

# Quick Start for HVR - Salesforce

Contents
<ul style="list-style-type: none"><li>• <a href="#">Create Demo Schemas, Tables and Objects</a><ul style="list-style-type: none"><li>• <a href="#">Create Schema</a></li><li>• <a href="#">Create Tables</a></li></ul></li><li>• <a href="#">Create Hub Database</a></li><li>• <a href="#">Grants/Access Privileges</a></li><li>• <a href="#">Download and Install Salesforce Data Loader</a></li><li>• <a href="#">Download and Install 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="#">Replicating from Oracle to Salesforce</a><ul style="list-style-type: none"><li>• <a href="#">Create Channel - Oracle to Salesforce</a></li><li>• <a href="#">Create Location Groups</a></li><li>• <a href="#">Select Table(s)</a></li><li>• <a href="#">Define Actions</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="#">Start Integrate Job</a></li><li>• <a href="#">Verify Replication</a></li></ul></li><li>• <a href="#">Replicating from Salesforce to Oracle</a><ul style="list-style-type: none"><li>• <a href="#">Create Channel - Salesforce to Oracle</a></li><li>• <a href="#">Add Table(s)</a></li><li>• <a href="#">Define Actions</a></li><li>• <a href="#">Initialize</a></li><li>• <a href="#">Start Capture Job</a></li><li>• <a href="#">Start Integrate Job</a></li><li>• <a href="#">Verify Replication</a></li></ul></li></ul>

This quick start guide helps you to get started with HVR for replicating data into Salesforce.

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

The example here demonstrates how to replicate tables from one Oracle schema (source location) to a Salesforce location (target location) and also replicate changes from Salesforce location to Oracle schema. For this replication two HVR channels are required, one for replicating changes from Oracle to Salesforce (**ora2sf**) and the other one for replicating changes from Salesforce to Oracle (**sf2ora**). For simplicity, the Oracle schema will be located on the same machine as the hub database.

Before proceeding with this example ensure that the requirements for using HVR with Oracle and Salesforce are met.

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

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

## Create Demo Schemas, Tables and Objects

This section describes the procedure for creating Oracle schemas, tables, and Salesforce objects.

1. In the Oracle database, create two schemas (**testdb1** and **testdb2**), one will be the source schema in the Oracle to Salesforce channel (ora2sf) and the other will be the target schema in the Salesforce to Oracle channel (**sf2ora**). Both of these schemas will contain one empty table (**dm51\_product**). Skip this step if you already have schemas with tables that you plan to use for this replication.

## Create Schema

```
create user testdb1 identified by hvr
default tablespace users
temporary tablespace temp
quota unlimited on users;
```

```
create user testdb2 identified by hvr
default tablespace users
temporary tablespace temp
quota unlimited on users;
```

## Create Tables

```
create table testdb1.dm51_product
(
  prod_id number(10) not null,
  prod_price number(10,2) not null,
  prod_descrip varchar2(100) not null
);
```

```
create table testdb2.dm51_product
(
  prod_id number(10) not null,
  prod_price number(10,2) not null,
  prod_descrip varchar2(100) not null
);
```

2. Create Salesforce object HVR Demo Product that corresponds to the Oracle's dm51\_product table. In Salesforce, database tables are called objects, columns are called fields, and rows are called records. Once the object is created, Salesforce assigns case-sensitive 'API names' for the object and its fields that must be matched to the corresponding 'base names' in the HVR channel (see section [Define Actions](#) below).
  - a. Log in to Salesforce and switch to Salesforce Classic.
  - b. In the top right menu bar, click **Setup**.
  - c. On the left side bar, navigate to **Build** and click **Create Objects New Custom Object**.
  - d. Enter the details for object **HVR Demo Product** as shown on the screenshot below.

For convenience, use the Oracle table name (**dm51\_product**) for the **Object Name**.



The **Record Name** is the standard field that will appear under the the **Standard Fields** section once you create the object. The **Record Name** field is always called "Name" when referenced via the API. In HVR, it will be mapped to column **prod\_id** in the Oracle table.

Ensure that the **Allow Bulk API Access** checkbox is selected.

salesforce 20 Salesforce is rolling out security enhancements that can impact your org. Re

Search...   Switch to Lightning Experience

Home Contacts Accounts Leads Opportunities Reports Dashboards Chatter Files Products Forecasts Quotes +

Quick Find / Search.   Help for this Page ?

Expand All | Collapse All

**Salesforce Mobile Quick Start**

**Home**

**Administrator**

- Manage Users
- Manage Apps
- Company Profile
- Data Classification
- Security Controls
- Domain Management
- Communication Templates
- Translation Workbench
- Data Management
- Mobile Administration
- Desktop Administration
- Outlook Integration and Sync
- Gmail Integration and Sync
- Email Administration
- Google Apps
- Analytics

**Build**

- Customize
- Create
  - Apps
  - Custom Labels
  - Objects
    - Big Objects
    - Picklist Value Sets
    - Packages
    - Report Types
    - Tabs
    - Service Cloud Launch Pad
    - Action Link Templates
    - Global Actions
    - Workflow & Approvals
  - Develop

**New Custom Object**

**Custom Object Definition Edit**

**Custom Object Information** ! = Required Information

The singular and plural labels are used in tabs, page layouts, and reports.

Label  Example: Account

Plural Label  Example: Accounts

Starts with vowel sound

The Object Name is used when referencing the object via the API.

Object Name  Example: Account

Description

Context-Sensitive Help Setting  Open the standard Salesforce.com Help & Training window  Open a window using a Visualforce page

Content Name

**Enter Record Name Label and Format**

The Record Name appears in page layouts, key lists, related lists, lookups, and search results. For example, the Record Name for Account is "Account Name" and for Case it is "Case Number". Note that the Record Name field is always called "Name" when referenced via the API.

Record Name  Example: Account Name

Data Type

**Optional Features**

- Allow Reports
- Allow Activities
- Track Field History
- Allow in Chatter Groups

**Object Classification**

When these settings are enabled, this object is classified as an Enterprise Application object. When these settings are disabled, this object is classified as a Light Application object. [Learn more.](#)

- Allow Sharing
- Allow Bulk API Access
- Allow Streaming API Access

e. Click **Save**.

f. The **HVR Demo Product** details page will open. Note that the **Standard Fields** section will be auto-populated once you create the table object. Now, you need to create custom fields **Price** and **Description** that correspond to the relevant columns in the Oracle's source tables.

g. In the **Custom Fields & Relationships** section, click **New**.

i. In step 1, choose the field type. In this case, select **Number**. Click **Next**.

ii. In step 2, enter details for the **prod\_price** field as shown on the image below. Click **Next**.

1. Enter 'Price' in **Field Label**. This is a label used on displays, page layout, reports, and list views.

2. Enter '10' in **Length** and '2' in **Decimal Places**. These are the length of the **Number** and the number of decimal places.

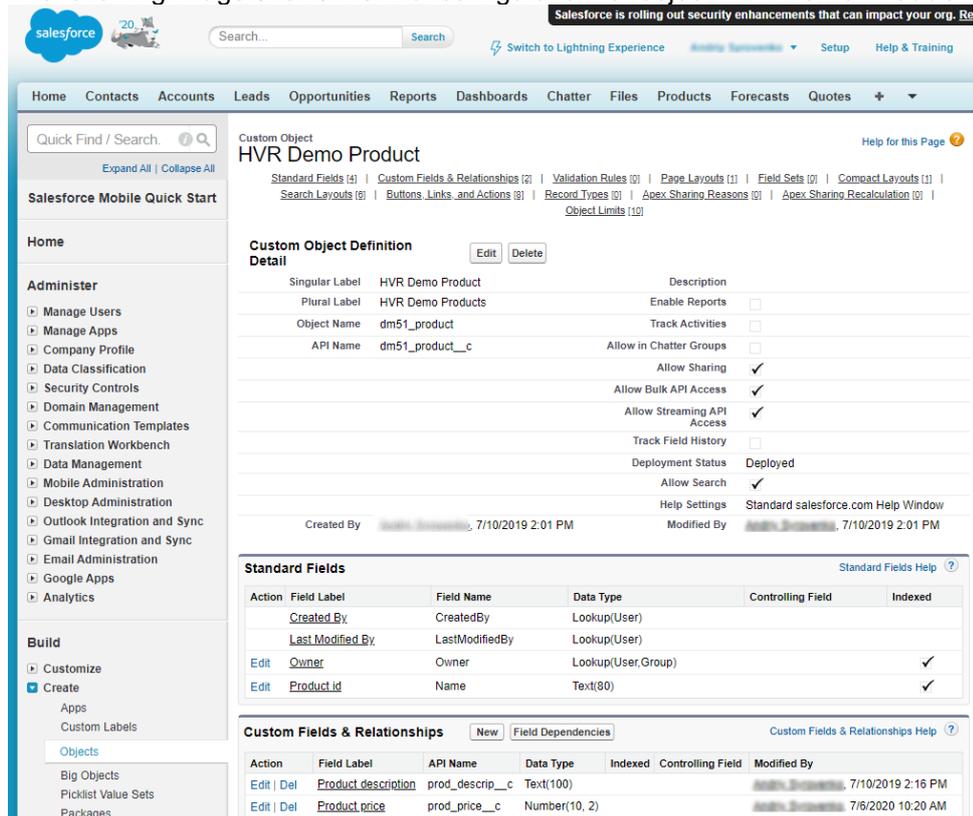
3. Enter 'prod\_id' in **Field Name**. This is an internal reference and is used for integration purposes, such as API.

The screenshot shows the 'New Custom Field' configuration page for a field named 'Price'. The page is titled 'Step 2. Enter the details' and is part of a 4-step process. The field label is 'Price'. The length is set to 10, and the decimal places are set to 2. The field name is 'prod\_price'. The description and help text fields are empty. The 'Required' checkbox is unchecked. The 'Unique' checkbox is unchecked. The 'External ID' checkbox is unchecked. The 'AI Prediction' checkbox is unchecked. The default value is set to 'Show Formula Editor'. The left sidebar shows the 'Administer' and 'Build' sections.

- iii. In step 3, establish field-level security as needed.
- iv. In step 4, select the page layouts that should include this field and click **Save & New**.
- v. Repeat the above steps to create new custom field **Description**.

The screenshot shows the 'New Custom Field' configuration page for a field named 'Description'. The page is titled 'Step 2. Enter the details' and is part of a 4-step process. The field label is 'Description'. The length is set to 100. The field name is 'prod\_descrip'. The description and help text fields are empty. The 'Required' checkbox is unchecked. The 'Unique' checkbox is checked, with the radio button selected for 'Treat "ABC" and "abc" as duplicate values (case insensitive)'. The 'External ID' checkbox is unchecked. The left sidebar shows the 'Administer' and 'Build' sections.

The following image shows the final configuration for object **HVR Demo Product**.



## 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. **myhub**) is created in Oracle.

Create the hub database (**myhub**) with password (**hvr**).

```
create user myhub
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 source schemas and hub database.

1. Configure the privileges for source schemas (**testdb1** and **testdb2**). For more information, see section [Grants for Log-Based Capture](#) in [Requirements for Oracle](#).

```
grant create session to testdb1;
grant create table to testdb1;
```

```
grant alter any table to testdb1;
grant select any dictionary to testdb1;
grant select any transaction to testdb1;

grant create session to testdb2;
grant create table to testdb2;
```

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

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

## Download and Install Salesforce Data Loader

1. Log into [www.salesforce.com](http://www.salesforce.com).
2. Click **Setup Administrative Setup Data Management Data Loader**.
3. Download and install the **ApexDataLoader.exe**.

## Download and Install HVR

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.

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. For more information about connecting to remote HVR installation, see [Configuring Remote Installation of HVR on Unix or Linux](#) and [Configuring Remote Installation of HVR on Windows](#).

## 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 **hvr** 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 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. **myhub**).
  - d. Enter the password for the hub database in **Password** (e.g. **hvr**).
7. Click **Connect**.

The screenshot shows the 'Register Hub' dialog box with the following configuration:

- Connect to HVR on remote machine
- Node: myserver
- Port: 4343
- Login: myserveradmin
- Password (masked)
- Prompt for password
- Encryption: [button]
- Class:
  - Oracle
  - Ingres / Vector(H)
  - SQL Server
  - DB2 Linux/Unix/Windows
  - DB2 for i
  - PostgreSQL/Aurora
  - MySQL/MariaDB/Aurora
  - HANA
  - Teradata
- Database Connection:
  - ORACLE\_HOME: d:/oracle/1210 [browse button]
  - ORACLE\_SID: ORA1210
  - TNS: [empty]
  - User: myhub
  - Password: [masked]

Buttons at the bottom: Connect, Cancel, Help.

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 two Oracle locations **ora1** and **ora2** (one for each channel).
  1. In the navigation tree pane, right-click **Location Configuration** **New Location**.
  2. Enter the **Location** name (**ora1**) and optionally the description for the location.
  3. Select **Oracle** in **Class**.
  4. Provide **Database Connection** details. For more information on **Database Connection** fields, see section [Location Connection](#) in [Requirements for Oracle](#).
    - a. Enter the directory path for **ORACLE\_HOME**. You can also click browse to select the directory path.
    - b. Enter **Oracle System ID** in **ORACLE\_SID**, or **TNS** credential, or **RAC** credential.  
  
For **RAC** connectivity, ensure to provide remote machine connection details under the **Connection** tab.
    - c. Enter a username of the source schema in **User**.
    - d. Enter a password for source schema in **Password**.
  5. Click **Test Connection** to verify the connection to location database.
  6. Click **OK**.

Location ora1

Location

Location

Description

Connection Group Membership

Connect to HVR on remote machine

Node  Login

Port  Password

Class

- Oracle
- Ingres / Vector(H)
- SQL Server
- DB2 Linux/Unix/Windows
- DB2 for i
- DB2 for z/OS
- PostgreSQL/Aurora
- MySQL/MariaDB/Aurora
- HANA
- Teradata
- Snowflake
- Greenplum
- Redshift
- Hive ACID
- File / FTP / Sharepoint
- Azure DLS
- Azure DLS Gen2

Database Connection

ORACLE\_HOME

ORACLE\_SID

TNS

RAC

SCAN  1521

Service

User

Password

Above is for Redo access only;  
either for Data Guard (non-active) Standby or Root Container for LogMiner

Test Connection OK Cancel Help

7. Repeat steps 1-6 to create the second Oracle location (**ora2**).

Before creating the Salesforce location, set the **\$JAVA\_HOME** environment variable to the Java installation directory using action **Environment**.

1. In the navigation tree pane, right-click channel **ora2sf**, click **New Action Environment**.
2. In the **Name** field, type 'JAVA\_HOME'.
3. In the **Value** field, click the browse button and select the directory.
4. Ensure to select all channels (\*) so that this action is applied to both channels **ora2sf** and **sf2ora**.

Action: Environment

Channel \* Group \*  
Table \* Location \*

Configuration Action

/Name JAVA\_HOME  
 /Value /Salesforce/zulu11.39.15-ca-fx-jdk11.0.7-win\_x64/zulu11.39.15-ca-fx-jdk11.0.7-win\_x64 ...

Regular Text

OK Cancel Help

- Create Salesforce location **sf**.
  1. In the navigation tree pane, right-click **Location Configuration New Location**.
  2. Enter the **Location** name (**sf**) and optionally the description for the location.
  3. Select **Salesforce** in **Class**.
  4. Provide **Database Connection** details. For more information on **Database Connection** fields , see section [Location Connection](#) in [Requirements for Salesforce](#).
    - a. Enter the URL for **Endpoint**.
    - b. Enter the Salesforce username in **Login**.
    - c. Enter the Salesforce password in **Password**.
    - d. Enter the directory path where the **dataloader.jar** file is located in **Dataloader**. The default installation directory on Windows is **C:\Program Files (x86)\salesforce.com\Data Loader**, and the name of the **.jar** file depends on the version of the Data Loader.
  5. Click **Test Connection** to verify the connectivity.

## 6. Click OK.

The screenshot shows the 'New Location' dialog box with the following details:

- Location:** sf
- Description:** Salesforce location
- Connection:** Group Membership
- Options:**
  - Connect to HVR on remote machine
  - /SslRemoteCertificate
  - /CloudLicense
- Salesforce Location:**
  - Endpoint: https://login.salesforce.com
  - Login: myuser@company
  - Password: [Masked]
  - Dataloader: C:\Program Files\salesforce.com\Data Loader\Dataloader.jar
- Left Pane (Selected):** Salesforce

## Replicating from Oracle to Salesforce

This section demonstrates how to replicate tables from an Oracle database on a source location to a Salesforce database on a target location.

### Create Channel - Oracle to Salesforce

To create a [channel](#) (**ora2sf**) in HVR:

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

The screenshot shows the 'New Channel' dialog box with the following details:

- Channel:** ora2sf
- Description:** Oracle to Salesforce channel
- Buttons:** OK, Cancel

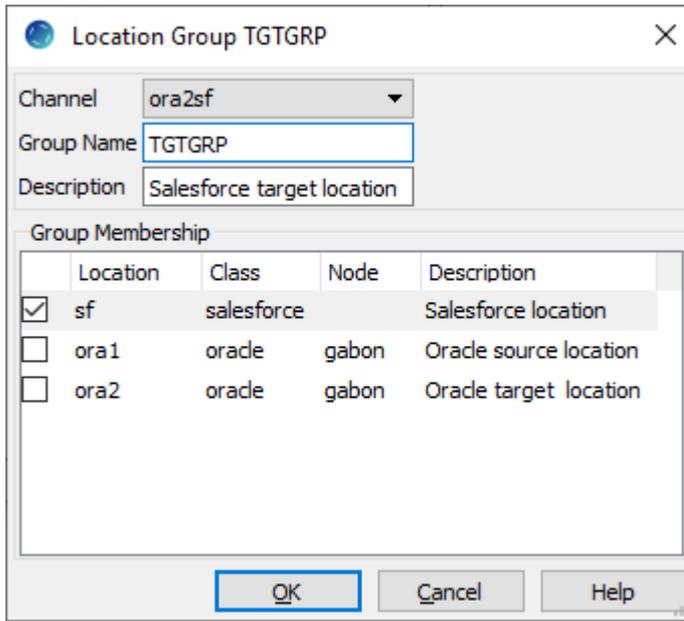
### Create Location Groups

This section describes how to create location groups in the channel. 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.

In this example, 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 (**ora2sf**).

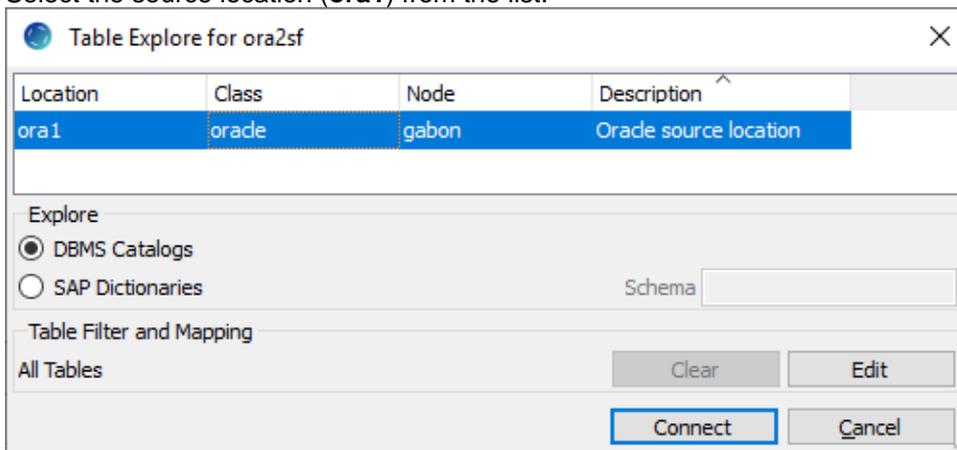
2. Create the source location group (**SRCGRP**):
  - a. Right-click **Location Groups New Group**.
  - b. Enter the group name and optionally the description for the location group.
  - c. Select the source location (**ora1**) in **Group Membership**.
  - d. Click **OK**.
3. Create the target location group (**TGTGRP**):
  - a. Right-click **Location Groups New Group**.
  - b. Enter the group name and optionally the description for the location group.
  - c. Select the target location (**sf**) from **Group Membership**.
  - d. Click **OK**.



## Select Table(s)

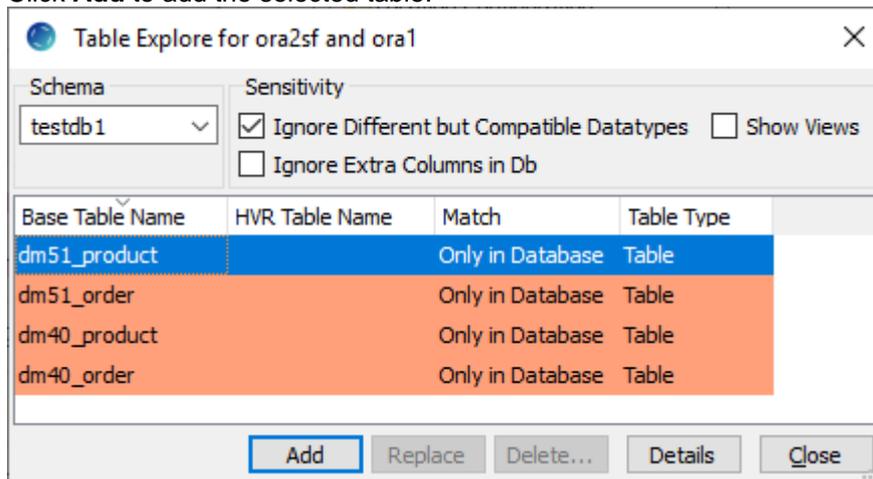
This section describes how to select the table (**dm51\_product**) from the source location for replication. Table Explore allows you to select schema(s) and/or table(s) for replication.

1. Right-click **Tables Table Explore**.
2. Select the source location (**ora1**) from the list.

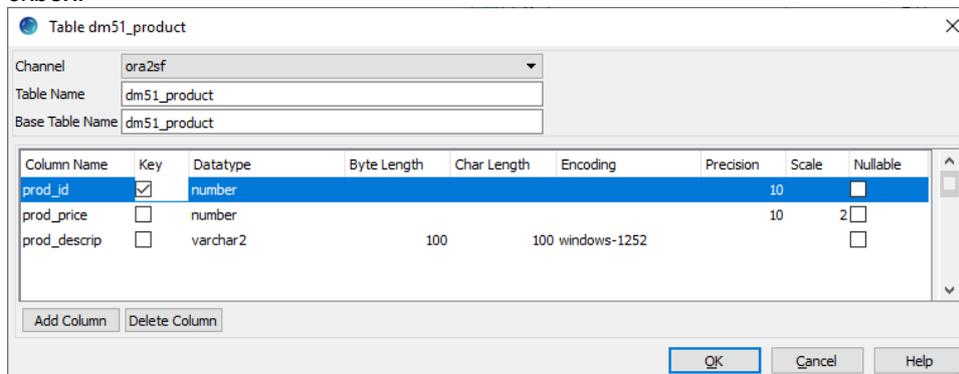


3. Click **Connect**.
4. Select table **dm51\_product** in the **Table Explore** dialog. Press the **Shift** key if you need to select multiple tables or **Ctrl+A** to select all tables.

- Click **Add** to add the selected table.



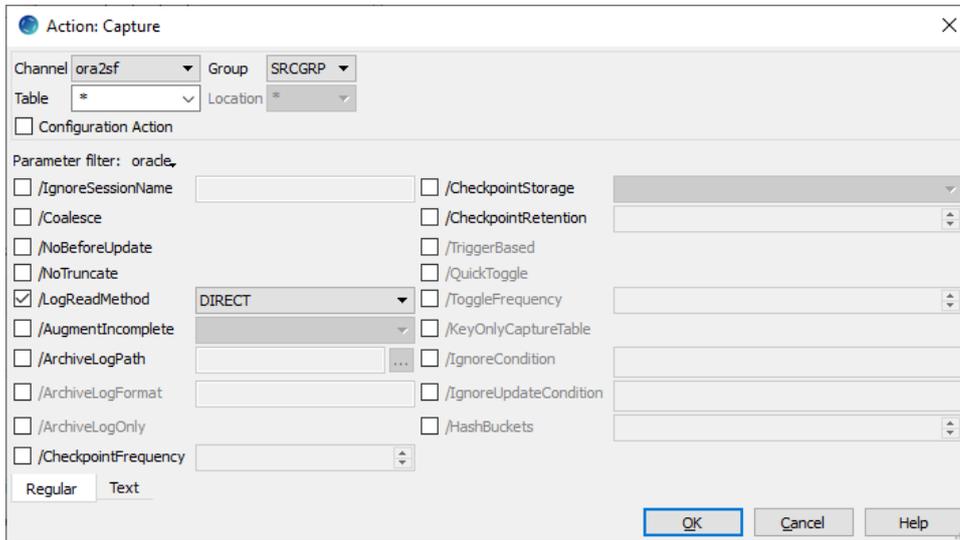
- Click **OK** in the **HVR Table Name** dialog.
- Click **Close** in the **Table Explore** dialog.
- Double-click table **dm51\_product** and make column **prod\_id** a key column by selecting the **Key** checkbox.



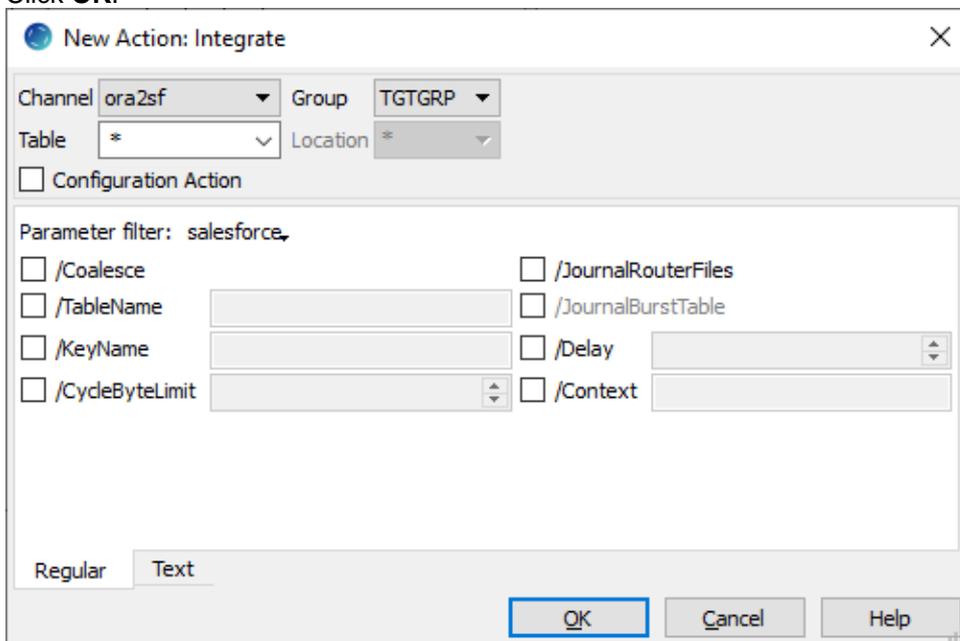
## Define Actions

This section describes how to define **actions** on the location groups (**SRCGRP** and **TGTGRP**). Actions define the behavior of the replication activity. The new channel needs two actions to indicate the direction of replication, and actions to map the database table and column names in Oracle to Salesforce objects and Field API names.

- Define action **Capture** with parameter **/LogReadMethod=DIRECT** to capture changes from the table in the source location group (**SRCGRP**).
  - In the navigation tree pane, right-click source location group **SRCGRP** **New Action Capture**.
  - Click **OK**.



2. Define action **Integrate** to integrate changes to the table object in the target location group (**TGTGRP** ).
  - a. In the navigation tree pane, right-click target location group **TGTGRP** **New Action Integrate**.
  - b. Click **OK**.



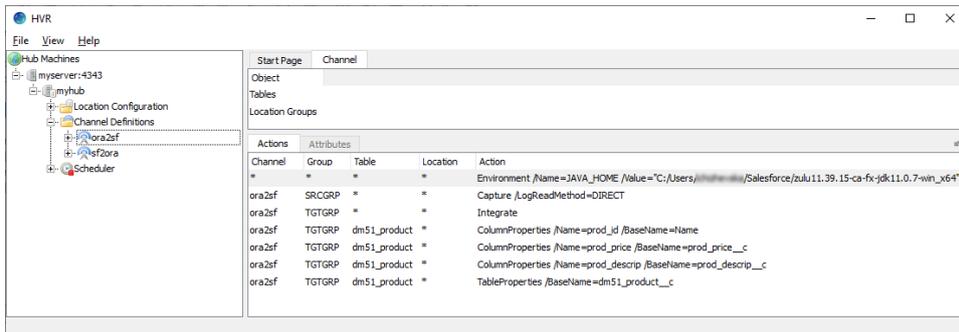
- Action **Integrate** only integrates insert and update statements, not deletes.
3. Define actions **TableProperties** and **ColumnProperties** to map the source table and columns in Oracle to the relevant target table objects and fields in Salesforce.
    - a. In the navigation tree pane, right-click group **TGTGRP** **New Action TableProperties**. In the **TableProperties** dialog, choose table **dm51\_product**, then select **/BaseName** and enter the Salesforce API Name **dm51\_product\_\_c**.
    - b. Right-click group **TGTGRP** **New Action ColumnProperties**. In the **ColumnProperties** dialog, choose table **dm51\_product**, then select **/Name**, click the browse button and select column **prod\_id**. Select **/BaseName** and enter the Salesforce API Name Name.
    - c. Right-click group **TGTGRP** **New Action ColumnProperties**. In the **ColumnProperties** dialog, choose table **dm51\_product**, then select **/Name**, click the browse button and select column **prod\_price**. Select **/BaseName** and enter the Salesforce API Name **prod\_price\_\_c**.
    - d. Right-click group **TGTGRP** **New Action ColumnProperties**. In the **ColumnProperties** dialog, choose table **dm51\_product**, then select **/Name**, click the browse button and select column **prod\_descrip**. Select **/BaseName** and enter the Salesforce API Name **prod\_descrip\_\_c**.

Salesforce API names are case-sensitive.

Channel	Group	Table	Location	Action
---------	-------	-------	----------	--------

ora2sf	TGTGRP	dm51_ product	*	TableProperties /BaseName=dm51_product__c
ora2sf	TGTGRP	dm51_ product	*	ColumnProperties /Name=prod_id /BaseName=Name
ora2sf	TGTGRP	dm51_ product	*	ColumnProperties /Name=prod_price /BaseName=prod_price__c
ora2sf	TGTGRP	dm51_ product	*	ColumnProperties /Name=prod_descrip /BaseName=prod_descrip__c

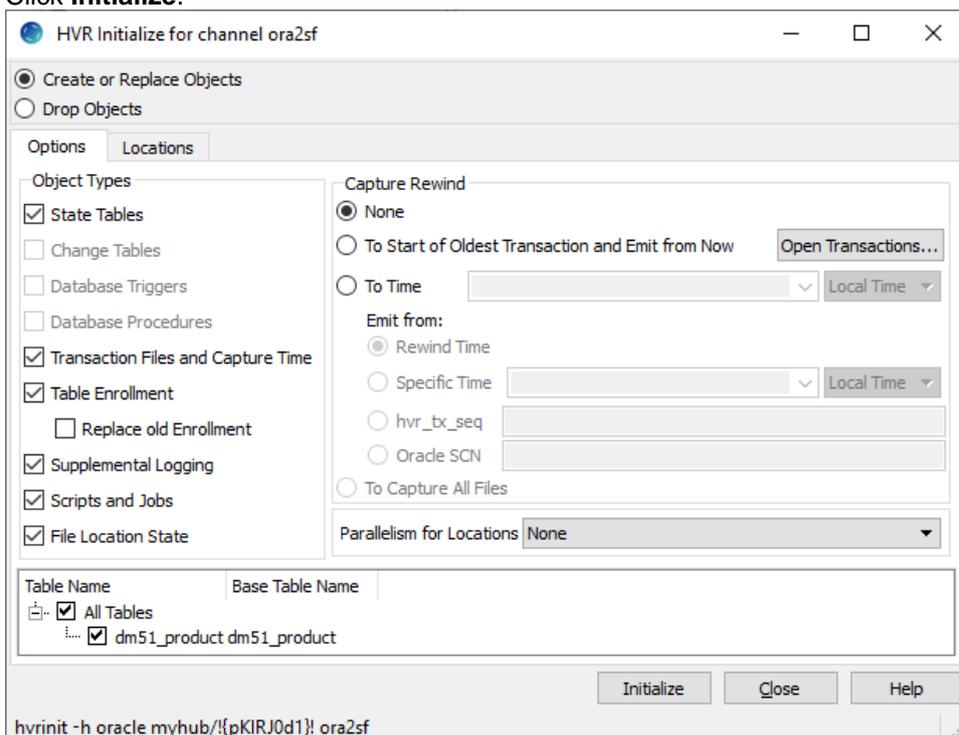
- In the navigation tree pane, click channel **ora2sf** to see all configuration actions defined for the channel.



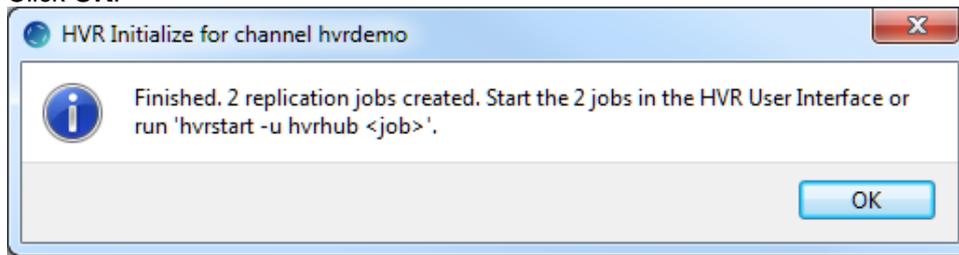
## Initialize

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

- Right-click channel **ora2sf** **HVR Initialize**.
- Select **Create or Replace Objects** in the **HVR Initialize** dialog.
- Click **Initialize**.



4. Click **OK**.

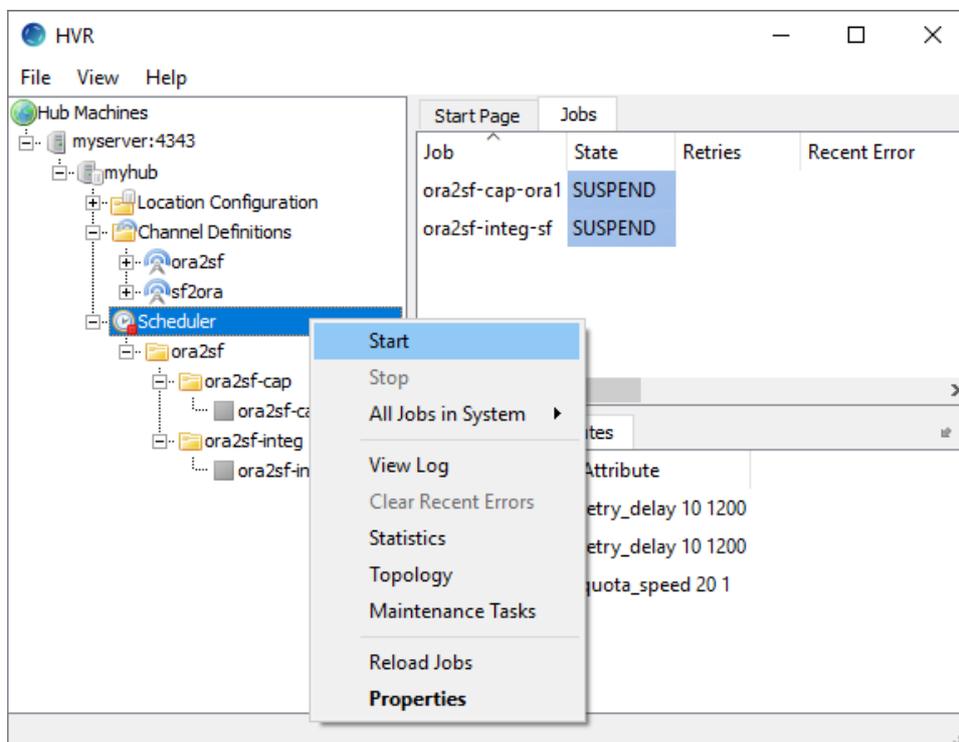


5. Click **Close** in the **HVR Initialize for channel ora2sf** dialog.
6. Expand the **Scheduler** node in the navigation tree pane to view the capture and integrate jobs. For more information about initiating replication in HVR, see section [Replication Overview](#).

## Start Scheduler

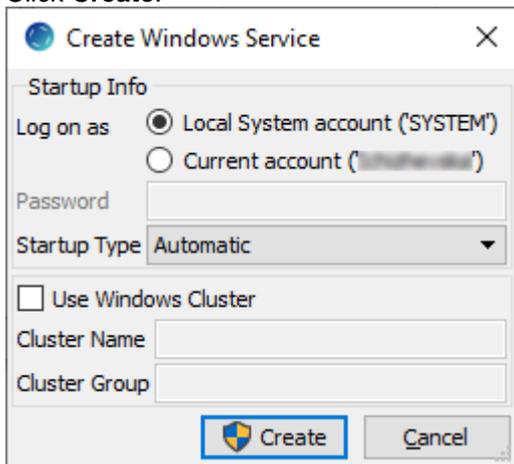
This section describes how to start the **HVR Scheduler**. The **HVR Scheduler** is a process which runs jobs defined in the catalog table **HVR\_JOB**. This catalog table can be found in the hub database. On Unix or Linux, the **HVR Scheduler** runs as a daemon. On Windows, the **HVR Scheduler** runs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

1. Start the Scheduler. 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\_myhub**.
  - b. Select **Local System Account ('SYSTEM')** in the **Create Windows Service** dialog.

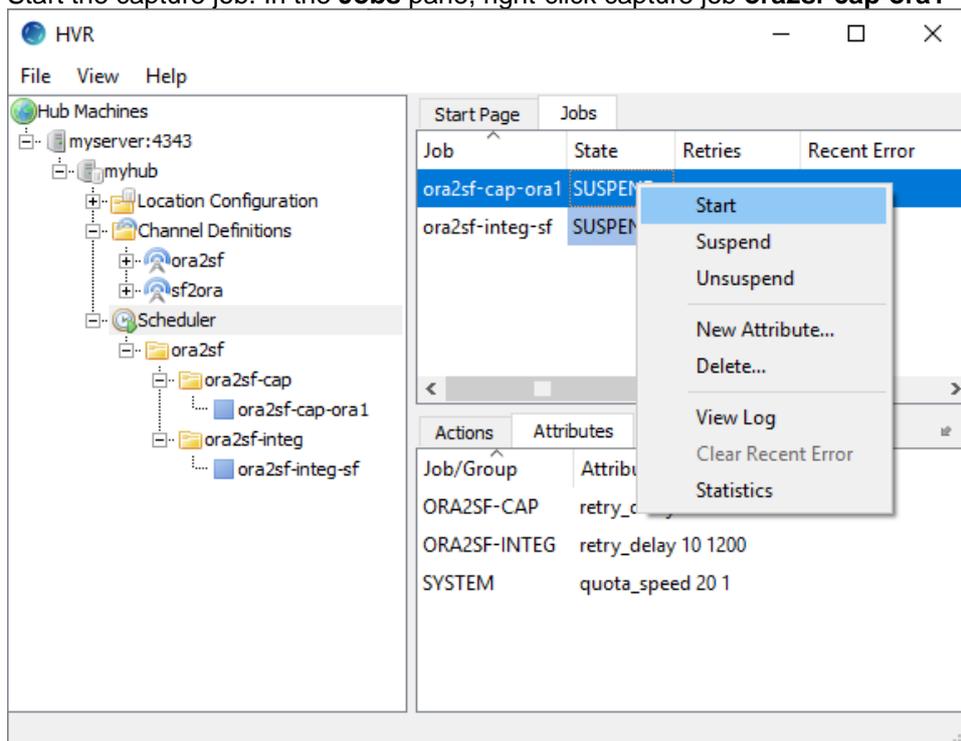
c. Click **Create**.



## Start Capture Job

This section describes how to start the job for capturing changes from the source location (**ora1**). 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 the 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 **ora2sf-cap-ora1** **Start**.



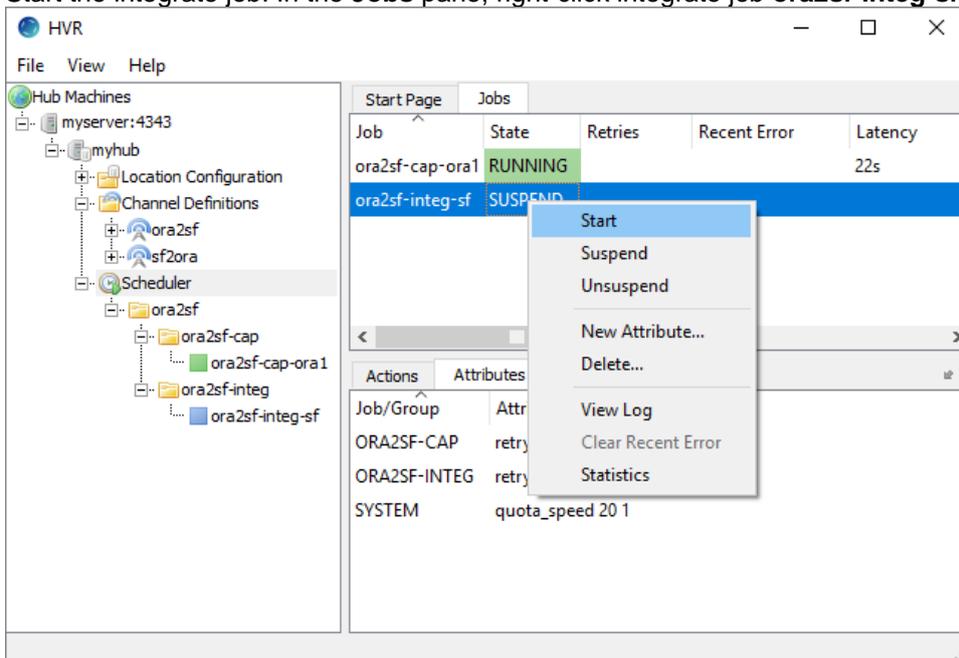
3. Click **Yes** in the **Start** dialog. On starting the capture job (**ora2sf-cap-ora1**) successfully, the status of the job changes from **SUSPEND** to **RUNNING**.

## Start Integrate Job

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

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

2. Start the integrate job. In the **Jobs** pane, right-click integrate job **ora2sf-integ-sf** **Start**.



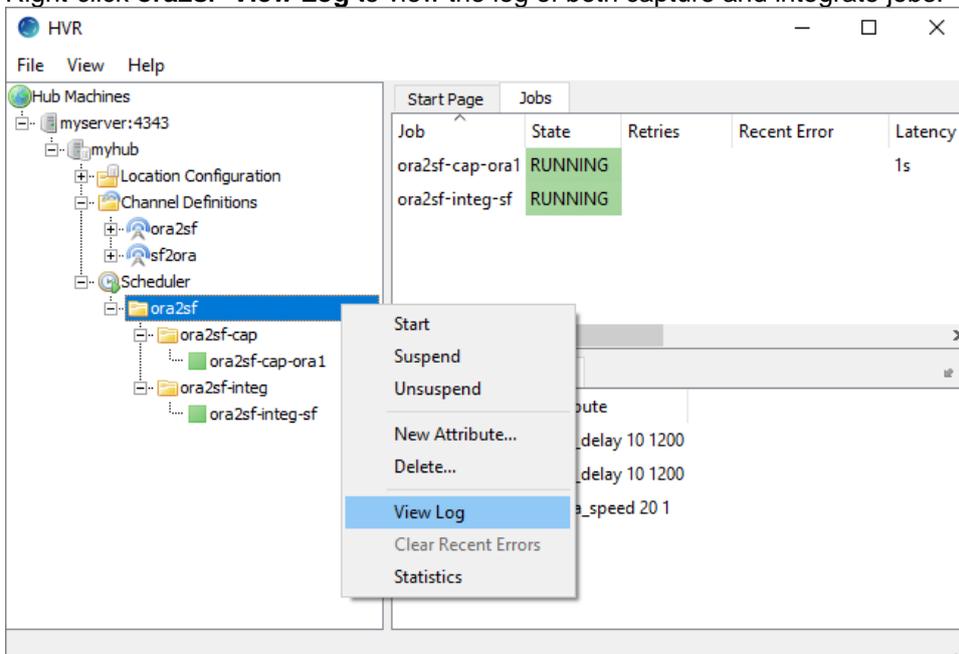
3. Click **Yes** in the **Start** dialog. On starting the integrate job (**ora2sf-integ-sf**) successfully, the status of the job changes from **SUSPEND** to **RUNNING**.

## Verify Replication

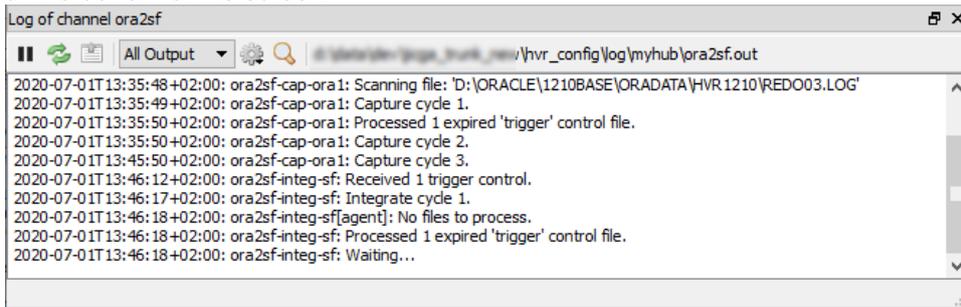
Replication can be verified by inspecting the channel log file. HVR creates separate log files for the hub, channel (**ora2sf**), and for each replication jobs (**ora2sf-cap-ora1** and **ora2sf-integ-sf**). These log files contain the details of the changes captured and integrated.

To view the replication activity log:

1. In the navigation tree pane, click **+** next to the **Scheduler**.
2. Right-click **ora2sf** **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 ora2sf**) at the bottom of the screen.



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

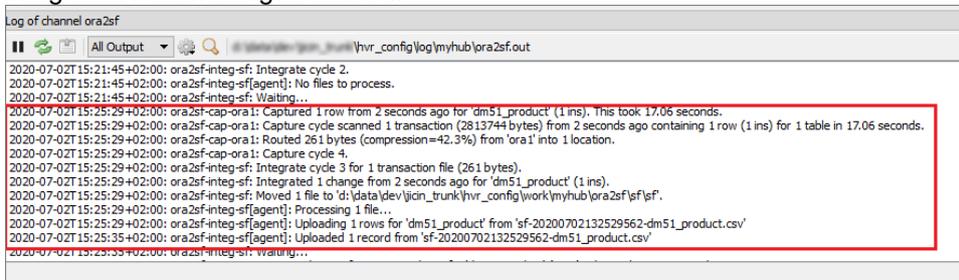


Right-clicking a particular job and selecting **View Log** displays logs related to that job alone.

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

```
insert into testdb1.dm51_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:



This indicates that the jobs replicated the original change to Salesforce. You can also verify the replication by looking at the Object data in Salesforce.

## Replicating from Salesforce to Oracle

This section demonstrates how to replicate tables from a Salesforce database on the source location to an Oracle database on the target location.

### Create Channel - Salesforce to Oracle

To retrieve data from Salesforce, create a second channel **sf2ora**. Follow the steps above to [create the channel](#) and two [location groups](#) **SRCGRP** and **TGTGRP**; source group **SRCGRP** includes location **sf**, and target group **TGTGRP** includes location **ora2**.

### Add Table(s)

For the Salesforce source location, you need to add tables manually.

- In the navigation tree pane, right-click **Tables** and click **New Table**.
- Enter the name of the table in the Oracle location in the **Table Name** field.
- Enter the name of the table in the Salesforce location in the **Base Table Name** field.

- Click **Add Column** to add columns to the table. The columns should correspond to the columns in Salesforce (**Name**, **prod\_price\_\_c**, **prod\_descrip\_\_c**).

Column Name	Key	Datatype	Byte Length	Char Length	Encoding	Precision	Scale	Nullable
Name	<input type="checkbox"/>	number				10		<input type="checkbox"/>
prod_price__c	<input type="checkbox"/>	number				10	2	<input type="checkbox"/>
prod_descrip__c	<input type="checkbox"/>	varchar		20				<input type="checkbox"/>

## Define Actions

This section describes how to define **actions** on the location groups (**SRCGRP** and **TGTGRP**). Actions define the behavior of the replication activity. The new channel needs two actions to indicate the direction of replication, and actions to map database table and column names in Oracle to Salesforce objects and Field API names.

- Define action **Capture** to capture changes from the tables in the source location group (**SRCGRP**).
  - In the navigation tree pane, right-click source location group **SRCGRP** **New Action Capture**.
  - Click **OK**.
- Define action **LocationProperties** with parameter **/BulkAPI** to the source group **SRCGRP** to use Salesforce Bulk API (instead of the SOAP interface).
  - In the navigation tree pane, right-click source location group **SRCGRP** **New Action LocationProperties**.
  - Select **/BulkAPI**.
  - Click **OK**.

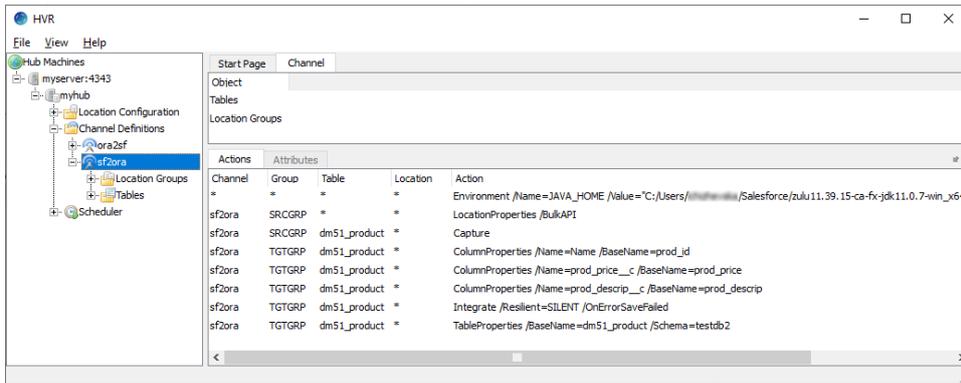
Channel	Group	Table	Location	Action
sf2ora	SRCGRP	dm51_produ ct	*	Capture
sf2ora	SRCGRP	dm51_produ ct	*	LocationProperties /BulkAPI

- Define action **Integrate** to integrate changes to the table objects in the target location group (**TGTGRP**). Select **/OnErrorSaveFailed** and **/Resilient=SILENT**, this affects how replication errors are handled.
  - In the navigation tree pane, right-click target location group **TGTGRP** **New Action Integrate**.
  - Select **/OnErrorSaveFailed**.
  - Select **/Resilient** and pick **SILENT** from the drop-down menu.
  - Click **OK**.
- Define action **TableProperties** and **ColumnProperties** to map the source object and fields in Salesforce to the relevant target table and columns in Oracle.
  - In the navigation tree pane, right-click group **TGTGRP** **New Action TableProperties**. In the **TableProperties** dialog, choose table **dm51\_product**, then select **/Schema** and enter the schema name for the Oracle table.
  - Right-click group **TGTGRP** **New Action ColumnProperties**. In the **ColumnProperties** dialog, choose table **dm51\_product**, then select **/Name**, click the browse button and select column **prod\_id**. Select **/BaseName** and enter the Salesforce API Name **Name**.
  - Right-click group **TGTGRP** **New Action ColumnProperties**. In the **ColumnProperties** dialog, choose table **dm51\_product**, then select **/Name**, click the browse button and select column **prod\_price**. Select **/BaseName** and enter the Salesforce API Name **prod\_price\_\_c**.
  - Right-click group **TGTGRP** **New Action ColumnProperties**. In the **ColumnProperties** dialog, choose table **dm51\_product**, then select **/Name**, click the browse button and select column **prod\_descrip**. Select **/BaseName** and enter the Salesforce API Name **prod\_descrip\_\_c**.

Salesforce API names are case-sensitive.

Channel	Group	Table	Location	Action
sf2ora	TGTGRP	dm51_product	*	Integrate /Resilient=SILENT /OnErrorSaveFailed
sf2ora	TGTGRP	dm51_product	*	TableProperties /BaseName=dm51_product /Schema=testdb2
sf2ora	TGTGRP	dm51_product	*	ColumnProperties /Name=prod_id /BaseName=Name
sf2ora	TGTGRP	dm51_product	*	ColumnProperties /Name=prod_price /BaseName=prod_price__c
sf2ora	TGTGRP	dm51_product	*	ColumnProperties /Name=prod_descrip /BaseName=prod_descrip__c

- In the navigation tree pane, click channel **sf2ora** to see all configuration actions defined for the channel.

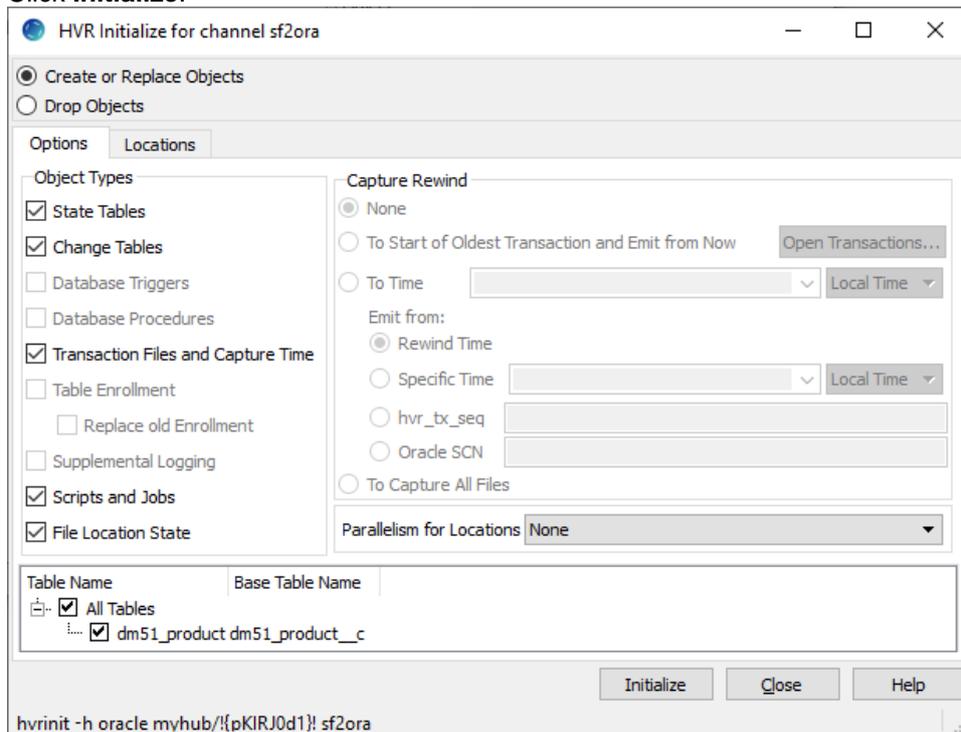


## Initialize

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

- Right-click channel **sf2ora** **HVR Initialize**.
- Select **Create or Replace Objects** in **HVR Initialize** dialog.

### 3. Click **Initialize**.



### 4. Click **OK**.

### 5. Click **Close** in the **HVR Initialize for channel sf2ora** dialog.

### 6. Expand the **Scheduler** node in the navigation tree pane to view the capture and integrate jobs. For more information about initiating replication in HVR, see section [Replication Overview](#).

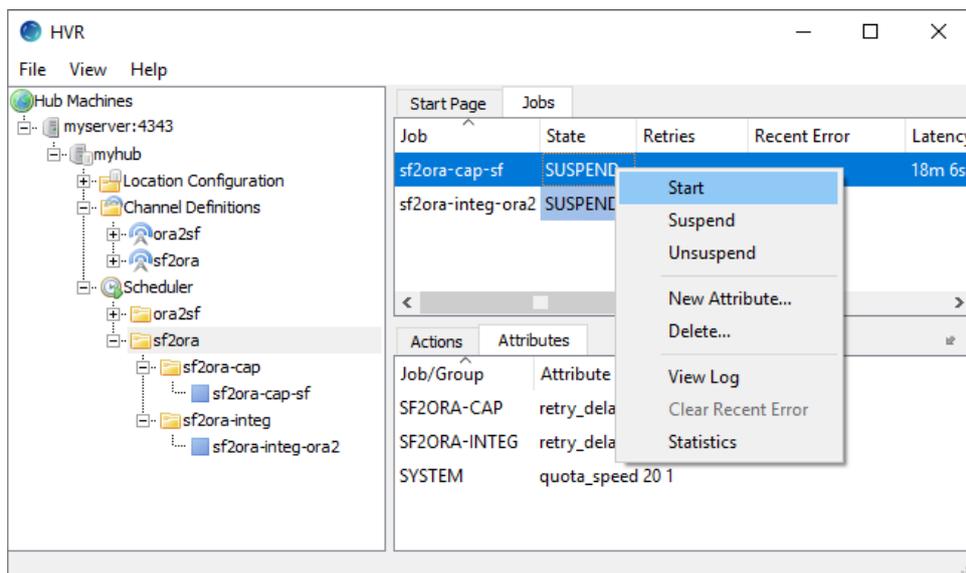
### 7. Right-click **Scheduler** and select **Start**.

## Start Capture Job

This section describes how to start the job for capturing changes from the source location (**sf**). By starting the **Capture** job under the **HVR Scheduler**, HVR begins capturing all changes since the moment HVR Initialize was executed. This 'capture begin moment' can be modified using the 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 **sf2ora-cap-sf** **Start**.

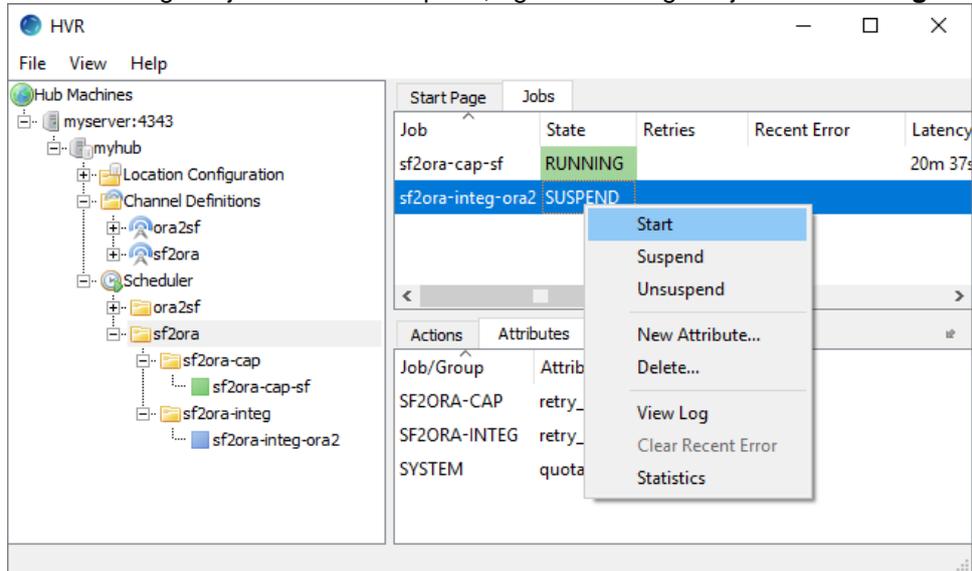


### 3. Click **Yes** in the **Start** dialog. On starting the capture job (**sf2ora-cap-sf**) successfully, the status of the job changes from **SUSPEND** to **RUNNING**.

## Start Integrate Job

This section describes how to start the job to integrate changes into the target location (**ora2**).

1. In the navigation tree pane, click **Scheduler**.
2. Start the integrate job. In the **Jobs** pane, right-click integrate job **sf2ora-integ-ora2** **Start**.



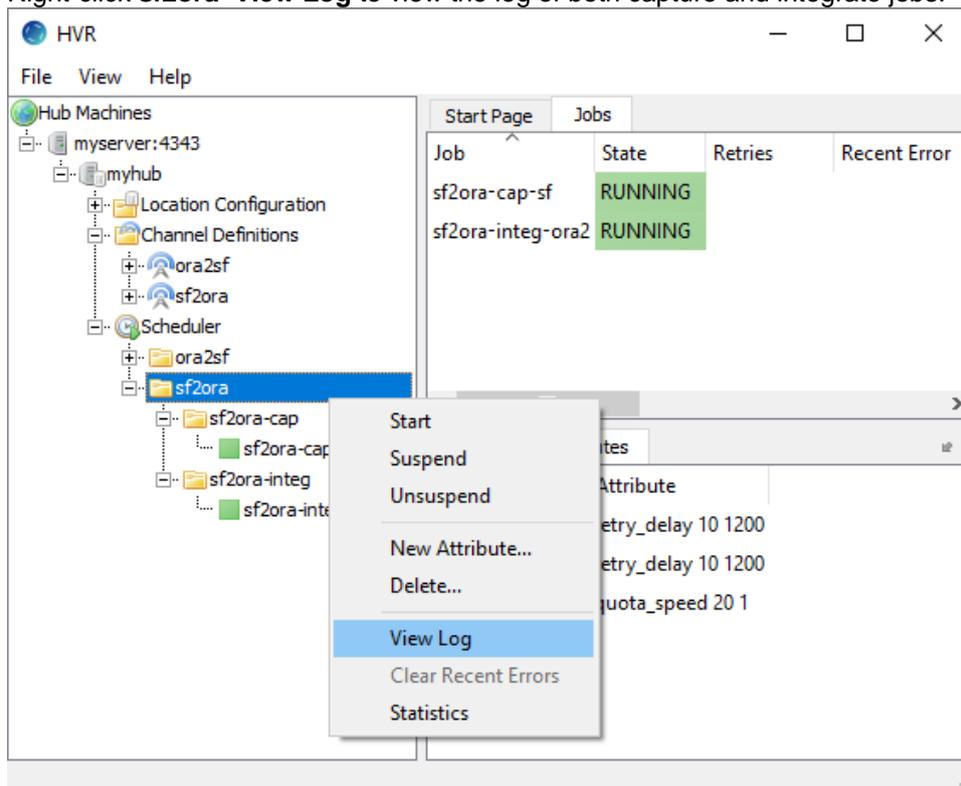
3. Click **Yes** in the **Start** dialog. On starting the integrate job (**sf2ora-integ-ora2**) successfully, the status of the job changes from **SUSPEND** to **RUNNING**.

## Verify Replication

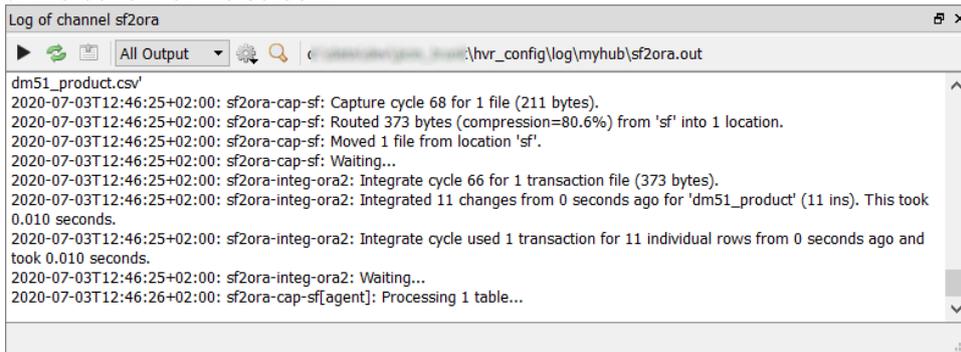
Replication can be verified by inspecting the channel log file. HVR creates separate log files for the hub, channel (**sf2ora**), and for each replication jobs (**sf2ora-cap-fs** and **sf2ora-integ-ora2**). These log files contain the details of the changes captured and integrated.

To view the replication activity log:

1. In the navigation tree pane, click + next to the **Scheduler**.
2. Right-click **sf2ora** **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 sf2ora**) at the bottom of the screen.

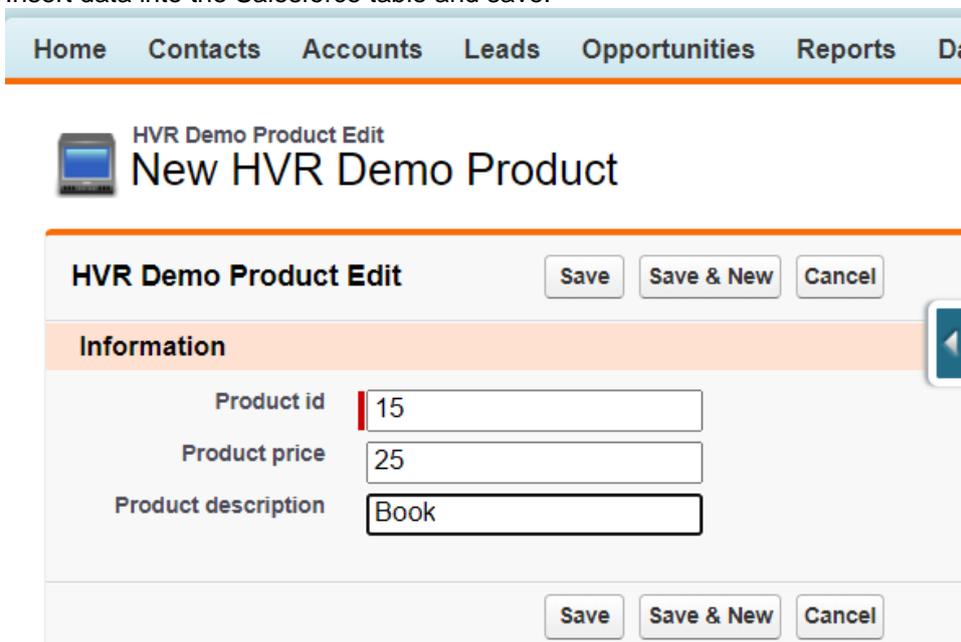


The directory path for the HVR log files is displayed in the log tab.



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

- Insert data into the Salesforce table and save.



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

