Oracle® Business Intelligence Applications

Security Guide

Version 7.9.5.1

E13768-01

January 2009

ORACLE®

Oracle Business Intelligence Applications Security Guide, Version 7.9.5.1

E13768-01

Contents

Index

Preface

Oracle Business Intelligence Applications (Oracle BI Applications) are comprehensive prebuilt solutions that deliver pervasive intelligence across an organization, empowering users at all levels — from front line operational users to senior management — with the key information they need to maximize effectiveness. Intuitive and role-based, these solutions transform and integrate data from a range of enterprise sources, including Siebel, Oracle, PeopleSoft, and corporate data warehouses — into actionable insight that enables more effective actions, decisions, and processes.

Oracle BI Applications are built on Oracle Business Intelligence Suite Enterprise Edition, a comprehensive next-generation BI and analytics platform.

Oracle BI Applications includes the following application families:

- Oracle Financial Analytics
- Oracle Human Resources Analytics
- Oracle Supply Chain and Order Management Analytics
- Oracle Procurement and Spend Analytics
- Oracle Sales Analytics
- Oracle Service Analytics
- Oracle Contact Center Telephony Analytics
- Oracle Marketing Analytics
- Oracle Partner Analytics
- Oracle Pricing Analytics

Oracle Business Intelligence Applications Security Guide contains information about the security features in Oracle BI Applications release 7.9.5.1.

Oracle recommends reading the *Oracle Business Intelligence Applications Release Notes* before installing, using, or upgrading Oracle BI Applications. The *Oracle Business Intelligence Applications Release Notes* are available:

- On the Oracle Business Intelligence Applications CD-ROM.
- On the Oracle Technology Network at `http://www.oracle.com/technology/documentation/bi_apps.html` (to register for a free account on the Oracle Technology Network, go to `http://www.oracle.com/technology/about/index.html`).

Audience

This document is intended for BI managers and implementors of Oracle BI Applications.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at http://www.oracle.com/accessibility/.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Related Documents

For more information, see the following documents in the Oracle BI Applications Release 7.9.5 documentation set (available at http://www.oracle.com/technology/documentation/bi_apps.html):

- *Oracle Business Intelligence Applications Release Notes*
- *System Requirements and Supported Platforms for Oracle Business Intelligence Applications*
- *Oracle Business Intelligence Applications Installation Guide for Informatica PowerCenter Users*
- *Oracle Business Intelligence Configuration Guide for Informatica PowerCenter Users*
- *Oracle Business Intelligence Data Warehouse Administration Console Guide*
- *Oracle Business Intelligence Applications Upgrade Guide for Informatica PowerCenter Users*
- *Oracle Business Analytics Warehouse Data Model Reference*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

What's New in This Release

The *Oracle Business Intelligence Applications Security Guide* contains information about implementing security in Oracle BI Applications that previously appeared in *Oracle Business Intelligence Applications Installation and Configuration Guide*.

For this release, quality enhancements were made to many sections in *Oracle Business Intelligence Applications Security Guide*.

2

Integrating Security for Oracle BI Applications

This section describes the security features in Oracle BI Applications. It contains the following main topics:

2.1 Overview of Security in Oracle BI Applications

This section contains the following topics:

2.1.1 Oracle BI Applications Security Types

Oracle BI Applications integrates tightly with the security model of the operational source system to allow the right content to be shown to the right user. Oracle BI Applications has many options that an administrator can use to authenticate and show critical business data to the right people.

Security in Oracle BI Applications can be classified broadly into three different categories:

- Data security

  Data security controls the visibility of data (content rendered in subject areas, dashboards, Oracle BI Answers, and so on) based on the user's association to data in the transactional system.

- Object security

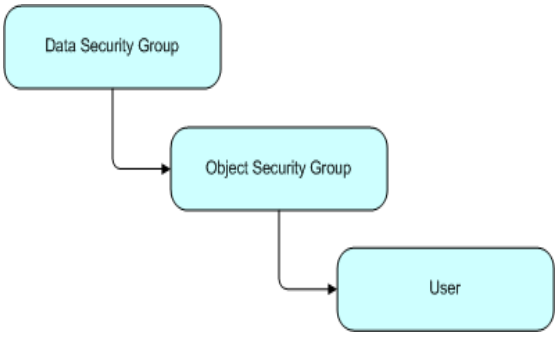
  Object security controls the visibility to business logical objects based on a user's role. You can set up object-level security for metadata repository objects, such as subject areas and presentation tables, and for Web objects, such as dashboard objects, defined in the Presentation Catalog.

- User security (authentication of users)

  User security refers to authentication and confirmation of the identity of a user based on the credentials provided.

2.1.2 Use of Security Groups in Oracle BI Applications

Object- and data-level security are implemented in Oracle BI Applications using security groups. These security groups are defined using the Security Manager in the Oracle BI Administration Tool. The standard hierarchical structure of security groups and users in Oracle BI Applications is the following: data security group, then object security group, then user, as shown in the following figure.



1. Creating security groups in the Oracle BI Repository with the same names as existing responsibilities or groups in the source applications. These security groups are added as members to Oracle BI-specific security groups, and the users will inherit this membership based on their own responsibilities or roles in the OLTP application.
2. Adding new Oracle BI-specific responsibilities (Oracle EBS and Siebel CRM Applications) or roles (PeopleSoft Enterprise applications) in the source applications, making sure their names match the object security groups in Oracle BI Applications, and assigning OLTP users to these new groups. The users will then inherit the security group membership in the same way as described in the first method above.

Note: Users should always be created in the operational application databases or directory services, such as LDAP, and never in the Oracle BI Repository. If users are created in the Oracle BI Repository, the security mechanism does not work.

For details on integrating security with source applications, see the following topics:

- Section 2.4, "Integrating Data Security for Oracle EBS"
- Section 2.5, "Integrating Data Security for Oracle's PeopleSoft Enterprise Applications"
- Section 2.6, "Integrating Data Security for Oracle's Siebel CRM"

2.1.3 Checking Oracle BI Applications User Responsibilities

An administrator can check a user's responsibility in the following ways:

- In the Siebel or Oracle EBS operational applications, go to the Responsibilities view.
- In PeopleSoft applications, go to the Roles view to check a user's roles.
- In the Oracle BI application, click on Settings/My Account link. The Presentation Services group membership for the user is shown near the bottom of the Web page. These are the Presentation Services groups, defined in the Presentation Services Catalog only. These groups are usually used to control the ability to perform actions (privileges). If a Presentation Services group has the same name as an Oracle BI Server security group, and the user is a member of the latter, than he will become automatically a member of the corresponding Presentation Services group.

2.1.4 Registering a New User Responsibility in Oracle Business Intelligence

When you add a new responsibility to a user in Oracle BI Presentation Services, the change is not immediately reflected in the Oracle Business Intelligence environment. In order to register the new user responsibility, both the administrator and the user must perform the following tasks:

1. The Oracle BI administrator must reload the Oracle BI Server metadata through Oracle BI Presentation Services.
2. Then the user must log out from the Oracle BI application (or from Siebel or Oracle EBS operational application if the user is looking at Oracle BI dashboards using an embedded application) and then log in again.

2.1.5 Default Security Settings in Oracle Business Intelligence

User Administrator and Group Administrator are a special user and group, respectively, that do not have any restrictions and do not go through the Siebel, Oracle EBS, or Peoplesoft databases. The User SADMIN is also a special user, similar to Administrator.

The Administrator group is set up as a member of the supergroup Administrators, so members of this group have no restrictions.

Note: Be sure to change the default password before migrating to production.

2.2 Data-Level Security In Oracle BI Applications

This section describes the data-level security features in Oracle BI Applications. It contains the following topics:

- Section 2.2.1, "Overview of Data-Level Security in Oracle BI Applications"
- Section 2.2.2, "Implementing Data-Level Security in the Oracle BI Repository"
- Section 2.2.3, "Initialization Blocks Used for Data-Level Security in Oracle BI Applications"

2.2.1 Overview of Data-Level Security in Oracle BI Applications

Data-level security defines what a user in an OLTP application can access inside a report. The same report, when run by two different users, can bring up different data. This is similar to how the My Opportunities view in an operational application displays different data for different users. However, the structure of the report is the

same for all users, unless a user does not have access to a column in a report, in which case the column is not displayed for that user.

Table 2–1 shows the security groups that are supported in Oracle BI Applications. During installation and configuration, you must make sure the correct security group and initialization blocks are set up for your environment.

Table 2–1 Summary of Supported Security Groups by Source System

	Oracle EBS	PeopleSoft Financials	PeopleSoft HR	Siebel
Operating Unit Org-Based security	Available since 7.9.3	Available since 7.9.3		Available since 7.5
Inventory Org-Based Security	Available since 7.9			
Company Org-Based Security	Available in 7.9.3 and obsolete in 7.9.4	Available in 7.9.3	Available in 7.9.3	
Business Group Org-Based Security	Available since 7.9.3			
HR Org-based Security			Available since 7.9.3	
Payables Org-Based Security		Available since 7.9.3		
Receivables Org-Based Security		Available since 7.9.3		
SetID-Based Security		Available since 7.9.3	Available since 7.9.3	
Position-Based Security	Available since 7.9.4 for HRMS		Available since 7.9.3	Available since 7.5
Ledger-Based Security	Available since 7.9.4	Available since 7.9.4		

2.2.2 Implementing Data-Level Security in the Oracle BI Repository

Data-level security in Oracle BI Applications is implemented in three major steps:

1. Set up initialization blocks that obtain specific security-related information when a user logs in, for example, the user's hierarchy level in the organization hierarchy, or the user's responsibilities.

   See Section 2.2.3, "Initialization Blocks Used for Data-Level Security in Oracle BI Applications".

2. Set up the joins to the appropriate security tables in the metadata physical and logical layers.

3. Set up the filters for each security group on each logical table that needs to be secured.

2.2.3 Initialization Blocks Used for Data-Level Security in Oracle BI Applications

In the Oracle BI Repository, the initialization blocks are set up for obtaining a given user's primary position, primary organization, and the owner ID, as described below:

- **Authorization**

  This initialization block is used to associate users with all security groups to which they belong. It obtains a user's responsibilities or roles from the source OLTP application, matches them with Oracle BI Applications security groups, and determines the user's applicable object security during the session. This initialization block populates a variable set called GROUP.

- **Business Groups**

  This initialization block is used to retrieve the business groups from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called BUSINESS_GROUP, which is used to drive security permissions for business group org-based security.

- **Companies**

  This initialization block is used to retrieve the companies from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called COMPANY, which is used to drive security permissions for company org-based security.

- **HR Organizations**

  This initialization block is used to retrieve the HR organizations from the OLTP application to which the corresponding login user has access. This initialization block populates a variable set called HR_ORG, which is used to drive security permissions for HR analysts.

- **Inventory Organizations**

  This initialization block is used to retrieve the inventory organizations from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called INV_ORG, which is used to drive security permissions for inventory org-based security.

- **Ledgers**

  This initialization block is used to retrieve the ledgers from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called LEDGER, which is used to drive security permissions for ledger-based security.

- **Operating Unit Organizations**

  This initialization block is used to retrieve the operating unit organizations from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called OU_ORG, which is used to drive security permissions for operating unit org-based security.

- **Orgs for Org-Based Security**

  This initialization block is used to retrieve the organizations reporting to the current user's business unit, from the Siebel CRM OLTP application.

  This initialization block populates a variable set called ORGANIZATION, which is used to drive primary org-based security.

- **Payable Organizations**

This initialization block is used to retrieve the payable organizations from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called PAYABLE_ORG, which is used to drive security permissions for payable org-based security.

- **Primary Owner ID**

  This initialization block obtains the owner ID for the given user. It obtains this information from the Siebel OLTP and populates the PR_OWNER_ID variable.

- **Payables Organizations**

  This initialization block is used to retrieve the payables organizations from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called PAYABLE_ORG, which is used to drive security permissions for payables org-based security.

- **SetID**

  This initialization block is used to retrieve the set IDs from the OLTP application to which the corresponding login responsibility has access. This initialization block populates a variable set called SET_ID, which is used to drive security permissions for Set ID-based security.

- **User Hierarchy Level**

  This initialization block obtains the fixed hierarchy level of the given user, based on the user's login, from W_POSITION_DH. It populates the variable HIER_ LEVEL. The SQL used by the block is run against the data warehouse. Therefore, it reflects the hierarchy level at the time of the last ETL run that populated this table (W_POSITION_DH).

- **User HR Organizations**

  This initialization block is used to retrieve the current HR organization from OLTP application to which the current user belongs. This initialization block populates a variable called USER_HR_ORG.

2.2.3.1 Data Security Groups in Oracle BI Applications

The table below describes the security groups used in Oracle BI Applications and the application to which they apply. Some selected security groups that share the same name as responsibilities (for Siebel CRM and Oracle EBS applications) and roles (for PeopleSoft applications). A user who has any of these responsibilities or roles in the source application will be a member of the corresponding data security group automatically when he logs in to the Oracle BI application. Other security groups based on similar objects in the source application can be added to the Oracle BI Repository and added to these data-level security groups, if you need the corresponding data filters to apply to any additional group of users. Table 2–2 shows the security groups that are supported in Oracle BI Applications.

Table 2–2 Data Security Groups in Oracle BI Applications

Security Group Name	Supported Source Application	Description	Associated Initialization Block Name
Business Group Org-Based Security	Oracle EBS, PeopleSoft HR	A business group is the highest level in the organization structure and is usually used to represent the entire enterprise or a major division. A business group can have several sets of books.	Business Groups

Table 2–2 (Cont.) Data Security Groups in Oracle BI Applications

Security Group Name	Supported Source Application	Description	Associated Initialization Block Name
Company Org-Based Security	PeopleSoft HR and Financials	This security group filters data based on the GL business units associated to the user that is logged in. The business unit is the highest level key structure in PeopleSoft.	Companies
HR Org-Based Security	PeopleSoft HR	This security group filters data based on the HR business units associated to the user that is logged in. The business unit is the highest level key structure in PeopleSoft.	HR Organizations
Human Resource Personnel Data Security	Oracle EBS, PeopleSoft HR	This security group gives HR staff access to all business groups that they are allowed to see except for their own business group.	HR Organizations User HR Organizations
Inventory Org-Based Security	Oracle EBS	An inventory organization tracks inventory transactions and balances, and/or manufactures or distributes products or components. This security group filters data based on the inventory orgs associated to the user that is logged in.	Inventory Organizations
Ledger-Based Security	Oracle EBS, PeopleSoft Financials	A ledger is essentially a reporting organization that uses a common chart of accounts, functional currency, fiscal calendar, and accounting method. This security group filters data based on the ledgers associated to the user that is logged in.	Ledgers
Operating Unit Org-Based Security	Oracle EBS, PeopleSoft Financials, Siebel CRM	This security group filters data based on the organizations associated to the user that is logged in.	Operating Unit Organizations
Payables Org-Based Security	PeopleSoft Financials	This security group filters data based on the payables business units associated to the user that is logged in. The business unit is the highest level key structure in PeopleSoft.	Payables Organizations
Primary Employee/Position Hierarchy-Based Security	Oracle EBS, PeopleSoft HR, Siebel CRM	This security group allows only the record owner and any employee up in his hierarchy chain to see the record.	User Hierarchy Level
Primary Owner-Based Security	Siebel CRM	This security group filters data based on the user that is logged.	Primary Owner ID
Receivables Org-Based Security	PeopleSoft Financials, Siebel CRM	This security group filters data based on the receivables business units associated to the user that is logged in. The business unit is the highest level key structure in PeopleSoft.	Receivables Organizations
SET ID-Based Security	PeopleSoft Financials, Oracle EBS	This security group filters data based on the Set IDs associated to the user that is logged in.	Set ID

2.3 Object-Level Security in Oracle BI Applications

This section describes the object-level security features in Oracle BI Applications. It contains the following topics:

2.3.1 Metadata Object-Level Security (Repository Groups)

Repository groups control access to metadata objects, such as subject areas, tables and columns.

2.3.1.1 Where is Repository Group Security Configured?

Metadata object security is configured in the Oracle BI Repository, using the Oracle BI Administration Tool. The Everyone user group is denied access to each of the subject areas. Each subject area is configured to give explicit read access to selected related responsibilities. This access can be extended to tables and columns.

Note: By default in Oracle BI Applications, only permissions at the subject area level have been configured.

Note: The Siebel Communications and Financial Analytics industry applications have tables and columns specific to these two industries that are industry-specific, and, therefore, hidden from other groups.

Oracle Business Intelligence supports hierarchies within the groups in the Oracle BI Repository. In the repository there are certain groups that are parent groups, which define the behavior of all the child groups. Inheritance is used to let permissions ripple through to child groups. The parent groups and their purpose are shown in Table 2–3.

Table 2–3 Repository Parent Groups

Parent Group	Permissions Inherited By
Finance	All Financial applications groups
Insurance	All Insurance applications groups
CM General	All Communications applications
Consumer Sector	Consumer Sector groups
Pharma	Life Sciences/Pharmaceuticals applications groups
Channel Managers	All Channel applications groups
Partner Managers	All Partner application groups

2.3.2 Metadata Object-Level Security (Presentation Services)

Presentation Services objects, such as dashboards and pages, are controlled using Presentation Services groups, which have the same name as the Siebel responsibilities. Access to dashboards and pages is controlled using the Presentation Services groups. If you log on as a user who belongs to the Presentation Services group Field Sales Representative Analytics, then you see only the Overview, Forecasting, and Details pages within the Pipeline Dashboard. In a similar fashion, you see only dashboards that allow you access to at least one page within that dashboard. These groups are customized in the Oracle BI Presentation Services interface.

For Oracle Business Intelligence integrated with Siebel operational applications, Presentation Services security makes use of the following principles:

- Security in Presentation Services has been preconfigured for the groups listed in Table 2–3 for each application.
- Permissions to each dashboard in Presentation Services are matched with the permissions of each related Siebel operational application view. In the Siebel operational application, views are controlled through responsibilities. However, in Oracle Business Intelligence Presentation Services, access to dashboards for each group is controlled through Web Administration. If the two access setups do not match, both of the following situations can occur:
  - If users have access to a view in the Siebel operational application, but do not have access to the corresponding dashboard, then they receive an error message indicating that they do not have access to the dashboard.
  - If users try to access a dashboard containing reports based on a subject area to which they do not have access, they see a dashboard with no reports.

2.4 Integrating Data Security for Oracle EBS

This section explains how security in Oracle BI Applications is deployed with Oracle EBS. Read this section if you want to understand how the default security settings are configured so that you can change the way security is implemented if required. This section contains the following topics:

2.4.1 Oracle BI Applications Authorization for Oracle EBS

The authorization process of Oracle BI Applications fetches a user's responsibilities from source Oracle EBS applications, matches them with all Oracle BI Applications security groups, and determines the user's applicable object security during a user's session. The initialization block Authorization is used to fetch roles and assign the result set to a special session variable called GROUP. The initialization block SQL is the following:

```
SELECT DISTINCT 'GROUP', RESPONSIBILITY_NAME FROM

FND_USER ,FND_USER_RESP_GROUPS, FND_RESPONSIBILITY_VL

WHERE

FND_USER.user_id=FND_USER_RESP_GROUPS.user_id

AND FND_USER_RESP_GROUPS.RESPONSIBILITY_ID = FND_RESPONSIBILITY_
VL.RESPONSIBILITY_ID

AND FND_USER_RESP_GROUPS.RESPONSIBILITY_APPLICATION_ID = FND_
RESPONSIBILITY_VL.APPLICATION_ID AND

FND_USER_RESP_GROUPS.START_DATE < SYSDATE AND
```

```
(CASE WHEN FND_USER_RESP_GROUPS.END_DATE IS NULL THEN SYSDATE
ELSE TO_DATE(FND_USER_RESP_GROUPS.end_Date) END) >= SYSDATE

AND FND_USER.user_id = (SELECT USER_ID FROM FND_USER WHERE USER_
NAME = ':USER')
```

2.4.2 Operating Unit-Based Security for Oracle EBS

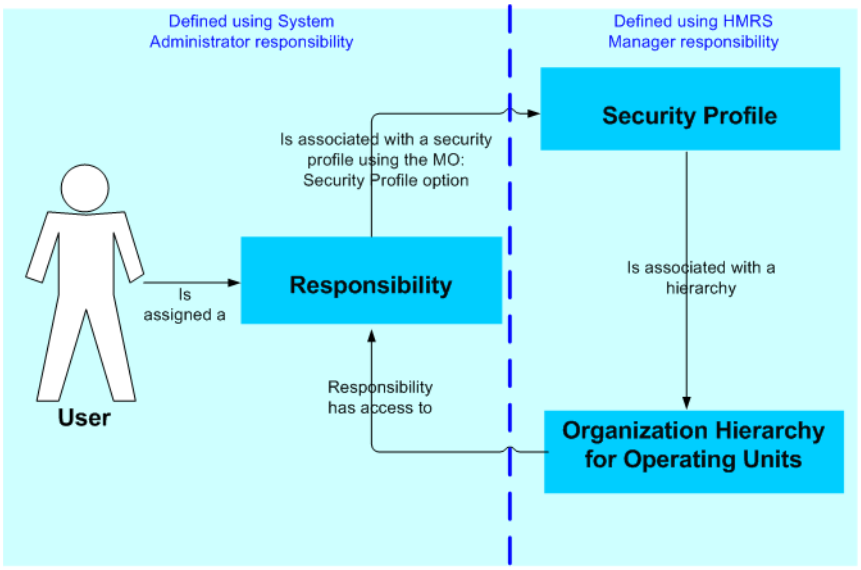
This section contains the following topics:

- Section 2.4.2.1, "About Operating Unit-Based Security for Oracle EBS"
- Section 2.4.2.2, "Implementation Steps for Operating Unit-Based Security for Oracle EBS"

2.4.2.1 About Operating Unit-Based Security for Oracle EBS

Operating units are secured by attaching a security profile to a user ID or responsibility. In turn, a security profile is associated with an organization hierarchy, which also has access to the user ID or responsibility (see Figure 2–1). The user ID or responsibility is defined using the System Administrator responsibility. The security profile and organization hierarchy are defined using the HRMS Manager responsibility.

Figure 2–1 Operating Unit-Based Security for Oracle EBS



Operating Unit assignment is decided by looking at the profiles set at the following levels, with the order of precedence indicated:

1. User
2. Responsibility
3. Application
4. Site

In other words, if a value is set in the profile at the user level and at the site level, the value set at the user level takes precedence.

2.4.2.2 Implementation Steps for Operating Unit-Based Security for Oracle EBS

The sequence for operating unit-based security for Oracle EBS is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   `USER (System variable)`

2. The 'EBS Single Sign-on Integration' session variable is initialized in the 'EBS Single Sign-on Integration' initialization block:

   `EBS_SSO_INTEGRATION_MODE`

   This session can be initialized with two possible values, 'Integrated' or 'Not Integrated', to indicate whether or not Oracle BI Applications is integrated with EBS SSO.

3. The 'EBS Security Context' initialization block then populates these session variables:

   `OLTP_EBS_RESP_ID`

   The session variable is initialized with the responsibility of the user's session in Oracle EBS if Oracle BI Applications is integrated with EBS; otherwise, it is defaulted to a random value, which will be ignored.

   `OLTP_EBS_RESP_APPL_ID`

   The session variable is initialized with the responsibility application of the user session in EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

4. The Oracle BI Server will get the operating unit corresponding to the USER from FND_USER_RESP_GROUPS. The following session variable is set automatically:

   `OU_ORG (Row-wise variable)`

   The initialization block 'Operating Unit Org', which sets the value for this variable, is shown below.

   Initialization block -- 'Operating Unit Org'

   The initialization block 'Operating Unit Org' sets the value for variable OU_ORG using the following SQL:

```
SELECT DISTINCT 'OU_ORG', TO_CHAR(PER_ORGANIZATION_
LIST.ORGANIZATION_ID)

FROM PER_ORGANIZATION_LIST,

(SELECT FND_PROFILE.VALUE_SPECIFIC('XLA_MO_SECURITY_PROFILE_
LEVEL', USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_
APPLICATION_ID) PROFILE_ID

FROM (SELECT USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_
APPLICATION_ID

FROM FND_USER_RESP_GROUPS

WHERE START_DATE < SYSDATE

AND (CASE WHEN END_DATE IS NULL THEN SYSDATE ELSE TO_
DATE(END_DATE) END) >= SYSDATE

AND USER_ID = (SELECT USER_ID FROM FND_USER WHERE USER_NAME =
':USER')
```

```
AND RESPONSIBILITY_ID = (CASE WHEN VALUEOF(NQ_SESSION.EBS_
SSO_INTEGRATION_MODE) = 'Integrated' THEN

VALUEOF(NQ_SESSION.OLTP_EBS_RESP_ID) ELSE RESPONSIBILITY_ID
END)

AND RESPONSIBILITY_APPLICATION_ID = (CASE WHEN VALUEOF(NQ_
SESSION.EBS_SSO_INTEGRATION_MODE) = 'Integrated' THEN
VALUEOF(NQ_SESSION.OLTP_EBS_RESP_APPL_ID) ELSE
RESPONSIBILITY_APPLICATION_ID END)))

WHERE PER_ORGANIZATION_LIST.SECURITY_PROFILE_ID = PROFILE_ID

UNION

SELECT DISTINCT 'OU_ORG', FND_PROFILE.VALUE_SPECIFIC('ORG_
ID', USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_APPLICATION_
ID) ORGANIZATION_ID

FROM (SELECT USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_
APPLICATION_ID

FROM FND_USER_RESP_GROUPS

WHERE START_DATE < SYSDATE

AND (CASE WHEN END_DATE IS NULL THEN SYSDATE ELSE TO_
DATE(END_DATE) END) >= SYSDATE

AND USER_ID = (SELECT USER_ID FROM FND_USER WHERE USER_NAME =
':USER')

AND RESPONSIBILITY_ID = (CASE WHEN VALUEOF(NQ_SESSION.EBS_
SSO_INTEGRATION_MODE) = 'Integrated' THEN VALUEOF(NQ_
SESSION.OLTP_EBS_RESP_ID) ELSE RESPONSIBILITY_ID END)

AND RESPONSIBILITY_APPLICATION_ID = (CASE WHEN VALUEOF(NQ_
SESSION.EBS_SSO_INTEGRATION_MODE) = 'Integrated' THEN
VALUEOF(NQ_SESSION.OLTP_EBS_RESP_APPL_ID) ELSE
RESPONSIBILITY_APPLICATION_ID END))
```

2.4.3 Inventory Org-Based Security for Oracle EBS

This section contains the following topics:

- Section 2.4.3.1, "About Inventory Org-Based Security for Oracle EBS"
- Section 2.4.3.2, "Implementation Steps for Inventory Org-Based Security for Oracle EBS"

2.4.3.1 About Inventory Org-Based Security for Oracle EBS

With inventory org-based security, the organization that a user belongs to determines which rows of data they can access. Inventory org-based security is applied based on the current logged-in responsibility rather than the current user. With Oracle EBS sources, an inventory organization can be associated with multiple responsibilities.

2.4.3.2 Implementation Steps for Inventory Org-Based Security for Oracle EBS

The sequence for inventory org-based security for Oracle EBS is described below:

1. When a user logs in to Oracle BI Applications, the following session variable is set automatically.

```
USER (System variable)
```

2. The 'EBS Single Sign-on Integration' session variable is initialized in the 'EBS Single Sign-on Integration' initialization block:

   ```
   EBS_SSO_INTEGRATION_MODE
   ```

   This session can be initialized with two possible values, 'Integrated' or 'Not Integrated', to indicate whether Oracle BI Applications is integrated with EBS SSO or not.

3. The 'EBS Security Context' initialization block then populates these session variables:

   ```
   OLTP_EBS_RESP_ID
   ```

   The session variable is initialized with the responsibility of the user session in Oracle EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

   ```
   OLTP_EBS_RESP_APPL_ID
   ```

   The session variable is initialized with the responsibility application of the user session in EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

4. The Oracle BI Server will get the inventory org corresponding to the USER from FND_USER_RESP_GROUPS. The following session variable is set automatically:

   ```
   INV_ORG (Row-wise variable)
   ```

   The initialization block 'Inventory Organizations', which sets the value for this variable, is shown below.

   Initialization block -- 'Inventory Organizations'

   The initialization block 'Inventory Organizations' sets the value for variable INV_ORG using the following SQL:

   ```
   SELECT DISTINCT 'INV_ORG', BIS_ORGANIZATIONS_V.ID

   FROM FND_USER_RESP_GROUPS, BIS_ORGANIZATIONS_V

   WHERE FND_USER_RESP_GROUPS.RESPONSIBILITY_ID = BIS_
   ORGANIZATIONS_V.RESPONSIBILITY_ID

   AND FND_USER_RESP_GROUPS.START_DATE < SYSDATE

   AND (CASE WHEN FND_USER_RESP_GROUPS.END_DATE IS NULL THEN
   SYSDATE ELSE

   AND FND_USER_RESP_GROUPS.USER_ID = (SELECT USER_ID FROM FND_
   USER WHERE USER_NAME = ':USER')

   AND RESPONSIBILITY_ID = (CASE WHEN VALUEOF(NQ_SESSION.EBS_
   SSO_INTEGRATION_MODE) = 'Integrated' THEN

   VALUEOF(NQ_SESSION.OLTP_EBS_RESP_ID) ELSE RESPONSIBILITY_ID
   END)

   AND RESPONSIBILITY_APPLICATION_ID = (CASE WHEN VALUEOF(NQ_
   SESSION.EBS_SSO_INTEGRATION_MODE) =

   'Integrated' THEN VALUEOF(NQ_SESSION.OLTP_EBS_RESP_APPL_ID)
   ELSE RESPONSIBILITY_APPLICATION_ID END)
   ```

2.4.4 Ledger-Based Security for Oracle EBS

Ledger-based security for Oracle EBS was introduced in Oracle BI Applications release 7.9.4. It replaces the company-based security to support the Oracle EBS GL set of books.

This section contains the following topics:

2.4.4.1 About Ledger-Based Security for Oracle EBS

In Oracle EBS Release 11i, a set of books is essentially a reporting entity that defines the reporting context including a chart of accounts, a functional currency, and an accounting calendar. A set of books can be assigned to a user, a responsibility, or to the site as the default for all responsibilities. Each user is associated with a single set of books when they log in to the application under a given responsibility in Oracle Applications. Ledger-based security filters data based on the set of books associated with the user that is logged in.

In Oracle EBS Release 12, the set of books is replaced by the ledger. A ledger determines the currency, chart of accounts, accounting calendar, ledger processing options and subledger accounting method. The data access set assigned to the user's responsibility controls what ledgers the user can access. A user may be able to access multiple ledgers from a responsibility. Ledger-based security filters data based on the ledgers associated with the user that is logged in.

2.4.4.2 Implementation Steps for Ledger-Based Security for Oracle EBS

The sequence for ledger-based security for Oracle EBS is described below:

1. When a user logs in to Oracle Business Intelligence Enterprise Edition, the session variable below is set automatically.

   `USER (System variable)`

2. The 'EBS Single Sign-on Integration' session variable is initialized in the 'EBS Single Sign-on Integration' initialization block:

   `EBS_SSO_INTEGRATION_MODE`

   This session can be initialized with two possible values, 'Integrated' or 'Not Integrated', to indicate whether Oracle BI Applications is integrated with EBS SSO or not.

3. The 'EBS Security Context' initialization block then populates these session variables:

   `OLTP_EBS_RESP_ID`

   The session variable is initialized with the responsibility of the user session in Oracle EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

   `OLTP_EBS_RESP_APPL_ID`

   The session variable is initialized with the responsibility application of the user session in EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

4. Then this session variable would be initialized in another init block, "Ledgers", which gets the ledgers (which is essentially the set of books in EBS) corresponding

to the USER and OLTP_EBS_RESP_ID and OLTP_EBS_RESP_APPL_ID, via table FND_USER_RESP_GROUPS and procedure FND_PROFILE.

Row-wise variable:

```
LEDGER (Row-wise variable)
```

5. The Oracle BI server gets the set of books or ledgers corresponding to the USER and OLTP_EBS_RESP_ID from the OLTP. The 'Ledgers' initialization block then populates these session variables.

The Ledgers initialization block should be set according to the Oracle EBS release, as follows:

- If you are using EBS release 12 or after, the following SQL applies as the data source in the initialization block:

```
SELECT DISTINCT 'LEDGER', TO_CHAR(GAL.LEDGER_ID)

FROM GL_ACCESS_SET_LEDGERS GAL, (SELECT FND_PROFILE.VALUE_
SPECIFIC('GL_ACCESS_SET_ID',USER_ID, RESPONSIBILITY_ID,
RESPONSIBILITY_APPLICATION_ID) PROFILE_VALUE

FROM (SELECT USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_
APPLICATION_ID

FROM FND_USER_RESP_GROUPS

WHERE START_DATE < SYSDATE AND (CASE WHEN END_DATE IS NULL
THEN SYSDATE ELSE

TO_DATE(END_DATE) END) >= SYSDATE AND USER_ID = (CASE WHEN

'VALUEOF(NQ_SESSION.EBS_SSO_INTEGRATION_MODE)' = 'Integrated'

THEN VALUEOF(NQ_SESSION.OLTP_EBS_USER_ID) ELSE (SELECT USER_
ID FROM FND_USER WHERE

USER_NAME = 'OPERATIONS') END) AND RESPONSIBILITY_ID = (CASE
WHEN

'VALUEOF(NQ_SESSION.EBS_SSO_INTEGRATION_MODE)' = 'Integrated'

THEN VALUEOF(NQ_SESSION.OLTP_EBS_RESP_ID) ELSE
RESPONSIBILITY_ID END)

AND RESPONSIBILITY_APPLICATION_ID = (CASE WHEN

'VALUEOF(NQ_SESSION.EBS_SSO_INTEGRATION_MODE)' = 'Integrated'

THEN VALUEOF(NQ_SESSION.OLTP_EBS_RESP_APPL_ID) ELSE
RESPONSIBILITY_APPLICATION_ID END)

))WHERE GAL.ACCESS_SET_ID = PROFILE_VALUE
```

- If you are using Oracle EBS 11i, the following SQL applies as the data source in the Ledger initialization block:

```
SELECT DISTINCT 'LEDGER', FND_PROFILE.VALUE_SPECIFIC('GL_SET_
OF_BKS_ID', USER_ID,

RESPONSIBILITY_ID, RESPONSIBILITY_APPLICATION_ID)

FROM (SELECT USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_
APPLICATION_ID FROM

FND_USER_RESP_GROUPS
```

```
WHERE START_DATE < SYSDATE

AND (CASE WHEN END_DATE IS NULL THEN SYSDATE ELSE TO_
DATE(END_DATE) END) >= SYSDATE

AND USER_ID IN (CASE WHEN VALUEOF(NQ_SESSION.EBS_SSO_
INTEGRATION_MODE) = 'Integrated'

THEN VALUEOF(NQ_SESSION.OLTP_EBS_USER_ID) ELSE (SELECT USER_
ID FROM FND_USER WHERE USER_NAME = ':USER') END)

AND RESPONSIBILITY_ID = (CASE WHEN VALUEOF(NQ_SESSION.EBS_
SSO_INTEGRATION_MODE) = 'Integrated'

THEN VALUEOF(NQ_SESSION.OLTP_EBS_RESP_ID) ELSE
RESPONSIBILITY_ID END)

AND RESPONSIBILITY_APPLICATION_ID = (CASE WHEN

VALUEOF(NQ_SESSION.EBS_SSO_INTEGRATION_MODE) = 'Integrated'

THEN VALUEOF(NQ_SESSION.OLTP_EBS_RESP_APPL_ID) ELSE
RESPONSIBILITY_APPLICATION_ID END))
```

2.4.5 Business Group Org-Based Security for Oracle EBS

This section contains the following topics:

- Section 2.4.5.1, "About Business Group Org-Based Security for Oracle EBS"
- Section 2.4.5.2, "Implementation Steps for Business Group Org-Based Security for Oracle EBS"

2.4.5.1 About Business Group Org-Based Security for Oracle EBS

A business group is the highest level in the organization structure. It is usually used to represent the entire enterprise or a major division. A business group can have several sets of books.

2.4.5.2 Implementation Steps for Business Group Org-Based Security for Oracle EBS

The sequence for business group org-based security for Oracle EBS is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The 'EBS Single Sign-on Integration' session variable is initialized in the 'EBS Single Sign-on Integration' initialization block:

   ```
   EBS_SSO_INTEGRATION_MODE
   ```

   This session can be initialized with two possible values, 'Integrated' or 'Not Integrated', to indicate whether Oracle BI Applications is integrated with EBS SSO or not.

3. The 'EBS Security Context' initialization block then populates these session variables:

   ```
   OLTP_EBS_RESP_ID
   ```

The session variable is initialized with the responsibility of the user session in Oracle EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

```
OLTP_EBS_RESP_APPL_ID
```

The session variable is initialized with the responsibility application of the user session in EBS if Oracle BI Applications is integrated with EBS; otherwise it is defaulted to a random value, which will be ignored.

4. The Oracle BI Server will get the set of books corresponding to the USER and OLTP_EBS_RESP_ID from FND_USER_RESP_GROUPS. The following session variable is set automatically:

   ```
   BUSINESS_GROUP (Row-wise variable)
   ```

   The initialization block 'Business Groups', which sets the value for this variable, is shown below.

   Initialization block -- 'Business Groups'

   The initialization block 'Business Groups' sets value for variable INV_ORG using the following SQL:

   ```
   SELECT DISTINCT 'BUSINESS_GROUP',

   TO_CHAR(FND_PROFILE.VALUE_SPECIFIC('PER_BUSINESS_GROUP_ID',
   USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_APPLICATION_ID))

   FROM (SELECT USER_ID, RESPONSIBILITY_ID, RESPONSIBILITY_
   APPLICATION_ID FROM FND_USER_RESP_GROUPS WHERE START_DATE <
   SYSDATE AND (CASE WHEN END_DATE IS NULL THEN SYSDATE ELSE TO_
   DATE(END_DATE) END) >= SYSDATE AND USER_ID = (SELECT USER_ID
   FROM FND_USER WHERE USER_NAME = ':USER')

   AND RESPONSIBILITY_ID = (CASE WHEN VALUEOF(NQ_SESSION.EBS_
   SSO_INTEGRATION_MODE) = 'Integrated' THEN VALUEOF(NQ_
   SESSION.OLTP_EBS_RESP_ID) ELSE RESPONSIBILITY_ID END)

   AND RESPONSIBILITY_APPLICATION_ID = (CASE WHEN VALUEOF(NQ_
   SESSION.EBS_SSO_INTEGRATION_MODE) = 'Integrated' THEN
   VALUEOF(NQ_SESSION.OLTP_EBS_RESP_APPL_ID) ELSE
   RESPONSIBILITY_APPLICATION_ID END))
   ```

Note: The 'Business Group Org-Based Security' security group contains all the data access permission filters.

2.4.6 HR Org-Based Security for Oracle EBS

The sequence for HR org-based security for Oracle EBS is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server gets the HR organizations corresponding to the USER from the following tables:
   - FND_USER_RESP_GROUPS
   - FND_USER
   - PER_SECURITY_PROFILES

- PER_SEC_PROFILE_ASSIGNMENTS
- PER_PERSON_LIST

  Note: Before the PER_PERSON_LIST table can be used, you must ensure that you have run the Oracle EBS HRMS Security List Maintenance process.
- PER_ALL_ASSIGNMENTS_F

3. The following session variable is set automatically:

   `HR_ORG (Row-wise variable)`

   The initialization block 'HR Organizations', which sets the value for this variable, is shown below.

   Initialization block -- 'HR Organizations'

   The initialization block 'HR Organizations' sets value for variable HR_ORG using the following SQL. The actual SQL query differs depending on whether Multiple Security Group (MSG) is set up or not.

   The following SQL should be used when MSG is not in place:

```
SELECT
 DISTINCT 'HR_ORG'
,TO_CHAR(SEC_DET.ORGANIZATION_ID)
FROM
(
SELECT
  'HR_ORG',
  ASG.ORGANIZATION_ID
FROM
 FND_USER_RESP_GROUPS URP
,FND_USER USR
,PER_SECURITY_PROFILES PSEC
,PER_PERSON_LIST PER
,PER_ALL_ASSIGNMENTS_F ASG
WHERE
    URP.START_DATE < TRUNC(SYSDATE)
AND (CASE WHEN URP.END_DATE IS NULL THEN TRUNC(SYSDATE) ELSE TO_DATE(URP.END_
DATE) END) >= TRUNC(SYSDATE)
AND USR.USER_NAME = ':USER'
AND USR.USER_ID = URP.USER_ID
AND TRUNC(SYSDATE)
   BETWEEN URP.START_DATE AND NVL(URP.END_DATE, HR_GENERAL.END_OF_TIME)
AND PSEC.SECURITY_PROFILE_ID = FND_PROFILE.VALUE_SPECIFIC('PER_SECURITY_
PROFILE_ID', URP.USER_ID, URP.RESPONSIBILITY_ID, URP.RESPONSIBILITY_
APPLICATION_ID)
AND PER.SECURITY_PROFILE_ID = PSEC.SECURITY_PROFILE_ID
AND PER.PERSON_ID = ASG.PERSON_ID
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND URP.RESPONSIBILITY_ID = DECODE(FND_GLOBAL.RESP_ID,
                                   -1, URP.RESPONSIBILITY_ID,
                                   NULL, URP.RESPONSIBILITY_ID,
                                   FND_GLOBAL.RESP_ID)
UNION
SELECT DISTINCT 'HR_ORG',
      ORGANIZATION_ID
FROM PER_ALL_ASSIGNMENTS_F ASG,
     FND_USER USR
WHERE ASG.PERSON_ID = USR.EMPLOYEE_ID
AND USR.USER_NAME = ':USER'
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
```

```
AND ASG.PRIMARY_FLAG = 'Y'
) SEC_DET
```

The following SQL should be used when MSG is in place:

```
SELECT
 DISTINCT 'HR_ORG'
,TO_CHAR(SEC_DET.ORGANIZATION_ID)
FROM
(
SELECT 'HR_ORG',
ASG.ORGANIZATION_ID
FROM FND_USER_RESP_GROUPS URP,
  FND_USER USR,
  PER_SEC_PROFILE_ASSIGNMENTS SASG,
  PER_SECURITY_PROFILES PSEC,
  PER_PERSON_LIST PER,
  PER_ALL_ASSIGNMENTS_F ASG
WHERE  URP.START_DATE < TRUNC(SYSDATE)
AND (CASE WHEN URP.END_DATE IS NULL THEN TRUNC(SYSDATE) ELSE TO_DATE(URP.END_
DATE) END) >= TRUNC(SYSDATE)
AND USR.USER_NAME = ':USER'
AND URP.SECURITY_GROUP_ID = SASG.SECURITY_GROUP_ID
AND URP.USER_ID = USR.USER_ID
AND TRUNC(SYSDATE)
    BETWEEN URP.START_DATE AND NVL(URP.END_DATE, HR_GENERAL.END_OF_TIME)
AND URP.USER_ID = SASG.USER_ID
AND URP.RESPONSIBILITY_ID = SASG.RESPONSIBILITY_ID
AND URP.RESPONSIBILITY_APPLICATION_ID = SASG.RESPONSIBILITY_APPLICATION_ID
AND PSEC.SECURITY_PROFILE_ID = SASG.SECURITY_PROFILE_ID
AND PSEC.SECURITY_PROFILE_ID = PER.SECURITY_PROFILE_ID
AND PER.PERSON_ID = ASG.PERSON_ID
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND TRUNC(SYSDATE) BETWEEN SASG.START_DATE AND NVL(SASG.END_DATE, HR_
GENERAL.END_OF_TIME)
AND URP.RESPONSIBILITY_ID = DECODE(FND_GLOBAL.RESP_ID,
                                   -1, URP.RESPONSIBILITY_ID,
                                   NULL, URP.RESPONSIBILITY_ID,
                                   FND_GLOBAL.RESP_ID)
UNION
SELECT DISTINCT 'HR_ORG', ORGANIZATION_ID
FROM PER_ALL_ASSIGNMENTS_F ASG,
     FND_USER USR
WHERE ASG.PERSON_ID = USR.EMPLOYEE_ID
AND USR.USER_NAME = ':USER'
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND ASG.PRIMARY_FLAG= 'Y'
) SEC_DET
```

Note: The 'HR Org-Based Security' security group contains all the data access permission filters. When users create ad-hoc reports, they see the data that is assigned with their permissions. For reports involved with the tables defined above, users are restricted to the data pertaining to their visibility in the organization structure.

2.4.7 Human Resource Personnel Data Analyst Security for Oracle EBS

HR personnel need to see all data for the internal organizations for which they are responsible and the data for their subordinates in their own organization. The 'Human Resource Personnel Data Security' security group supports this requirement. The security mechanism for this group uses the following metadata elements:

- HR_ORG variable. This variable is defined by the row-wise initialization block HR Organizations. This data set stores all the organizations the user is responsible for, plus the user's own organization, which is the same as the organization selected in USER_HR_ORG. The query for populating this data set is:

  Note: The actual SQL query differs depending on whether Multiple Security Group (MSG) is set up or not.

  The following SQL is used when MSG is not in place:

```
SELECT
 DISTINCT 'HR_ORG'
,TO_CHAR(SEC_DET.ORGANIZATION_ID)
FROM
(
SELECT
  'HR_ORG',
  ASG.ORGANIZATION_ID
FROM
 FND_USER_RESP_GROUPS URP
,FND_USER USR
,PER_SECURITY_PROFILES PSEC
,PER_PERSON_LIST PER
,PER_ALL_ASSIGNMENTS_F ASG
WHERE
    URP.START_DATE < TRUNC(SYSDATE)
AND (CASE WHEN URP.END_DATE IS NULL THEN TRUNC(SYSDATE) ELSE TO_DATE(URP.END_
DATE) END) >= TRUNC(SYSDATE)
AND USR.USER_NAME = ':USER'
AND USR.USER_ID = URP.USER_ID
AND TRUNC(SYSDATE)
   BETWEEN URP.START_DATE AND NVL(URP.END_DATE, HR_GENERAL.END_OF_TIME)
AND PSEC.SECURITY_PROFILE_ID = FND_PROFILE.VALUE_SPECIFIC('PER_SECURITY_
PROFILE_ID', URP.USER_ID, URP.RESPONSIBILITY_ID, URP.RESPONSIBILITY_
APPLICATION_ID)
AND PER.SECURITY_PROFILE_ID = PSEC.SECURITY_PROFILE_ID
AND PER.PERSON_ID = ASG.PERSON_ID
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND URP.RESPONSIBILITY_ID = DECODE(FND_GLOBAL.RESP_ID,
                                   -1, URP.RESPONSIBILITY_ID,
                                   NULL, URP.RESPONSIBILITY_ID,
                                   FND_GLOBAL.RESP_ID)
UNION
SELECT DISTINCT 'HR_ORG',
      ORGANIZATION_ID
FROM PER_ALL_ASSIGNMENTS_F ASG,
     FND_USER USR
WHERE ASG.PERSON_ID = USR.EMPLOYEE_ID
AND USR.USER_NAME = ':USER'
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND ASG.PRIMARY_FLAG = 'Y'
) SEC_DET
```

  The following SQL is used when MSG is in place:

```
SELECT
 DISTINCT 'HR_ORG'
,TO_CHAR(SEC_DET.ORGANIZATION_ID)
FROM
(
SELECT 'HR_ORG',
```

```
ASG.ORGANIZATION_ID
FROM FND_USER_RESP_GROUPS URP,
  FND_USER USR,
  PER_SEC_PROFILE_ASSIGNMENTS SASG,
  PER_SECURITY_PROFILES PSEC,
  PER_PERSON_LIST PER,
  PER_ALL_ASSIGNMENTS_F ASG
WHERE  URP.START_DATE < TRUNC(SYSDATE)
AND (CASE WHEN URP.END_DATE IS NULL THEN TRUNC(SYSDATE) ELSE TO_DATE(URP.END_
DATE) END) >= TRUNC(SYSDATE)
AND USR.USER_NAME = ':USER'
AND URP.SECURITY_GROUP_ID = SASG.SECURITY_GROUP_ID
AND URP.USER_ID = USR.USER_ID
AND TRUNC(SYSDATE)
   BETWEEN URP.START_DATE AND NVL(URP.END_DATE, HR_GENERAL.END_OF_TIME)
AND URP.USER_ID = SASG.USER_ID
AND URP.RESPONSIBILITY_ID = SASG.RESPONSIBILITY_ID
AND URP.RESPONSIBILITY_APPLICATION_ID = SASG.RESPONSIBILITY_APPLICATION_ID
AND PSEC.SECURITY_PROFILE_ID = SASG.SECURITY_PROFILE_ID
AND PSEC.SECURITY_PROFILE_ID = PER.SECURITY_PROFILE_ID
AND PER.PERSON_ID = ASG.PERSON_ID
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND TRUNC(SYSDATE) BETWEEN SASG.START_DATE AND NVL(SASG.END_DATE, HR_
GENERAL.END_OF_TIME)
AND URP.RESPONSIBILITY_ID = DECODE(FND_GLOBAL.RESP_ID,
                                   -1, URP.RESPONSIBILITY_ID,
                                   NULL, URP.RESPONSIBILITY_ID,
                                   FND_GLOBAL.RESP_ID)
UNION
SELECT DISTINCT 'HR_ORG', ORGANIZATION_ID
FROM PER_ALL_ASSIGNMENTS_F ASG,
     FND_USER USR
WHERE ASG.PERSON_ID = USR.EMPLOYEE_ID
AND USR.USER_NAME = ':USER'
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND ASG.PRIMARY_FLAG= 'Y'
) SEC_DET
```

- USER_HR_ORG variable. This variable is defined using the initialization block User HR Organizations. This variable stores the user's own organization. The query for populating this variable is:

```
SELECT DISTINCT 'USER_HR_ORG', ORGANIZATION_ID
FROM PER_ALL_ASSIGNMENTS_F ASG,
     FND_USER USR
WHERE ASG.PERSON_ID = USR.EMPLOYEE_ID
AND USR.USER_NAME = ':USER'
AND TRUNC(SYSDATE) BETWEEN ASG.EFFECTIVE_START_DATE AND ASG.EFFECTIVE_END_DATE
AND ASG.PRIMARY_FLAG= 'Y'
```

- Human Resources Analyst security group. The data filter defined for this group is the following:

```
Core."Dim - Employee Organization"."Employee Organization
Number" = VALUEOF(NQ_SESSION."HR_ORG") AND (Core."Dim -
Employee Organization"."Employee Organization Number" <>
VALUEOF(NQ_SESSION."USER_HR_ORG") OR Core."Dim - Security
Dimension"."Hierarchy Based Column" = VALUEOF(NQ_
SESSION."USER"))
```

  This filter joins the fact table used in the report to the Employee Organization dimension to get the organization number for the employee owner of the fact

record. If this organization is among the HR orgs, then it will be compared next to the user's own organization. If they are different, then there is no further check, and the record is selected. If they are the same, then an additional filter is applied based on the employee hierarchy, to make sure the employee owner of this fact record is one of the user's subordinates.

2.4.8 Employee-Based Security for Oracle EBS

Employee-based security restricts data visibility of the records to the owner of that record, and all employees he or she reports to in the company's employee hierarchy. This security mechanism uses data from the data warehouse database, and shares the metadata components with other supported applications (Siebel CRM and PeopleSoft). By default, this type of security supports only HR Analytics facts. For more information on how this security mechanism works, see Section 2.6.4, "Primary Position-Based Security for Siebel CRM Industry Applications".

2.5 Integrating Data Security for Oracle's PeopleSoft Enterprise Applications

This section explains how security is implemented for Oracle's PeopleSoft Enterprise Applications in Oracle BI Applications. Read this section if you want to understand how the default security settings are configured so that you can change the way security is implemented if required. This section contains the following topics:

2.5.1 Oracle BI Applications Authorization for PeopleSoft

The authorization process of Oracle BI Applications fetches a user's role from the source PeopleSoft application, matches the role with all Oracle BI Applications security groups, and determines the user's applicable object security during a user's session. The initialization block 'Authorization' is used to fetch roles and assign the result set to a special session variable called 'GROUP', which the Oracle BI Server then uses for matching. The initialization block SQL is the following:

```
SELECT DISTINCT

'GROUP', ROLENAME

FROM

PSROLEUSER

WHERE
```

```
ROLEUSER = ''':USER'''
```

2.5.2 Operating Unit-Based Security for PeopleSoft Financials

The sequence for operating unit-based security for PeopleSoft Financials is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server then gets the operating units (or the general ledger business units in PeopleSoft Financials) corresponding to the USER from the following tables:

   – PS_SEC_BU_OPR

   – PS_BUS_UNIT_TBL_GL

   – PS_INSTALLATION_FS

   – PS_SEC_BU_CLS

   The following session variable is set automatically:

   ```
   OU_ORG (Row-wise variable)
   ```

   The initialization block 'Operating Unit Organizations', which sets the value for this variable, is shown below.

   Initialization block -- 'Operating Unit Org'

   The initialization block 'Operating Unit Org' sets value for variable OU_ORG using the following SQL:

   ```
   SELECT DISTINCT 'OU_ORG', S1.BUSINESS_UNIT
   FROM PS_SEC_BU_OPR S1, PS_BUS_UNIT_TBL_GL A, PS_INSTALLATION_
   FS I
   WHERE S1.OPRID = ''':USER'''
   AND S1.BUSINESS_UNIT = A.BUSINESS_UNIT
   AND I.SECURITY_TYPE = 'O'
   AND I.BU_SECURITY = 'Y'
   UNION
   SELECT DISTINCT 'OU_ORG', S2.BUSINESS_UNIT
   FROM PS_SEC_BU_CLS S2,
   PS_BUS_UNIT_TBL_GL A,
   PS_INSTALLATION_FS I2,
   PSOPRDEFN P
   WHERE P.OPRID = ''':USER'''
   AND S2.BUSINESS_UNIT = A.BUSINESS_UNIT
   AND P.OPRCLASS = S2.OPRCLASS
   AND I2.SECURITY_TYPE = 'C'
   AND I2.BU_SECURITY = 'Y'
   ```

Note: The 'Operating Unit Org-Based Security' security group contains all the data access permission filters.

2.5.3 Company Org-Based Security for PeopleSoft Financials and PeopleSoft HR

The sequence for company org-based security for PeopleSoft Financials and PeopleSoft HR is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server then gets the companies or business units corresponding to the USER from the following tables:
   - PS_SEC_BU_OPR
   - PS_BUS_UNIT_TBL_GL
   - PS_SCRTY_TBL_DEPT
   - PS_BU_DEPT_VW
   - PS_BUS_UNIT_TBL_GL
   - PSOPRDEFN for PeopleSoft HR
   - PS_INSTALLATION_FS for PeopleSoft Financials
   - PSOPRDEFN for PeopleSoft Financials
   - PS_SEC_BU_CLS for PeopleSoft Financials

   The following session variable is set automatically:

   ```
   COMPANY (Row-wise variable)
   ```

   The initialization block 'Companies', which sets the value for this variable, is shown below.

   Initialization block -- 'Companies'

   The initialization block 'Companies' sets value for variable COMPANY using the following SQL:

   For PeopleSoft Financials:

   ```
   SELECT DISTINCT 'COMPANY', S1.BUSINESS_UNIT
   FROM PS_SEC_BU_OPR S1,PS_BUS_UNIT_TBL_GL A, PS_INSTALLATION_
   FS I
   WHERE S1.OPRID = ':USER'
   AND S1.BUSINESS_UNIT = A.BUSINESS_UNIT
   AND I.SECURITY_TYPE = 'O'
   UNION
   SELECT DISTINCT 'COMPANY', S2.BUSINESS_UNIT
   FROM PS_SEC_BU_CLS S2,
   PS_BUS_UNIT_TBL_GL A,
   PS_INSTALLATION_FS I2,
   PSOPRDEFN P
   ```

```
WHERE P.OPRID = ':USER'

AND S2.BUSINESS_UNIT = A.BUSINESS_UNIT

AND P.OPRCLASS = S2.OPRCLASS

AND I2.SECURITY_TYPE = 'C'

AND I2.BU_SECURITY = 'Y'
```

For PeopleSoft HR:

```
SELECT DISTINCT 'COMPANY',C.BUSINESS_UNIT

FROM PSOPRDEFN A, PS_SCRTY_TBL_DEPT B, PS_BU_DEPT_VW C, PS_
BUS_UNIT_TBL_GL D

WHERE

A.ROWSECCLASS = B.ROWSECCLASS AND

B.ACCESS_CD = 'Y' AND

B.DEPTID = C.DEPTID AND

C.BUSINESS_UNIT = D.BUSINESS_UNIT AND

A.OPRID = ':USER'
```

Note: The 'Company Org-Based Security' security group contains all the data access permission filters.

2.5.4 Ledger-Based Security for PeopleSoft Financials

Ledger data in PeopleSoft is reference data that is secured by and shared by business units. The Ledger table includes the SetID field and uses the TableSet feature in PeopleTool. In addition, ledger data access is controlled by row-level security, which enables you to implement security to restrict individual users or permission lists from specific rows of data that are controlled by the ledger. Ledger-based security filters data based on the ledgers associated with the user that is logged in.

When you set up ledger-based security for a PeopleSoft application, you should also set up the company org-based security for PeopleSoft. Ledger-based security does not automatically restrict the data by the GL business unit.

The sequence for ledger-based security for PeopleSoft Financials is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server gets the ledgers corresponding to the USER from the following tables:

   - PS_LED_DEFN_TBL
   - PS_INSTALLATION_FS
   - PS_SEC_LEDGER_CLS
   - PS_LED_GRP_TBL
   - PSOPRDEFN
   - PSROLEUSER
   - PSROLECLASS

The following session variable is set automatically:

```
LEDGER (Row-wise variable)
```

The initialization block 'Ledgers', which sets the value for this variable, is set as follows.

```
SELECT DISTINCT'LEDGER', LG.SETID || SO.LEDGER

FROM PS_SEC_LEDGER_OPR SO, PS_LED_DEFN_TBL LG, PS_
INSTALLATION_FS IFS

WHERE SO.LEDGER = LG.LEDGER AND IFS.SECURITY_TYPE = 'O'

AND IFS.LEDGER_SECURITY = 'Y' AND SO.OPRID = ':USER'

UNION

SELECT distinct 'LEDGER', LG.SETID || SC.LEDGER

FROM PS_SEC_LEDGER_CLS SC, PS_LED_GRP_TBL LG, PSOPRDEFN OP,
PSROLEUSER ORL, PSROLECLASS RCL, PS_INSTALLATION_FS IFS

WHERE SC.LEDGER_GROUP = LG.LEDGER_GROUP AND SC.OPRCLASS =
RCL.CLASSID AND OP.OPRID = ORL.ROLEUSER

AND ORL.ROLENAME = RCL.ROLENAME and IFS.SECURITY_TYPE = 'C'
AND IFS.LEDGER_SECURITY = 'Y' AND OP.OPRID = ':USER'
```

Note: The 'Ledger-Based Security' security group contains all the data access permission filters.

2.5.5 HR Org-Based Security for PeopleSoft HR

The sequence for HR org-based security with PeopleSoft HR is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server gets the HR business units corresponding to the USER from the following tables:
   - PSOPRDEFN
   - PS_SCRTY_TBL_DEPT
   - PS_BU_DEPT_VW
   - PS_BUS_UNIT_TBL_HR

   The following session variable is set automatically:

   ```
   HR_ORG (Row-wise variable)
   ```

   The initialization block 'HR Organizations', which sets the value for this variable, is shown below.

   Initialization block -- 'HR Organizations'

   The initialization block 'HR Organizations' sets value for variable HR_ORG using the following SQL:

   ```
   SELECT DISTINCT 'HR_ORG', C.BUSINESS_UNIT

   FROM PSOPRDEFN A, PS_SCRTY_TBL_DEPT B, PS_BU_DEPT_VW C, PS_
   BUS_UNIT_TBL_HR D
   ```

```
WHERE
A.ROWSECCLASS = B.ROWSECCLASS AND
B.ACCESS_CD = 'Y' AND
B.DEPTID = C.DEPTID AND
C.BUSINESS_UNIT = D.BUSINESS_UNIT AND
A.OPRID = ':USER'
```

Note: The 'HR Org-Based Security' security group contains all the data access permission filters. When users create ad-hoc reports, they see the data that is assigned with their permissions. For reports involved with the tables defined above, users are restricted to the data pertaining to their visibility in the organization structure.

2.5.6 Payables Org-Based Security for PeopleSoft Financials

The sequence for payables org-based security for PeopleSoft Financials is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server gets the Payables business units corresponding to the USER from the following tables:
   - PSOPRDEFN
   - PS_SEC_BU_OPR
   - PS_SEC_BU_CLS
   - PS_INSTALLATION_FS
   - PS_BUS_UNIT_TBL_AP

   The following session variable is set automatically:

   ```
   PAYABLES_ORG (Row-wise variable)
   ```

   The initialization block 'Payables Organizations', which sets the value for this variable, is shown below.

   Initialization block -- 'Payables Organizations'

   The initialization block 'Payables Organizations' sets value for variable PAYABLES_ORG using the following SQL:

   ```
   SELECT DISTINCT 'PAYABLES_ORG', s1.BUSINESS_UNIT
   FROM PS_SEC_BU_OPR s1, PS_BUS_UNIT_TBL_AP a, PS_INSTALLATION_
   FS i
   WHERE s1.OPRID = ':USER'
   AND s1.BUSINESS_UNIT = a.BUSINESS_UNIT
   AND i.SECURITY_TYPE = 'O'
   AND i.BU_SECURITY = 'Y'
   UNION
   SELECT DISTINCT 'PAYABLES_ORG', s2.BUSINESS_UNIT
   ```

```
FROM PS_SEC_BU_CLS s2, PS_BUS_UNIT_TBL_AP a, PS_INSTALLATION_
FS i2, PSOPRDEFN p

WHERE p.OPRID = ':USER'

AND s2.BUSINESS_UNIT = a.BUSINESS_UNIT

AND p.OPRCLASS = s2.OPRCLASS

AND i2.SECURITY_TYPE = 'C'

AND i2.BU_SECURITY = 'Y'
```

Note: The 'Payables Org-Based Security' security group contains all the data access permission filters. When users create ad-hoc reports, they see the data that is assigned with their permissions. For reports involved with the tables defined above, users are restricted to the data pertaining to their visibility in the organization structure.

2.5.7 Receivables Org-Based Security for PeopleSoft Financials

The sequence for receivables org-based security for PeopleSoft Financials is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server gets the Receivables business units corresponding to the USER from the following tables:

   - PS_SEC_BU_OPR
   - PS_SEC_BU_CLS
   - PS_INSTALLATION_FS
   - PS_BUS_UNIT_TBL_AR

   The following session variable is set automatically:

   ```
   RECEIVABLES_ORG (Row-wise variable)
   ```

   The initialization block 'Receivables Organizations', which sets the value for this variable, is shown below.

   Initialization block -- 'Receivables Organizations'

   The initialization block 'Receivables Organizations' sets value for variable RECEIVABLES_ORG using the following SQL:

   ```
   SELECT DISTINCT 'RECEIVABLES_ORG', s1.BUSINESS_UNIT

   FROM PS_SEC_BU_OPR s1, PS_BUS_UNIT_TBL_AR a, PS_INSTALLATION_
   FS i

   WHERE s1.OPRID = ':USER'

   AND s1.BUSINESS_UNIT = a.BUSINESS_UNIT AND i.SECURITY_TYPE =
   'O'

   AND i.BU_SECURITY = 'Y'

   UNION

   SELECT DISTINCT 'RECEIVABLES_ORG', s2.BUSINESS_UNIT
   ```

```
FROM PS_SEC_BU_CLS s2, PS_BUS_UNIT_TBL_AR a, PS_INSTALLATION_
FS i2, PSOPRDEFN p

WHERE p.OPRID = ':USER'

AND s2.BUSINESS_UNIT = a.BUSINESS_UNIT AND p.OPRCLASS =
s2.OPRCLASS AND i2.SECURITY_TYPE = 'C'

AND i2.BU_SECURITY = 'Y'
```

Note: The 'Receivables Org-Based Security' security group contains all the data access permission filters. When users create ad-hoc reports, they see the data that is assigned with their permissions. For reports involved with the tables defined above, users are restricted to the data pertaining to their visibility in the organization structure.

2.5.8 SetID-Based Security for PeopleSoft HR and Financials

The sequence for SetID-based security for PeopleSoft Financials is described below:

1. When a user logs in to Oracle BI Applications, the session variable below is set automatically.

   ```
   USER (System variable)
   ```

2. The Oracle BI Server gets the SetIDs corresponding to the USER from the following tables:

   - PS_SEC_SETID_OPR
   - PS_SEC_SETID_CLS
   - PS_INSTALLATION_FS
   - PSOPRDEFN

   The following session variable is set automatically:

   ```
   SET_ID (Row-wise variable)
   ```

   The initialization block 'Set ID' sets value for variable SET_ID using the following SQL:

   For PeopleSoft Financials:

   ```
   SELECT DISTINCT 'SET_ID', s1.SETID

   FROM PS_SEC_SETID_OPR s1, PS_INSTALLATION_FS i

   WHERE s1.OPRID = ':USER'

   AND i.SECURITY_TYPE = 'O'

   AND i.SETID_SECURITY = 'Y'

   UNION

   SELECT DISTINCT 'SET_ID', s2.SETID

   FROM PS_SEC_SETID_CLS s2, PS_INSTALLATION_FS i2, PSOPRDEFN p

   WHERE p.OPRID = ':USER'

   AND p.OPRCLASS = s2.OPRCLASS

   AND i2.SECURITY_TYPE = 'C'

   AND i2.SETID_SECURITY = 'Y'
   ```

Note: The 'Set ID-Based Security' security group contains all the data access permission filters.

2.5.9 Human Resource Personnel Data Analyst Security for PeopleSoft HR

HR personnel need to see all data for the internal organizations for which they are responsible for and the data for their subordinates in their own organization. The 'Human Resource Personnel Data Security' security group supports this requirement. The security mechanism for this group uses the following metadata elements:

- HR_ORG variable. This variable is defined by the row-wise initialization block HR Organizations. This data set stores all the organizations the user is responsible for, plus the user's own organization, which is the same organization as the one selected in USER_HR_ORG. The query for populating this data set is the following:

```
SELECT DISTINCT

'HR_ORG',

C.BUSINESS_UNIT

FROM

PSOPRDEFN A, PS_SCRTY_TBL_DEPT B, PS_BU_DEPT_VW C, PS_BUS_
UNIT_TBL_HR D

WHERE

A.ROWSECCLASS = B.ROWSECCLASS AND

B.ACCESS_CD = 'Y' AND

B.DEPTID = C.DEPTID AND

C.BUSINESS_UNIT = D.BUSINESS_UNIT AND

A.OPRID = ':USER'

UNION

SELECT DISTINCT 'HR_ORG', FINAL_JOB.BUSINESS_UNIT

FROM (

SELECT X.EMPLID, MAX(X.BUSINESS_UNIT) BUSINESS_UNIT FROM

(

SELECT A.EMPLID, A.EMPL_RCD, A.EFFDT,EFFSEQ, A.JOB_
INDICATOR,A.EMPL_STATUS, A.BUSINESS_UNIT

FROM PS_JOB A ,

(SELECT EMPLID, MAX(EFFDT) MAX_EFFDT

FROM PS_JOB

WHERE

JOB_INDICATOR = 'P' AND EMPL_STATUS IN ('A', 'L', 'P', 'W')

GROUP BY EMPLID) B

WHERE

A.EMPLID = B.EMPLID

AND A.EFFDT = B.MAX_EFFDT
```

```
AND A.JOB_INDICATOR = 'P' AND A.EMPL_STATUS IN ('A', 'L',
'P', 'W')

AND A.EFFSEQ = (SELECT MAX (C.EFFSEQ)

FROM PS_JOB C

WHERE

C.EMPLID = A.EMPLID AND

C.EMPL_RCD = A.EMPL_RCD AND

C.EFFDT = A.EFFDT AND

C.JOB_INDICATOR = 'P' AND C.EMPL_STATUS IN ('A', 'L', 'P',
'W'))

) X

GROUP BY X.EMPLID

) FINAL_JOB, PSOPRDEFN

WHERE

FINAL_JOB.EMPLID = PSOPRDEFN.EMPLID AND

PSOPRDEFN.OPRID = ':USER'
```

- USER_HR_ORG variable. This variable is defined using the initialization block User HR Organizations. This variable stores the user's own organization. The query for populating this variable is the following:

```
SELECT DISTINCT FINAL_JOB.BUSINESS_UNIT

FROM (

SELECT X.EMPLID, MAX(X.BUSINESS_UNIT) BUSINESS_UNIT FROM

(

SELECT A.EMPLID, A.EMPL_RCD, A.EFFDT,EFFSEQ, A.JOB_INDICATOR,
A.EMPL_STATUS, A.BUSINESS_UNIT

FROM PS_JOB A ,

(SELECT EMPLID, MAX(EFFDT) MAX_EFFDT

FROM PS_JOB

WHERE

JOB_INDICATOR = 'P' AND EMPL_STATUS IN ('A', 'L', 'P', 'W')

GROUP BY EMPLID) B

WHERE

A.EMPLID = B.EMPLID

AND A.EFFDT = B.MAX_EFFDT

AND A.JOB_INDICATOR = 'P' AND A.EMPL_STATUS IN ('A', 'L',
'P', 'W')

AND A.EFFSEQ = (SELECT MAX (C.EFFSEQ)

FROM PS_JOB C

 WHERE
```

```
C.EMPLID = A.EMPLID AND

C.EMPL_RCD = A.EMPL_RCD AND

C.EFFDT = A.EFFDT AND

C.JOB_INDICATOR = 'P' AND C.EMPL_STATUS IN ('A', 'L', 'P',
'W'))

) X

GROUP BY X.EMPLID

) FINAL_JOB, PSOPRDEFN

WHERE

FINAL_JOB.EMPLID = PSOPRDEFN.EMPLID AND

PSOPRDEFN.OPRID = ':USER'
```

- Human Resources Analyst security group. The data filter defined for this group is the following:

  ```
  Core."Dim - Employee Organization"."Employee Organization
  Number" = VALUEOF(NQ_SESSION."HR_ORG") AND (Core."Dim -
  Employee Organization"."Employee Organization Number" <>
  VALUEOF(NQ_SESSION."USER_HR_ORG") OR Core."Dim - Security
  Dimension"."Hierarchy Based Column" = VALUEOF(NQ_
  SESSION."USER"))
  ```

This filter joins the fact table used in the report to the Employee Organization dimension to get the organization number for the employee owner of the fact record. If this organization is among the HR orgs, then it will be compared next to the user's own organization. If they are different, then there is no further check, and the record is selected. If they are the same, then an additional filter is applied based on the employee hierarchy, to make sure the employee owner of this fact record is one of the user's subordinates.

2.5.10 Employee-Based Security for PeopleSoft

Employee-based security restricts data visibility of the records to the owner of that record, and all employees he or she reports to in the company's employee hierarchy. This security mechanism uses data from the Oracle Business Analytics Warehouse database, and shares the metadata components with other supported applications (for example, Oracle EBS, Siebel CRM, or PeopleSoft). By default, this type of security supports only HR Analytics facts. For more information about how this security mechanism works, see Section 2.6.4, "Primary Position-Based Security for Siebel CRM Industry Applications".

2.6 Integrating Data Security for Oracle's Siebel CRM

This section explains how security in Oracle BI Applications is deployed with Siebel CRM. Read this section if you want to understand how the default security settings are configured so that you can change the way security is implemented if required. This section contains the following topics:

- Section 2.6.1, "Primary Position-Based Security"
- Section 2.6.2, "Partner Analytics Security Settings"
- Section 2.6.3, "Usage Accelerator Analytics Security Settings"

2.6.1 Primary Position-Based Security

This section covers primary position-based security. It contains the following topics:

2.6.1.1 Introduction

Primary position-based security restricts data visibility for a fact or dimension record to the primary owner of this record and those above him in the hierarchy. The primary owner of a record could be a position or an employee. Primary position-based security uses a flattened hierarchy table called W_POSTION_DH, which is based on W_POSITION_D and is treated as a slowly changing dimension.

For Siebel CRM-based data, W_POSITION_D is populated from the Position table in Siebel CRM, so a new record is created for the same position every time a new employee is associated with this position as the primary employee.

Consequently, every record in the source tables can be represented by more than one record in W_POSITION_DH, but only one record can have the value of CURRENT_FLG as 'Y' at any time. The W_POSITION_DH table also contains one set of columns prefixed with CURRENT, and another set of columns not prefixed with CURRENT. The columns that are prefixed with CURRENT reflect the current hierarchy structure for the position or employee record at any time. The columns that are not prefixed with CURRENT reflect the hierarchy structure for the same position or employee record during the period between EFFECTIVE_START_DT and EFFECTIVE_END_DT. This latter set of columns is used to enable fact records to be visible to the owner of a record and his upper level managers at the time the record was created, even after he changes position or managers in the company hierarchy.

Facts join to this dimension by the record owner; for example, W_REVN_F is joined using PR_POSITION_DH_WID, where PR_POSITION_DH_WID is the primary position on the revenue line in the source application. Another example is W_PAYROLL_F is joined using EMP_POSTN_DH_WID, where EMP_POSTN_DH_WID is the employee owner of this payroll record.

2.6.1.2 Primary Employee/Position Hierarchy-Based Security Group

This security group uses the following metadata elements in the repository:

- HIER_LEVEL session variable. This variable is populated by the initialization block 'User Hierarchy Level' using the following SQL:

  ```
  Select round(FIXED_HIER_LEVEL) FROM VALUEOF(OLAPTBO).W_
  POSITION_DH WHERE BASE_LOGIN= ':USER' AND CURRENT_FLG='Y'
  ```

  The HIER_LEVEL value can be a number between 0 and 9 and designates the level of the user in the position or employee hierarchy of the company. For example, the CEO of the company is the only employee whose HIER_LEVEL takes the value 9, if the employee hierarchy is a full tree.

- Dim - Security logical dimension. This logical dimension is joined to the supported fact tables. It is defined using on the physical table W_POSITION_DH.
- Hierarchy-Based Column logical column. This column is a logical column in the Dim - Security logical dimension. It is defined as follows:

  ```
  "INDEXCOL(VALUEOF(NQ_SESSION."HIER_LEVEL"), "Core"."Dim -
  Security Dimension"."Current Base Level Login", "Core"."Dim -
  Security Dimension"."Current Level 1 Login", "Core"."Dim -
  Security Dimension"."Current Level 2 Login", "Core"."Dim -
  Security Dimension"."Current Level 3 Login", "Core"."Dim -
  Security Dimension"."Current Level 4 Login", "Core"."Dim -
  Security Dimension"."Current Level 5 Login", "Core"."Dim -
  Security Dimension"."Current Level 6 Login", "Core"."Dim -
  Security Dimension"."Current Level 7 Login", "Core"."Dim -
  Security Dimension"."Current Level 8 Login", "Core"."Dim -
  Security Dimension"."Current Top Level Login")".
  ```

  The IndexCol function in this definition makes the Hierarchy-Based Column default to one of the logical columns in the list based on the value of HIER_ LEVEL. So, if the value of HIER_LEVEL is 0, the new column will default to the first column in the list, and so on.

- A filter in the security group 'Primary Employee/Position Hierarchy-Based Security' defined as follows: ("Core"."Dim - Security Dimension"."Hierarchy Based Column" = VALUEOF(NQ_SESSION."USER")).

  A user needs to be a member of the security group 'Primary Employee/Position Hierarchy-Based Security', through one of his responsibilities (for Siebel and Oracle EBS applications) and Roles (for PeopleSoft applications), for the data security filters to apply. Users are assigned to this security group based on their responsibilities, using the Authorization initialization block, as described in Section 2.2.3, "Initialization Blocks Used for Data-Level Security in Oracle BI Applications." By default, this initialization block is populated using the following SQL:

  ```
  Select 'GROUP', R.NAME

  from VALUEOF(TBO).S_RESP R, VALUEOF(TBO).S_PER_RESP P,
  VALUEOF(TBO).S_USER U

  where U.LOGIN=Upper(':USER') and U.ROW_ID=P.PER_ID and
  P.RESP_ID=R.ROW_ID

  UNION

  select 'GROUP', CASE VALUEOF(NQ_SESSION.HIER_LEVEL)

  WHEN 0 THEN 'Hierarchy Level (Base)'

   when 1 then 'Hierarchy Level 1'

  when 2 then 'Hierarchy Level 2'

  when 3 then 'Hierarchy Level 3'

  when 4 then 'Hierarchy Level 4'

  when 5 then 'Hierarchy Level 5'

   when 6 then 'Hierarchy Level 6'

  when 7 then 'Hierarchy Level 7'

  when 8 then 'Hierarchy Level 8'
  ```

```
When 9 then 'Hierarchy Level (Top)'

ELSE'NOGROUP'END from VALUEOF(TBO).S_DUAL
```

The first part of this SQL selects the user's responsibilities from the Siebel CRM application. The user will be assigned automatically to the security groups with the same name in the Oracle BI Repository.

The second part of this SQL assigns the user to one of the Oracle BI-specific security groups, such as Hierarchy Level (Base), Hierarchy Level 1 through 8, and Hierarchy Level (Top), based on the variable HIER_LEVEL. These security groups are not used for data security purposes; they are used for Presentation column purposes, in conjunction with the Web Choose function defined in some reports. The purpose of this function is to allow a multi-user report to show different position columns to the user, based on his hierarchy level. This is very similar to the IndexCol function described in Section 2.6.1.2, "Primary Employee/Position Hierarchy-Based Security Group."

2.6.1.3 Configuring Oracle BI Repository Table Joins for Primary Employee/Position Hierarchy-Based Security

The procedures below provide instructions for adding primary position-based security to a new dimension or fact table. The following procedures use the W_AGREE_D (Agreement) dimension as an example.

To add primary position-based security to a dimension table

1. Create an alias on W_POSITION_DH specifically to join to the underlying physical table.
2. Configure the join in the physical layer.
3. Add the W_POSITION_DH alias to the dimension's Logical table source.
4. Add new logical columns CURRENT_BASE_LOGIN, CURRENT_LVL1ANC_ LOGI; etc. to the logical table, and map them to the corresponding physical columns.
5. Add the Hierarchy column 'Hierarchy Based Column', as defined in described in section 3.2 above.
6. Open the security group screen using Manage/Security in Oracle BI Administrator.
   a. Right-click the group 'Primary Employee/Position Hierarchy-Based Security'.and choose Properties.
   b. In the Properties dialog, click the Permissions box and select the Filter tab.
   c. To add a new filter, click on the Add button.
   d. In the new dialog, select the Business Model tab, and find the logical table: Dim - Agreement.

      A new record will be added to the list of Filters automatically.
   e. Click on the ellipsis box, and add the filter condition "Core"."Dim - Customer"."Hierarchy Based Login" = VALUEOF(NQ_SESSION."USER") in the Security Filter Expression Builder and click OK.

To add primary position-based security support to a fact table

1. Join the underlying physical table to Dim_W_POSITION_DH_Position_Hierarchy.

This assumes you already created the appropriate foreign key in the fact table and populated it correctly.

2. Join the logical table to the Dim - Security Dimension.
3. Open the security group screen using Manage/Security in Oracle BI Administrator.
   a. Right-click the group 'Primary Employee/Position Hierarchy-based Security', and choose Properties.
   b. In the Properties dialog, click the Permissions box and select the Filter tab.
   c. To add a new filter, click on the Add button.
   d. In the new dialog, select the Business Model tab, and find the logical table: Dim - Agreement.

      A new record will be added to the list of Filters automatically.
   e. Click on the ellipsis box, and add the condition "Core"."Dim - Security Dimension"."Hierarchy Based Column" = VALUEOF(NQ_SESSION."USER") in the Security Filter Expression Builder and click OK.

2.6.2 Partner Analytics Security Settings

Oracle Partner Analytics incorporates the concept of role-based analytics. Role-based analytics provides brand owners the ability to display dashboards and pages to users based on their specific roles. For example, a sales manager would have the ability to view dashboards related to pipeline and sales effectiveness, whereas the marketing manager would have the ability to view dashboards related to campaigns. Oracle Partner Analytics also includes flexible security mechanisms to control access to subject areas and to data.

Oracle Partner Analytics roles map to Siebel responsibilities in the Siebel operational application. This section describes the roles and associated dashboards and pages for both Partner Manager and Partner Portal applications. It also includes subject area and data-level security settings for responsibilities.

2.6.2.1 PRM Partner Portal Role-Based Interactive Dashboards Mapping

The dashboard and page tab mapping for specific responsibilities in the PRM Partner Portal application are shown in Table 2–4.

Table 2–4 Responsibilities for PRM Partner Portal Analytics

Responsibility	Dashboard	Page Tab Name
Partner Executive Analytics User	Partner Executive	Pipeline
	Partner Executive	Products
	Partner Executive	Sales Effectiveness
	Partner Executive	Service
Partner Operations Analytics User	Partner Commerce	Overview
	Partner Commerce	Products
	Partner Marketing	Overview
	Partner Marketing	ROI
	Partner Sales	Pipeline

Table 2–4 (Cont.) Responsibilities for PRM Partner Portal Analytics

Responsibility	Dashboard	Page Tab Name
	Partner Sales	Revenue
	Partner Service	Customer Sat
	Partner Service	Overview
	Partner Service	Service Requests
	Partner Training	Training
Partner Sales Manager Analytics User	Partner Commerce	Orders
	Partner Commerce	Overview
	Partner Commerce	Quotes
	Partner Sales	Pipeline
	Partner Sales	Revenue
	Partner Sales	Subordinates
	Partner Training	Subordinates
Partner Sales Rep Analytics User	Partner Commerce	Orders
	Partner Commerce	Overview
	Partner Commerce	Quotes
	Partner Sales	Pipeline
	Partner Sales	Revenue
	Partner Sales	Subordinates
	Partner Training	Subordinates
Partner Service Manager Analytics User	Partner Service	Customer Sat
	Partner Service	Overview
	Partner Service	Service Requests
	Partner Service	Subordinates
	Partner Training	Subordinates
Partner Service Rep Analytics User	Partner Service	Overview
	Partner Service	Service Requests
	Partner Service	Subordinates
	Partner Training	Subordinates

2.6.2.2 Partner Manager Role-Based Interactive Dashboards Mapping

Table 2–5 provides the dashboard and page tab mapping for specific responsibilities in the Siebel PRM Partner Manager application.

Table 2–5 Siebel Responsibilities for PRM Analytics

Responsibility	Dashboard	Page Tab Name
Channel Account Manager Analytics User	Channel Customers	Overview
	Channel Customers	Sales
	Channel Sales	Products
	Channel Sales	Sales
	Channel Service	Products
	Channel Service	Service
	Channel Training	Training Profile
Channel Executive Analytics User	Channel Customers	Customer Profile
	Channel Executive	Customer Satisfaction
	Channel Executive	Pipeline
	Channel Executive	Product
	Channel Executive	Program
	Channel Executive	Revenue
	Channel Executive	Service
	Channel Segmentation	Channel Mix
	Channel Segmentation	Partner Territory
	Channel Segmentation	Partner Tier
	Channel Segmentation	Partner Type
Channel Marketing Manager Analytics User	Channel Customers	Overview
	Channel Customers	Sales
	Customer Marketing	Effectiveness
	Customer Marketing	Responses
	Customer Marketing	ROI
Channel Operations Analytics User	Channel Commerce	Orders
	Channel Commerce	Overview
	Channel Commerce	Quotes
	Channel Commerce	Products
	Channel Customers	Overview
	Channel Customers	Sales
	Channel Customers	Service
	Channel Marketing	Effectiveness
	Channel Marketing	Overview

Table 2–5 (Cont.) Siebel Responsibilities for PRM Analytics

Responsibility	Dashboard	Page Tab Name
	Channel Sales	Margins
	Channel Sales	Pipeline
	Channel Sales	Revenue
	Channel Sales	Sales Cycle
	Channel Sales	Wins
	Channel Segmentation	Partner Territory
	Channel Segmentation	Partner Tier
	Channel Segmentation	Partner Type
	Channel Service	Customer Satisfaction
	Channel Service	Overview
	Channel Service	Products
	Channel Service	Resolution Time
	Channel Service	Service Requests
	Channel Training	Overview
	Channel Training	Performance

2.6.2.3 PRM Analytics Subject Area Mappings

Ad hoc queries in Siebel PRM Analytics are built by the user, depending on user responsibilities and based on columns in subject areas in the Oracle BI application. By restricting visibility to subject areas based on responsibilities, PRM Analytics provides brand owners a flexible way to deploy role-based analytics.

The subject area visibility for responsibilities in Partner Manager are shown in Table 2–6, where an X indicates that the subject area is visible for the user holding that responsibility.

Table 2–6 Responsibilities for PRM Partner Manager Analytics

Subject Area	Channel Executive Analytics User	Channel Operations Analytics User	Channel Account Manager Analytics User	Channel Marketing Manager Analytics User
Activities	X	X	X	X
Assets	X	X	X	
Campaigns	X	X	X	X
Consumers	X	X	X	X
Customer Satisfaction	X	X	X	
Customers	X	X	X	X
Orders	X	X	X	X
Partner Training	X	X	X	

Table 2–6 (Cont.) Responsibilities for PRM Partner Manager Analytics

Subject Area	Channel Executive Analytics User	Channel Operations Analytics User	Channel Account Manager Analytics User	Channel Marketing Manager Analytics User
Partners	X	X	X	X
Pipeline	X	X	X	X
Pricing	X	X	X	X
Products	X	X	X	X
Real-Time Activity				
Real-Time Assets				
Service Requests	X	X	X	

2.6.2.4 PRM Analytics Subject Area Visibility

The subject area visibility for roles in Partner Portal is shown in Table 2–7, where an X indicates that subject area is visible for the user holding that responsibility.

Table 2–7 Subject Area Visibility for PRM Partner Portal

Subject Area	Partner Executive Analytics User	Partner Operations Manager Analytics User	Partner Sales Manager Analytics User	Partner Sales Rep Analytics User	Partner Service Manager Analytics User	Partner Service Rep Analytics User
Activities	X	X	X	X	X	X
Assets	X	X			X	X
Campaigns	X	X				
Consumers	X	X				
Customer Satisfaction	X	X			X	X
Customers	X	X	X	X	X	X
Orders	X	X	X	X	X	X
Partner Training	X	X	X	X	X	X
Partners	X	X				
Pipeline	X	X	X	X		
Pricing						
Products	X	X	X	X	X	X
Real-Time Activity						
Real-Time Assets						
Service Requests	X	X			X	X

2.6.2.5 PRM Analytics Data-Level Visibility

PRM Analytics also provides brand owners the ability to restrict security based on the user's organization or position. This security mechanism makes sure that one user

does not have access to another user's data. It also makes sure that one partner does not have access to another partner's data. Data-level security is administered for responsibilities. Details regarding setting up data -level visibility are provided in the topic Section 2.2.2, "Implementing Data-Level Security in the Oracle BI Repository."

Table 2–8 shows the data-level security settings included for the responsibilities in Partner Manager and Partner Portal.

Table 2–8 Oracle PRM Data-Level Security Settings

Responsibility	Data-Level Security	Type	Comments
Channel Executive Analytics User	No	N/A	N/A
Channel Operations Analytics User	No	N/A	N/A
Channel Account Manager Analytics User	No	N/A	N/A
Channel Marketing Manager Analytics User	No	N/A	N/A
Partner Executive Analytics User	Yes	Organization	Displayed records should match organization of the user.
Partner Sales Manager Analytics User	Yes	Organization	Displayed records should match organization of the user.
Partner Sales Rep Analytics User	Yes	Position	Displayed records should match position of the user.
Partner Service Manager Analytics User	Yes	Organization	Displayed records should match organization of the user.
Partner Service Rep Analytics User	Yes	Position	Displayed records should match position of the user.

2.6.3 Usage Accelerator Analytics Security Settings

Table 2–9 describes the additional security configurations that may be necessary and the particular responsibilities associated with the Oracle Usage Accelerator dashboards.

Table 2–9 Usage Accelerator Responsibilities and Dashboards

User Responsibility	Data-Level Security	Dashboard Name (View)	Dashboard Page
Usage Accelerator–Sales Rep	Primary Position Data-Level Security	Score Card	Individual Scorecard
Usage Accelerator–Financial Services Sales Rep		Action Plan	Account Coverage Contact Coverage Opportunity Coverage Financial Account Coverage—Financial Services only Account Completeness Contact Completeness Opportunity Updates

Table 2–9 (Cont.) Usage Accelerator Responsibilities and Dashboards

User Responsibility	Data-Level Security	Dashboard Name (View)	Dashboard Page
Usage Accelerator–Sales Manager	No position-based Security	Score Card	Team Scorecard Individual Scorecard
		Action Plan	Account Coverage (Team) Contact Coverage (Team) Opportunity Coverage (Team) Financial Account Coverage (Team) —Financial Services only Account Completeness (Team) Contact Completeness (Team) Opportunity Updates (Team)
Usage Accelerator–Financial Services Sales Manager		Coverage	Account Coverage Account Coverage (Team) Contact Coverage Opportunity Coverage Financial Account Coverage—Financial Services only
		Completeness	Account Completeness Contact Completeness
		Opportunity Updates	Opportunity Updates
		User Adoption	Active Users Application Usage—excluded for Financial Services Application Usage—Financial Services only*
Usage Accelerator–Sales Executive	No position-based Security	Scorecard	Organization Scorecard Individual Scorecard
Usage Accelerator–Financial Services Sales Executive		Action Plan	Account Coverage (Org) Contact Coverage (Org) Opportunity Coverage (Org) Financial Account Coverage (Org)—Financial Services only Account Completeness (Org) Contact Completeness (Org) Opportunity Updates (Org)

2.6.4 Primary Position-Based Security for Siebel CRM Industry Applications

This section covers primary position-based security for CRM Industry Applications. It contains the following topics:

- Section 2.6.4.1, "Consumer Sector Analytics Security Settings"
- Section 2.6.4.2, "Communications, Media, and Energy (CME) Analytics Security Settings"
- Section 2.6.4.3, "Financial Services Analytics Security Settings"
- Section 2.6.4.4, "Pharma Sales Analytics and Pharma Marketing Analytics Security Settings"

2.6.4.1 Consumer Sector Analytics Security Settings

Table 2–10 describes the consumer sector responsibilities associated with each CS Dashboard.

Table 2–10 Consumer Sector Responsibilities Associated with Each CS Dashboard

Responsibility	Dashboard	Pages
VP Sales	VP Sales	Business Overview, Product Overview
	Sales Performance	Sales Volume Planning, Hierarchy, Trends, Growth
	Promotion	Plan Year to Date, Corporate
Key Account Manager	Key Account Manager	Business, Category
	Promotion	Plan year to date, Key account
	Funds	Account
	Retail Audit	Last audit, Trends
	Sales Performance	Sales Volume Planning, Hierarchy, Trends, Growth

2.6.4.2 Communications, Media, and Energy (CME) Analytics Security Settings

Oracle's CME family of products (Oracle Communications, Media and Energy Sales Analytics, Oracle Communications, Media and Energy Service Analytics, Oracle Communications, Media and Energy Marketing Analytics) use the Siebel operational applications security model; that is, it uses Siebel operational applications responsibilities (and corresponding repository and Presentation Services groups) for controlling access to Siebel operational applications objects (both metadata and Presentation Services objects). This security model is described in the topic Section 2.1, "Overview of Security in Oracle BI Applications."

In addition to responsibilities provided by the operational applications, Oracle Communications, Media, and Energy (CME) provides additional responsibilities, and responsibility-specific security, as indicated in Table 2–11.

Table 2–11 CME Responsibilities Associated with Each CME Dashboard

CME Responsibility	CME Dashboard	Dashboard Pages
CM Marketing Analytics User CM Marketing Analytics Administrator	Loyalty Management	▪ Customer Lifetime Value ▪ Churn Propensity ▪ Selling Propensity ▪ Financial Risk ▪ Actual Churn
CM Sales Analytics User CM Sales Analytics Administrator	Revenue Management	▪ Sales Portal ▪ Service Activations ▪ Service Modifications ▪ Service Disconnections

Table 2–11 (Cont.) CME Responsibilities Associated with Each CME Dashboard

CME Responsibility	CME Dashboard	Dashboard Pages
	Account Management	■ Sales Portal ■ Service Activations ■ Service Modifications ■ Service Disconnections
CM Service Analytics User CM Service Analytics Administrator	Account Management	■ Trouble Tickets ■ Customer Satisfaction

2.6.4.3 Financial Services Analytics Security Settings

The following applications use the Siebel operational applications security model:

- The Financial Analytics family of products (Finance Sales Analytics, Finance Service Analytics, Finance Marketing Analytics, Finance Institutional Analytics, Finance Retail Analytics).
- The Insurance Analytics family of products (Insurance Partner Manager Analytics, Insurance Sales Analytics, Insurance Service Analytics, Insurance Marketing Analytics, Insurance Partner Manager Analytics).

In addition to responsibilities provided by the Siebel operational applications, these applications provide additional responsibilities, and responsibility-specific security, as indicated in Table 2–12.

For the Financial Services products, the Siebel operational applications security model has been extended in the following ways:

- **Financial Analytics user**

  A finance-specific responsibility (and corresponding repository and Presentation Services group) that must be used in conjunction with Siebel operational applications responsibilities and groups to control access to Finance-specific objects in Financial Analytics.

- A user in the Insurance Analytics family of products (Insurance Partner Manager Analytics, Insurance Sales Analytics, Insurance Service Analytics, Insurance Marketing Analytics, Insurance Partner Manager Analytics)

  An insurance-specific responsibility (and corresponding repository and Presentation Services group) that must be used to control access to the Insurance and Healthcare-specific objects in Insurance and the Healthcare Analytics family of products (Healthcare Sales Analytics, Healthcare Service Analytics, Healthcare Marketing Analytics, Healthcare Partner Manager Analytics).

For example, when you give a salesperson all horizontal Sales responsibilities and also include the finance responsibility Financial Analytics User, this user is able to see, in addition to all horizontal sales objects (dashboards, subject areas, folders in the Presentation layer, and so on), all finance-specific Sales objects. Similarly, in order to see Insurance and Healthcare-specific objects, you need to add one of the Insurance Analytics family of products (Insurance Partner Manager Analytics, Insurance Sales Analytics, Insurance Service Analytics, Insurance Marketing Analytics, Insurance Partner Manager Analytics) user responsibilities to this user.

2.6.4.3.1 Parent and Child Group Behavior Oracle BI Applications supports hierarchies in the repository groups, and certain groups within the Oracle BI Repository are parent groups that define the behavior of all the child groups. For Financial Services Analytics, the parent groups are the following:

- **Finance**

  Parent group for all Financial applications groups. Financial Analytics User is a child group of Finance group.

- **Insurance**

  Parent group for all Insurance applications groups. Insurance Analytics User is a child group of Insurance group.

Inheritance is used to let permissions ripple through to child groups. The parent groups for Financial Services and their purpose are shown in Table 2–12.

> **Note:** A Financial Services Analytics user is provided as a child to both Finance and Insurance. Therefore, this user has permissions available to both Finance and Insurance. If you have purchased both Financial Analytics and one of the Oracle Insurance Analytics family of products (Insurance Partner Manager Analytics, Insurance Sales Analytics, Insurance Service Analytics, Insurance Marketing Analytics, Insurance Partner Manager Analytics), you should use the Financial Services Analytics user responsibilities to view all relevant dashboards.

Table 2–12 shows the additional responsibilities, and responsibility-specific security in Oracle's Financial Analytics family of products (Finance Sales Analytics, Finance Service Analytics, Finance Marketing Analytics, Finance Institutional Analytics, Finance Retail Analytics), the Oracle Insurance Analytics family of products (Insurance Partner Manager Analytics, Insurance Sales Analytics, Insurance Service Analytics, Insurance Marketing Analytics, Insurance Partner Manager Analytics), and the Oracle Healthcare Analytics family of products (Healthcare Sales Analytics, Healthcare Service Analytics, Healthcare Marketing Analytics, Healthcare Partner Manager Analytics).

If you are also deploying Usage Accelerator, Financial Services-specific Usage Accelerator responsibilities are shown in Table 2–9.

Table 2–12 Financial Services Responsibility Required to View FS Dashboards

FS Responsibilities	Dashboards
Financial Analytics User	Credit
	Credit Card
	Private Banking
	Consumer Banking
	Corporate and Commercial Banking
	Investment Holdings
	Separate Account Management
	Wealth Management
	Institutional Sales

Table 2–12 (Cont.) Financial Services Responsibility Required to View FS Dashboards

FS Responsibilities	Dashboards
	Investment Banking
	Finance Marketing
	Finance Executive
User in one of the Oracle Insurance Analytics family of products (Oracle Insurance Partner Manager Analytics, Oracle Insurance Sales Analytics, Oracle Insurance Service Analytics, Oracle Insurance Marketing Analytics, Oracle Insurance Partner Manager Analytics)	Policy Sales
	Policy Service
	Insurance Marketing
	Insurance Executive
	Insurance Claims
	Health Plan Sales
	Health Plan Service
	Health Plan Marketing
	Health Plan Executive
	Insurance Agents / Partners

2.6.4.4 Pharma Sales Analytics and Pharma Marketing Analytics Security Settings

Data-level security in Pharma Sales Analytics and Pharma Marketing Analytics is based on the Siebel position ID for all Pharma Analytics responsibilities except PH Executive Analytics. The Siebel position ID is always resolved through the fact table.

Data visibility is unconstrained for administrative roles. For other roles, data visibility is controlled by the position ID. The Oracle Business Analytics Warehouse uses the table W_POSITION_DH for user position-based security control. A user sees only the data that is available to that user's positions. This security model is enforced for all queries, with the exception of queries that deal exclusively with dimension data, such as:

- Time period
- Product
- Invitee status

Table 2–13 shows Pharma Analytics responsibilities and functions.

Table 2–13 Pharma Analytics Responsibilities and Functions

Responsibility	Use
LS Administrator	Administrator privileges to all options on Pharma Analytics.
PH Call Activity Analytics Admin	Administrator privileges to Call Activity Analytics option.

Table 2–13 (Cont.) Pharma Analytics Responsibilities and Functions

Responsibility	Use
PH EMEA Call Activity Analytics User	Enables brick-based metrics to be used in the Presentation Services for Pharma subject areas. Note that in the 7.7 Analytics Release, all report columns use position-based hierarchies, where in earlier releases, report columns used alignment-based sales hierarchies. All brick-based alignment pages have been removed from the reports. Therefore, if you want to use brick-based position hierarchies, you must reconfigure the reports to maintain the alternate hierarchy.
PH EMEA Executive Analytics User	Enables brick-based metrics to be used in the Presentation Services for Pharma subject areas. Note that in the 7.7 Analytics Release, all report columns use position-based hierarchies, where in earlier releases, report columns used alignment-based sales hierarchies. All brick-based alignment pages have been removed from the reports. Therefore, if you want to use brick-based position hierarchies, you must reconfigure the reports to maintain the alternate hierarchy.
PH EMEA Marketing Analytics User	Enables brick-based metrics to be used in the Presentation Services for Pharma subject areas. Note that in the 7.7 Analytics Release, all report columns use position-based hierarchies, where in earlier releases, report columns used alignment-based sales hierarchies. All brick-based alignment pages have been removed from the reports. Therefore, if you want to use brick-based position hierarchies, you must reconfigure the reports to maintain the alternate hierarchy.
PH EMEA Sales Analytics User	Enables brick-based metrics to be used in the Presentation Services for Pharma subject areas. Note that in the 7.7 Analytics Release, all report columns use position-based hierarchies, where in earlier releases, report columns used alignment-based sales hierarchies. All brick-based alignment pages have been removed from the reports. Therefore, if you want to use brick-based position hierarchies, you must reconfigure the reports to maintain the alternate hierarchy.
PH Executive Analytics Admin	Unrestricted access to all Pharma Analytics options with ZIP territories.
PH Marketing Analytics Administrator	Administrator privileges to Pharma ROI, Call Activity Profit & Loss Report, Pharma Promotional Effectiveness Subject Area, and Medical Education Effectiveness Subject Area.
PH Medical Education Analytics Admin	Administrator privileges to Medical Education Analytics option.
PH Medical Education Analytics User	Enables access to Medical Education Analytics option.
PH Disconnected Analytics Admin	Administrator privileges to the PH Disconnected Manager Analytics User and Sales Rep Analytics Dashboards.
PH Disconnected Analytics User	Enables the Pharma Disconnected Analytics Home Page. Allows access to Sales Rep Dashboard as part of the Sales Rep Analytics option.
PH Disconnected Manager Analytics Admin	Administrator privilege to the PH Disconnected Manager Analytics User and District Manager Analytics Dashboards.
PH Disconnected Manager Analytics User	Enables the Pharma Disconnected Analytics Home Page. Allows access to the District Manager Dashboard as part of the Sales Rep Analytics option.

Table 2–13 (Cont.) Pharma Analytics Responsibilities and Functions

Responsibility	Use
PH Sales Analytics Administrator	Administrator privileges to Rx Sales Analytics option.
PH US Call Activity Analytics User	Enables access to Call Activity Analytics Option for ZIP territory alignments.
PH US Executive Analytics User	Unrestricted access to all Pharma Disconnected Analytics options with ZIP-based territories.

2.6.5 Primary Owner-Based Security

Primary owner-based security is supported through the "Primary Owner-Based Security" security group. This type of security mechanism allows records to be visible only to their primary owner. By default, this type of security supports a few dimensions in the Core business model, but other tables can be added if they have a primary owner's source Integration ID column.

The security filter in this security group is defined as:

"Core"."Dim - Activity"."VIS_PR_OWN_ID" = VALUEOF(NQ_SESSION."PR_OWNER_ID")

The session variable PR_OWNER_ID is a single value variable, populated by the Primary Owner ID initialization block. This initialization block runs the following SQL, for the Siebel OLTP data source, to populate the variable:

```
select PAR_ROW_ID

from VALUEOF(TBO).S_USER

where LOGIN = ':USER'
```

2.6.6 Business Unit-Based Security

Business unit-based security is supported through the "Primary Org-Based Security" security group. By default, only a few dimensions in the Core, Workforce Analytics and Forecasting business models support this data security type. Other fact and dimension tables can be added to this security group if they have the column VIS_PR_BU_ID column populated.

The security filter in this security group is defined as:

```
"Core"."Dim - Order"."VIS_PR_BU_ID" = VALUEOF(NQ_
SESSION."ORGANIZATION")
```

The session variable ORGANIZATION is a Row-wise variable, initialized using the Initialization block: Orgs for Org-Based Security. This Init Block runs the following SQL for the Siebel OLTP data source, to populate the ORGANIZATION variable:

```
select distinct 'ORGANIZATION', PRR.SUB_PARTY_ID

from VALUEOF(TBO).S_POSTN P, VALUEOF(TBO).S_USER U,
VALUEOF(TBO).S_PARTY_PER PP,    VALUEOF(TBO).S_PARTY_RPT_REL PRR

where U.ROW_ID=PP.PERSON_ID and P.ROW_ID=PP.PARTY_ID and
PRR.PARTY_ID = P.BU_ID and PRR.PARTY_TYPE_CD = 'Organization'
and U.LOGIN = ':USER'
```

Index