



## Repository Views

---

A view is a stored query accessible as a virtual table composed of the result set of a query. Unlike ordinary tables (base tables) in a relational database, a view does not form part of the physical schema; it is a dynamic and virtual table computed or collated from data in the database. Changing the data in a table alters the data shown in subsequent invocations of the view.

The advantages of repository views are as follows:

- Data security: Provides an additional level of table security by restricting access to a pre-determined set of rows and/or columns of a table.
- Provides an easy way to query data from different data sources like a single table.
- Useful when developing complex reports based on multiple tables.

This appendix contains the following sections:

- [Creating Repository Views, page D-1](#)
- [Using Views in Prime Provisioning, page D-2](#)

## Creating Repository Views

This section describes how to create views in Sybase repository and Oracle repository.

- [Creating Views Sybase Repository, page D-1](#)
- [Creating Views in Oracle Repository, page D-2](#)

## Creating Views Sybase Repository

### New and Upgrade Installation

All the views available in Prime Provisioning (see the [Using Views in Prime Provisioning, page D-2](#)) are created as part of the new and upgrade installation of Prime Provisioning 6.5.

# Creating Views in Oracle Repository

## New and Upgrade Installation

To create repository views (see the [Using Views in Prime Provisioning, page D-2](#)) in new and upgrade installation of Prime Provisioning 6.5, follow these steps:

---

**Step 1** Copy the **schema.tar** file to the Oracle server and then extract all files into a directory.



**Note** The schema information is held in the schema.tar file in the software package. Obtain the correct package (schemas can change between packages) and extract the **schema.tar** file from the package.

---

**Step 2** Navigate to the directory containing the expanded schema, then go to the **ddl/6.0** sub-directory.

**Step 3** Run the command **sqlplus**.

**Step 4** Log in as sysdba and provide the DBA privileges to the Prime Provisioning user using the command: **GRANT DBA, CONNECT, RESOURCE TO <isc\_user>;**

**Step 5** Log in with the username and password previously created.

**Step 6** Enter the SQL command **start DBViews.sql;**  
This will create all the views in Oracle repository.

---

## Using Views in Prime Provisioning

The different views available in Prime Provisioning are as follows:

- [Summary View, page D-2](#)
- [Site View, page D-4](#)
- [Customer View, page D-5](#)
- [Region View, page D-5](#)

## Summary View

You can query using the column name for summary view. [Table D-1](#) describes the column name and its type name.

**Table D-1** Summary View Column Names

| Column Name           | Type Name |
|-----------------------|-----------|
| SR_Number             | Integer   |
| SR_STATE              | Integer   |
| SR_Last_Modified_Time | Varchar   |

**Table D-1** Summary View Column Names (continued)

| Column Name            | Type Name |
|------------------------|-----------|
| PE_Name                | Varchar   |
| PE_Interface           | Varchar   |
| PE_Interface_IPAddress | Varchar   |
| CE_Name                | Varchar   |
| CE_Interface           | Varchar   |
| CE_Interface_IPAddress | Varchar   |
| CE_Type                | Integer   |
| CE_Site_ID             | Integer   |
| CE_Site_Name           | Varchar   |
| VPN_Name               | Varchar   |
| VRF_Name               | Varchar   |
| Customer_ID            | Integer   |
| Customer_Name          | Varchar   |
| JOB_DESCRIPTION        | Varchar   |

The description of the column name is as follows:

- **SR\_Number**—Service Request Number, represents the service request JOB ID that is available on the Service Request page in the Prime Provisioning GUI
- **SR\_STATE**—State of the Service Request and the following table maps the value in the database and its associated state:

| Database Value | Associated State |
|----------------|------------------|
| -1             | UNKNOWN          |
| 0              | All States       |
| 1              | Requested        |
| 2              | Pending          |
| 3              | Failed Deploy    |
| 4              | In Valid         |
| 5              | Deployed         |
| 6              | Broken           |
| 7              | Functional       |
| 8              | Lost             |
| 9              | Closed           |
| 10             | Failed Audit     |
| 11             | Wait Deploy      |
| 12             | In Progress      |

- SR\_Last\_Modified\_Time—last modified time of SR based on the current state of the SR
- PE\_Name—PE Host Name
- PE\_Interface—PE Interface Name associated with SR.
- PE\_Interface\_IPAddress—IP address of the PE interface
- CE\_Name—CE Host Name
- CE\_Interface—CE interface name associated with SR
- CE\_Interface\_IPAddress—IP address of the CE interface
- CE\_Type—Management type of the CE Device, the following table maps the value in the database and the CE Management Type:

| Database Value | CE Management Type                 |
|----------------|------------------------------------|
| -1             | UNKNOWN                            |
| 0              | Managed                            |
| 1              | UnManaged                          |
| 2              | Managed - Management LAN           |
| 3              | UnManaged - Management LAN         |
| 4              | Directly Connected                 |
| 5              | Directly Connected Management Host |
| 6              | Multi-VRF                          |
| 7              | Un Managed Multi-VRF               |

- CE\_Site\_ID—Site ID of the CE
- CE\_Site\_Name—Site name of the CE
- VPN\_Name—VPN name associated with SR
- VRF\_Name—VRF name associated with SR
- Customer\_ID—Customer ID
- Customer\_Name—Customer Name
- JOB\_DESCRIPTION—Job description of MPLS SR

An example for the summary view query is as follows:

```
select SR_Number, PE_Name, CE_Name, VPN_Name from Summary_View;
```

## Site View

You can query using the column name for site view. [Table D-2](#) describes the column name and its type name.

**Table D-2**      **Site View Column Names**

| Column Name | Type Name |
|-------------|-----------|
| SITE_ID     | Integer   |
| SITE_NAME   | Varchar   |
| CPE_Name    | Varchar   |
| LINK_ID     | Integer   |

The description of the column name is as follows:

- SITE\_ID—Site ID
- SITE\_NAME—Site Name
- CPE\_Name—CPE name associated with the site
- LINK\_ID—Link ID of the CPE associated to a SR

An example for the site view query is as follows:

```
select Site_Id, Site_Name, CPE_Name, Link_ID from Site_View;
```

## Customer View

You can query using the column name for customer view. [Table D-3](#) describes the column name and its type name.

**Table D-3**      **Customer View Column Names**

| Column Name      | Type Name |
|------------------|-----------|
| CUSTOMER_ID      | Integer   |
| CUSTOMER_CONTACT | Varchar   |

The description of the column name is as follows:

- CUSTOMER\_ID—Customer ID
- CUSTOMER\_CONTACT—Information about the customer

An example for the customer view query is as follows:

```
select * from Customer_View;
```

## Region View

You can query using the column name that is available for region view. [Table D-4](#) describes the column name and its type name.

**Table D-4**      **Region View Column Name**

| Column Name | Type Name |
|-------------|-----------|
| PROVIDER_ID | Integer   |
| REGION_ID   | Integer   |
| PE_NAME     | Varchar   |

The description of the column name is as follows:

- PROVIDER\_ID—Provider ID
- REGION\_ID—Region ID of the provider
- PE\_NAME—PE Host Name associated to this Region

An example for the region view query is as follows:

```
select Region_Id, PE_Name from Region_View;
```