema in Source Location

Create source schema (**sourcedb**):

```
create schema sourcedb owned by hvruser;
```

HVR supports capturing only from column-storage tables in HANA. For this demonstration, create two column-storage tables called **product** and **orders**. Create Tables in Source Schema

```
create COLUMN table sourcedb.product
(
  prod_id number(10) not null,
  prod_price number(10,2) not null,
  prod_descrip varchar2(100) not null
);
```

```
create COLUMN table sourcedb.orders
(
  prod_id number(10) not null,
  ord_id number(10) not null,
  cust_name varchar2(100) not null,
  cust_addr varchar2(100)
);
```

## Insert Values in Source Tables

```
insert into sourcedb.product
values (100, 90, 'Book');
```

```
insert into sourcedb.orders
values (100, 123, 'Customer1', 'P.O. Box 122, Anytown, Anycountry');
```

## Target Location

HVR automatically creates tables in the target database during [HVR Refresh](#), and it is the recommended method for creating tables in the target schema.

## Create Hub Database

This section describes how to create a hub database (schema). The hub database is a repository database that HVR uses to control its replication activities. It contains HVR [catalog tables](#) that hold all specifications of replication such as the names of the replicated databases, the replication direction and the list of tables to be replicated. For more information about HVR hub server and database, see section [Hub Server](#) in [System Requirements](#).

HVR supports the creation of a hub database on certain databases (location classes) only. For the list of supported location classes, see section [Hub Database](#) in [Capabilities](#).

For this demonstration, the hub database (e.g. **hvrhub**) is created in Oracle.

- Create the hub database (**hvrhub**) with password (**hvr**).

```
create user hvrhub
identified by hvr
default tablespace users
temporary tablespace temp
quota unlimited on users;
```

## Grants/Access Privileges

This section describes the grants/access privileges required for the hub, source, and target databases.

1. Configure the privileges for the Oracle hub database (**hvrhub**). For more information, see section [Grants for Hub Schema](#) in [Requirements for Oracle](#).

```
grant create session to hvrhub;  
grant create table to hvrhub;  
grant create procedure to hvrhub;  
grant create trigger to hvrhub;  
grant execute on dbms_alert to hvrhub;
```

A user with **dba** privileges should grant **execute on dbms\_alert** to the hub database user (e.g. *hvruser*).

2. Configure the privileges for the HANA source database (**sourcedb**). For more information, see section [Grants for Capture](#) in [Requirements for HANA](#).

The following grants are required for capturing changes from HANA:

- To read from tables which are not owned by HVR User (using **TableProperties/Schema**), the **User** must be granted select privilege.  
grant select on *tbl* to *hvruser*

```
grant select on tbl to hvruser
```

- The **User** should also be granted select permission from some system table and views. In HANA, however, it is impossible to directly grant permissions on system objects to any user. Instead, special wrapper views should be created, and **User** should be granted read permission on this views. To do this:
  - a. Connect to the HANA database as user **SYSTEM**.
  - b. Create a schema called **\_HVR**.

```
create schema _HVR;
```

- c. Grant **SELECT** privilege on this schema to **User**.

```
grant select on schema _HVR to hvruser;
```

- d. Execute the **hvrhanaviews.sql** script from the **\$HVR\_HOME/sql/hana** directory. This will create a set of views that HVR uses to read from system dictionaries on HANA.
3. Configure the privileges for the HANA target database (**targetdb**). For more information, see section [Grants for Integrate and Refresh Target](#) in [Requirements for HANA](#).

The following grants are required for integrating changes into HANA:

- The **User** should have permission to read and change replicated tables

```
grant select, insert, update, delete on tbl to hvruser
```

- The **User** should have permission to create and alter tables in the target schema

```
grant create any, alter on schema schema to hvruser
```

- The **User** should have permission to create and drop HVR state tables

# Download, Install, and Configure HVR

This section describes how to download HVR, install it on either of Windows, Linux, or Unix machines, as well as configure an HVR remote agent for HVR hub to connect to the remote source and target locations.

## 1. Download and Install HVR

a. For this demonstration, an HVR distribution should be installed:

- On the HANA machine. This will be an HVR Remote agent for capture and integrate. This HVR remote agent will establish a connection to the HANA database and the hub can connect to it via a running [HVR remote listener](#).

If the source and target databases are on different machines, the HVR remote agent should be installed on both source and target location machines.

- On the hub machine.

An HVR distribution is available for download at <https://www.hvr-software.com/account/>. To request a trial version, visit <https://www.hvr-software.com/free-trial/>.

Install HVR on a hub machine. For details on installing HVR, see the respective operating system sections:

- [Installing HVR on UNIX or Linux](#)
- [Installing HVR on Windows](#)
- [Installing HVR on macOS](#)

The HVR distribution requires a license key in order for the software to operate. Please see the HVR [licensing page](#) for more details on how to install the HVR license.

## 2. Configure HVR remote agent

After installing the HVR remote agent on the HANA machine, it needs to be configured to connection from the HVR hub. For the detailed configuration steps, see the following sections:

- [Configuring Remote Installation of HVR on Unix or Linux](#)
- [Configuring Remote Installation of HVR on Windows](#)

After the installation, you can control HVR using the HVR graphical user interface ([HVR GUI](#)).

- If the hub machine is Windows, then [HVR GUI](#) can be executed directly on the hub machine.
  - To control HVR remotely from your PC, connect to the hub machine using Windows Remote Desktop Connection and launch [HVR GUI](#) on the hub machine.
- If the hub machine is Linux, then [HVR GUI](#) can be executed directly on the hub machine. However, an application like X Server or VNC viewer must be installed to run [HVR GUI](#) directly on Linux.
  - To control HVR remotely from your PC, install HVR on the PC (with Windows or macOS) and configure the [HVR Remote Listener](#) on the hub machine.
- If the hub machine is Unix, then [HVR GUI](#) should typically be run remotely from a PC to control HVR installed on the hub machine. To do this, install HVR on the PC (with Windows or macOS) and configure the [HVR Remote Listener](#) on the hub machine.

The [HVR Remote Listener](#) allows you to connect [HVR GUI](#) available on your PC to the remote HVR hub machine.

## Launch HVR GUI

This section describes how to launch [HVR GUI](#) on various operating systems.

- On Windows and macOS, double-click the HVR shortcut icon available on the desktop or execute command **hvrgui** in the CLI.
- On Linux, double-click the hvrgui file available in the HVR\_extracted\_path/bin directory or execute command **hvrgui** in the CLI.

Linux requires applications like X server or VNC viewer to execute **HVR GUI**.

- On Unix, **HVR GUI** is not supported. So, **HVR GUI** should be run on a remote PC (with Windows, Linux, or macOS) to control HVR installed on the Unix machine.

## Register Hub

This section describes how to connect [HVR GUI](#) to the hub database.

When you launch HVR GUI for the first time, the **Register Hub** dialog is displayed automatically. The **Register Hub** dialog can also be accessed from menu **File** by selecting **Register Hub**. Skip steps 1 to 4 if you want to run **HVR GUI** directly on the hub machine.

1. Click **Connect to HVR on remote machine**.

To connect **HVR GUI** on a PC to a remote HVR hub machine, the **HVR Remote Listener** must be configured and running on the HVR hub machine.

2. Enter the name or IP address of the hub machine in the **Node** field (e.g. **myserver**).
3. Enter the port number (defined in the **HVR Remote Listener** of the hub machine) in the **Port** field (e.g. **4343**).
4. Enter the **Login** (e.g. **myserveradmin**) and **Password** for the hub machine. By default, this is the operating system login credentials of the hub machine.
5. Select **Oracle** in the **Class** pane.
6. Specify **Database Connection** details.
  - a. Enter the directory path in **ORACLE\_HOME**. You can also click the browse button to select the directory path.
  - b. Enter the Oracle System ID in **ORACLE\_SID** or **TNS** credentials.
  - c. Enter the user name of the hub database in **User** (e.g. **hvrhub**).
  - d. Enter the password for the hub database in **Password** (e.g. **hvr**).

7. Click **Connect**.

8. Click **Yes** in the prompt dialog asking to create catalog tables in the hub database.

HVR displays this prompt when connecting to a hub database for the first time.

On connecting successfully to the hub database, the navigation tree pane displays the hub machine and the hub database. **Location Configuration**, **Channel Definitions**, and **Scheduler** are displayed under the hub database.

## Create Locations

This section describes how to create locations in HVR GUI. Location is a storage place (for example, database or file storage) from where HVR captures (source location) or integrates (target location) changes.

- Create one source location connected to the source schema (**sourcedb**).
  1. In navigation tree pane, right-click **Location Configuration New Location**.
  2. Enter **Location** name (e.g. **src**) and **Description** for the location.
  3. Click **Connect to HVR on remote machine**, because HVR needs to connect to the HVR remote agent installed. Enter the details of the virtual machine running the HVR Remote Agent there (in our example, the HVR Remote Agent runs on the same machine as the HANA server).
    - a. Enter the name or IP address of the HVR remote agent machine in the **Node** field (e.g. **myserver**).
    - b. Enter the port number (defined in the [HVR Remote Listener](#) of the HVR remote agent machine) in the **Port** field (e.g. **4343**).
    - c. Enter the **Login** (e.g. **hvr**) and **Password** for the HVR remote agent machine. By default, this is the operating system login credentials of the HVR remote agent machine.

The HVR agent on the HANA server needs to run an HVR Remote Listener on port **4343** for this connection to work.
  4. Select **HANA** in **Class**.
  5. Provide **Database Connection** details. For more information on **Database Connection** fields, see section [Location Connection](#).
    - a. Enter the hostname or ip-address of the HANA server in **Node**.
    - b. Select the **Mode** as **Multiple containers - Tenant database**.
    - c. Enter the database **Instance Number**.
    - d. Enter the the specific name of the **Database** in a multiple-container environment. For example, **sourcedb**.
    - e. Enter the username for **Database** in **User**.
    - f. Enter the password for **User** in **Password**.
  6. Click **Test Connection** to verify the connection to location database.

7. Click **OK**.

- Create one target location connected to the target schema (**targetdb**).
  1. In the navigation tree pane, right-click **Location Configuration New Location**.
  2. Enter **Location** name (e.g. **tgt**) and **Description** for the location.
  3. Click **Connect to HVR on remote machine**, because HVR needs to connect to the HVR remote agent installed. Enter the details of the virtual machine running the HVR Remote Agent there (in our example, the HVR remote agent runs on the same machine as the HANA server).
    - a. Enter the name or IP address of the HVR remote agent machine in the **Node** field (e.g. **myserver**).
    - b. Enter the port number (defined in the [HVR Remote Listener](#) of the HVR remote agent machine) in the **Port** field (e.g. **4343**).
    - c. Enter the **Login** (e.g. **hvr**) and **Password** for the HVR remote agent machine. By default, this is the operating system login credentials of the HVR remote agent machine.

The HVR agent on the HANA server needs to run an HVR Remote Listener on port 4343 for this connection to work.
  4. Select **HANA** in **Class**.
  5. Provide **Database Connection** details. For more information on **Database Connection** fields, see section [Location Connection](#).
    - a. Enter the hostname or ip-address of the HANA server in **Node**.
    - b. Select the **Mode** as **Multiple containers - Tenant database**.
    - c. Enter the database **Instance Number**.
    - d. Enter the the specific name of the **Database** in a multiple-container environment. For example, **targetdb**.
    - e. Enter the username for the **Database** in **User**.
    - f. Enter the password for the **User** in **Password**.
  6. HVR only supports capture from HANA on Linux. For a default installation of the ODBC Driver Manager Library, HVR will automatically determine the default path to the library and the **odbc.ini** and **odbcinst.ini** files. Otherwise, the directories must be specified in the **Driver Manager Library** and **ODBCSYSINI** fields. For more information on **Linux** fields, see section [Location Connection](#).
  7. Click **Test Connection** to verify the connection to location database.

8. Click **OK**.

## Create Channel

This section describes how to create a channel (e.g. **hvrdemo**) in HVR. A channel groups together the locations and tables that are involved in replication. It also contains actions that control the replication.

1. In the navigation tree pane, right-click **Channel Definitions** **New Channel**.
2. In the **New Channel** dialog, enter **Channel** name and **Description** for the channel.
3. Click **OK**.

## Create Location Groups

This section describes how to create location groups in a channel. The location groups are used for defining actions on the location. Typically, a channel contains two location groups - one for the source location and one for the target location. Each location group can contain multiple locations.

For this demonstration, create one source location group (e.g. **SRCGRP**) and one target location group (e.g. **TGTGRP**).

1. In the navigation tree pane, click **+** next to the channel (**hvrdemo**).
2. Create source location group (**SRCGRP**):
  - a. Right-click **Location Groups** **New Group**.
  - b. Enter **Group Name** and **Description** for the location group.

- c. Select source location (**src**) in **Group Membership**. Click **OK**.

Location Group SRC

Channel: hvrdemo

Group Name: SRCGRP

Description: Source location group

	Location	Class	Node	Description
<input checked="" type="checkbox"/>	src	hana	sap_hana_server	SAP HANA source location
<input type="checkbox"/>	tgt	hana	sap_hana_server	SAP HANA target location

Buttons: OK, Cancel, Help

3. Create target location group (**TGTGRP**):
- Right-click **Location Groups** **New Group**.
  - Enter **Group Name** and **Description** for the location group.
  - Select target location (**tgt**) in **Group Membership**. Click **OK**.

Location Group TGT

Channel: hvrdemo

Group Name: TGTGRP

Description: Target location group

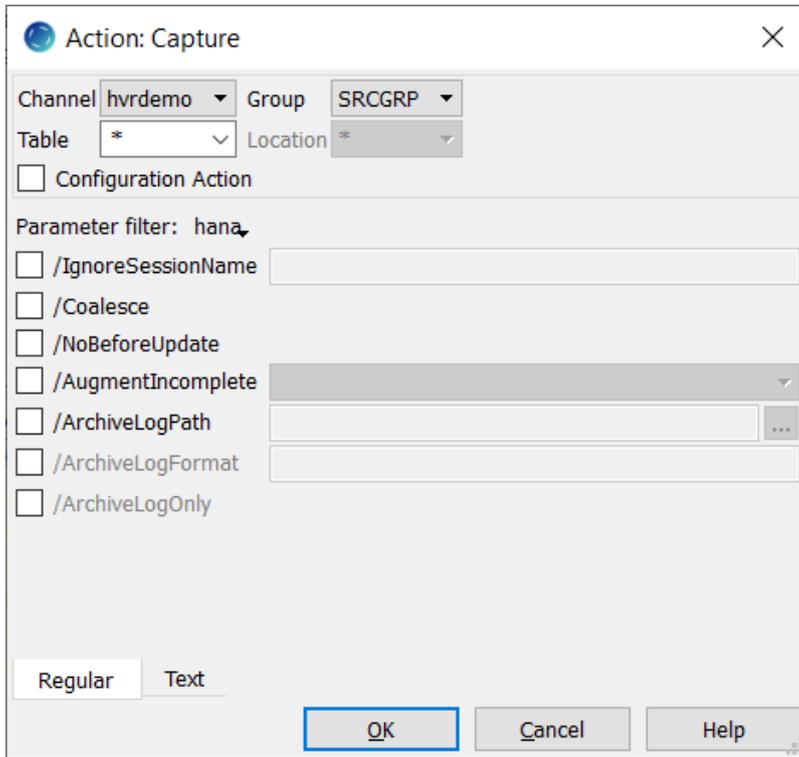
	Location	Class	Node	Description
<input checked="" type="checkbox"/>	tgt	hana	sap_hana_server	SAP HANA target location
<input type="checkbox"/>	src	hana	sap_hana_server	SAP HANA source location

Buttons: OK, Cancel, Help

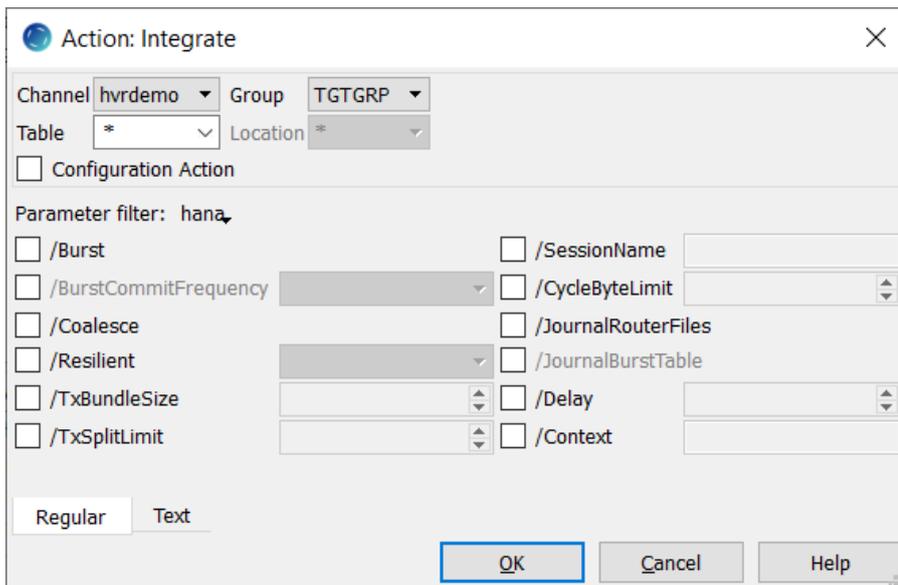
## Define Actions

This section describes how to define **actions** on the location groups (**SRCGRP** and **TGTGRP**). Actions define the behavior of a replication activity.

- Define action **Capture** to capture changes from all tables in the source location group.
  - Right-click source location group **SRCGRP** **New Action** **Capture**. Click **OK**.



2. Define action **Integrate** to integrate changes into all tables in the target location group.
  - a. Right-click target location group **TGTGRP** **New Action Integrate**. Click **OK**.

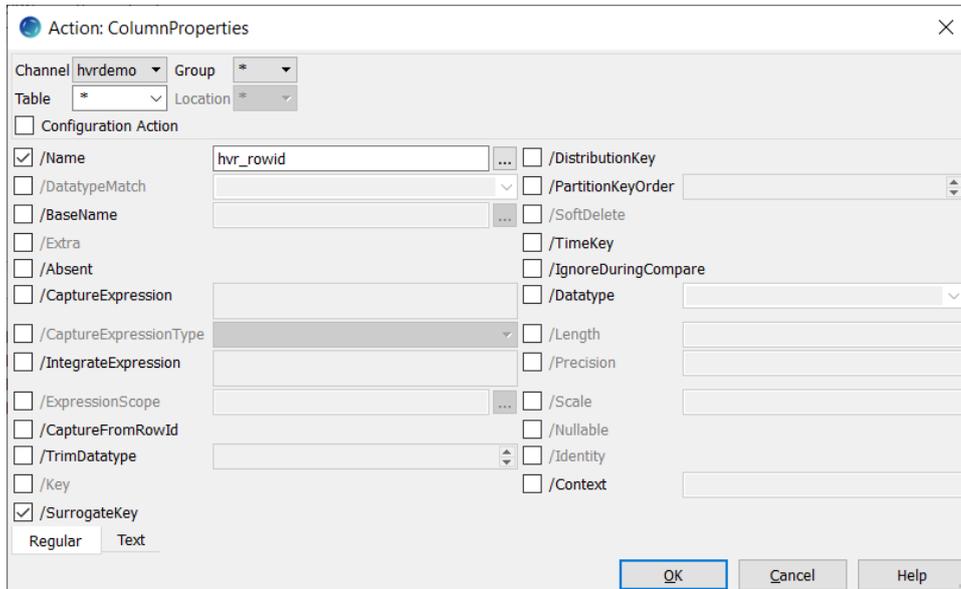


3. For a HANA location, HVR requires a temporary staging directory to store replicated data before loading it to a target database. Therefore, on both source (**SRCGRP**) and target (**TGTGRP**) location groups, define action **LocationProperties** with parameters **/StagingDirectoryHvr** and **/StagingDirectoryDb** to specify the directories for loading staging files:
  - **/StagingDirectoryHvr** is the location where HVR will create the temporary staging files, this should be the directory on the machine, from which HVR connects to HANA.

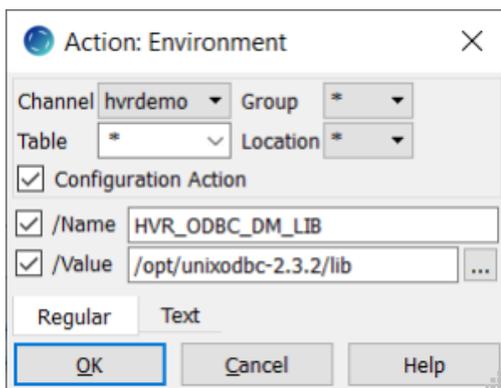
- **/StagingDirectoryDb** is a local directory on the HANA machine which is configured for importing data by HANA.

4. Since it is not possible to enable 'supplemental logging' on HANA, the real key values are not available to HVR during capture. Therefore, the following actions must be defined prior to adding tables to the channel to instruct HVR to capture Row ID values and use them instead of the regular keys during replication. On the source location group (**SRCGRP**), define action **ColumnProperties** with parameters **/Name** and **/CaptureFromRowId**. On both source (**SRCGRP**) and target (**TGTGRP**) location groups, define action **ColumnProperties** with parameters **/Name** and **/SurrogateKey**.
  - a. Right-click source location group **SRCGRP** **New Action ColumnProperties**.
  - b. Select parameter **/Name**. This parameter defines the name for the extra column in the source location. Enter **hvr\_rowid**.
  - c. Select parameter **CaptureFromRowId**. It instructs HVR to capture values from HANA tables' row-id column.

- d. Right-click channel **hvrdemo** **New Action ColumnProperties**.
- e. Select parameter **/Name**. This parameter defines the name for the extra column in the target location. Enter **hvr\_rowid**.
- f. Select parameter **/SurrogateKey**. It instructs HVR to use column **hvr\_rowid** instead of the regular key during replication.



5. For the source (**SRCGRP**) and target (**TGTGRP**)location groups, define action **Environment** to use the Unix ODBC Driver Manager Library for connecting to HANA:
  - a. Right-click channel **hvrdemo** **New Action Environment**.
  - b. Select parameter **/Name** and specify the name of the environment variable **HVR\_ODBC\_DM\_LIB**.
  - c. Select parameter **/Value**.This parameter defines a directory where the driver is installed. Select the directory using the browse button.

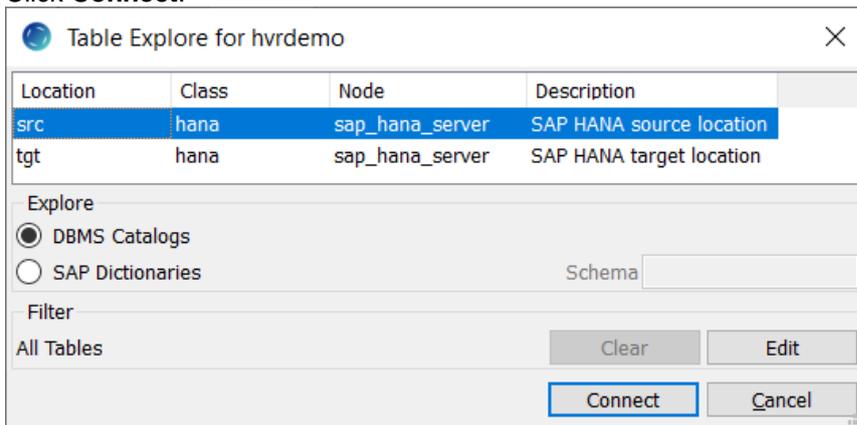


## Select Table(s)

This section describes how to select the tables (**product** and **orders**) from the source location for replication. The **Table Explore** dialog allows you to select schema(s) and/or table(s) for replication.

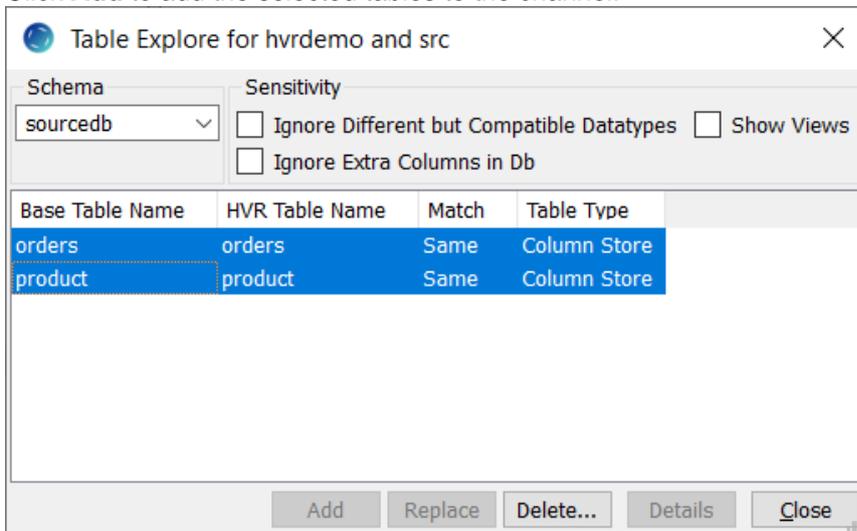
1. Right-click **Tables** **Table Explore**.
2. Select source location (**src**) from the list.

3. Click **Connect**.



4. Select tables from the **Table Explore** dialog. Use the **Shift** or **Ctrl** key to select multiple tables or **Ctrl I+A** to select all tables.

5. Click **Add** to add the selected tables to the channel.



6. Click **OK** in **HVR Table Name** dialog.

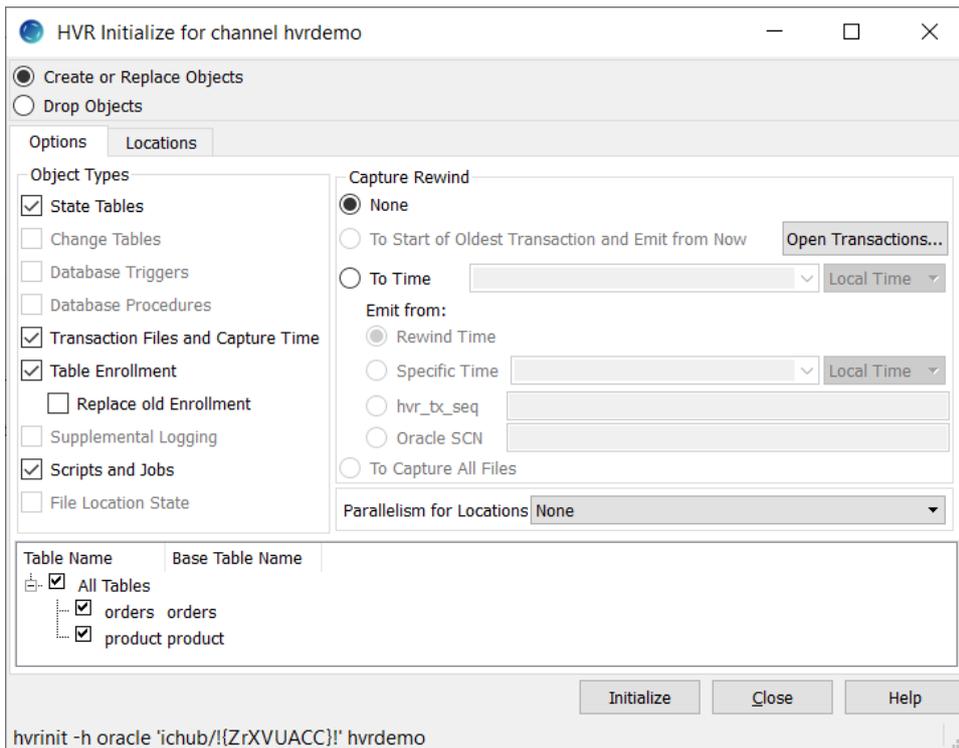
7. Click **Close** in **Table Explore** dialog.

## Initialize

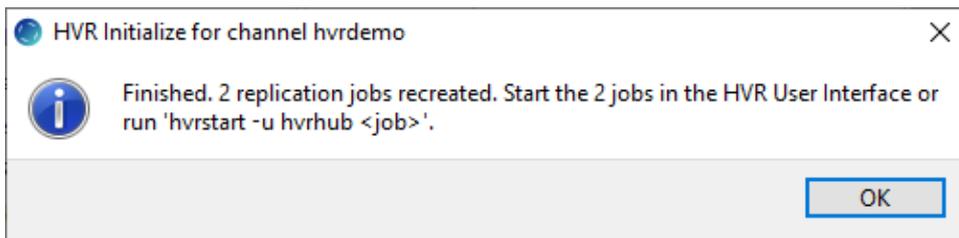
This section describes how to initialize the replication. **HVR Initialize** first checks the channel and creates the replication jobs in the **HVR Scheduler**.

1. Right-click channel **hvrdemo** **HVR Initialize**.
2. Select **Create or Replace Objects** in the **HVR Initialize** dialog.

3. Click **Initialize**.



4. Click **OK**.



5. Click **Close** in the **HVR Initialize** dialog.

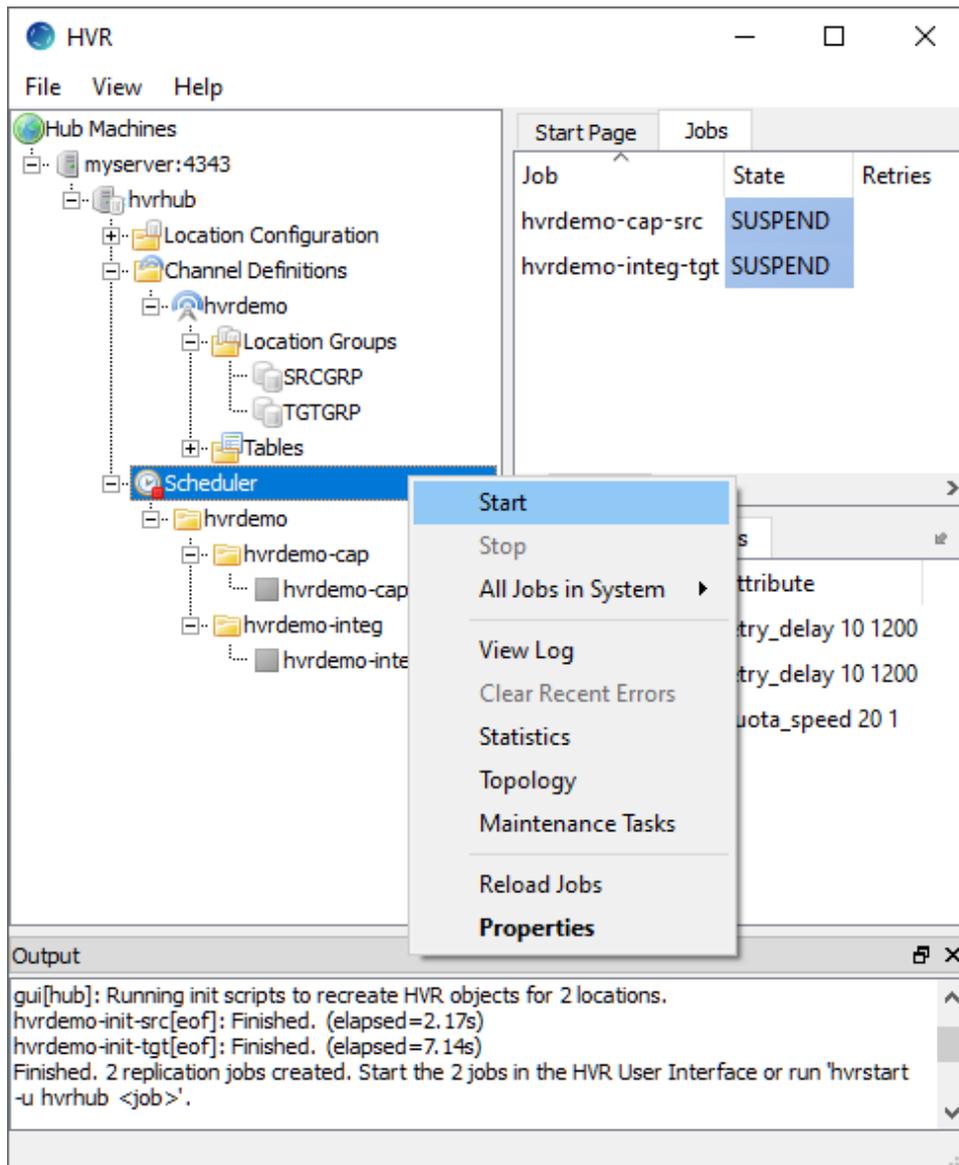
Click the **Scheduler** node in the navigation tree pane to view the capture and integrate jobs in the **Jobs** pane.

For more information about initiating replication in HVR, see section [Replication Overview](#).

## Start Scheduler

This section describes how to start the **HVR Scheduler**. The **HVR Scheduler** is a process which runs jobs defined in the catalog table **HVR\_JOB**. This catalog table can be found in the hub database. On Unix or Linux, the **HVR Scheduler** runs as a daemon. On Windows, the **HVR Scheduler** runs as a system service.

1. In the navigation tree pane, right-click **Scheduler Start**.



2. On Windows, the following steps are required to create the **HVR Scheduler** system service.

- a. Click **Create...** in the prompt asking to create the service **hvr\_scheduler\_hvrhub**.

[SC-Hvr-QSG-ADLSG2\\_SchedulerWinService](#)

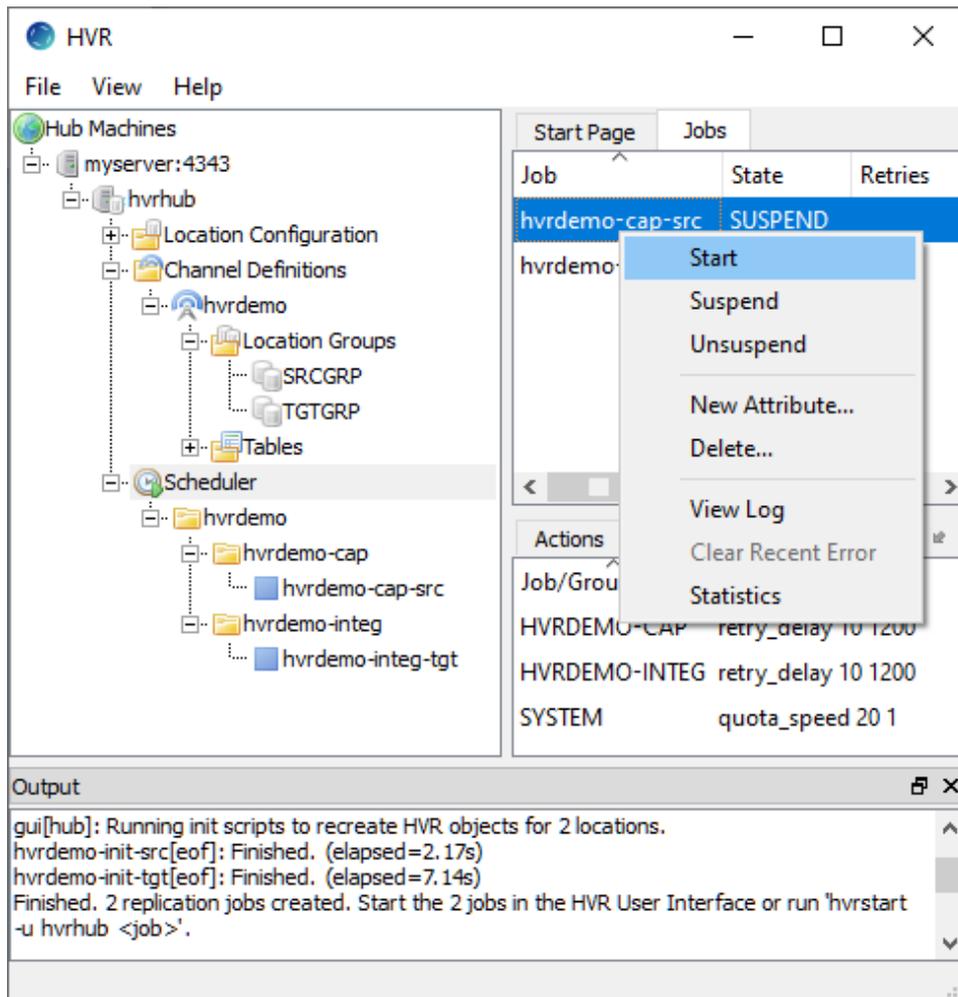
- b. In the **Create Windows Service** dialog, select **Local System Account ('SYSTEM')** and click **Create**.

## Start Capture Job

This section describes how to start the job for capturing changes from the source location (**src**). By starting the **Capture** job in **HVR Scheduler**, HVR begins capturing all changes since the moment **HVR Initialize** was executed. This 'capture begin moment' can be modified using option **Capture Rewind** available in the **Advanced Options** tab of the **HVR Initialize** dialog.

1. In the navigation tree pane, click **Scheduler**.

2. Start the capture job. In the **Jobs** pane, right-click capture job **hvrdemo-cap-src** **Start**.



3. Click **Yes** in the **Start** dialog.

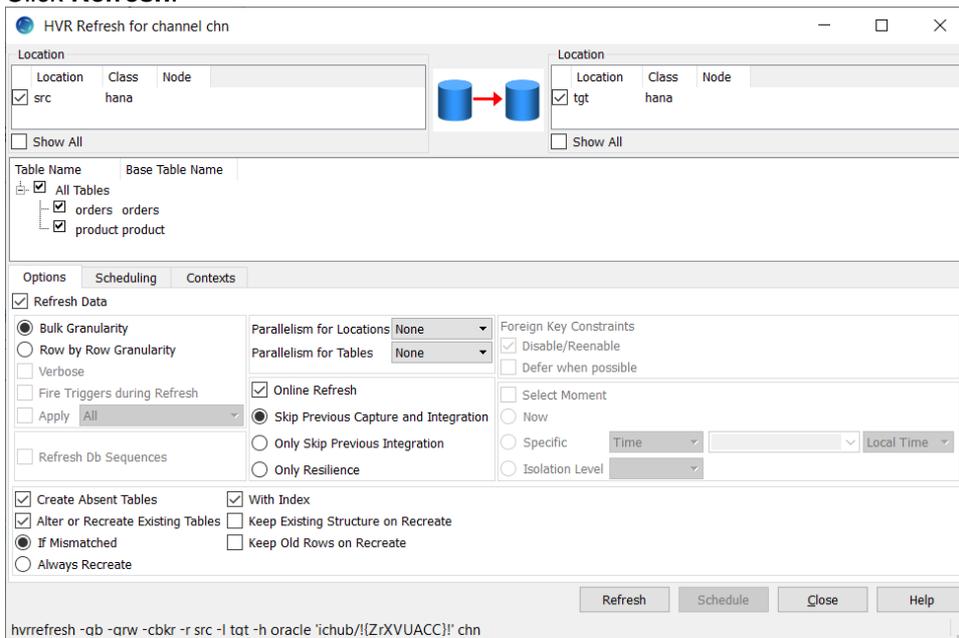
On starting the capture job (**hvrdemo-cap-src**) successfully, the status of the job changes from **SUSPEND** to **RUNNING**.

## Refresh

This section describes how to perform initial load into the target database. **HVR Refresh** copies all existing data from the source location (**src**) to the target location (**tgt**) and optionally creates new tables and keys in the target location.

1. In the navigation tree pane, right-click channel **hvrdemo** **HVR Refresh**.
2. Select the table(s) that needs to be copied from the source location to the target location.
3. Select **Create Absent Tables** in **HVR Refresh** dialog.

#### 4. Click Refresh.



5. Click **Yes** to begin **HVR Refresh**. When the refresh is completed, the **Refresh Result** dialog displays the total number of rows replicated from the selected tables.

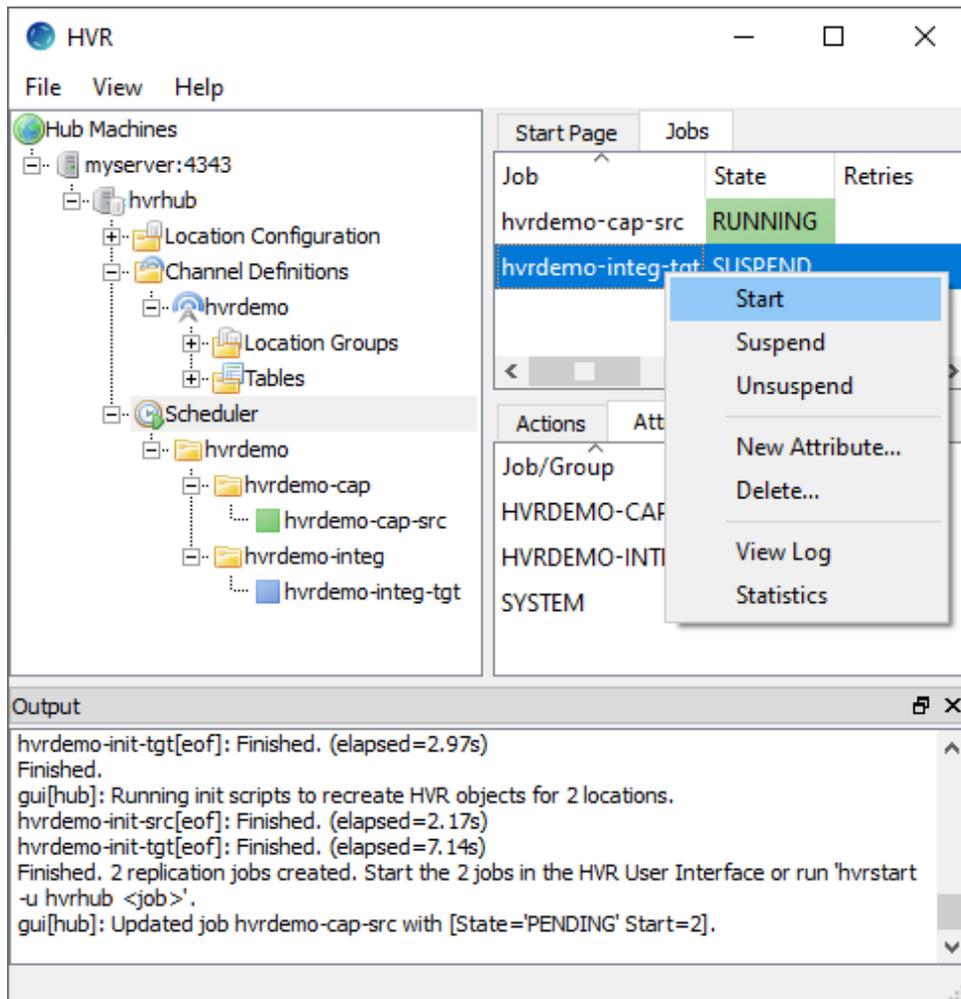
6. Click **Close** in the **Refresh Result** dialog and then in the **HVR Refresh** dialog.

## Start Integrate Job

This section describes how to start the job to integrate changes into the target location (**tgt**).

1. In the navigation tree pane, click **Scheduler**.

2. Start the integrate job. In the **Jobs** pane, right-click integrate job **hvrdemo-integ-tgt** **Start**.



3. Click **Yes** in the **Start** dialog.

On starting the integrate job (**hvr\_demo-integ-tgt**) successfully, the status of the job changes from **SUSPEND** to **RUNNING**.

## Verify Replication

This section describes two methods for verifying the HVR's replication activity.

- [Viewing Log File](#)
- [Using HVR Compare](#)

### Viewing Log File

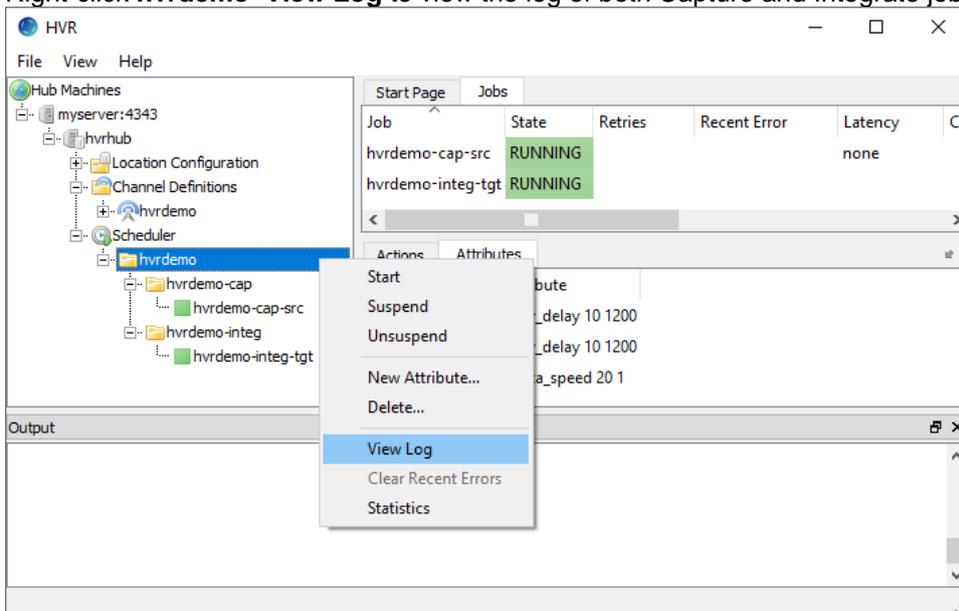
HVR creates separate log files for the hub, channel (**hvrdemo**), and for each replication jobs (**hvrdemo-cap-src** and **hvrdemo-integ-tgt**). These log files contain the details of the changes captured and integrated.

Replication can be verified by inspecting the channel log file.

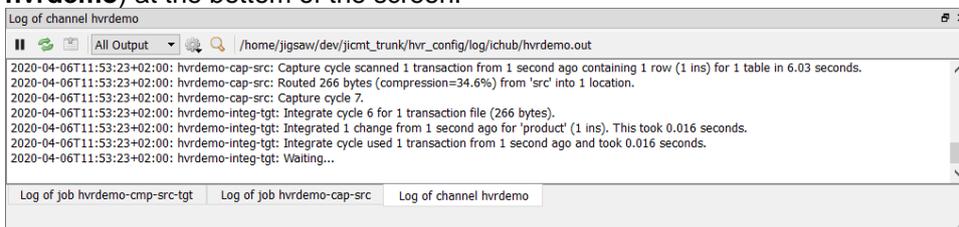
To view the replication activity log:

1. In the navigation tree pane, click **+** next to the **Scheduler**.

- Right-click **hvrdemo** **View Log** to view the log of both Capture and Integrate jobs.



- The logs for both Capture and Integrate jobs are displayed in the logs pane (**Log of channel hvrdemo**) at the bottom of the screen.



The directory path for the HVR log files is displayed in the log tab.

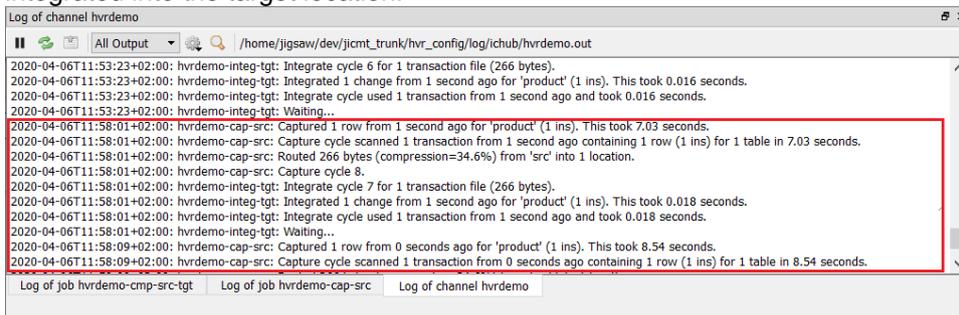


Right-clicking a particular job and selecting **View Log** displays logs related to that job alone.

- Insert, update, or delete value(s) in the source location database. For example:

```
insert into dm01_product
values (101, 91, 'Pencil')
```

- The output log is updated and indicates that the change is captured from the source location and integrated into the target location:



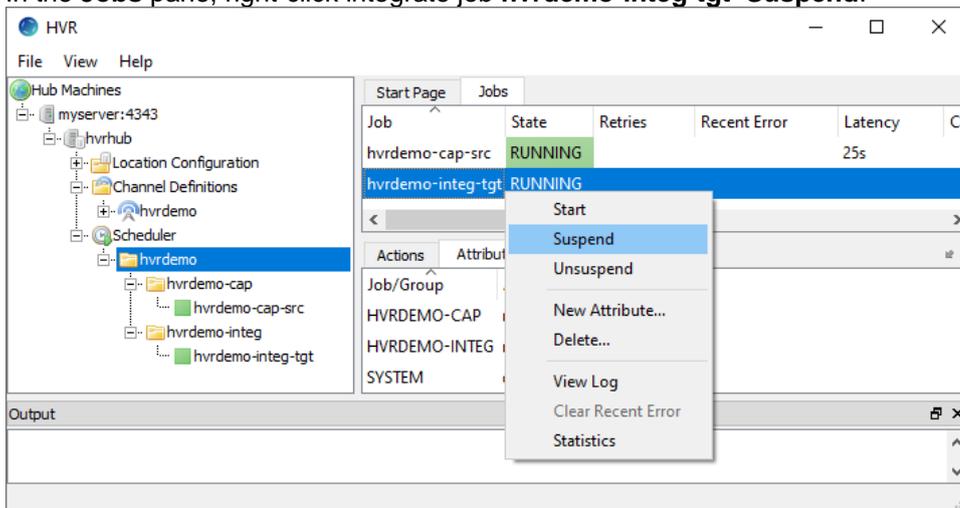
## Using HVR Compare

**HVR Compare** allows you to verify the replication activity by comparing the data in source and target locations.

To compare the source and target locations:

- Suspend the integrate job (**hvrdemo-integ-tgt**):
  - In the navigation tree pane, click **Scheduler**.

- b. In the **Jobs** pane, right-click integrate job **hvrdemo-integ-tgt Suspend**.



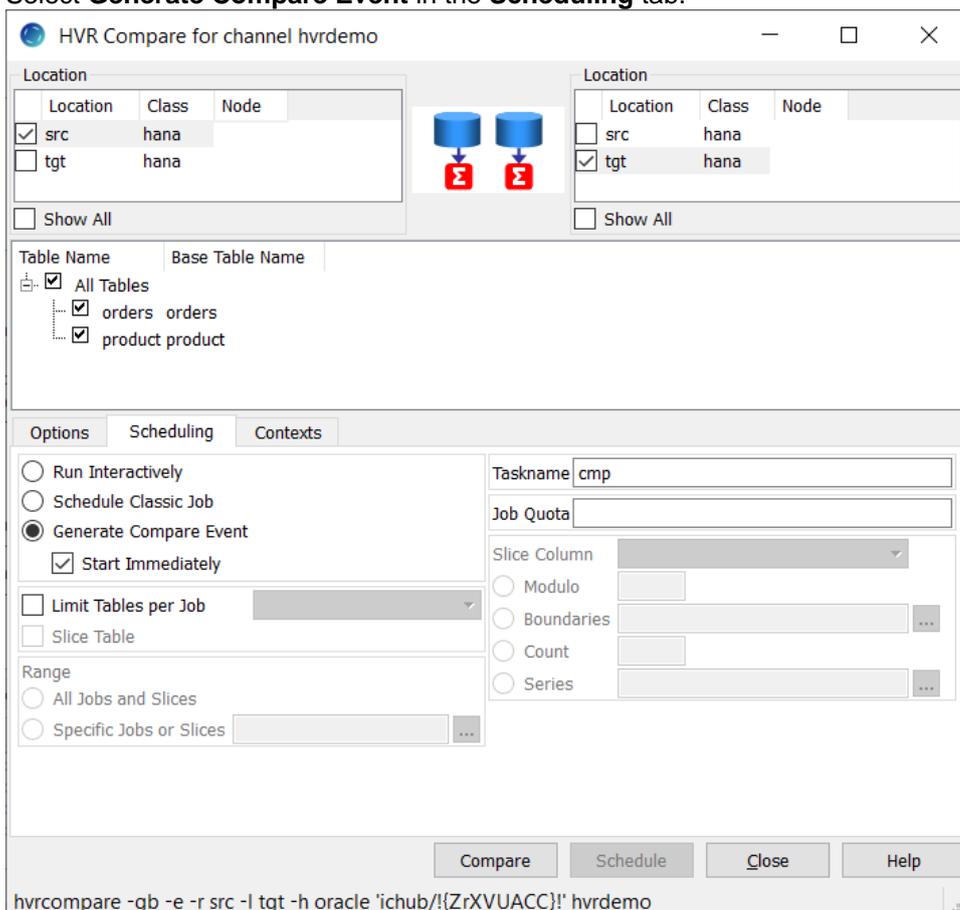
- c. Click **Yes** in the **Suspend** dialog.

2. Insert, update, or delete value(s) in the source location database. For example:

```
insert into sourcedb.product
values (101, 91, 'Pencil');
```

3. Execute **HVR Compare**:

- a. In the navigation tree pane, right-click channel **hvrdemo HVR Compare**.  
 b. Select the source location (**src**) on the left side and the target location (**tgt**) on the right side.  
 c. Select **Generate Compare Event** in the **Scheduling** tab.



- d. Click **Compare**.

- e. On completion, the compare result is displayed in the web browser. If the **State** column displays **DONE/DIFFERENT**, it indicates the data in the source and target locations are not identical.

channel: hvrdemo    granularity: bulk  
source\_loc: src    > tables  
target\_loc: tgt

**Results**

EVENT DURATION	EVENT SPEED	NUMBER OF DIFFERENT TABLES	NUMBER OF TABLES	ROWS PROCESSED DURING EVENT	SUBTASKS DONE(BUSY)/TOTAL	
0.6s	9 rows/s	1	2	6	8(+0)/8	

TABLE	STATE	SOURCE ROWS SELECTED	TARGET ROWS SELECTED	DURATION	SPEED
orders	DONE/IDENTICAL	2	2	0s	34 rows/s
product	DONE/DIFFERENT	4	3	0.1s	26 rows/s

- f. **Start Integrate Job (hvrdemo-integ-tgt)**, the changes made in source location (in step 2) will be integrated to the target location now.
- g. Execute **HVR Compare** again (step 3). In the compare result screen, if the **State** column displays **DONE/IDENTICAL**, it indicates the changes are replicated successfully.

channel: hvrdemo    granularity: bulk  
source\_loc: src    > tables  
target\_loc: tgt

**Results**

EVENT DURATION	EVENT SPEED	NUMBER OF DIFFERENT TABLES	NUMBER OF TABLES	ROWS PROCESSED DURING EVENT	SUBTASKS DONE(BUSY)/TOTAL	
0.5s	11 rows/s	0	2	6	8(+0)/8	

TABLE	STATE	SOURCE ROWS SELECTED	TARGET ROWS SELECTED	DURATION	SPEED
orders	DONE/IDENTICAL	2	2	0s	34 rows/s
product	DONE/IDENTICAL	4	4	0s	68 rows/s