

ellucian™

# Banner Operational Data Store Handbook

Release 8.4  
August 2012



Banner®, Colleague®, PowerCAMPUS®, Luminis® and Datatel® are trademarks of Ellucian or its affiliates and are registered in the U.S. and other countries. Ellucian, Advance, DegreeWorks, fsaATLAS, Course Signals, SmartCall, Recruiter, MOX, ILP, and WCMS are trademarks of Ellucian or its affiliates. Other names may be trademarks of their respective owners.

©2004-2012 Ellucian. All rights reserved. The unauthorized possession, use, reproduction, distribution, display or disclosure of this material or the information contained herein is prohibited.

Contains confidential and proprietary information of Ellucian and its subsidiaries. Use of these materials is limited to Ellucian licensees, and is subject to the terms and conditions of one or more written license agreements between Ellucian and the licensee in question.

In preparing and providing this publication, Ellucian is not rendering legal, accounting, or other similar professional services. Ellucian makes no claims that an institution's use of this publication or the software for which it is provided will guarantee compliance with applicable federal or state laws, rules, or regulations. Each organization should seek legal, accounting and other similar professional services from competent providers of the organization's own choosing.

Prepared by: Ellucian  
4375 Fair Lakes Court  
Fairfax, Virginia 22033  
United States of America

## Revision History

<b>Publication Date</b>	<b>Summary</b>
August 2012	New version that supports Banner Operational Data Store 8.4 software.

## Contents

<b>Chapter 1</b>	<b>Architecture</b> .....	<b>1-1</b>
	<b>BPRA product architecture</b> . . . . .	<b>1-1</b>
	Source system database. . . . .	1-2
	Target database. . . . .	1-3
	Banner Operational Data Store . . . . .	1-3
	Banner Enterprise Data Warehouse . . . . .	1-4
	Performance Management applications . . . . .	1-4
	Advancement Performance . . . . .	1-5
	Banner Recruiting and Admissions Performance . . . . .	1-5
	Banner Student Retention Performance . . . . .	1-6
	<b>Data replication</b> . . . . .	<b>1-6</b>
	Schemas and users. . . . .	1-7
	Banner ODS schemas . . . . .	1-8
	Banner EDW schemas . . . . .	1-9
	Banner RAP schemas . . . . .	1-10
	Banner SRP schemas . . . . .	1-10
	<b>Oracle Streams framework</b> . . . . .	<b>1-11</b>
	DDL Handler. . . . .	1-13
	<b>Oracle Materialized Views framework</b> . . . . .	<b>1-13</b>
	Database links. . . . .	1-15
	Public database link . . . . .	1-15
	Private database link . . . . .	1-15
	Source to target data flow . . . . .	1-16
	Staging infrastructure. . . . .	1-18
	Materialized views staging objects . . . . .	1-19
	P_STAGE_MVIEW procedure. . . . .	1-19
	P_UNSTAGE_MVIEW procedure. . . . .	1-22
	makeMVs.sql script . . . . .	1-24
	Refresh Materialized Views . . . . .	1-24

<b>Extract, Transform, and Load process (ETL)</b> . . . . .	1-26
ETL components . . . . .	1-27
Stage tables . . . . .	1-27
Database triggers . . . . .	1-27
Trigger packages . . . . .	1-28
Change tables . . . . .	1-28
Change table triggers . . . . .	1-30
Composite views and functions or packages . . . . .	1-30
OWB mappings . . . . .	1-31
Composite and slotted tables . . . . .	1-32
Reporting views . . . . .	1-33
Run ETL load processes . . . . .	1-33
Run ETL Load or Refresh jobs in parallel . . . . .	1-34
Run ETL Refresh jobs in parallel in Materialized Views framework . . . . .	1-34
Incremental refresh process . . . . .	1-34
<b>Multi-Entity Processing</b> . . . . .	1-35
<b>Administrative User Interface</b> . . . . .	1-35
<b>Banner ODS data model</b> . . . . .	1-36
<b>Multiple source databases</b> . . . . .	1-36
Source Alias . . . . .	1-36
Add a source database . . . . .	1-37
<b>Validation table data and incremental refresh</b> . . . . .	1-38
<b>Table indexes</b> . . . . .	1-39
<b>Product-specific information</b> . . . . .	1-40
Banner Common . . . . .	1-40
Banner Finance . . . . .	1-44
Banner Student . . . . .	1-45
Composite views and meta data . . . . .	1-57
<b>Naming conventions</b> . . . . .	1-58
Banner ODS standards (ODSMGR schema) . . . . .	1-58
Front-end views: reporting style . . . . .	1-58
Front-end views: Object:Access style . . . . .	1-59
Front-end composite tables . . . . .	1-59
Indexes . . . . .	1-60
Administrative standards (IA_ADMIN schema) . . . . .	1-60
Administrative tables . . . . .	1-60

Administrative packages . . . . .	1-61
Meta data tables and views . . . . .	1-61
Sequences . . . . .	1-62

**Chapter 2 Administrative User Interface . . . . . 2-1**

<b>Set up Users and PINS . . . . .</b>	<b>2-3</b>
Create Users and PINS. . . . .	2-3
Update Existing Users . . . . .	2-4
Update User Roles . . . . .	2-4
User Roles . . . . .	2-5
<b>Data Display Rules . . . . .</b>	<b>2-6</b>
Positional display rules. . . . .	2-6
Hierarchical display rules . . . . .	2-6
Display Rule Information in Published Meta Data. . . . .	2-9
Display Rule Cross-Reference Chart. . . . .	2-9
Set up a Display Rule . . . . .	2-12
Update Display Rules . . . . .	2-15
Duplicate Display Rules . . . . .	2-16
Reload using a Single Extract Transform and Load (ETL) Slot Process . . . . .	2-16
<b>Set up Fine-Grained Access Security . . . . .</b>	<b>2-17</b>
Set up and Maintain Organizational Areas. . . . .	2-19
Create a Banner ODS Organizational Area . . . . .	2-20
Update a Banner ODS Organizational Area . . . . .	2-20
Delete a Banner ODS Organizational Area . . . . .	2-20
Banner User ID Translations. . . . .	2-21
Create Banner User ID Translations . . . . .	2-22
Update Banner User ID Translations . . . . .	2-23
Delete Banner User ID Translations . . . . .	2-24
Set up Business Profiles . . . . .	2-24
Create a Business Profile . . . . .	2-25
Associate Business Profiles with a User . . . . .	2-25
Associate Users with a Business Profile . . . . .	2-26
View, Update or Delete a Business Profile. . . . .	2-26
Set up and Maintain Security Rules . . . . .	2-27
Determine Data Security Requirements . . . . .	2-28
Set up a Security Rule . . . . .	2-28
Update or Delete a Security Rule . . . . .	2-35
Assign Security Rules . . . . .	2-35

Security Predicates . . . . .	2-47
Policy Management. . . . .	2-48
Policies for all Tables . . . . .	2-48
Policies for a Single Table . . . . .	2-49
<b>Administrative User Interface Data Access. . . . .</b>	<b>2-52</b>
<b>Set up and Synchronize Data . . . . .</b>	<b>2-53</b>
<b>Set up Parameters . . . . .</b>	<b>2-54</b>
Cascade filter . . . . .	2-55
Create a parameter. . . . .	2-55
Update or Delete a Parameter. . . . .	2-56
System Parameters. . . . .	2-57
Parameters . . . . .	2-58
<b>Schedule a Process . . . . .</b>	<b>2-60</b>
Process descriptions and details . . . . .	2-61
Details . . . . .	2-61
Edit . . . . .	2-62
Banner ODS Processes . . . . .	2-62
Schedule Banner ODS Mappings. . . . .	2-62
Banner ODS Utilities . . . . .	2-62
Schedule a Single Process . . . . .	2-63
Schedule Multiple Processes . . . . .	2-63
View and Remove a Scheduled Process. . . . .	2-66
Configure an Account and Stop a Running Job/Process. . . . .	2-67
Configure a User Account to Kill a Job/Process. . . . .	2-67
Kill a Running Job/Process . . . . .	2-68
Run a Process from Outside the Administrative UI . . . . .	2-69
Set up Customized Scheduled Processes . . . . .	2-71
Schedule a Process Parameters . . . . .	2-75
INSTALLED PROCESS Parameter. . . . .	2-77
PROCESS INFO Parameter. . . . .	2-78
SUBPROCESS Parameter . . . . .	2-78
SUBPROCESS INFO Parameter . . . . .	2-80
JOB Parameter . . . . .	2-82
ETL MAP PACKAGE Parameter . . . . .	2-84
ETL MAP PACKAGE LOAD PURGE Parameter . . . . .	2-86
ETL MAP PACKAGE LOGIC Parameter . . . . .	2-88
ETL MAP PACKAGE RECONCILE LOGIC Parameter. . . . .	2-88
ETL SLOT PACKAGE Parameter. . . . .	2-89

ETL CONTROL GROUP Parameter . . . . .	2-90
EVENT parameter . . . . .	2-92
PARAMETER Parameter. . . . .	2-92
<b>Banner ODS Utilities . . . . .</b>	<b>2-94</b>
Report Banner ODS Source Change Table Counts. . . . .	2-95
Change Tables and Control Reports . . . . .	2-95
Synchronize Comments for Multiple Reporting Views . . . . .	2-96
Synchronize Comments for a Single Reporting View. . . . .	2-96
Banner ODS Checks and Balances . . . . .	2-97
Reconcile a Single Table. . . . .	2-98
Transfer Banner Fine-Grained Access . . . . .	2-99
Cleanup Reconcile Tables . . . . .	2-101
Reconcile a Group of Tables. . . . .	2-102
Create a Reconcile Group . . . . .	2-102
Run a Reconcile Group job . . . . .	2-105
Reconcile temporary table names . . . . .	2-106
Reconcile Multiple Tables . . . . .	2-106
<b>Materialized Views. . . . .</b>	<b>2-108</b>
Materialize a View . . . . .	2-108
Materialized View Considerations. . . . .	2-109
Materialized Views and Patch Releases . . . . .	2-109
Create an Index of Materialized View. . . . .	2-109
Create a Materialized View . . . . .	2-110
Maintain a Materialized View . . . . .	2-111
Delete a Materialized View. . . . .	2-112
Materialized View Status Report . . . . .	2-113
Report Materialized View Status . . . . .	2-113
Materialized View Control Reports . . . . .	2-113
<b>View Control Reports . . . . .</b>	<b>2-114</b>
Error Messages . . . . .	2-115
Banner ODS Checks and Balances Process . . . . .	2-116
Materialized View Control Report Messages . . . . .	2-119
Freeze Process . . . . .	2-120
Publish Meta Data (PUBLISH_META_DATA) . . . . .	2-121
Reconcile (RECONCILE_JOB, RECONCILE_SINGLE_JOB). . . . .	2-121
Oracle Warehouse Builder Runtime Audit Browser Integration . . . . .	2-122
Integration Setup . . . . .	2-122
RAB Authentication . . . . .	2-123

<b>Set up E-mail Notification</b> . . . . .	<b>2-124</b>
<b>Freeze Data Maintenance</b> . . . . .	<b>2-125</b>
Set up Banner ODS Freeze Data Lists . . . . .	2-125
Add a Table/View to a Banner ODS Freeze Data List . . . . .	2-128
Delete, Rename or Duplicate Banner ODS Freeze Data. . . . .	2-129
Freeze a Single Banner ODS Table/View . . . . .	2-130
Freeze Multiple Banner ODS Tables/Views at the Same Time . . . . .	2-131
Create a dynamic Freeze List parameter . . . . .	2-132
Update or Freeze Recurring Banner ODS Data. . . . .	2-135
Update Banner ODS Daily . . . . .	2-135
Update Banner ODS on Specific Dates and Times . . . . .	2-136
<b>Meta Data</b> . . . . .	<b>2-137</b>
Baseline and Local Meta Data. . . . .	2-140
Create Meta Data . . . . .	2-140
Set up Institutional Meta Data Publish Preferences. . . . .	2-140
Meta Data Parameter Set up for Publishing Reports . . . . .	2-141
Configure Publishing Parameters and Create Meta Data Web Directory. . . . .	2-142
Edit Target Meta Data Properties . . . . .	2-145
Add Target Views and Target Columns. . . . .	2-146
Edit Target Views and Target Columns. . . . .	2-147
Synchronize Meta Data Comments with Reporting Views . . . . .	2-147
Delete Local Target Properties . . . . .	2-148
Delete Local Target Columns . . . . .	2-149
Edit Source Meta Data Properties . . . . .	2-150
Add Source Names and Source Columns . . . . .	2-151
Edit Source Names and Source Columns . . . . .	2-152
Delete Local Source Properties . . . . .	2-152
Delete Local Source Columns. . . . .	2-153
Add and Delete Source to Target Meta Data Local Mappings . . . . .	2-154
Import Target and Source Meta Data. . . . .	2-155
CSV Export . . . . .	2-156
<b>Publish Meta Data from the Administrative UI</b> . . . . .	<b>2-157</b>
Publish Meta Data for an Entire Subject Area . . . . .	2-157
Publish Meta Data for One Source or Target . . . . .	2-158

<b>Publish Meta Data Reports.</b> . . . . .	<b>2-158</b>
Publish Meta Data by Scheduling a Process . . . . .	2-159
Publish Meta Data from the Command Line . . . . .	2-159
View Published Meta Data . . . . .	2-160
Reporting View Meta Data . . . . .	2-160
Composite View Meta Data . . . . .	2-160
Metamodel . . . . .	2-161
Banner ODS Meta Data Object Types . . . . .	2-163
Source Meta Data Tables . . . . .	2-163
Target Meta Data Tables . . . . .	2-165
Source and Target Meta Data Tables . . . . .	2-167
Reporting Meta Data Views . . . . .	2-170
<b>Staging and data replication.</b> . . . . .	<b>2-176</b>
Staging options . . . . .	2-176
Materialized Views staging options . . . . .	2-177
Maintain stage tables . . . . .	2-177
Add a non-baseline staging table to the Banner ODS . . . . .	2-177
Remove a non-baseline staging table from the Banner ODS . . . . .	2-178
Add a schema . . . . .	2-179
Remove a schema . . . . .	2-179
Staging Area Status. . . . .	2-179
Streams framework . . . . .	2-180
Materialized Views framework. . . . .	2-180
Run Staging Area Status report . . . . .	2-180
Reconcile and restage stage tables. . . . .	2-183
Run stage table reconcile . . . . .	2-185
Maintain Oracle Streams framework . . . . .	2-186
Create Streams Framework . . . . .	2-187
Remove Streams Framework . . . . .	2-187
Configure Streams Replication for Baseline Tables. . . . .	2-187
Remove a Baseline Staging Table from the Banner ODS . . . . .	2-188
Start or Stop the Streams Capture Process . . . . .	2-188
Start or Stop the Streams Propagation Schedule . . . . .	2-189
Start or Stop the Streams Apply Process. . . . .	2-189
Required Source Archived Logs. . . . .	2-190
Monitor Streams for Apply Errors . . . . .	2-191
Monitor Source Capture Queue for Growth . . . . .	2-192
Monitor the Status of Source Propagation Jobs . . . . .	2-192
Set the Checkpoint Frequency and Retention Time (optional). . . . .	2-193
Avoid NOLOGGING and UNRECOVERABLE Keywords in Source . . . . .	2-194
Use Streams Tags with Batch Processes . . . . .	2-194

Use Data Dictionary Views to Display Streams Information . . . . .	2-194
Maintain Materialized Views framework . . . . .	2-195
Refresh groups . . . . .	2-195
Staging Refresh Collections . . . . .	2-196
Associate Staging Collections with Refresh Jobs . . . . .	2-196
Refresh materialized views . . . . .	2-196
Refresh Staging Collection job control reports. . . . .	2-198
<b>Web Tailor Administration . . . . .</b>	<b>2-200</b>
Functions. . . . .	2-200
Customize a Web Menu or Procedure . . . . .	2-201
Customize a Graphic Element. . . . .	2-201
Customize a Set of Information Text . . . . .	2-201
Customize a Set of Menu Items. . . . .	2-201
Update User Roles . . . . .	2-202
Customize a Web Module . . . . .	2-202
Customize Web Rules . . . . .	2-202
Customize Web Tailor Parameters . . . . .	2-202
Customize a Login Return Location . . . . .	2-203
Customize Web Tailor Overrides . . . . .	2-203
Customize Global User Interface Settings . . . . .	2-203

## **Chapter 3 Banner ODS Business Concepts . . . . . 3-1**

<b>Banner ODS Business Concepts . . . . .</b>	<b>3-2</b>
Diagrams. . . . .	3-5
Accounts Receivable . . . . .	3-6
Receivable Customer. . . . .	3-6
Receivable Revenue . . . . .	3-7
Advancement . . . . .	3-8
Advancement Prospect. . . . .	3-8
Advancement Rating . . . . .	3-9
Annual Giving . . . . .	3-10
Annual Giving Comparison . . . . .	3-11
Campaign Giving History. . . . .	3-12
Campaign Management . . . . .	3-13
Constituent. . . . .	3-14
Constituent Entity . . . . .	3-15
Designation . . . . .	3-16
Designation Giving History. . . . .	3-17
Gift . . . . .	3-18

Organizational Constituent . . . . .	3-19
Pledge . . . . .	3-20
Solicitation Effort . . . . .	3-21
Common . . . . .	3-22
Event . . . . .	3-22
Institution . . . . .	3-23
Organization Entity . . . . .	3-24
Person Demographic . . . . .	3-25
Person Role . . . . .	3-26
Person Supplemental . . . . .	3-27
Relationship . . . . .	3-28
Finance. . . . .	3-29
Account Index Audit . . . . .	3-29
Budget Availability Comparison . . . . .	3-30
Budget Availability Ledger . . . . .	3-31
Budget Detail . . . . .	3-32
Cashier Session Analysis . . . . .	3-33
Encumbrance . . . . .	3-34
Endowment Distribution . . . . .	3-35
Endowment Units . . . . .	3-36
Fixed Asset . . . . .	3-37
General Ledger . . . . .	3-38
Grant and Project . . . . .	3-39
Grant Ledger . . . . .	3-40
Grant-Contract and Proposal . . . . .	3-41
Invoice Payable . . . . .	3-42
Operating Ledger . . . . .	3-43
Operating Ledger Comparison . . . . .	3-44
Purchasing Payable . . . . .	3-45
Transaction History . . . . .	3-46
Financial Aid. . . . .	3-47
Financial Aid Application . . . . .	3-47
Financial Aid Award and Disbursement . . . . .	3-48
Financial Aid Fund . . . . .	3-49
Loan Disbursement . . . . .	3-50
Human Resources . . . . .	3-51
Employee . . . . .	3-51
Employee and Position. . . . .	3-52
Human Resource Application . . . . .	3-53
Human Resource Faculty . . . . .	3-54
Payroll . . . . .	3-55
Position. . . . .	3-56
Student . . . . .	3-57
Active Registration . . . . .	3-57
Admissions Application. . . . .	3-58
Advisor Student List. . . . .	3-59

Course Catalog . . . . .	3-60
Enrollment Management . . . . .	3-61
Enrollment Management Subset . . . . .	3-62
Faculty Assignment . . . . .	3-63
Faculty Subset. . . . .	3-64
Government Reporting . . . . .	3-65
Recruitment Information . . . . .	3-66
Residential Life . . . . .	3-67
Schedule Offering. . . . .	3-68
Student Detail . . . . .	3-69
Student Detail Subset. . . . .	3-70
Travel and Expense . . . . .	3-71
Authorization . . . . .	3-71
Reimbursement . . . . .	3-72

**Chapter 4 Third Party Reporting Tools . . . . . 4-1**

<b>Cognos Business Intelligence . . . . .</b>	<b>4-1</b>
Framework Manager models . . . . .	4-2
Metadata layers . . . . .	4-2
Database view metadata layer . . . . .	4-2
Business view metadata layer . . . . .	4-3
Presentation view metadata layer. . . . .	4-3
Cognos and BPRA Meta Data Integration . . . . .	4-4
View BPRA Meta Data in Cognos. . . . .	4-4
View BPRA Meta Data details. . . . .	4-6
Packages . . . . .	4-8
Banner ODS packages. . . . .	4-9
Lists of Values. . . . .	4-21
Prompts . . . . .	4-21
Filters. . . . .	4-22
Banner ODS Pre-defined Cognos Filters. . . . .	4-22
Cognos Security Integration . . . . .	4-22
Cognos Authorization and fine-grained access . . . . .	4-22
Cognos and BPRA authorization . . . . .	4-23
Luminis authentication (single sign-on). . . . .	4-27
Transaction history tracking process . . . . .	4-44
View and save transaction history . . . . .	4-45
Play back transactions from a log file. . . . .	4-45
Brand Cognos Connection page . . . . .	4-47
Customize the welcome splash screen. . . . .	4-47

<b>Oracle Business Intelligence Discoverer (Banner ODS)</b> . . . . .	<b>4-48</b>
Delivered EULs . . . . .	4-48
Lists of Values . . . . .	4-49
Lists of Values – Item Classes . . . . .	4-49
Conditions . . . . .	4-53
Date Hierarchies . . . . .	4-54

**Chapter 5 Data Models** . . . . . 5-1

<b>Entity Relationship Diagrams (ERD)</b> . . . . .	<b>5-1</b>
ERD Relationship Legend . . . . .	5-2
Identifying Relationships . . . . .	5-2
Optional Non-Identifying Relationship . . . . .	5-4
Special Relationships . . . . .	5-5
Accounts Receivable . . . . .	5-6
Receivable Customer . . . . .	5-6
Receivable Revenue . . . . .	5-7
Advancement . . . . .	5-8
Advancement Prospect . . . . .	5-8
Advancement Rating . . . . .	5-9
Annual Giving . . . . .	5-10
Campaign Management . . . . .	5-11
Campaign Giving History . . . . .	5-12
Constituent . . . . .	5-13
Constituent Entity . . . . .	5-14
Designation . . . . .	5-15
Designation Giving History . . . . .	5-16
Gift . . . . .	5-17
Organizational Constituent . . . . .	5-18
Pledge . . . . .	5-19
Solicitation Effort . . . . .	5-20
Common . . . . .	5-21
Event . . . . .	5-21
Institution . . . . .	5-22
Organization Entity . . . . .	5-23
Person Demographic . . . . .	5-24
Person Role . . . . .	5-25
Person Supplemental . . . . .	5-26
Relationship . . . . .	5-27
Finance . . . . .	5-28
Account Index Audit . . . . .	5-28
Budget Availability Ledger . . . . .	5-29
Budget Detail . . . . .	5-30

Encumbrance . . . . .	5-31
Endowment Distribution . . . . .	5-32
Endowment Units . . . . .	5-33
Fixed Asset . . . . .	5-34
General Ledger . . . . .	5-35
Grant and Project . . . . .	5-36
Grant Ledger . . . . .	5-37
Invoice Payable . . . . .	5-38
Operating Ledger . . . . .	5-39
Purchasing Payable . . . . .	5-40
Transaction History . . . . .	5-41
Financial Aid . . . . .	5-42
Financial Aid Application . . . . .	5-42
Financial Aid Award and Disbursement . . . . .	5-43
Financial Aid Fund . . . . .	5-44
Loan Disbursement . . . . .	5-45
Human Resources . . . . .	5-46
Employee . . . . .	5-46
Human Resource Application . . . . .	5-47
Human Resource Faculty . . . . .	5-48
Payroll . . . . .	5-49
Position . . . . .	5-50
Student . . . . .	5-51
Active Registration . . . . .	5-51
Admissions Application . . . . .	5-52
Advisor Student List . . . . .	5-53
Course Catalog . . . . .	5-54
Enrollment Management . . . . .	5-55
Faculty Assignment . . . . .	5-56
Government Reporting . . . . .	5-57
Recruitment Information . . . . .	5-58
Residential Life . . . . .	5-59
Schedule Offering . . . . .	5-60
Student Detail . . . . .	5-61
Travel and Expense . . . . .	5-62
Authorization . . . . .	5-62
Reimbursement . . . . .	5-63
Travel and Expense Reporting Views . . . . .	5-64
<b>Reporting Views . . . . .</b>	<b>5-67</b>
<b>List of Value Views . . . . .</b>	<b>5-67</b>

<b>Chapter 6</b>	<b>Self-Service Reporting</b>	<b>6-1</b>
	<b>Navigation Quick Reference</b>	<b>6-2</b>
	<b>Search Criteria Page</b>	<b>6-3</b>
	Recommended and Required Search Criteria	6-4
	Dependant Search Criteria Filters	6-4
	List of Values Search Criteria Filters	6-5
	Show SQL	6-5
	<b>List Page</b>	<b>6-5</b>
	Export	6-6
	Email	6-6
	Individual	6-6
	Entire List	6-6
	Sort	6-6
	Records Per Page (Display Setting)	6-7
	Access Detail Reports	6-7
	<b>Detail Reports Page</b>	<b>6-7</b>
	Detail Reports Drop-down List	6-8
	Sort	6-8
	Export	6-8
	<b>Query Search Criteria</b>	<b>6-8</b>
	<b>Search Rules</b>	<b>6-9</b>
	Create and Save a Search Rule	6-9
	Load a Search Rule	6-10
	Update a Search Rule	6-10
	Save as Another Search Rule (Save As)	6-11
	Rename a Search Rule	6-12
	Delete a Search Rule	6-13
	<b>Banner ODS Populations</b>	<b>6-13</b>
	Create Banner ODS Populations	6-14
	Refresh (Update) Banner ODS Populations	6-15
	Delete a Banner ODS Population	6-16
	Use Populations with Banner ODS	6-17

<b>Report Templates</b> . . . . .	<b>6-18</b>
Accounts Receivable Report Templates . . . . .	6-19
Receivable Customer. . . . .	6-19
Advancement Report Templates . . . . .	6-21
Advancement Person. . . . .	6-21
Finance Report Templates . . . . .	6-22
General Ledger . . . . .	6-22
Operating Ledger . . . . .	6-25
Financial Aid Report Templates . . . . .	6-28
Financial Aid Awards . . . . .	6-28
Human Resources Report Templates . . . . .	6-30
Employee . . . . .	6-30
Student Report Templates . . . . .	6-32
Advisor's Students . . . . .	6-32
Admissions Application. . . . .	6-34
Enrolled Students . . . . .	6-36
<b>Self-Service Reporting Configuration Parameters</b> . . . . .	<b>6-38</b>
<b>Security</b> . . . . .	<b>6-38</b>
Oracle Account Security . . . . .	6-39
Authentication . . . . .	6-39
Authorization. . . . .	6-40
APEX Built-in Authentication. . . . .	6-42
Other Security Options. . . . .	6-45
<b>Glossary</b> . . . . .	<b>G-1</b>
<b>Index</b> . . . . .	<b>I-1</b>

# 1 Architecture

---



The Banner Operational Data Store (Banner ODS) and the Banner Enterprise Data Warehouse (Banner EDW) are the data warehouse components of the Banner Performance Reporting and Analytics Business Intelligence platform. The following sections describe the architecture of this platform and the roles and integration of ODS and EDW with the other components.

## BPRA product architecture

---



The complete suite of BPRA products provides comprehensive content across areas such as Student, Financial Aid, Finance, Accounts Receivables, Human Resources and Advancement giving your institution the ability to take full advantage of the data stored in your source system by turning it into applied knowledge in the warehouse. You can use the BPRA products together to help you make informed decisions, to guide strategic institutional planning and forecasting based on analysis of historical trends, and to enhance institutional performance.

The BPRA solution set includes the following products:

- Banner Operational Data Store (Banner ODS)
- Banner Enterprise Data Warehouse (Banner EDW)
- Advancement Performance (AP)
- Banner Recruiting and Admissions Performance (Banner RAP)
- Banner Student Retention Performance (Banner SRP)

### Note

Your institution may license some or all of the BPRA products. If you do license multiple BPRA products, it is important that you understand the relationship among all of the products as you use them. ■



The following figure illustrates the components of the BPR suite of products.

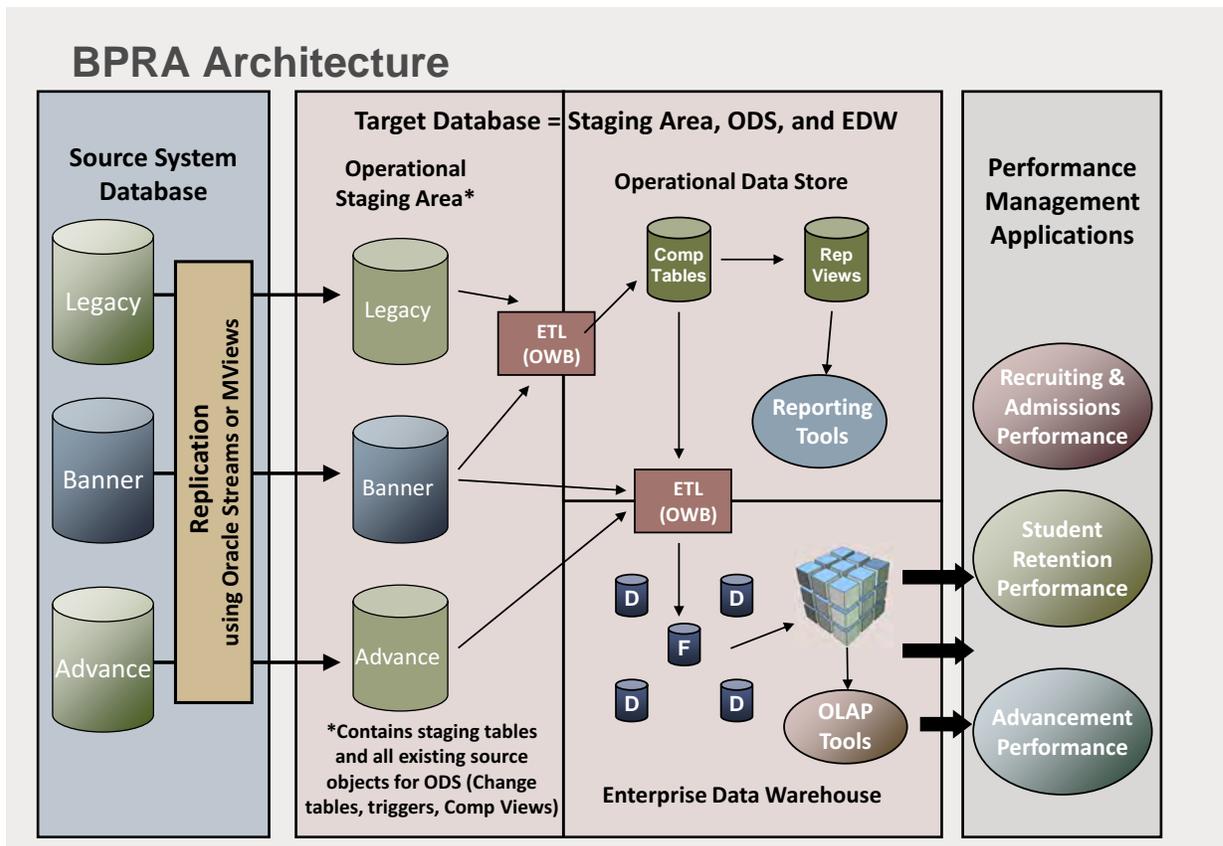


Figure 1: BPR product architecture

## Source system database

The starting point for any performance or reporting analysis solution is your source system data. The information stored in the source transactional database is ultimately the information that you want to analyze.

The BPR products are specifically designed to accept information from the Banner and Advance products. However, the BPR products use an open design and can accept information from other sources as well. References to the “source” database refer to whichever source product you use, typically Banner or Advance.

## Target database

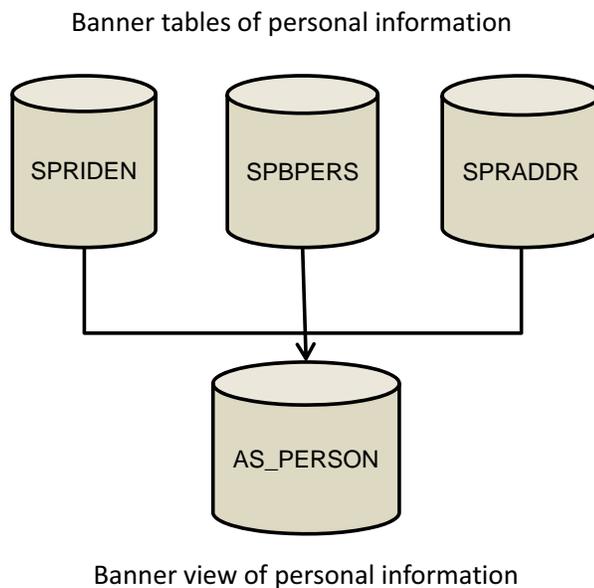
The “target” database refers to the database where you load information from the source database. Depending on the way you license your BPRAs products, this may be the Banner ODS or Banner EDW database or both.

## Banner Operational Data Store

The Banner ODS enables you to extract information from your source administrative systems and reorganize the information into a simplified set of tables in the Banner ODS database. End users can then create and deploy operational and ad hoc reports.

Banner ODS provides an extensive and flexible data store and business-organized reporting views with fewer columns and improved performance. You can use these views alone, or in combination with other views. SunGard Higher Education also uses the supported third party reporting metadata layers to deliver an enterprise business area with many prejoined conditions to enhance operational and ad hoc reporting.

In the Banner ODS information from complex and normalized source tables are grouped into more simplified, denormalized tables that are grouped by concept. The following picture illustrates how data from Banner tables of person-related information are combined into one Banner ODS table named AS\_PERSON.



**Figure 2: Banner to Banner ODS table consolidation**

In Banner, to properly access the data, you need to understand the rules used to store the data in each table and the rules used to properly join the tables. Using the Banner ODS, you can access replicated Banner data in the ODS without the need to understand the complexities of the data structure because you can retrieve the data from the view.

## Banner Enterprise Data Warehouse

The Banner EDW is a multi-dimensional database that gives you a complete picture of your institution's current and past business conditions. The Banner EDW offers comprehensive reporting and analysis capabilities by providing the following data objects:

- Operational/Aggregate stars that you can refresh with current data on a daily basis at both a summary and detail level
- Snapshot stars that offer a historical snapshots of the data at institution-specific points-in-time at a summary level

This combination of current and historical data allows you to do comparative reporting and analysis. Banner EDW includes prebuilt metadata integration with the IBM Cognos BI software to enable fast deployment of reports and analytics.

## Performance Management applications

The Performance Management products are a subset of BPRA products that you can license and use in conjunction with the Banner ODS and/or Banner EDW to monitor and manage your institutions business objectives and analyze outcomes. The following picture illustrates the Performance Management products and high-level features.

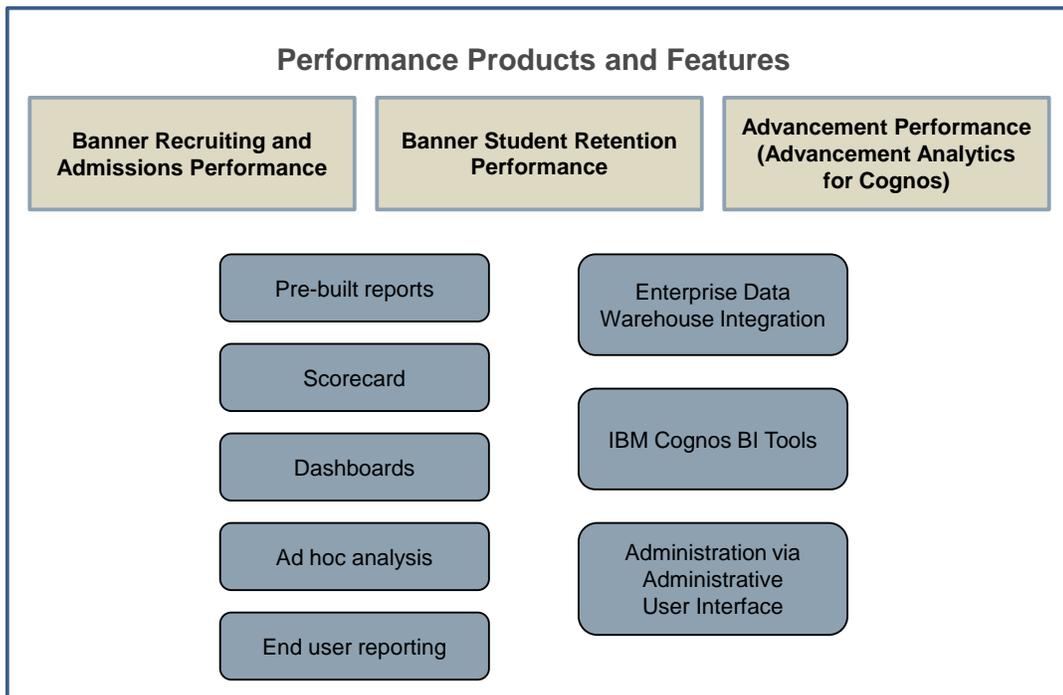


Figure 3: Performance Management products and features

Each Performance Management product includes the following types of objects built using the IBM Cognos Business Intelligence application:

- Business Concept Packages - reporting metadata layer
- Cubes - predefined reporting structures for quick analysis of summary measures by many attributes
- Reports - display trends of outcomes, summaries of current outcomes, and detailed information about students, applicants, recruits, or constituents (depending on the product)
- Dashboards - display several graphical performance charts for a specific business area on a single screen that you can review at a glance
- Scorecards - display institutional goals and objectives including Key Performance Indicators (KPIs) that monitor progress toward your goals and objectives and a set of strategic initiatives that are needed to produce desired outcomes

The data for these objects is stored in the Banner EDW. These objects are intended to illustrate the kind of analysis you can perform on the warehouse data. You can use the reports, dashboards, and scorecards as delivered or you can modify them to reflect the specific information you need to analyze and monitor your institution's progress.

## **Advancement Performance**

The Advancement Performance solution provides Advancement organizations (Banner Advancement and Advance users) with innovative ways to manage prospects and campaigns, drive fundraising, engage alumni and other constituents, and more. The Advancement Performance solution is comprised of the following products:

- Advancement Analytics for Cognos
- Enterprise Data Warehouse (Advancement data)

The Advancement Analytics for Cognos product provides the performance application content and tools and uses the Banner EDW multi-dimensional database that gives you a complete picture of your institution's current and past business conditions. This permits your institution to report both current and historical data for summary, trend and detail reporting and analysis

## **Banner Recruiting and Admissions Performance**

Banner Recruiting and Admissions Performance is the reporting analytics and performance portion of the Banner Relationship Management Suite that lets you easily access recruitment, admissions, and selected financial aid information and use it to create reports.

Banner Recruiting and Admissions Performance uses the Banner EDW multi-dimensional database that gives you a complete picture of your institution's current and past business

conditions. This permits your institution to report both current and historical data for summary, trend and detail reporting and analysis.

## **Banner Student Retention Performance**

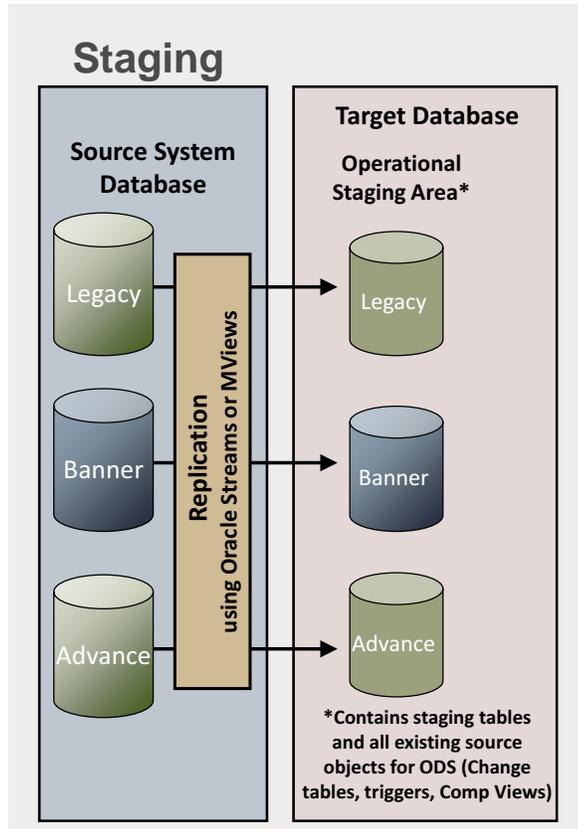
You can use Banner Student Retention Performance to monitor student retention, student success (performance and progress) and student engagement to satisfy institution goals and objectives; extend and modify performance monitoring capabilities; and create operational reports and ad hoc queries that meet the specific needs of your institution.

Banner Student Retention Performance uses the Banner EDW multi-dimensional database that gives you a complete picture of your institution's current and past business conditions. This permits your institution to report both current and historical data for summary, trend and detail reporting and analysis.

## **Data replication**

---

The replication of data between the source and target databases is key to the usefulness of the warehouse solution and in turn the reports built off the target database. Data replication is referred to as the “staging” process, which simply means to copy tables in the source database into the operational staging area of the target database as illustrated by the following picture.



**Figure 4: Data replication - staging data**

You have two options for staging data in the target database. You can use Oracle Streams or Oracle Materialized Views as the framework for staging data. During the initial installation or upgrade process, your institution chose which staging approach to implement.

Refer to one of the following sections to learn more about managing the staging environment in the framework used by your institution.

- [“Oracle Streams framework” on page 1-11](#)
- [“Oracle Materialized Views framework” on page 1-13](#)

You can also refer to the *Oracle Streams Operations Supplement* or the *Materialized Views Operations Supplement* for additional information about setting up, configuring, and administering one of the frameworks.

## Schemas and users

Each schema in the source system needs to have an identical schema in the target database to successfully synchronize data between the two. The following sections list the components owned by the various schemas for each of the BPRA products.

## Banner ODS schemas

The following schemas exist in the Banner ODS.

Schema	Owns
ODSSRC	<ul style="list-style-type: none"><li>• All composite views</li><li>• Database packages that contain business logic used in the composite views and trigger logic.</li></ul> <p><b>Note:</b> These are all the ETL objects that used to live in the ODSMGR schema of Banner prior to ODS 8.2.</p>
ODSMGR	<ul style="list-style-type: none"><li>• All composite tables, and the reporting views that sit on top of the composite tables</li><li>• Database packages that contain business logic used in the reporting views</li><li>• OWB mapping packages</li></ul>
IA_ADMIN	<ul style="list-style-type: none"><li>• Metadata tables</li><li>• Any objects used or associated with the Administrative User Interface like the parameter table, the data display rules table, and the security tables.</li></ul>
ODSSTG	<p>Depending on the architecture framework you use, the ODSSTG schema owns one of the following:</p> <ul style="list-style-type: none"><li>• Oracle Streams objects</li></ul> <p><i>Or</i></p> <ul style="list-style-type: none"><li>• Materialized Views objects (packages)</li></ul> <p>This user is created in Banner and the ODS.</p>
ODSEUL	<ul style="list-style-type: none"><li>• Discoverer End User Layer tables</li></ul> <p><b>Note:</b> ODSEUL is the default name of this schema; however, you can rename the schema.</p>
ODSLOV	<ul style="list-style-type: none"><li>• List Of Value views that are created from the MGT_VALIDATION table</li></ul> <p>These views are used as part of the Cognos, Discoverer and Self-Service Reporting (SSR) tool metadata layers to build List of Values that can be used for reporting.</p>
SSRMGR	<ul style="list-style-type: none"><li>• All objects related to building the SSR application</li></ul>

If you source the target database from a Banner source database, the following additional schemas may exist depending on which Banner products you license and stage in your target database.

- ALUMNI
- FAISMGR
- FIMSMGR
- FTAEMGR
- GENERAL
- PAYROLL
- POSNCTL
- SATURN
- TAISMGR

These schemas would house the staging tables (materialized views), change tables, and triggers.

### **ODSSTG password management**

You must pay special attention when changing the password to the ODSSTG database user account on either the source or target database because the ODSSTG account in the Banner ODS has an Oracle DB Link back to the source account. If you change any user account passwords for schemas on the source database, for example, in Banner ODSSTG, SATURN, GENERAL, you must also update the DB link in the Banner ODS database to match the password for the related Banner account schema.

Refer to FAQ 1-AXRVD8, which describes the process and steps to alter passwords for any of the Banner ODS related database accounts.

### **Banner EDW schemas**

The following schemas exist in the Banner EDW.

<b>Schema</b>	<b>Owns</b>
EDWMGR	<ul style="list-style-type: none"><li>• All fact, dimension, aggregate tables</li><li>• OWB mappings to load those tables</li></ul>
EDWSTG	<ul style="list-style-type: none"><li>• EDW stage/input &amp; clean tables</li><li>• OWB mappings to load those tables</li><li>• Table function packages to load input tables</li></ul>

## Banner RAP schemas

The RAP product includes the schemas listed for Banner EDW as well as the following additional schema.

Schema	Owns
RELATEMGR	<ul style="list-style-type: none"><li>• Staging tables that house the Relationship Management information</li><li>• Triggers and change tables associated with the staging tables</li></ul>

---

## Banner SRP schemas

The SRP product includes the schemas listed for Banner EDW as well as the following additional schema.

Schema	Owns
RELATEMGR	<ul style="list-style-type: none"><li>• Staging tables that house the Relationship Management information</li><li>• Triggers and change tables associated with the staging tables</li></ul>

---

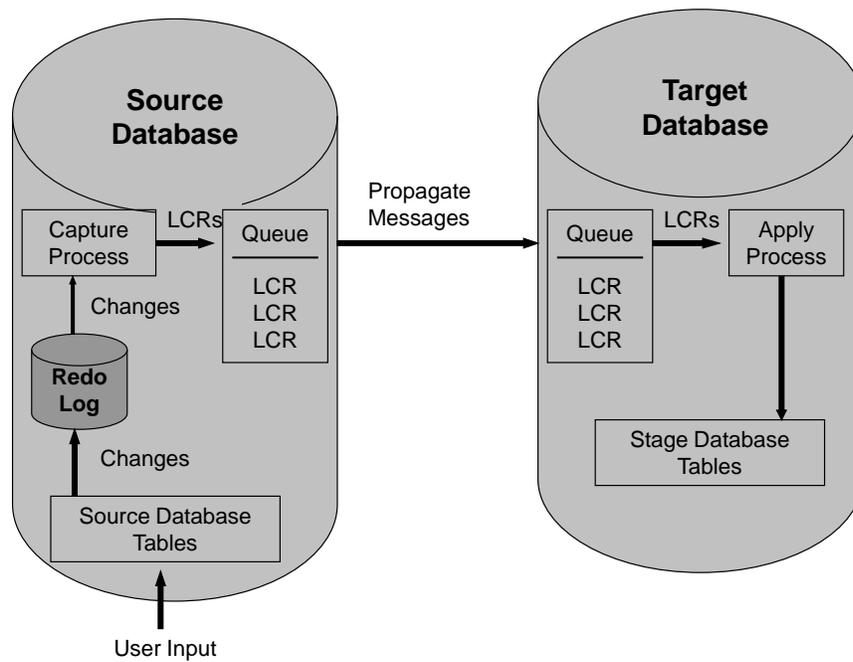
# Oracle Streams framework

 **Note**

Refer to the *BPRA Oracle Streams Operations Supplement* for more information about maintaining the Oracle Streams framework. ■

The Oracle Streams framework uses Oracle Streams functionality to replicate data from the source to target database. Any insert, update, or delete actions performed on the source database tables are also performed on the tables in the staging area of the target database to synchronize the source and destination databases. The existing table triggers, change tables, and packages to create change records for the target database reside in that database.

The following picture shows the components used to replicate data between the source database and the target database.



**Figure 5: Oracle Streams data replication between source and target databases**

The replication process uses the following Oracle Streams components.

Oracle Streams Component	What it does...
Capture Process	The Streams capture process mines changes from the redo log on the Banner database whenever possible and from the archived logs if it falls behind the generated redo logs. Changes in the redo log that match specified rules are converted into messages called logical change records (LCRs), which are placed in a queue associated with the capture process.
Propagation Schedule	The Streams propagation schedule moves the messages from the source database queue to a queue on the Banner ODS. Each message remains in the source queue until the destination confirms that it has received it. This confirmation guarantees messages are never lost during the propagation stage.
Apply Process	The Streams apply process in the Banner ODS removes the messages from the queue and applies them directly to the destination tables in the Banner ODS. Any errors encountered while applying the messages are placed into an error queue. The messages in the error queue can be reprocessed once any issues have been resolved.

During the installation or upgrade process, the system creates all the components required to set up the Oracle Streams framework. This includes setting up the Streams queues, propagation schedule, capture and apply processes, and populating the staging tables in the warehouse with data from the source Banner tables.

Refer to the following Oracle documents for more information about maintaining and troubleshooting the Oracle Streams environment.

- “Oracle Streams Concepts and Administration Guide”
- “Oracle Streams Replication Administrator's Guide”
- “Streams Complete Reference FAQ” (MetaLink Document ID: 752871.1)

## DDL Handler

A DDL handler is assigned to each Streams apply handler to replicate DDL statements from the source database to the staged tables in the target database. DDL statements run against a source table are replicated to warehouse unless a statement includes table dependencies. This means DDL statements executed on the source database that create, alter, or drop columns, non-foreign key constraints, and indexes are replicated to the target database; however, using the DDL handler allows the replication process to ignore the same types of statements for table triggers and foreign keys. These changes will not be replicated in the target database.

The user who executes a DDL statement on the source database must also exist in the target database for the change to be replicated successfully in the target.

When a DDL command is executed on the source database and the object schema is not specified, then the DDL command will only be applied successfully on the target database if the source user who executed the command also exists in the target database.

### Warning

Be aware that source objects with system-generated names will have different names in the target database. This means that DDL statements involving these objects will not be replicated successfully, and will result in DBA\_APPLY\_ERROR records being created. The majority of the system-generated names in the source system that may be affected are NOT NULL constraints on table columns. ■

## Oracle Materialized Views framework

---

### Note

Refer to the *BPRA Materialized Views Operations Supplement* for more information about maintaining the Oracle Materialized Views framework. ■

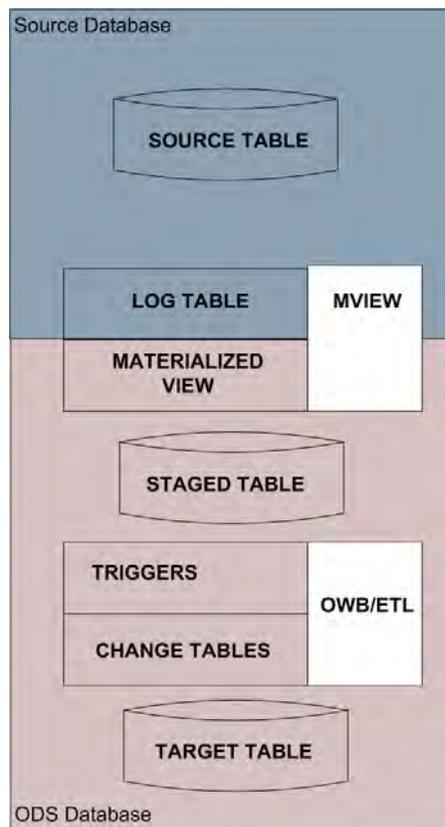
The Materialized Views architecture uses Oracle materialized views (mviews) to initially stage data in the Banner ODS database and keep that data synchronized with the source database. In the Materialized Views architecture the staging tables in Banner ODS are actually materialized views that have the same names as the tables in the source database and get their data from the source database tables over a DB link. These materialized views are implemented as physical database tables that include triggers.

The Banner ODS upgrade process creates materialized views for all of the tables in your source database that are associated with Banner ODS. The upgrade process also creates an mview log for each table in the source database that doesn't already have an mview log associated with it. These mview logs track changes that are made to the source tables. The processes that refresh the Banner ODS database read the changes recorded in the mviews logs and update the materialized views (staging tables) in the Banner ODS staging area accordingly. This keeps the staging area synchronized with the source database.

The Operational Staging Area refers to those schemas in the Banner ODS/EDW database where the staging tables (replicated copies of the source tables) reside. In the mviews framework, the staging tables are implemented as read-only materialized views. This reduces the chance of a conflict between the source tables and the stage tables (materialized views) because the materialized views cannot be updated. The only way that a materialized view in the Banner ODS would be out of sync with its master table in the source system is if the source table has had a change that hasn't yet been applied to the Banner ODS via the refresh process.

There is an ODSSTG schema in the Banner ODS that houses the staging infrastructure and an ODSSRC schema that houses the ODS ETL components, for example, composite views. Refer to the [“Multi-Entity Processing”](#) section later in this document for information about schema relationships and how they relate between the source and target databases.

The following picture illustrates where materialized views fit into the Banner ODS architecture. For every source table that is used to create a materialized view, there is an associated materialized view log where changes are tracked for refresh purposes.



**Figure 6: Materialized Views in Banner ODS architecture**

## Database links

The Materialized Views framework uses database links (db links) to connect to the source system and create or refresh the materialized views in the Banner ODS. To support the use of Oracle Fine Grained Access Control (FGAC) in the source system the Oracle user associated with the DB Links (the ODSSTG user in the source system) must be defined as FGA policy exempt.

There are two options for creating the required database links between the Banner ODS and the source system: a single public database link or multiple private database links. The choice between public and private db links has potential database security implications. Your institution should consider both options, and then decide which one is the best choice for your institution and internal policy requirements.

### Public database link

You can create the db link as a public database link. You create this link as the ODSSTG user in the BPRA instance and connect to the source database as ODSSTG. This option simplifies the administration of the Materialized Views environment since it requires only a single link. However, a public db link may pose a security concern because it is “public”.

### Private database link

You can create the db links as private database links. This requires creating a private db link for each of the source system schema owners. Each of these database links will be owned by the individual schema owners and will connect to the ODSSTG user in the source system. Using private db links provides somewhat tighter security because the links are private, but it also adds to the potential administrative overhead because of the number of links that you need to maintain. Additionally, if private db links are used there will also be a private db link owned by ODSSTG connecting to the ODSSTG user in the source system.

Each schema requires only one db link from Banner ODS to the source when you choose to use private database links. The following figure illustrates some of the schemas that would be in place for a Banner ODS environment with Banner as the source system.

 **Note**

This illustration does not include all schemas. ■

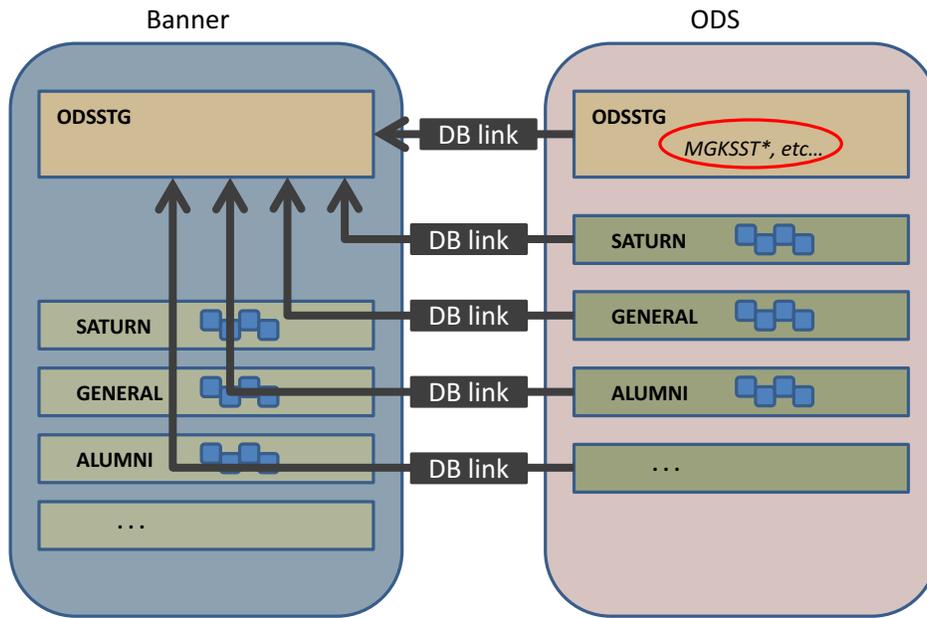
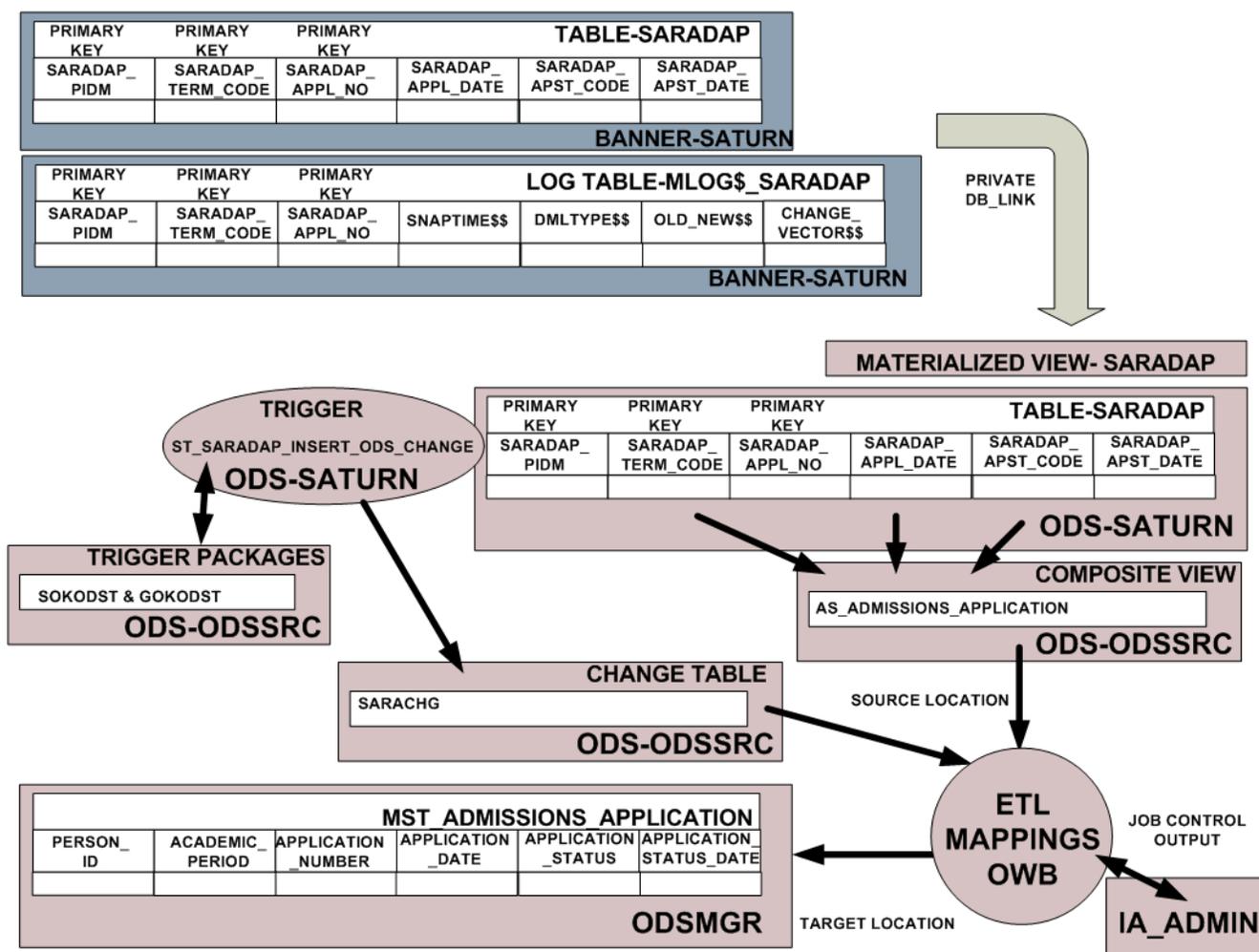


Figure 7: Source and Banner ODS schemas

## Source to target data flow

The following picture illustrates the detailed data flow of information from the source to target database using Materialized Views. This example identifies a Banner table and its components related to the materialized view framework. An Advance source system would share a similar data flow replacing the Banner-specific tables and codes with comparable Advance tables and codes.



**Figure 8: Materialized Views with Banner and ODS data flow**

This example uses the Banner table SARADAP, which resides in the SATURN schema in the Banner database. The log table created for this Banner table is called MLOG\$\_SARADAP and also resides in the SATURN schema in the Banner database. This log is used to track INSERT, UPDATE, and DELETE actions occurring in the SARADAP table to be used by the FAST refresh of the materialized view. In this case, the SARADAP table in Banner has a Primary Key on it which is used for the FAST refresh. The MLOG\$\_SARADAP table will log records in it which contain the Primary Key columns. These columns will be used during the FAST refresh to compare data in the log to the data in the materialized view.

Records will continue to accumulate in the log tables until a materialized refresh is performed for SARADAP. The materialized view refresh process uses the DBLINK back to the Banner database to pull the records from the log and refresh the materialized view SARADAP in the ODS.

From this point forward, the ETL architecture will perform as it has previously using triggers and change tables to refresh the Banner ODS. The difference being that all of this logic has been taken out of the Banner database and now lives in the ODS database. The materialized view SARADAP will have an ODS trigger on it (ST\_SARADAP\_INSERT\_ODS\_CHANGE) which then populates the change table SARACHG. This change table, in conjunction with the AS\_ADMISSIONS\_APPLICATION Composite View, is used as part of the ODS refresh of the Composite Table MST\_ADMISSIONS\_APPLICATION.

## Staging infrastructure

As part of setting up and maintaining the staging area of the warehouse, you will create, load, and stage materialized views in the target database. In addition, you may need to remove and restage materialized views.

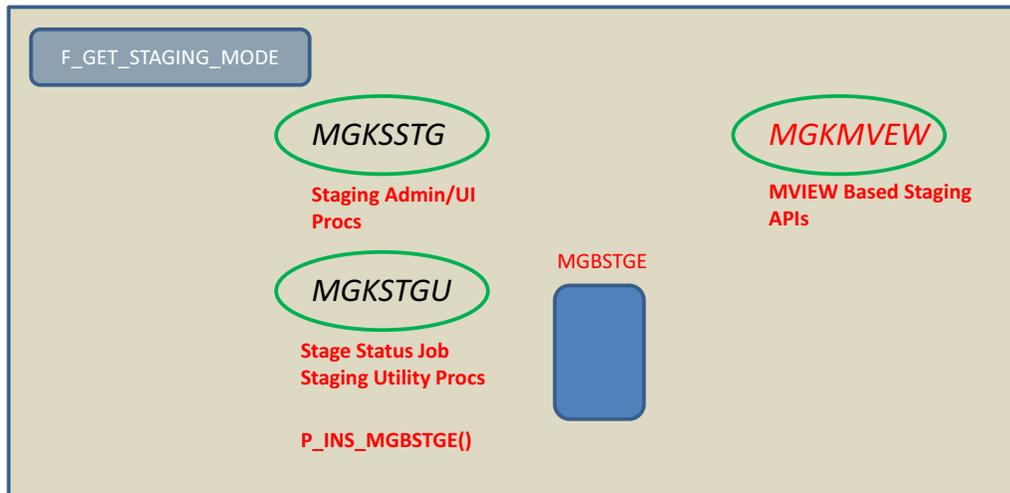
The initial installation of or upgrade to the Materialized Views framework performs the initial creation of materialized views and the database links that are needed to support the delivered Banner ODS features. These materialized views are the staging tables located in the staging area of the target database.

### Note

Refer to the [“Maintain Materialized Views framework”](#) section in [Chapter 2, “Administrative User Interface”](#) for more information about staging and maintaining the Materialized Views architecture. ■

## Materialized views staging objects

The system uses several objects (tables, packages, function, and procedures) to manage and let you monitor the Materialized Views framework. The following figure illustrates the staging objects used with the Materialized Views framework.



**Figure 9: Staging packages for Materialized Views**

The F\_GET\_STAGING\_MODE function returns a value showing whether tables have been staged in the Banner ODS. The MGBSTGE table contains a record for each Banner table that is needed to support the delivered Banner ODS processes. The MGKMVEW contains the procedures that are used to stage tables as materialized views. The packages MGKSSTG and MGKSTGU provide the functionality to maintain and monitor the materialized views from the BPRA Administrative User Interface (UI).

The Administrative UI tool offers you a user GUI interface to perform tasks that maintain and update the Materialized Views framework. Using the Administrative UI, you can create additional materialized views or refresh the materialized views. The procedures P\_STAGE\_MVIEW and P\_UNSTAGE\_MVIEW located in the MGKMVEW package are the actual components used to carry out the staging (data refresh) tasks.

### P\_STAGE\_MVIEW procedure

The system uses the P\_STAGE\_MVIEW procedure located in the MGKMVEW package to create, reload, or restage the materialized views that replicate source tables in the Banner ODS. All indexes on the source system master table are created on the materialized view in the Banner ODS. This procedure also creates the necessary synonyms in the ODS to support the ETL processes.

```
PROCEDURE P_STAGE_MVIEW(  
    SRC_ALIAS_IN          VARCHAR2 ,  
    {OWNER   VARCHAR2 ,
```

```

TABLE_IN    VARCHAR2
| TABLES_IN          ODSSTG.STAGING_TABTYPE,
}

OVERWRITE_IN          VARCHAR2 DEFAULT NULL,
MVLOG_TABLESPACE_IN  VARCHAR2 DEFAULT NULL,
MVIEW_TABLESPACE_IN  VARCHAR2 DEFAULT NULL,
MVINDX_TABLESPACE_IN VARCHAR2 DEFAULT NULL,
DIRECTORY_IN         VARCHAR2 DEFAULT NULL);

```

## **P\_STAGE\_MVIEW procedure arguments**

Following are descriptions of the arguments for the P\_STAGE\_MVIEW procedure.

### ***SRC\_ALIAS\_IN***

A logical name assigned to the database link owned by ODSSTG that points to the master site.

### ***OWNER\_IN, TABLE\_IN | TABLES\_IN***

The owner and tables to be staged in the ODS as materialized views. OWNER\_IN must be a single owner, while TABLE\_IN supports wildcards in order to stage a single or multiple tables.

The procedure is overloaded to alternatively accept a TABLES\_IN parameter. TABLES\_IN is a collection of objects defined as fields of OWNER and TABLE\_NAME, where both fields are of a VARCHAR2(30) data type.

### ***OVERWRITE\_IN***

An optional string specifying what to do when a materialized view is already staged in the ODS. Valid values are:

- 'N' - No overwrite. Do nothing if the materialized view already exists. (Default value)
- 'S' - Synchronize (or reload) the materialized view. This option disables any table triggers on the materialized view, purges the materialized view from the materialized view log, performs a complete refresh of the staged data, and enables the table triggers.
- 'Y' - Overwrite (or restage) the materialized view. This option saves all table triggers for the materialized view, safely drops the materialized view log, drops the materialized view, recreates the materialized view log, recreates the materialized view, and restores all table triggers on the materialized view.

### ***MVLOG\_TABLESPACE\_IN***

An optional string specifying in which tablespace at the master site the materialized view log should be created. If no value is specified, the log will be created in the default tablespace for the owner of the master table.

### ***MVIEW\_TABLESPACE\_IN***

An optional string specifying in which tablespace at the materialized view site the materialized view should be created. If no value is specified, the materialized view will be created in the default tablespace for the owner of the materialized view.

### ***MVINDX\_TABLESPACE\_IN***

An optional string specifying in which tablespace at the materialized view site the materialized view indexes should be created. If no value is specified, the indexes will be created in the default tablespace for the owner of the materialized view.

### ***DIRECTORY\_IN***

This procedure provides the ability to either create the materialized views for you, or generates scripts that can be run later. This is an optional parameter specifying where to generate the scripts. This must be a valid directory object name as can be viewed in the ALL\_DIRECTORIES database view.

Each call to the P\_STAGE\_MVIEW procedure also generates the following driver scripts needed to create and drop the materialized views:

- mvlogs\_create.sql - run at the master site to create the materialized view logs
- mviews\_create.sql - run at the materialized view site to create the materialized views and synonyms
- mvlogs\_drop.sql - run at the master site to drop the materialized view logs
- mviews\_drop.sql - run at the materialized view site to drop the materialized views and synonyms

### ***Example: Stage ALUMNI table***

The following command stages the ALUMNI.AABDUES table in the ODS. The materialized view log will be created in the ALUMNI user's default tablespace at the master site. The materialized view and indexes will be created in the ALUMNI user's default tablespace at the materialized view site.

```
exec mgkmvew.p_stage_mview('BPRA_BANNER', 'ALUMNI', 'AABDUES');
```

### ***Example: Restage tables***

The following commands will restage both the SATURN.SPRADDR and GENERAL.GURMAIL tables. All materialized view logs will be created in the MVLOG

tablespace at the master site. All materialized views and indexes will be created in the MVIEW and INDX tablespaces at the materialized view site, respectively.

```
Declare
    Tab odsstg.staging_tabtype;
Begin
    Tab(1) := 'SATURN.SPRADDR';
    Tab(2) := 'GENERAL.GURMAIL';
    Mgkmview.p_stage_mview(
        Src_alias_in => 'BPRA_BANNER',
        Tables_in => TAB,
        Overwrite_in => 'Y',
        Mvlog_tablespace_in => 'MVLOG',
        Mview_tablespace_in => 'MVIEW',
        Mvindx_tablespace_in => 'INDX');
End;
/
```

### **Example: Generate materialized view scripts**

The following commands will generate materialized view scripts for all Banner Finance materialized views. The scripts will be generated in the database server's directory associated with the DATA\_PUMP\_DIR directory object. The script will create all objects in their users' default tablespaces.

```
Begin
    mgkmview.p_stage_mview(
        src_alias_in => 'BPRA_BANNER',
        owner_in => 'FIMSMGR',
        table_in => '%',
        directory_in => 'DATA_PUMP_DIR');
End;
/
```

## **P\_UNSTAGE\_MVIEW procedure**

The P\_UNSTAGE\_MVIEW procedure located in the MGKMVIEW package safely drops the materialized view log from the master site (source), and drops the materialized view and synonyms from the Banner ODS.

```
PROCEDURE P_UNSTAGE_MVIEW(
    SRC_ALIAS_IN          VARCHAR2,
    {OWNER    VARCHAR2,
    TABLE_IN   VARCHAR2
    | TABLES_IN          ODSSTG.STAGING_TABTYPE, }
    PRESERVE_MV_TABS_IN  VARCHAR2 DEFAULT 'N');
```

## P\_UNSTAGE\_MVIEW procedure arguments

Following are descriptions of the arguments for the P\_UNSTAGE\_MVIEW procedure.

### **SRC\_ALIAS\_IN**

A logical name assigned to the database link owned by ODSSTG that points to the master site.

### **OWNER\_IN, TABLE\_IN | TABLES\_IN**

The owner and tables to be removed from the ODS. OWNER\_IN must be a single owner, while TABLE\_IN supports wildcards in order to remove a single or multiple tables.

The P\_UNSTAGE\_MVIEW procedure is overloaded to alternatively accept a TABLES\_IN parameter. TABLES\_IN is a collection of objects defined as fields of OWNER and TABLE\_NAME, where both fields are of a VARCHAR2(30) data type.

### **PRESERVE\_MV\_TABS\_IN**

An optional string when 'Y' specifies the materialized view should be dropped, but the underlying table structure and data should remain in the ODS. Once complete, this data can no longer be refreshed based on records in the materialized view log. When the value is 'N' then the underlying table and data will be dropped along with the materialized view. The default value is 'N'.

### **Example: Remove materialized views from staging area**

The following example will remove all of the Accounts Receivable materialized views from the Banner ODS staging area.

```
exec mgkmvew.p_unstage_mview('BPRA_BANNER', 'TAISMGR', '%');
```

### **Example: Remove table and materialized views from staging area, keep sources**

The following will remove the ALUMNI.AABDUES table and the POSNCTL.NHRDIST materialized views from the staging area, but the underlying tables and data will be retained.

```
Declare
    Tabodsstg.staging_tabtype;
Begin
    Tab(1) := 'ALUMNI.AABDUES';
    Tab(2) := 'POSNCTL.NHRDIST';
    Mgkmvew.p_unstage_mview(
        Src_alias_in => 'BPRA_BANNER',
        Tables_in => tab,
        Preserve_mv_tabs_in => 'Y');
End;
/
```

## makeMVs.sql script

The makeMVs.sql script is available in the `ia_admin\dbscripts\utility_scripts` directory. This is a sample script that shows how to mass-generate Materialized Views scripts for two schemas.

## Refresh Materialized Views

The Materialized Views architecture uses Oracle's Fast Refresh process, which tracks only the changes since the last refresh. This incremental refresh functionality speeds the process of refreshing the materialized views in Banner ODS. If there is a primary key on the source table, the Fast Refresh uses that key to perform the refresh. When a change happens in the source, the change is put in the log and the key is logged. If a source table doesn't have a primary key, the materialized view is refreshed using RowID.

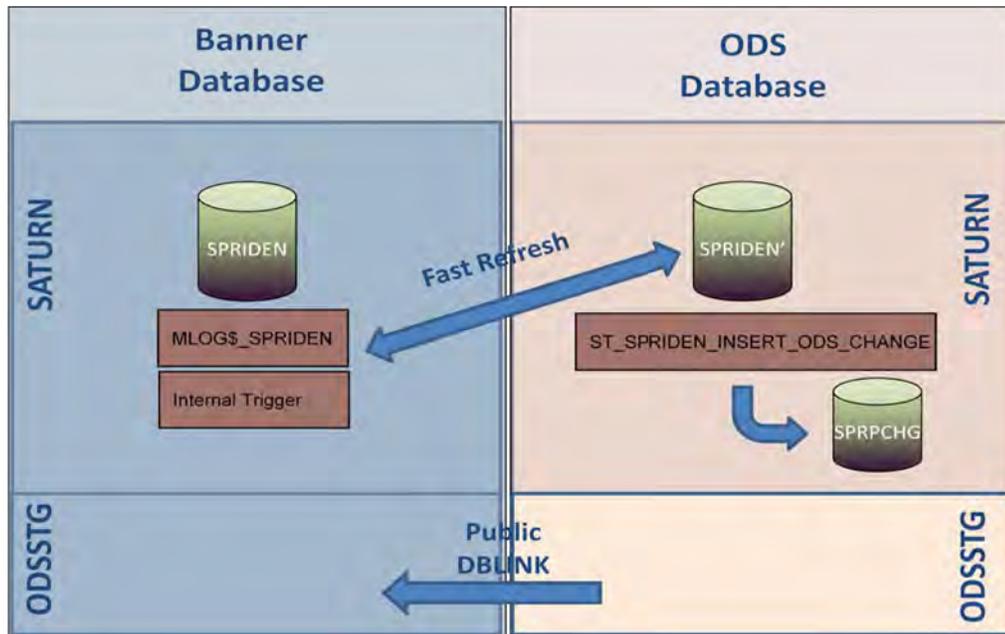
Because the materialized view architecture uses the Fast Refresh, a materialized view log is created for each table in the source system during the upgrade to the Materialized View architecture. The log specifies how to track changes. As part of the log creation, Oracle creates a trigger on the table. In the warehouse, when materialized views are created (with names same as in source database) the system gets all data and table structure from the source over the DB Link and includes the statement “refresh fast on demand with primary key”.

Materialized view refresh processing has been added to the beginning of the ETL jobs available in the Administrative User Interface (UI).

### Note

Refer to [Chapter 2, “Administrative User Interface”](#) for details about using the Administrative UI to refresh the materialized views and maintaining the materialized views framework. ■

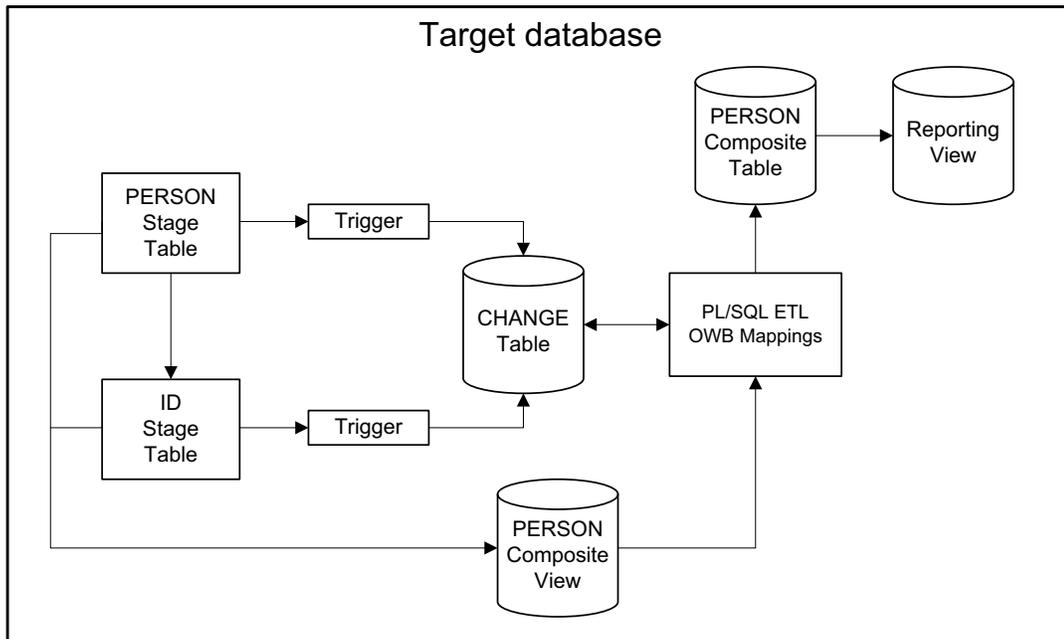
The following picture illustrates the objects used and actions performed during refresh of the SPRIDEN materialized views.



Changes to the SPRIDEN table in Banner fire the internal Oracle trigger on SPRIDEN and insert changes into the mview log (MLOG\$\_SPRIDEN). Changes continue to accumulate in the mview log until a materialized view refresh is performed. After a materialized view refresh occurs, the records from the mview log are pushed over to the SPRIDEN materialized view in Banner ODS to synchronize it with the SPRIDEN table in Banner. As changes are pushed into the materialized view, the ODS trigger (ST\_SPRIDEN\_INSERT\_ODS\_CHANGE) on the SPRIDEN materialized view fires and inserts a record into the ODS Change Table (SPRPCHG). These records are used to refresh the normal ODS Composite Table.

# Extract, Transform, and Load process (ETL)

The ETL process uses OWB, triggers, and change tables to load and refresh data from the staging tables to the composite tables in the target database. The following figure illustrates these components:



**Figure 10: Target database components**

The system uses the ETL processes to extract data from the staging tables and load it into the warehouse composite tables. All ETL activities are performed from within Oracle PL/SQL packages and deployed into a target database schema. The PL/SQL packages are created using Oracle Warehouse Builder (OWB). These packages are scheduled and run via the DBMS\_JOBS queue in Oracle.

All objects are created in the target database including all change tables, triggers, packages and composite views. All source tables needed to create the composite views are replicated in the target database with the same schema name as in the source. For example, the target database has a SATURN schema which contains replicated source (Banner) tables.

## Note

If you use Materialized Views to replicate data, you can schedule the Materialized Views refresh jobs to keep the source database tables and the target staging tables synchronized. ■

You can submit and monitor the ETL jobs using the Administrative User Interface. Typically referred to as ‘mappings’, the packages, when executed, delete, update and load data from the staging to the composite tables based on the type of mappings executed.

During the initial load of the target database, data is extracted from the source database using Oracle views that include specific business logic (for example, Enrolled or In State Resident indicators). The extracted data is then migrated into denormalized composite tables within the target database. These composite tables represent a conceptual organizational structure (for example, Student, an Employee, or a Receivable Customer). To provide for data value security, the Administrative UI allows you to create Oracle Fine Grained Access rules and apply them to the composite tables to prevent information from being viewed without authorization.

The final layer of data access is the reporting views. These views allow calculated columns and increased flexibility in managing what data the end users can access. In select instances, such as the slotted concepts, data display rules are applied to user and institution profiles which filter out unwanted data.

To ensure that the data is current, you can incrementally refresh the target database on a scheduled basis. OWB packages combine the business logic views with the change tables located in the product schemas to determine what updates are applied to the target database composite tables.

You can manage all data loads and updates, fine grained access rules, meta data management, data display rules, and freeze data processing using the Administrative UI.

## **ETL components**

The following section describe the various components used to accomplish the ETL process.

### **Stage tables**

Information from the source database tables is replicated in the target database stage tables.

### **Database triggers**

A single database trigger exists on each stage table, except for the validation tables. Triggers exist for all tables used in a view, including function tables. The triggers are created in the schema owner of the associated stage table.

Each trigger identifies Data Manipulation Language (DML) activity on the table. When a change is made to a source table, that change is replicated in the associated target database table. The change in the staging table causes the trigger on that table to fire. The trigger calls a stored PL/SQL procedure which inserts records into the appropriate change tables

to reflect the change in the replicated table. The triggers flag changes on Banner replicated tables and create records in the appropriate change tables.

Triggers are created on the actual source replicated tables that provide source data for the target database. The triggers are not delivered with the baseline Banner applications.

## Trigger packages

Trigger packages manage the trigger procedures. There is one procedure for each change table with each procedure managing a unique index on the change table. There is one package per product area within the target database, such as Student, Human Resources, Finance, Financial Aid, Advancement. ODSSRC owns the trigger packages.

As data is entered into the source database, it is typically processed one row at a time. For each field entered, the data is verified for field syntax, such as date or numeric format. Fields requiring additional verification are verified against rule tables. After the values are properly checked, the data is committed to the database table that will house the information. During the commit action, any Oracle triggers on the database table being updated are initiated and additional, but separate, logic is executed based on the parameters of the trigger (such as Before Insert and After Insert).

Triggers are built and enabled on all source database replicated tables that house information that is used in the target database. Therefore, when a target database trigger is fired, the trigger inserts the keys of the data being changed into the change tables along with a DML indicator. The existence of these rows in the change tables tells target database that the source has data waiting to be retrieved.

### Note

The change tables only maintain the most recent database activity for a row of information for a specific key. When multiple actions occur against the same source database table and row, only the last action is represented in the change table. This allows the replication process to work faster, and decreases the amount of data captured in the change tables. ■

## Change tables

Change tables maintain data about what tables and records have been changed, inserted, or deleted in the stage tables and the source database tables. There is not a one-to-one relationship between change tables and stage tables or between changes tables and composite tables. One change table exists for each logical group of information.

Change tables work like collector tables. They include four basic fields:

- Keys
- Table name
- Process ID

- Most recent DML.

Change tables reflect DML activity for specific target database stage tables, but are also used when multiple tables use the same key.

**Example:**

The SPRPCHG table stores DML activities for the Hold and the Person composite views.

Change tables are owned by their respective product schemas in the target database, and are identified using standard source table naming conventions. The column names start with the seven-character prefix of the table name. All columns in each of the change tables are identical with the exception of the key columns. Here, the key columns represent the product/database tables they are accessing, and also represent the keys that the target database uses when records change. All change tables are suffixed by ‘CHG’.

The columns that compose the change table are the key columns relative to the composite view(s) it supports, along with the TABLE\_NAME and the PROCESS\_ID columns. The last two columns allow inserts into the table with a null PROCESS\_ID by updates to Banner that take place during Incremental Refresh. Since the target database processes and deletes all rows in the change tables with a NOT NULL PROCESS\_ID, the null value allows the row to stay until the next update. This ensures that it is not bypassed or inadvertently deleted.

Typically, a second index is created in the format of TABLE\_NAME, PROCESS\_ID, and RECORD\_ACTION columns.

**Example:**

SPRPCHG – Change table for PIDM related Banner replicated tables

Column Name	Data Type	Column Comment
SPRPCHG_TABLE_NAME	VARCHAR2(30)	Used to identify which composite view (and/or target database table) is being populated by this specific row of data.
SPRPCHG_PIDM	NUMBER	The change table needs to hold as many keys as required to manage DELETE and UPDATE information in the target database. Keys do not need to identify a unique row, but must maintain some fields for comparison.
SPRPCHG_RECORD_ACTION	VARCHAR2(1)	Stores the last DML action for the key combination ( <i>I</i> , <i>U</i> , or <i>D</i> ).

Column Name	Data Type	Column Comment
SPRPCHG_PROCESS_ID	VARCHAR2(30)	Updated by the procedure UPDATE_CHANGE_TABLE which inserts non-null values to flag which rows are being processed during the incremental refresh process. This allows inserts to take place into the change table while replication is also taking place.
SPRPCHG_ACTIVITY_DATE	DATE	Reflects the actual date of insertion or update of the rows.

## Change table triggers

The target database maintains triggers on all source replicated tables used to incrementally refresh data into the target. Although the triggers are enabled on the actual source replicated tables, they are referred to as ‘change table triggers’ because they populate the target database change tables with DML information. The trigger inserts rows of information in one or more change tables by invoking a procedure that packages all trigger insert actions for the target database change tables.

The triggers use basic logic except that the Exception routines allow for continued processing when encountering a DUP\_VAL\_ON\_INDEX condition. This condition occurs when a row of data exists within the change table for the table’s unique index. When encountered, the procedure updates rather than inserts the information in the change table by overlaying the DML activity and the activity date. This action causes only the most recent DML activity to be stored in the change table.

All triggers are owned and maintained within the product schema of the table to which the triggers are added. For example, SATURN would own Student Triggers.

Change table Triggers comprise of the following procedures:

- Each Banner product has a procedure that manages all change table triggers for that product area. For example, GOKODST for General and SOKODST for SATURN.
- The triggers are owned by the ODSSRC schema.
- The names for each procedure follow Banner standard naming conventions.

## Composite views and functions or packages

Composite views exist in the target database under the ODSSRC schema. During the ETL process, when you perform a refresh of target data, the composite views are joined with the appropriate change tables and updated with the changed information.

In some cases, functions are used to calculate new data that is created from source data and loaded into the target database. Packages are used to group related procedures, functions, and cursors together. There is one package for each target database module of information; for example, Student, Finance, and Advancement. These packages are installed into the ODSSRC schema.

Composite views represent a composite (mixture) of the tables selected from Banner and allow for a single piece of data to be extracted row-by-row, with all the business logic included in the view itself. The column names are generic so that they can be used by all SunGard Higher Education product lines. Therefore, names familiar to Banner clients can appear to be more generic than the familiar Banner terminology. For example, Term becomes Academic Period, PIDM becomes UID (unique ID). The views are used for reporting in Banner. But, they are designed to become the Incremental Refresh data extraction view.

Views are created and maintained in the ODSSRC schema within the target database. Since these views are accessing data directly in the various source replicated tables, explicit SELECT grants are assigned to the schema when tables are staged in the target. Refer to the section [“Multi-Entity Processing” on page 1-35](#) to see a list of schemas and what they own.

## **OWB mappings**

Oracle Warehouse Builder (OWB) mappings, which are PL/SQL scripts, define the relationship of data between the composite views and composite tables. The Extract, Transform, and Load processes (ETL) built using OWB are the mappings that populate Banner ODS.

The OWB mappings are run during the initial load of Banner ODS and when you incrementally refresh Banner ODS. When run, the scripts load, update, or delete data in Banner ODS composite tables. Three scripts — Load, Update, and Delete — exist for each Banner ODS composite table. The different types of mappings perform the following functions:

- **LOAD mappings:** initially load Banner ODS composite tables by selecting all rows of data from the source system via the composite view.
- **DELETE mappings:** delete rows of data in Banner ODS when the change table reflects activity of any type for the key. This mapping uses the key in the change table since no data will be found in the composite view for deletes. This process also updates the `PROCESS_ID` value in the corresponding change table for all rows before any delete takes place.

- UPDATE mappings: insert records into Banner ODS based on keys in the composite view joined against rows in the corresponding change table.

 **Note**

It is mandatory that you run the DELETE mapping before the related UPDATE mapping, otherwise no records will process in the UPDATE mapping. ■

The OWB user interface contains graphical editors that enable you to design a complete logical model of your warehouse. The OWB helps you plan how to extract data from a variety of sources, transform the data, and configure the data for loading into Banner ODS. The OWB code generator lets you deploy and populate the Banner ODS without manual coding, and integrates with the Oracle database and query tools.

## Composite and slotted tables

Composite tables are the tables within Banner ODS that are loaded with data from the source system. Slotted tables store data values for a specific code related to a base table.

### Composite tables

The composite tables are populated during the initial install process, and are also updated during the incremental refresh process. The composite tables are used for the following purposes:

- Denormalized tables are used to store “conceptual” structures of data.
- Normalized tables are used for quick data filtering or for unlimited repeating values.
- The MGRSDAX rule table is used to load the composite tables.

### Slotted tables

The slotted tables have the following attributes:

- Used to denormalize Repeating Concepts (normalized tables.)
- Populated via rules from MGRSDAX.

Some GTVSDAX rules, but not values, are duplicated when MGRSDAX is initially populated. Use the Administrative UI to add or modify MGRSDAX rules' values to meet your institution's needs.

## Understanding composite tables and slotted tables

Banner ODS includes composite tables and slotted tables. Composite tables include the main data that is extracted from your source system and stored in Banner ODS. Slotted tables store data values for a specific code related to a base table.

### *Example*

The `TEST_SCORES_SLOTTED` table in Banner ODS stores all valid Test Score values that were loaded from your source system to Banner ODS. When a report is created against Banner ODS, the system pulls data from the composite tables. The system checks codes stored on the slotted tables, as needed, and pulls the appropriate code values. If you choose to use Business Profiles, the system pulls the appropriate values for the profile with which the user is associated. The default business profile of INSTITUTION is used when specific display rules are not established.

Using slotted tables optimizes the speed queries since the system only has to check for specific code values as needed.

## Updating slotted tables

It is important to keep data in the slotted tables synchronized with data in the composite tables. Whenever composite tables are updated, related slotted tables should also be updated.

Both composite and slotted tables are updated when refresh jobs are run to update Banner ODS data on a regular basis.

## Reporting views

Data from each Banner ODS composite table is presented in one or more reporting views. Banner ODS reporting views are the views that your users use to create reports within Banner ODS. Users point their report writing tool at these views and build reports.

## Run ETL load processes

You run load processes using the Administrative User Interface (UI) from the **Options>Schedule a Process>Select a Subprocess>Schedule Banner ODS Mappings** menu option. When you run a process, one or more LOAD mappings extract all the data from a composite view in the source system and move it into the corresponding target database composite table.

You can run a Load process periodically for one or more composite tables, for example, as an alternative to the Refresh process. To facilitate the use of a load at any time, the Load processes also purge the appropriate change tables that correspond to the composite tables being loaded.

You can disable the purge feature on Load mappings. To disable the change table purge for a Load mapping, you need to create records in the MTVPARM table. Refer to the [“ETL MAP PACKAGE LOAD PURGE Parameter”](#) section of the [“Administrative User Interface”](#) chapter for information about using this parameter to disable a change table purge for a Load mapping.

## Run ETL Load or Refresh jobs in parallel

You can schedule the ETL Load and Refresh jobs to run in parallel to reduce the time it takes to load or refresh the entire database.

## Run ETL Refresh jobs in parallel in Materialized Views framework

If the ETL Refresh jobs run at the same time, it's possible that materialized views shared by multiple areas of the warehouse may not get refreshed appropriately before the actual ETL Refresh job runs. To avoid this issue, you should refresh any related materialized views before you run the ETL Refresh jobs in parallel.

### *Example*

Suppose that your institution runs the ETL refresh jobs Refresh General, Refresh Student, and Refresh Accounts Receivable at the same time each night. The source Banner SPRIDEN table is used for refreshing both the Student and General subject areas. This means it's possible that the ETL Refresh jobs could load the warehouse before all of the related materialized views have been refreshed. To ensure that all related materialized views are refreshed in the staging area before the ETL Refresh jobs actually load the target database, in this case you should schedule the following mviews refresh jobs to run before the ETL Refresh jobs:

- AR Refresh Group and AR Validation Refresh Group
- General Refresh Group and General Validation Refresh Group
- Student Refresh Group and Student Validation Refresh Group

## Incremental refresh process

The term incremental refresh identifies how data synchronization occurs between source and target tables to ensure that accurate information is stored in the target database. Data that has changed in the source is captured and, using the ETL tool, is applied to the target database. During the process, the change tables bring over only the data that has changed, and then, using an ETL mapping, the change tables are deleted. This is followed by an update ETL mapping that inserts the new data. The incremental refresh process uses records in change tables to identify the records which need to be refreshed, and uses different mappings for load vs. refresh processes.

Typically, you will run a complete load, then run the refresh processes on a nightly basis to keep the target data synchronized with the source data. You should also run an incremental refresh process if data in Banner ODS has changed since the last time you ran the refresh.

## Multi-Entity Processing

---

The Multi-Entity Processing (MEP) framework is available for all target database composite views, composite tables, and reporting views. This enables all information from multiple sources (data sources, institutions, campuses, etc.) that is located in one database to be selectively assigned security access as needed in the target database.

### **Example:**

You can take existing data from one database for use in multiple institutions, move information into the target database, selectively restrict the user access to data by institution, and so on.

The MEP columns only appear on generated meta data reports in the Administrative UI if MEP is set up for your institution.

### **Note**

To use MEP with your source system and target database, Professional Services must provide the needed analysis, subsequent product enhancements, and set up. This includes identifying source tables that require MEP, and the target database objects to be modified. ■

## Administrative User Interface

---

The Administrative User Interface (UI) is Web-based and uses SGHE's Banner Web Tailor. The Administrative UI is used to set up and maintain the target database and warehouse, including initiating and monitoring ETL processes. Administrative functions include:

- Preferences and Security - Use to manage security, set global preferences, and set up user accounts.
- Options - Use to control the processes to extract and load data into Banner ODS, schedule a process, view control reports, view and/or remove scheduled processes, and maintain information about saving (freezing) data.
- Meta Data - Use to view and manage the meta data supporting the systems.
- New Banner Web Tailor Administration - Use to customize a Web menu, procedure, graphic element, set of information text, or a set of menu items. You can also update user roles, customize a Web module, Web rules, or Banner Web Tailor

parameters; customize a login return location; and customize Banner Web Tailor overrides or global user interface settings.

## Banner ODS data model

---

SunGard Higher Education has developed a data model that includes data from a number of higher education administrative system modules. The administrative system modules supported by the Banner ODS data model include Student, Financial Aid, Advancement, Human Resources and Finance — including Accounts Receivable. Each module, or area of information, includes a number of tables in the administrative systems. The data model brings the appropriate data elements, from multiple tables in the source system, into a different table structure in the Banner ODS to support the reporting needs of the entire institution.

The data model represents the data elements that are included in Banner ODS. Banner ODS shows the individual table and the relationship with other tables stored within the model. It further includes all the data elements available in Banner ODS composite tables and/or the reporting views related to the object described.

## Multiple source databases

---

The Banner EDW architecture supports stage tables from different source databases. The only requirement to load information from multiple sources into the Banner EDW stage tables is that the schema and table names in the source databases must be unique.

### Note

Because the schema and table names in the source databases must be unique, you cannot load information from two different Banner databases into the Banner EDW. ■

## Source Alias

The Source Alias (`source_alias`) uniquely identifies each source database. You specify the `source_alias` during the installation or upgrade process. The `source_alias` is then used to create a parameter in WebTailor, which associates each `source_alias` to a database link owned by ODSSTG. This approach allows the database link to the source to be changed while minimizing the disruption to the existing Banner EDW functionality.

### *Source Alias in Streams framework*

In the Streams framework, the Source Alias is used as a prefix when naming the various Streams components. The prefix identifies the source database and the suffix identifies the

Streams component. For example, the Streams component BANNER\$APP is associated with the source alias of BANNER, and is an apply process.

The following table lists the database location and suffix for each Streams component. The Source Alias is added to the beginning of each Name Suffix to uniquely identify the Streams component.

Streams Component	Database location	Name Suffix
Capture	Source	\$CAP
Capture queue	Source	\$CAPQ
Capture queue table	Source	\$CAPQT
Propagation	Source	\$PROP
Apply queue	Banner ODS	\$APPQ
Apply queue table	Banner ODS	\$APPQT
Apply	Banner ODS	\$APP

## Add a source database

Use the following steps to add subsequent source databases after an initial source database has been configured.

1. Run the source install steps on the source database. Refer to the Banner ODS Installation or Upgrade Guide for the source install steps.

This creates the ODSSTG administrative user with the necessary privileges, compiles the support package, and creates a database link from the ODSSTG user to the ODSSTG user on the Banner EDW.

2. Create a database link connecting the ODSSTG user on the Banner EDW instance to the ODSSTG user on the new source.

*Streams users only perform step 3; Mviews users can skip to step 4.*

3. As the ODSSTG user on the Banner EDW, execute the following procedure from SQL\*Plus.

```
SQL> SET SERVEROUTPUT ON
SQL> MGKSTRC.P_CREATE_LOCAL_ENV(database link, source
alias);
```

where you enter your institution's values for the parameter in parentheses.

4. Create schemas in the Banner EDW.

For each schema in the source database that includes tables that will be staged in the Banner EDW, create a schema in the Banner EDW with the same name.

5. Add the new schemas to the Banner EDW using the Administrative User Interface.
  - 5.1. Follow the steps in the section [“Add a schema”](#) in [Chapter 2, “Administrative User Interface”](#).
  - 5.2. Repeat step [5](#) for each new schema you want to add to the Banner EDW.
6. Stage new tables in the Banner EDW using the Administrative User Interface.
  - 6.1. Follow the steps in the section [“Add a non-baseline staging table to the Banner ODS”](#) in [Chapter 2, “Administrative User Interface”](#).
  - 6.2. Repeat step [6](#) for each new schema you added.

## Validation table data and incremental refresh

---

The Banner Operational Data Store (Banner ODS) was designed with validation table codes and descriptions stored on each individual data record. This design expedites the display of information as it eliminates the need for excessive joins of as many as ten or fifteen additional tables. During the design phase of Banner ODS, several methodologies on managing validation table change requirements were discussed with institutions. The consensus was that it is preferable to build internal institutional policies and procedures to ensure that the descriptions are not changed, but new codes are added.

This is similar to the way in which Banner Course Catalog process works. If the title of the course changes, the institution creates a new catalog record with the new title for the new effective term.

### **Example**

If a description such as “Bowling Basics” changes to “Bowling Fundamentals”, it is assigned a new code so that Banner ODS reflects the past data for “Bowling Basics” and the new values are reflected for “Bowling Fundamentals”.

To change a column description, the institution policy requires to either initiate a reload of all affected tables (time intensive) or create a script to update all columns in Banner ODS to alter the old value to the new value.

### **Note**

To ensure data integrity, do not apply updates to existing values in the validation tables once Banner ODS is in production and the incremental

refresh cycle is implemented. Else, there will be inconsistencies in the information displayed between the source system and Banner ODS. ■

To further explain the difficulty in incrementally refreshing tables based on coded description changes and not the result of data value changes, it is necessary to understand the efforts required to implement a validation to data table refresh. First, the source system would have to be enhanced to maintain triggers on each validation table to track all DML activity. While it is possible to apply triggers to each of these tables, the trigger event is likely to have performance impact on the source system. This is because it requires the trigger to populate an entry into a change table for every row in each source data table that is populated with the altered validation table value. This requires a full table scan of every affected source table as the source system does not maintain keyed links between the validation tables and the data tables.

For example, the validation table STVDEPT is used enterprise wide in Banner Student, Banner Advancement, and Banner HR systems in eighty four (84) different tables. If a value were to be changed in the STVDEPT table, then the trigger on the STVDEPT table would have to read all 84 of the source tables to identify the key(s) of each row that contained the altered DEPT value, and then populate that key into the change table. Given the size of many of these data tables, the commit time required for the end users to wait on the change of the validation table in Banner would freeze their Banner session until the change table population took place.

## Table indexes

---

Indexes are added based on the reporting needs of the Banner ODS tables as well as performance for the incremental refresh process. The IA\_ADMIN.MGBINDX table stores a list of the delivered indexes for tracking and documentation purposes. You populate this table using the following query for a release:

```
SELECT &sysid, x.table_name, x.index_name, column_name,
       uniqueness, descend, column_position, &relno, 'NO', SYSDATE, 1
FROM user_indexes x, user_ind_columns y
WHERE x.table_name = y.table_name AND
      x.index_name = y.index_name AND
      x.table_name LIKE 'M%' AND
      x.table_name NOT IN
      (SELECT DISTINCT table_name
       FROM all_tab_columns
       WHERE column_name LIKE '%FREEZE_EVENT%')
ORDER BY x.table_name, x.index_name, column_position;
```

The MGBINDX table is used in the Banner ODS Checks and Balances process to verify that baseline indexes are valid and present. If your institution has created additional indexes,

the differences are reported in the control report as warnings. To include the additional indexes in the Banner ODS Checks and Balances process, insert the new index information into the `IA_ADMIN.MGBINDX` table using SQL. Refer to the `MGBINDX_DATA_ODS.SQL` script in the `dbscripts` directory for a syntax example. Set the `LOCAL_IND = 'YES'` to identify this as your institution's index. The local records in this table will be preserved with future upgrades. We recommend that you do not delete baseline rows from the `MGBINDX` table.

The Banner ODS metadata also uses the delivered indexes when documenting the Recommended Search Columns. The script `update_recsearchconds.sql` (located in the `dbscripts/utility_scripts` directory) is used to generate that information based on the actual indexes in the database. If you add local indexes, it is recommended that you run the script (from the `IA_ADMIN` account) so the list of Recommended Search Columns accurately reflects the database.

## Product-specific information

---

This section discusses the Banner ODS information unique to individual Banner products.

### Banner Common

The Banner ODS `VALIDATION` reporting view provides access to all of the Banner product validation table values to be used when creating a pull-down list of values (LOV) for parameters. This reporting view can be used by a variety of reporting tools. The `MGT_VALIDATION` table is the source for the reporting view and is used to build the LOV views that reside in the `ODSLOV` schema. The source for the `MGT_VALIDATION` ODS composite table is a series of composite views listed below. These views retrieve the values from specific product validation tables that are used within the Banner ODS.

Performing a `select distinct` on a code within a reporting view may be a valid solution to generate a List of Values. However, this method will likely cause a performance impact on the system. The `VALIDATION` reporting view can instead be used as a pull-down list. It provides the appropriate Banner Validation Table name as a filter for `VALIDATION.TABLE_NAME`.

The information on the List and Detail Reports pages can be viewed online or exported to a CSV file (Microsoft Excel format) or XML file for printing or additional manipulation. Following are the composite views:

- `AA_VALIDATION`
- `AF_VALIDATION`
- `AG_VALIDATION`
- `AN_VALIDATION`
- `AP_VALIDATION`

- AR\_VALIDATION
- AS\_VALIDATION
- AT\_VALIDATION

Each of these Banner composite views extracts values from validation tables in their respective Banner product areas. Also included are the status indicators, effective dates, and sometimes the qualifiers.

Within Banner Finance, there are several groups of values stored within the FTVSDAT System Data Maintenance table. To properly represent some of these values, they have been pulled into the AF\_VALIDATION composite view with the TABLE\_NAME as follows:

- GRANT\_CATEGORY represents all grant categories stored within FTVSDAT.
- GRANT\_SUBCATEGORY represents all grant sub categories stored within FTVSDAT.
- GRANT\_TYPE represents all grant types stored within FTVSDAT.

Values have been added to table FTVFSPD to represent beginning and ending periods. The added values are '00', '13', and '14'. The FTVFSYR table has for its description, the Fiscal Year converted to a four-digit year.

In specific situations, Banner source tables were not used. The following is a compiled list of data element names used in place of Banner specific tables names.

The hard coded TABLE\_NAMES are as follows:

- ACADEMIC\_TITLE
- ACCOUNT\_ATTRIBUTE\_TYPE
- ACCOUNT\_ATTRIBUTE\_VALUE
- ACCOUNT\_CLASS
- ACCOUNT\_LEVEL\_1
- ACCOUNT\_LEVEL\_2
- ACCOUNT\_LEVEL\_3
- ACCOUNT\_LEVEL\_4
- ACCOUNT\_POOL
- ACCOUNT\_SET\_CODE
- ACCOUNT\_TYPE\_ATTR\_TYPE
- ACCOUNT\_TYPE\_ATTR\_VALUE

- ACCOUNT\_TYPE\_LEVEL\_1
- ACCOUNT\_TYPE\_LEVEL\_2
- ACCOUNT\_TYPE\_SET\_CODE
- ADVISOR\_NAME\_LFMI
- ASSIGNMENT\_GRADE
- CALENDAR\_MONTH
- CALENDAR\_YEAR
- COLLECTION\_AGENCY\_NAME
- CONTRACT\_NUMBER
- CONTRACT\_TYPE
- COURSE\_IDENTIFICATION
- COURSE\_REFERENCE\_NUMBER
- EMPLOYEE\_STATUS
- EMPLOYEE\_TIME\_STATUS
- ENDOWMENT\_FUND
- ENTITY\_TYPE
- FINANCIAL\_AID\_SOURCE\_TYPE
- FINANCIAL\_AID\_TYPE
- FINANCIAL\_MANAGER
- FISCAL\_QUARTER
- FUND\_ATTRIBUTE\_TYPE
- FUND\_ATTRIBUTE\_VALUE
- FUND\_LEVEL\_1
- FUND\_LEVEL\_2
- FUND\_LEVEL\_3
- FUND\_LEVEL\_4
- FUND\_LEVEL\_5
- FUND\_POOL
- FUND\_SET\_CODE
- FUND\_TYPE\_ATTR\_TYPE

- FUND\_TYPE\_ATTR\_VALUE
- FUND\_TYPE\_LEVEL\_1
- FUND\_TYPE\_LEVEL\_2
- FUND\_TYPE\_SET\_CODE
- GENDER
- INSTALLMENT\_PLAN
- INSTRUCTOR\_NAME
- INTENDED\_TIME\_STATUS
- INTERNAL\_ACCOUNT\_TYPE
- INTERNAL\_FUND\_TYPE
- LOCATION\_LEVEL\_1
- LOCATION\_LEVEL\_2
- LOCATION\_LEVEL\_3
- LOCATION\_LEVEL\_4
- LOCATION\_LEVEL\_5
- ORGANIZATION\_ATTR\_TYPE
- ORGANIZATION\_ATTR\_VALUE
- ORGANIZATION\_LEVEL\_1
- ORGANIZATION\_LEVEL\_2
- ORGANIZATION\_LEVEL\_3
- ORGANIZATION\_LEVEL\_4
- ORGANIZATION\_LEVEL\_5
- ORGANIZATION\_LEVEL\_6
- ORGANIZATION\_LEVEL\_7
- ORGANIZATION\_LEVEL\_8
- ORGANIZATION\_POOL
- ORGANIZATION\_SET\_CODE
- ORG\_FINANCIAL\_MANAGER
- POSITION\_STATUS
- POST\_SECONDARY\_SCHOOL

- PREF\_CLAS
- PRINCIPAL\_INVESTIGATOR
- PROGRAM\_ATTR\_TYPE
- PROGRAM\_ATTR\_VALUE
- PROGRAM\_LEVEL\_1
- PROGRAM\_LEVEL\_2
- PROGRAM\_LEVEL\_3
- PROGRAM\_LEVEL\_4
- PROGRAM\_LEVEL\_5
- PROGRAM\_SET\_CODE
- RECEIVABLE\_CONTRACT
- RECEIVABLE\_DELIQUENCY
- RECEIVABLE\_EXEMPTION
- SECONDARY\_SCHOOL
- SPORTS

## Banner Finance

The following table explains the use for FIELD\_CODE and LEDGER\_IND within the TRANSACTION\_HISTORY reporting view for Banner Finance. The LEDGER\_IND and FIELD\_CODE work together to drive what ledger amount field was updated.

LEDGER_IND	Ledger	FIELD_CODE	Amount Field Updated	Description
G	General	01	Sum_Periodic_Debits	Debits
		02	Sum_Periodic_Credits	Credits
O	Operating	01	Curr_Adopted_Budget	Current Period Original Budget
		02	Curr_Budget_Adjustments	Current Period Budget Adjustment
		03	Curr_Year_To_Date_Activity	Current Period Activity
		04	Curr_Encumbrances	Current Period Purchase Order and General Encumbrance
		05	Curr_Budget_Reservation	Current Period Requisition Budget Reservation

LEDGER_ IND	Ledger	FIELD_ CODE	Amount Field Updated	Description
		06	Curr_Accumulated_Budget	Current Period Accounted Budget
		07	Curr_Temporary_Budget	Current Period Temporary Budget
		08	Curr_Grant_Activity	Obsolete
E	Encumbrance	01	Original_Amount	Original Encumbrance Amount
		02	Curr_Adjustments	Encumbrance Adjustments
		03	Curr_Liquidations	Encumbrance Liquidations

## Banner Student

When a new base student record is created in Banner, a new record is created in the Banner ODS table MST\_BASE\_STUDENT. Each record in this table contains a range of academic periods in which the student status allows the student to register. If the status prevents the student from registering, then the beginning and ending academic periods in the Banner ODS record are the same and match the Banner effective term.

Additionally, the MST\_BASE\_STUDENT table contains information on each student's program of study. This table contains one record per student per effective term per.

### Banner Student data extraction for the MST\_GENERAL\_STUDENT composite table

Creating a new record within one of a number of Banner tables indicates to the Banner ODS that the student has activity within the specific term. As a result, a new record is created in the MST\_GENERAL\_STUDENT table for the student and term when the Banner ODS is loaded or refreshed.

Following is a list of Banner tables that define student activity in the Banner ODS:

- SGBSTDN - student base table
- SFBETRM - student registration table
- SHRTTRM - institutional course maintenance term header table
- SHRTRAM - attendance period by transfer institution table
- SHRDGMR - degree table
- SGRCHRT - student cohort table
- SGRSPRT - sport table

- SGRSATT - student attribute table
- SGRSACT - student activity table
- SGRCOOP - cooperative education table
- RPRATRM - applicant award by term table
- RORSTAT - applicant status table
- TBRACCD - account charge/payment detail table
- TBBCSTU - contract student authorization table

The MST\_GENERAL\_STUDENT table also contains information about each student's program of study. This table contains one record per student per academic period with student activity per curricula.

### **Additional 'Zero' record in the Banner ODS tables**

In Banner, the values for student classification and academic standing are specific for a student, academic period, and their primary program level only. In the Banner ODS, many reports require student classification and academic standing data for all student curricula, regardless of the level value. To create comprehensive reports while limiting the number of outer-joins used, a single record with a value of zero for the key fields (person\_uid, student\_level, and academic\_period) is inserted into the MST\_STUDENT\_CLASSIFICATION and MST\_ACADEMIC\_STANDING composite tables as a step in the load mappings. Existing student classification and academic standing values are displayed if they exist for a specified student, level, and academic period. Otherwise, the NULL values from this new record are displayed.

### **Key Banner Student views architecture**

Due to the complex architecture of some Banner Student views, the following flow charts illustrate how those Banner Student reporting views are built from Banner to the Banner ODS.

#### **Note**

These diagrams only refer to key views and key tables used within the reporting views. ■

**Academic Outcome and Academic Outcome Slot reporting view flow**

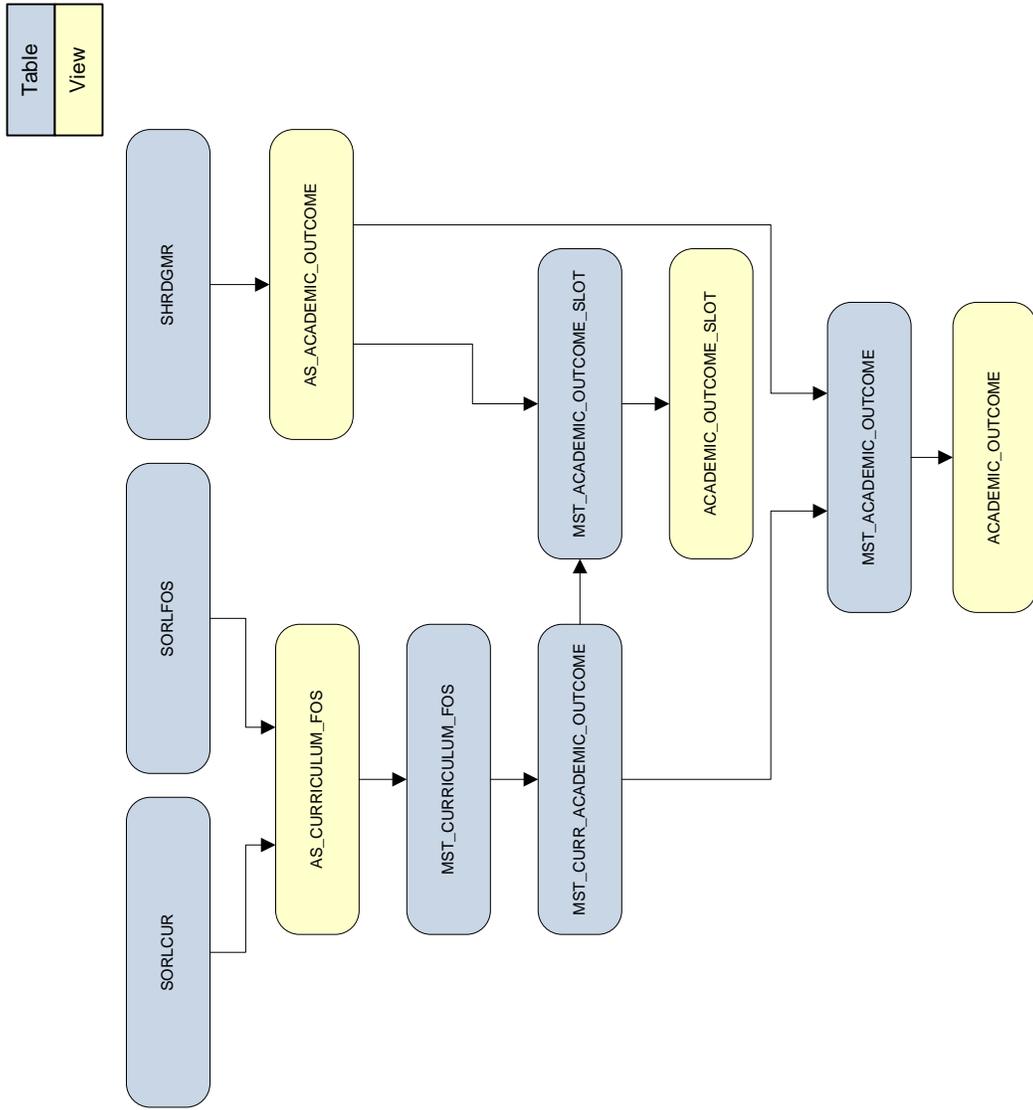


Table
View

**Academic Study reporting view flow**

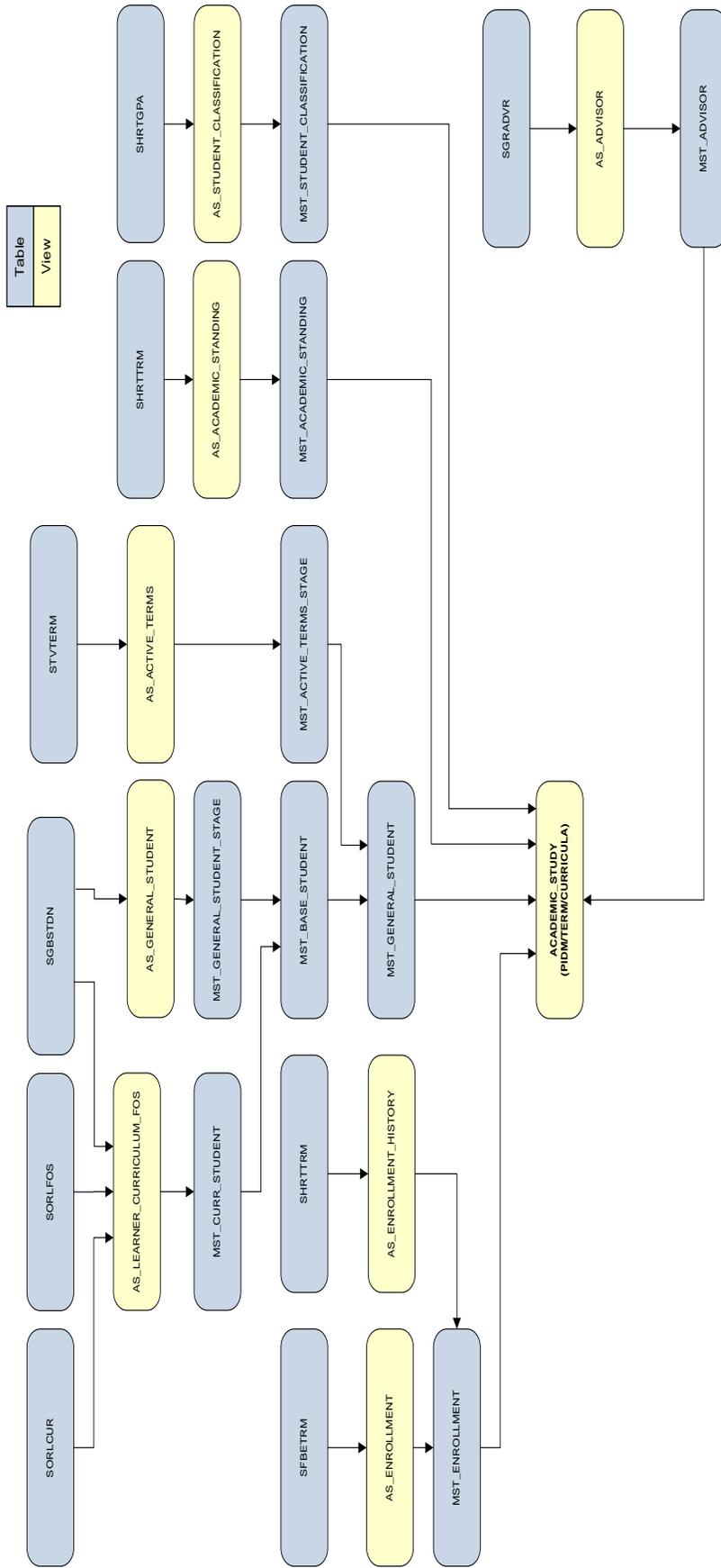
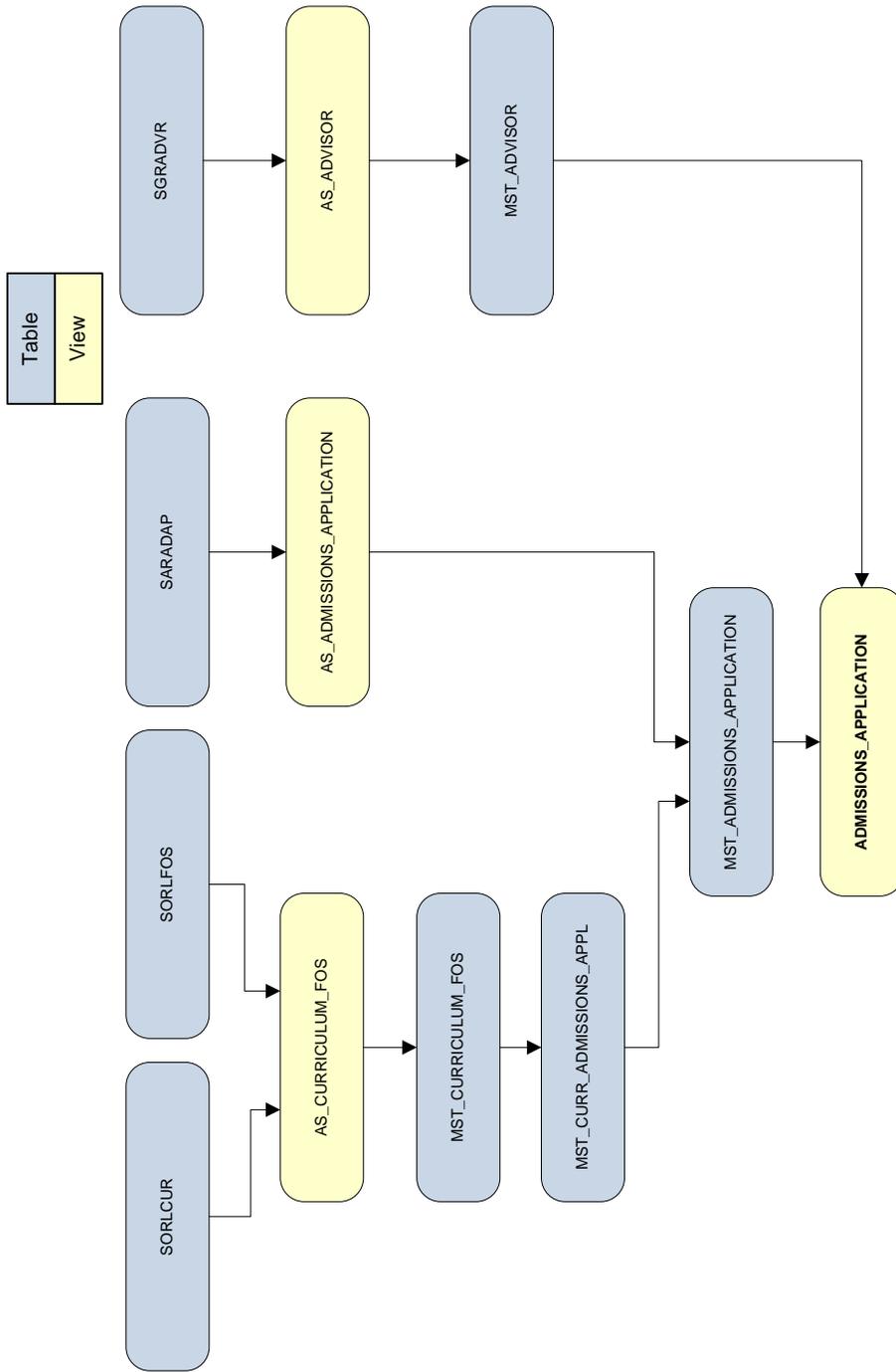


Table
View

*Admissions Application reporting view flow*



**Combined Academic Outcome reporting view flow**

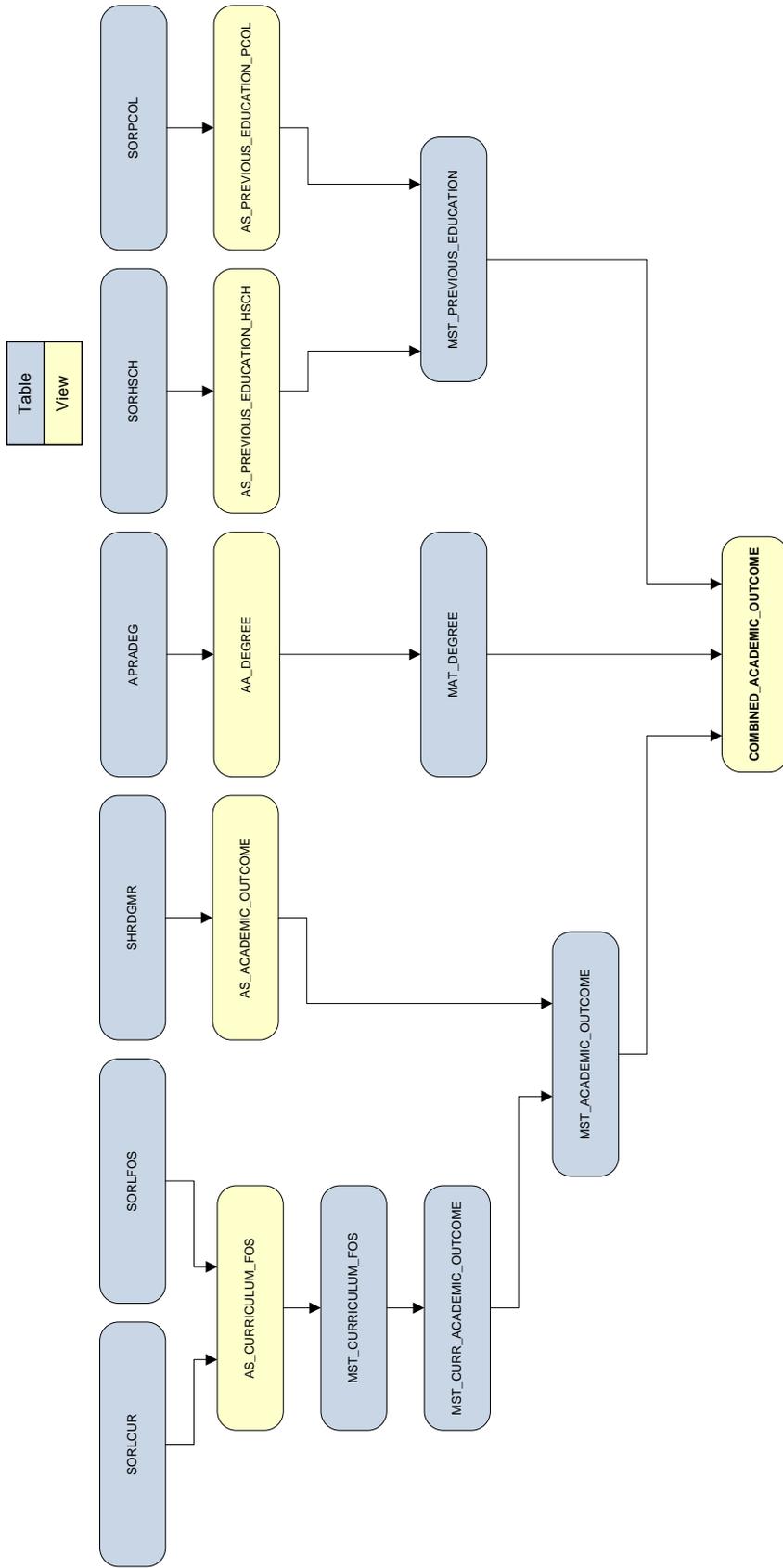


Table
View

*Field of Study reporting view flow*

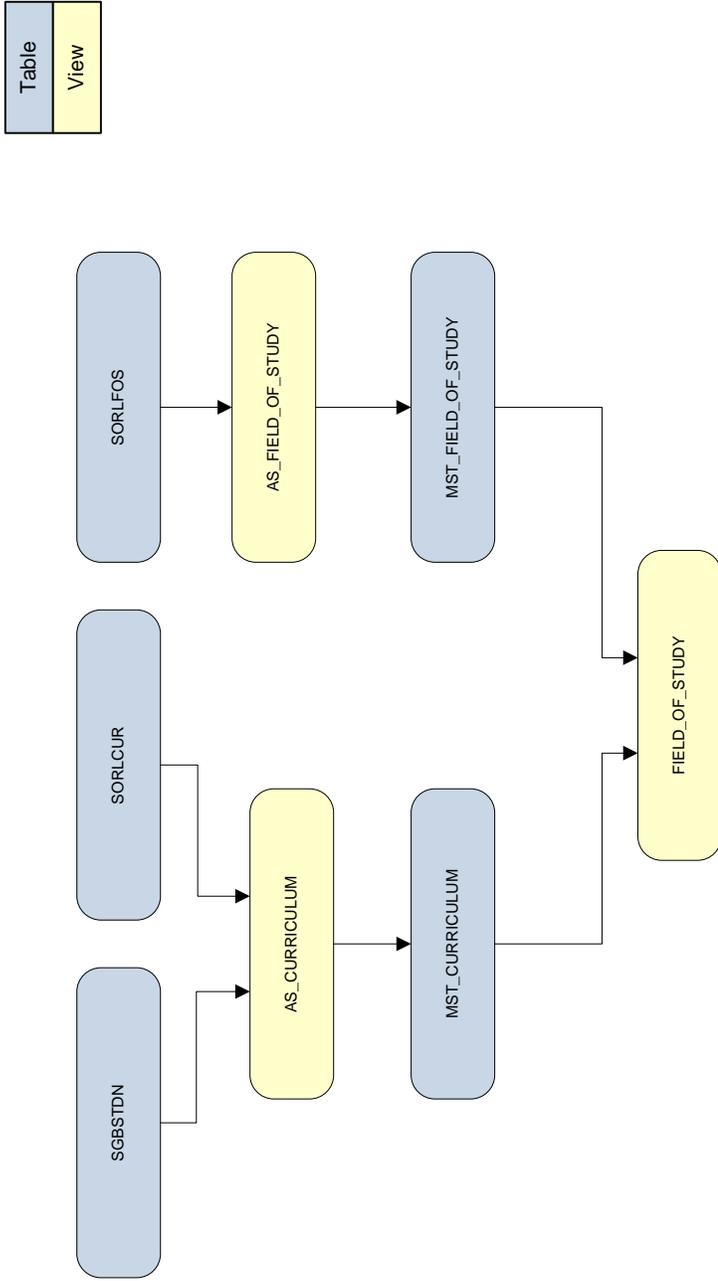
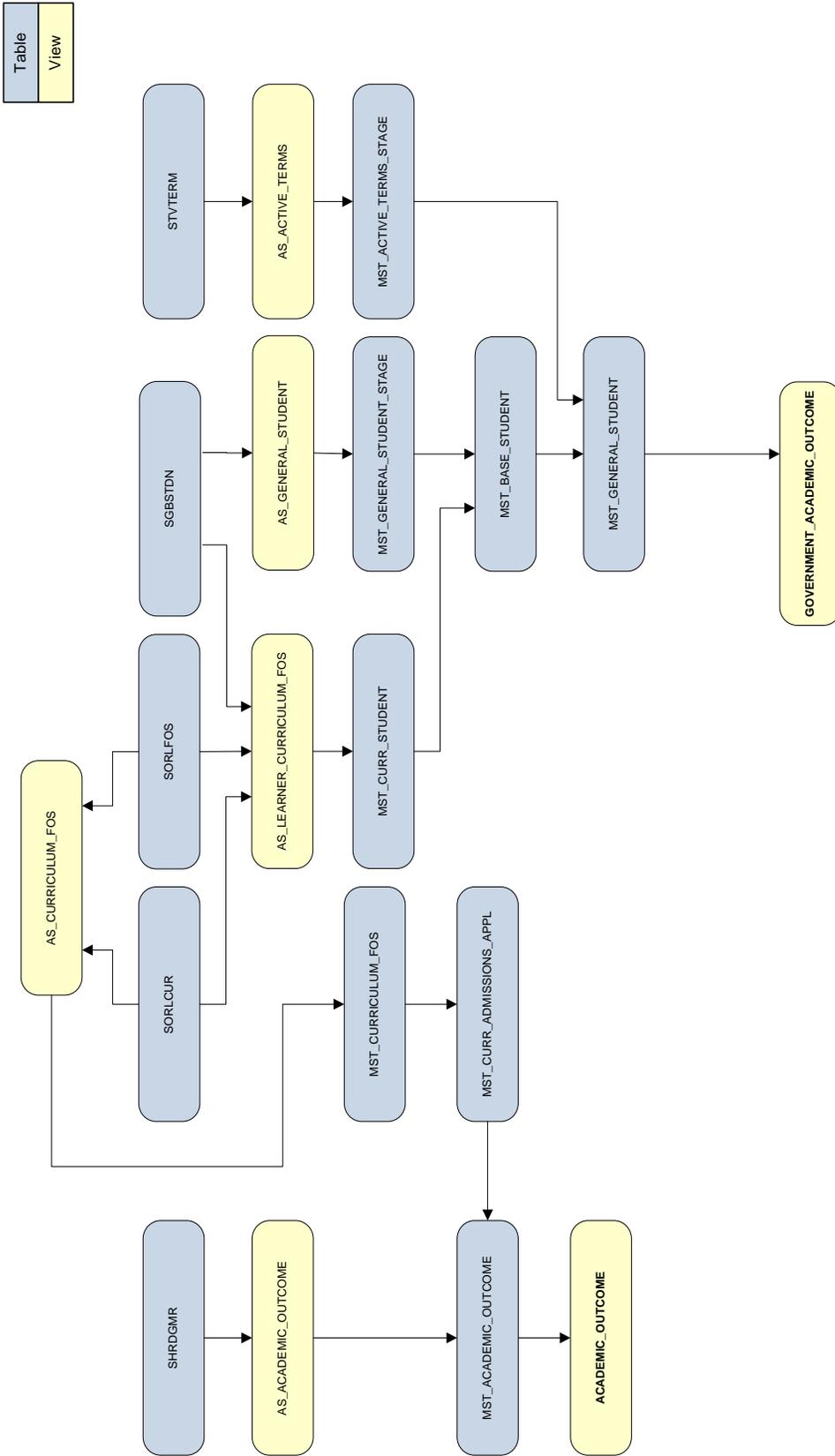


Table
View

**Government Academic Outcome reporting view flow**



**Government Admissions reporting view flow**

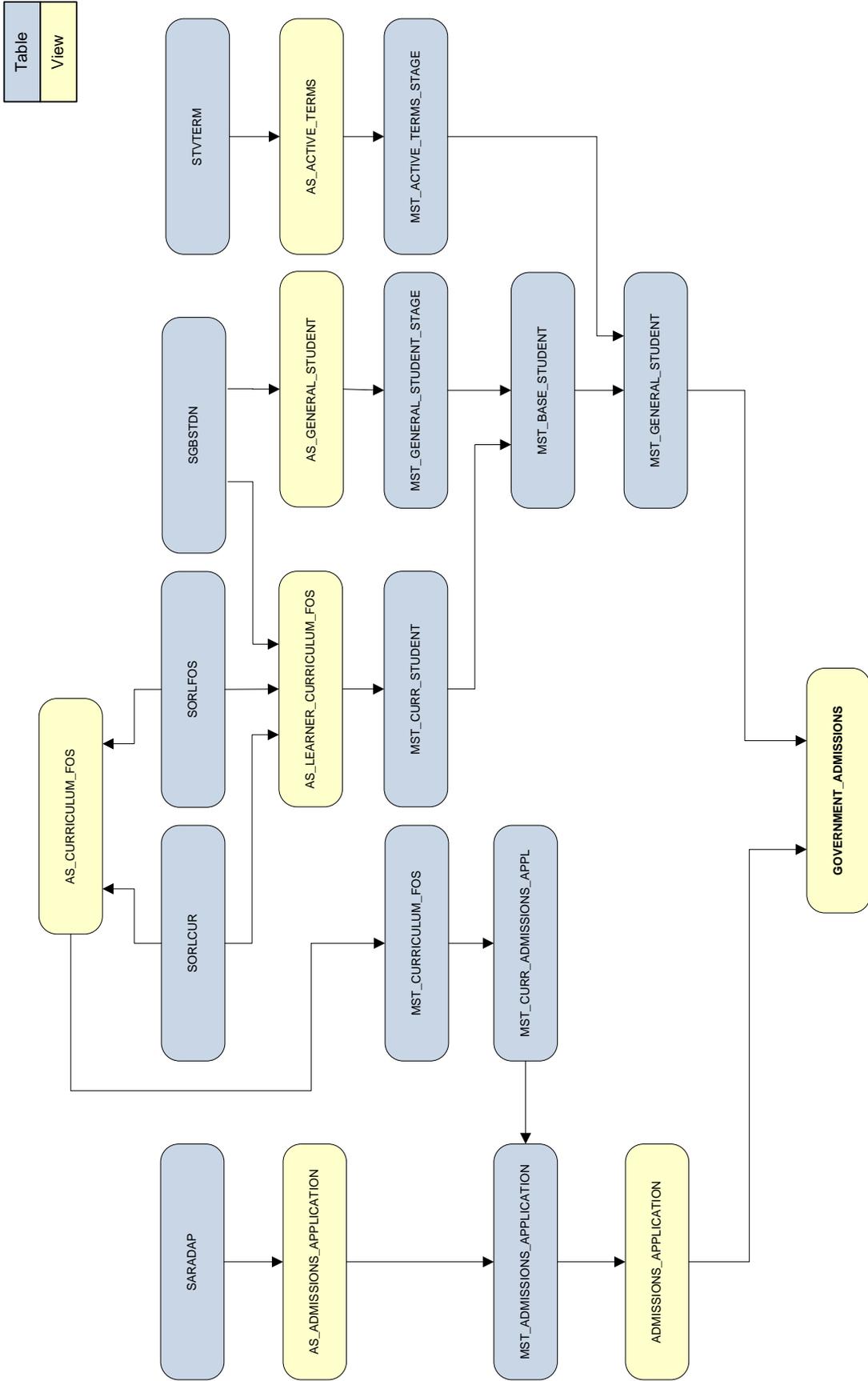
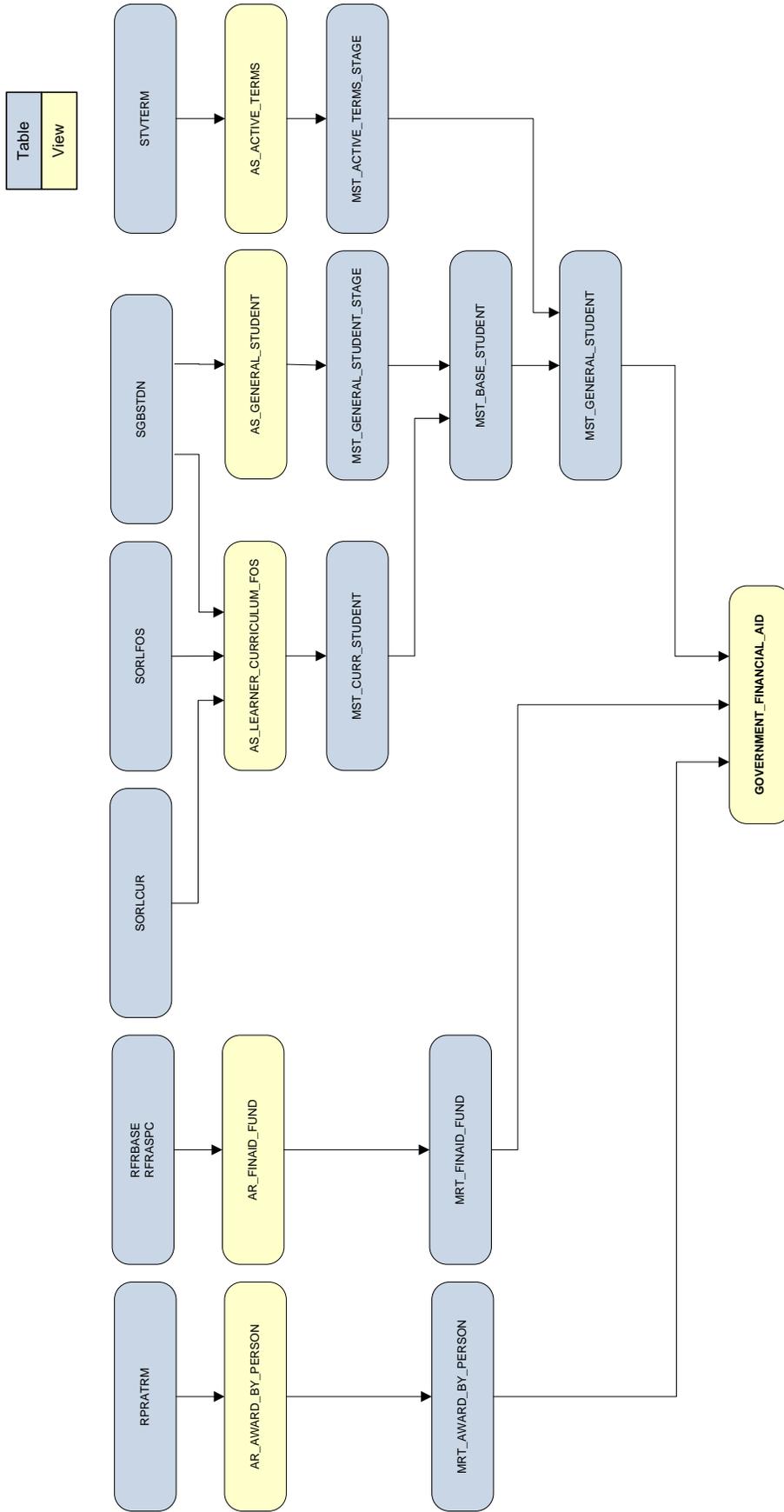
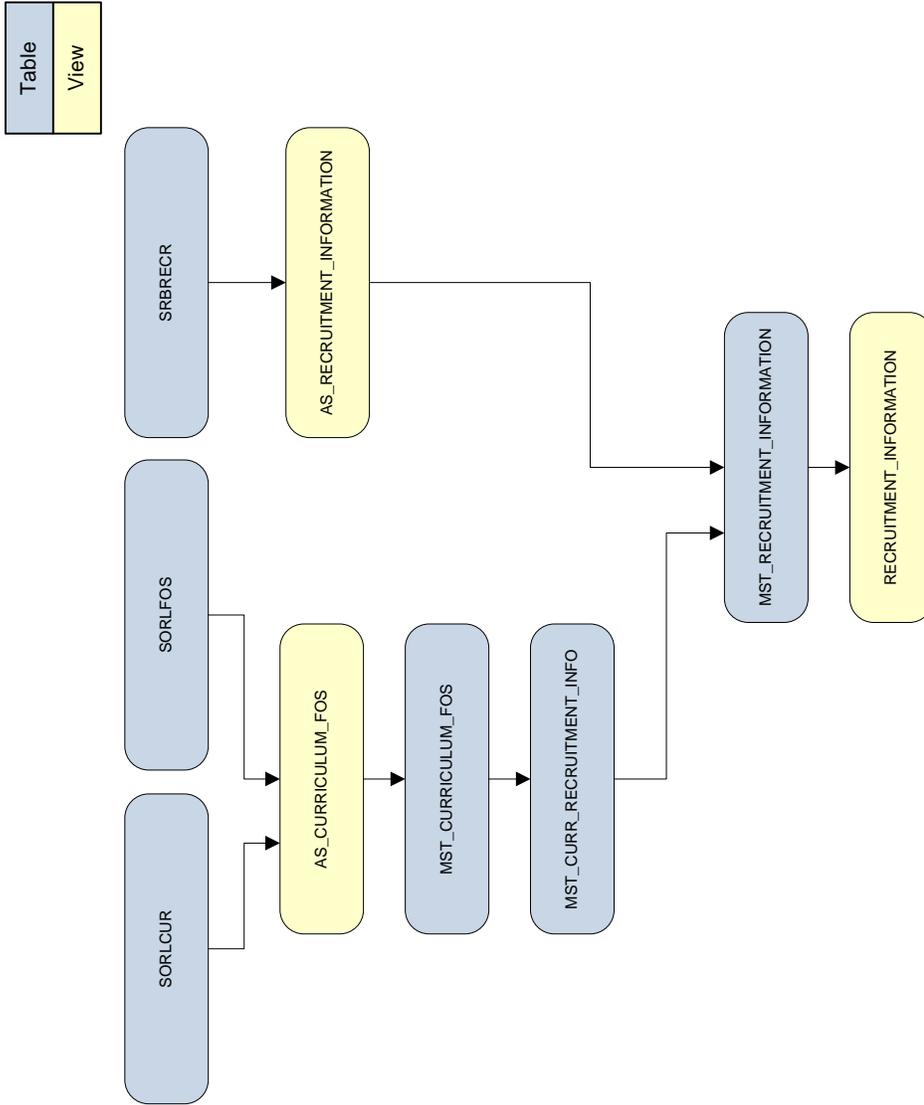


Table
View

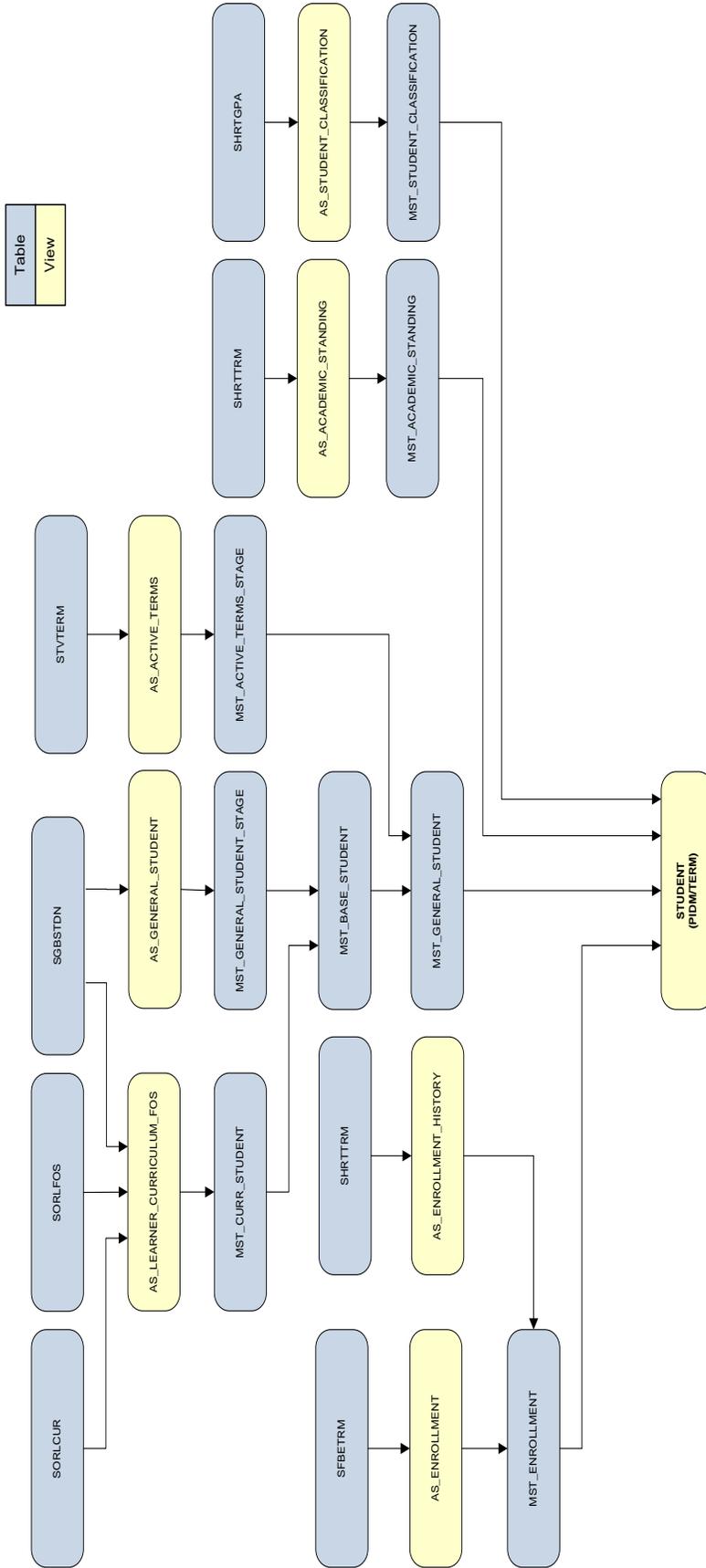
**Government Financial Aid reporting view flow**



*Recruitment Information reporting view flow*



**Student reporting view flow**



## Composite views and meta data

The composite views gather Banner source data necessary to populate and maintain the information stored in the Banner Operational Data Store (Banner ODS). This source information then updates the information that resides in the Banner ODS database.

### Note

Any institution change to a composite view impacts the Banner ODS maintenance processes. ■

The Banner ODS composite view meta data is also available as published meta data. Use the following steps to view published composite view meta data reports using the Administrative UI.

1. Select **Meta Data** from the Administrative menu.
2. Select **Banner Operational Data Store**.
3. Select the **Banner ODS Composite View Meta Data Reports** link located at the top right-hand corner of the page.
4. Select a subject area.

The Composite View Meta Data Reports page opens listing the view name and description.

5. To display the column details associated with the selected composite view, select one of the composite views. A description of each field on the report appears below:

Field	Description
Description	Business description of the composite view target, including the key and frequency of data returned by the view.
Target Column	Name of the column in the composite view target.
Business Definition	Definition of the target column in business terms.
Database Data Type	Used for formatting purposes when writing reports.
Business Data Type	Used to store client-specific data about a given column. This field is empty by default.
Domain Value	Used to store client-specific data about a given column. This field is empty by default.

Field	Description
Source Name	Name of the source table, FUNCTION, CONSTANT, or CALCULATION.
Source Column	Name of the source column from the source table or view, if the source is a table or view; name of the PL/SQL function, if the source is FUNCTION; description of the constant, if the source is CONSTANT; description of the calculation used, if the source is CALCULATION.

## Naming conventions

This section describes the naming conventions and standards applied to scripts and database objects used to create and maintain the BPRA solutions.

### Banner ODS standards (ODSMGR schema)

#### Front-end views: reporting style

##### Object name

Natural language naming conventions are acceptable. Maximum length is 30 characters.

##### *Examples:*

PERSON, STUDENT\_COURSE, CONSTITUENT

##### *Additional detail*

Script names *must* follow unique, 7-character naming standards. The first three characters are System Descriptor, Product ID, and Object\_ID. The next four characters are free form.

## Front-end views: Object:Access style

### Object name

Maximum length is 30 characters. See the table below.

1 <sup>st</sup> Character	A	O:A View Indicator
2 <sup>nd</sup> Character	A - Advancement F - Finance P - Payroll R - Financial Aid S - Student T - Accounts Receivable or Billing Receivable	Product Identifier
3 <sup>rd</sup> Character	_ (underscore)	
5 <sup>th</sup> -30 <sup>th</sup> Characters	Unique Descriptor	

### Examples:

AS\_STUDENT\_DATA, AA\_GIVING

### Additional detail

Script names are the same as the object name.

## Front-end composite tables

### Object name

Maximum length is 30 characters. See the table below.

1 <sup>st</sup> Character	M - Mart	System Descriptor
2 <sup>nd</sup> Character	A - Advancement G - General F - Finance P - Payroll R - Financial Aid S - Student T - Accounts Receivable or Billing Receivable	Product Identifier
3 <sup>rd</sup> Character	T - Table or V - View	Object Identifier
4 <sup>th</sup> Character	_ (underscore)	
5 <sup>th</sup> -30 <sup>th</sup> Characters	Unique Descriptor	

**Examples:**

MAT\_GIFT, MGT\_VALIDATION

**Additional detail**

Script names *must* follow unique 7-character naming standards. The first three characters are System Descriptor, Product ID, and Object ID. The next four characters are free form.

**Indexes**

**Primary key indexes**

**Object name:**

PK\_{table\_name} (For front-end tables, omit the first three identifiers). Maximum length is 30 characters.

**Additional indexes**

**Object name**

Index is either table name or abbreviation suffixed by “\_INDEX\_nn” where nn is a one-up number. Maximum length is 30 characters.

**Administrative standards (IA\_ADMIN schema)**

**Administrative tables**

**Object name**

Maximum length is 30 characters. See the table below.

1 <sup>st</sup> Character	M- Mart	System Descriptor
2 <sup>nd</sup> Character	D - Control Reports G, T - General Purpose	Table Purpose
3 <sup>rd</sup> Character	B - Base R - Repeating T - Temporary V - Validation	Table Type
4 <sup>th</sup> -7 <sup>th</sup> Characters	Unique Descriptor	

**Examples:**

MDBLOGH, MTVPARM

### ***Additional detail***

Script names *must* follow unique 7-character naming standards. Script names are the same as the object name.

## **Administrative packages**

### **Object name**

Maximum length is 30 characters. See the table below.

1 <sup>st</sup> Character	<i>M</i> - Mart	System Descriptor
2 <sup>nd</sup> Character	<i>G</i> - General Purpose	Product Identifier
3 <sup>rd</sup> Character	<i>K</i> - Package	Object Identifier
4 <sup>th</sup> -7 <sup>th</sup> Characters	Unique Descriptor	

### ***Examples:***

MGKSECR, MGKPARM

### ***Additional detail***

Script names *must* follow unique 7-character naming standards. Script names are the same as the object name.

## **Meta data tables and views**

### **Object name**

Maximum length is 30 characters. See the table below.

1 <sup>st</sup> Character	<i>W</i> - Warehouse	System Descriptor
2 <sup>nd</sup> Character	<i>M</i> - Meta Data	Table Purpose
3 <sup>rd</sup> Character	<i>T</i> - Table or <i>V</i> -View	Object Identifier
4 <sup>th</sup> Character	_ (underscore)	
5 <sup>th</sup> -30 <sup>th</sup> Characters	Unique Descriptor	

### ***Examples:***

WMT\_SOURCE, WMV\_TARGET\_OBJECT

### ***Additional detail***

Script names are the same as the object name.

## Sequences

### Object name

See the table below.

1 <sup>st</sup> Character	<i>M</i> - Mart	System Descriptor
2 <sup>nd</sup> Character	<i>G</i> - General Purpose	Product Identifier
3 <sup>rd</sup> Character	<i>S</i> - Sequence	Object Identifier
4 <sup>th</sup> -7 <sup>th</sup> Characters	Unique Descriptor	

### Examples:

MGSHOST, MGSPARM, MGSPIDM, MGSSDAX

### Additional detail

Script names *must* follow unique 7-character naming standards. Script names are the same as the object name.

1 <sup>st</sup> Character	<i>W</i> - Warehouse	System Descriptor
2 <sup>nd</sup> Character	<i>D</i> - Dimension <i>F</i> - Fact	Star Schema Table Type
3 <sup>rd</sup> Character	<i>T</i> - Table <i>Z</i> - Snapshot Table	Object Identifier
4 <sup>th</sup> Characters	_ (underscore)	
5-5 <sup>th</sup> -30 <sup>th</sup> Characters	Unique Descriptor	

1 <sup>st</sup> Character	<i>W</i> - Warehouse	System Descriptor
2 <sup>nd</sup> Character	<i>T</i> - Temporary	Warehouse Table Type
3 <sup>rd</sup> Character	<i>T</i> - Table	Object Identifier
4 <sup>th</sup> Characters	_ (underscore)	
5-5 <sup>th</sup> -30 <sup>th</sup> Characters	Unique Descriptor, ending in any:  _INPUT _CLEAN _ERROR _WKEYS	

1 <sup>st</sup> Character	W - Warehouse	System Descriptor
2 <sup>nd</sup> Character	D - Dimension	Product Identifier
3 <sup>rd</sup> Character	S - Sequence	Object Identifier
4 <sup>th</sup> Characters	_ (underscore)	
5-5 <sup>th</sup> -30 <sup>th</sup> Characters	Unique Descriptor, ending with _SEQ	
1 <sup>st</sup> Character	PK_ (underscore)	Primary Key Prefix
4 <sup>th</sup> -30 <sup>th</sup> Characters		Table Name or Abbreviation (includes the first 4 characters, e.g., WFT_)
1 <sup>st</sup> - 2 <sup>nd</sup> Character	FK	Foreign Key Prefix
3 <sup>rd</sup> Character	n	Where n is a one-up number
4 <sup>th</sup> Characters	_ (underscore)	
5 <sup>th</sup> -30 <sup>th</sup> Characters		Child Table Name (omits the first 4 characters, e.g., WFT_)



# 2 Administrative User Interface

---

The Administrative User Interface (UI) enables you to easily perform the tasks required to set up and maintain the Banner Operational Data Store (Banner ODS) at your institution. Your institution may licence either one or both products. This chapter includes information about using the Administrative UI to maintain both the Banner ODS and Banner EDW. You can reference the information that is appropriate for your institution's environment. Review the map below to become familiar with the location of the options on the Administrative UI menus.

<b>Preferences and Security</b>	<b>Meta Data</b>
Institutional Preferences	Banner Operational Data Store
Set Up Users & PIN	
Set Up Data Display Rules	Maintain Banner ODS Meta Data
Set Up Banner ODS Security Rules	
<b>Options</b>	<b>New Web Tailor Administration</b>
Set Up Parameters	Customize a Web Menu or Procedure
Schedule a Process	Customize a Graphic Element
View Control Reports	Customize a Set of Information Text
View and/or Remove Scheduled Processes	Customize a Set of Menu Items
Freeze Data Maintenance	Update User Roles
	Customize a Web Module
	Customize Web Rules
	Customize Web Tailor Parameters
	Customize a Login Return Location
	Customize Web Tailor Overrides
	Customize Global User Interface Settings
<b>Staging</b>	
Maintain Stage Tables	
Report Staging Area Status	
Refresh Staging Area Status	
Refresh Staging Tables	
Reconcile Stage Tables	

---

There are a number of tasks involved in setting up and maintaining Banner ODS. Some tasks are performed one time when you initially install and implement Banner ODS. Other tasks are performed during implementation and on an ongoing basis. Each task is listed below, and is described in detail in later sections of this guide.

- Set up institutional preferences
- [“Set up Users and PINS”](#) on page [2-3](#)
- [“Data Display Rules”](#) on page [2-6](#)
- [“Set up Fine-Grained Access Security”](#) on page [2-17](#)
- [“Set up and Synchronize Data”](#) on page [2-53](#)
- [“Set up Parameters”](#) on page [2-54](#)
- [“Schedule a Process”](#) on page [2-60](#)
- [“Freeze Data Maintenance”](#) on page [2-125](#)
- Review and maintain [“Meta Data”](#) on page [2-137](#)

You can also use Web Tailor to perform some security functions and set some security-related preferences. In addition, Web Tailor gives you some options for customizing the appearance and behavior. For more information on using Web Tailor, see the “Web Tailor User Guide.”

 **Warning**

Because Banner ODS contains sensitive business information, you should take standard precautions to prevent unauthorized access. User IDs and PINs should, of course, be kept secret, since anyone with a valid ID and PIN, and URL, can gain access to the system. ■

This section outlines all the tasks, and offers suggestions about when you want to perform them.

# Set up Users and PINS

---

Anybody using a reporting tool to access Banner ODS must be defined as an Oracle User. Use your normal Oracle processes procedures to create user accounts.

After a user account is created, that user can report against Banner ODS. Each user is listed in the Administrative UI on the View Banner ODS Business Profile and User Associations page. From that page, you can assign security rules for each user using a Business Profile. See [“Set up Fine-Grained Access Security” on page 2-17](#) for more information.

You should set up user accounts for Banner ODS users at your institution based on how each user needs to use Banner ODS. Banner ODS includes two types of users:

- Administrative Users—who require a user account so they can use the Administrative UI to set up and maintain Banner ODS.
- Oracle Users—who require an Oracle user account (set up in your source system) so that they can use a reporting tool to access Banner ODS and build reports.

Some users may be both Administrative and Oracle users, in which case they need a user account of both types. In these cases, you can use the same user ID in both systems (Administrative UI and Oracle).

PINs are disabled if the number of login attempts is exceeded (set on Web Tailor “Customize Web Rules” screen). They can be easily enabled on the Update a User Account screen using this checkbox.

## Create Users and PINS

Administrative UI users set up and maintain Banner ODS at your institution. Each Administrative user must have a unique ID and PIN created for them in order to gain access to the Administrative UI.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set Up Users & PIN**.
3. Click **Create a New User Account** from the Set Up User and PIN page. The Create a New User Account page opens.
4. Enter a User ID.

A User ID can be one to nine characters, is limited to numbers and upper case letters, and may not contain spaces. (If you enter lower case letters, they will be converted to upper case letters.)

5. Enter First, Middle, and Last Names (only Last Name is required.)
6. Enter a PIN (It must be exactly six numbers; it cannot contain letters or special characters.)
7. Indicate whether the PIN is enabled or disabled.
8. Click **Create**.

## Update Existing Users

Use this option to update misspelled or changed names, or to enable or disable a PIN.

If a user's login attempts are exceeded (as set up in Web Tailor, Customize Web Rules page), their PIN is disabled. Use this page to enable their PIN.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set Up Users & PIN**.
3. Click an entry from the **Name** column on the Set Up Users and PIN page.
4. Change the fields. Only the Last Name field is required.

### **Note**

The PIN must be exactly six numbers, and cannot contain letters or special characters. ■

5. Click **Update** to save. Or, click **Delete** to remove the User Account.

## Update User Roles

User roles define which tabs of the Administrative UI a user can access. In turn, the roles permit or restrict a user to perform various tasks within the Administrative UI. When you create an Administrative user, the user is assigned the following user roles in Web Tailor: BPR Meta Data, BPR Options, BPR Security, and Web Tailor Administration. This gives the user access to all options within both Banner ODS, except for Banner ODS Staging, and New Web Tailor Administration menus. You may want to change a user's access, for example, to disable a user's ability to change security settings.

Perform the following steps within the Administrative UI to assign roles to a user by changing the user's defined roles in WebTailor.

1. Click **New Web Tailor Administration** from the Administrative UI menu.
2. Click **Update User Roles**.

3. Enter or select the **User ID** to which you want to assign roles.
4. Click **Submit**.
5. Check which roles to assign to the user. Refer to the [“User Roles”](#) descriptions to determine which roles to assign to each user.
6. Click **Submit Changes**.

## User Roles

The following table defines each of the User Roles available within the Administrative User Interface.

User Role	Description
BPRA Meta Data	<p>Allows a user to access the Meta Data tab within the Administrative User Interface where the user can:</p> <ul style="list-style-type: none"> <li>• View the published Banner ODS meta data.</li> <li>• Update, edit, and publish the Banner ODS meta data.</li> <li>• View and recover baseline records marked for deletion.</li> </ul>
BPRA Options (Jobs, Parameters, FreezeData)	<p>Allows a user to access the Options tab within the Administrative User Interface where the user can:</p> <ul style="list-style-type: none"> <li>• Define internal system parameters.</li> <li>• Run jobs to load and update data in the Banner ODS.</li> <li>• Manage aspects of freezing data in the Banner ODS.</li> <li>• Define and maintain data cleansing translation rules.</li> <li>•</li> </ul>
BPRA Security	<p>Allows a user to access the Preferences &amp; Security tab within the Administrative User Interface where the user can:</p> <ul style="list-style-type: none"> <li>• Set institutional preferences.</li> <li>• Create and delete users and reset user’s PINs.</li> <li>• Create and maintain data display rules</li> <li>• Create and maintain fine-grained access security rules defining which values a user can view in the Banner ODS.</li> </ul>

User Role	Description
BPRA Staging	Allows a user to access the Staging tab within the Administrative User Interface where the user can: <ul style="list-style-type: none"> <li>• Add and remove staging tables in the Banner ODS.</li> <li>• Run the Staging Area Status report and view a list of stage tables, the status of processes, and details of staging errors.</li> </ul>
Web Tailor Administrator	Allows a user to access the Web Tailor Administration tab within the Administrative User Interface where the user can customize various aspects of the Administrative Tool.

## Data Display Rules

Display rules enable you to control and customize how data stored in the Banner ODS composite tables is displayed in the reporting views. There are two types of display rules: positional rules and hierarchical rules.

There are also a number of display rules used to determine a value stored in either a Banner ODS composite table or displayed in a Banner ODS Reporting view. All display rules are stored in Banner ODS database table - MGRSDAX.

### Positional display rules

Positional display rules define the specific location (by position) of data in a view. Slotted views or tables require a set of positional display rules to store information in a meaningful way.

#### *Example - Positional Display Rules (for Slotted views)*

The TEST view in Banner ODS displays all valid test score values loaded from your source system to Banner ODS. This data is stored in a vertical presentation as “one row per person per test”. The corresponding TEST\_SLOT view provides an alternative horizontal presentation, that ‘flattens’ the data to “one row per person with the details of (up to) seven test scores.” Positional display rules are required to define which seven test scores will be included, and in what position or order they will appear within this “slotted” presentation. These Display rules are used to build the underlying MST\_TEST\_SLOT table.

### Hierarchical display rules

Hierarchical display rules define a specific order in which to retrieve a set of related data. Hierarchical display rules are required for a subset of (non-slotted) Reporting views.

### **Example – Hierarchical Display Rules (for applicable non-slotted views)**

The PERSON\_ADDRESS and ADDRESS\_BY\_RULE view displays one address per entity per ADDRESS\_RULE (stored in MGRSDAX as an Internal Code under the Internal Group of ADDRESS, and must end in ADDR) to be used for mailing purposes.

The mailing address displayed is based on the hierarchical display rules created to determine which address types should be retrieved for the mailing address. You can create a series of hierarchical display rules based on priority, so that if “address type 1” does not exist, get “address type 2” and so on.

To invoke the ADDRESS\_BY\_RULE reporting view rule, add a Filter/WHERE clause that states “where ADDRESS\_RULE = IC\_REG\_ADDR.” This will retrieve the first current address found in the source system for the hierarchy you created.

When Banner ODS is first installed, MGRSDAX (Banner ODS table that stores display rules) is populated with specific rules from your source system, as well as rules delivered with the product. The records (or display rules) in MGRSDAX match external codes (institution specific values) with internal codes (system defined values). After Banner ODS system is installed, you must then use the Administrative UI Preferences and Security option and Set Up a Display Rule to review and update the display rules in MGRSDAX. This ensures that display rules match your criteria, and are set up to meet your reporting needs.

Multiple display rules can also be managed, or assigned, using business profiles. (See [“Set up a Display Rule” on page 2-12](#) for information on setting up business profiles.)

#### **Note**

Business profiles are only used when more than one Oracle user is used to access the data from your institution supported report writer. ■

If business profiles are used, then the system pulls the appropriate values for the profile with which the user is associated, if a rule exists for that profile.

#### **Warning**

If multiple profiles exist for that user, then the first profile with a matching display rule is used. ■

#### **Note**

If no display rules are found for any profiles assigned to the user, the display rule for the default profile (INSTITUTION) is used. ■

For reporting views such as the TEST\_SLOT view, use business profiles to designate unique sets of test score data and the positional order of that data within the view for different business offices and users at the institution.

For hierarchical reporting views such as the PERSON\_ADDRESS view, business profiles enable you to designate unique sets of (mailing) address type hierarchies for different business offices and users.

**Example:**

A display rule consists of one or more related records in MGRSDAX. Records that share the same Profile Code, Internal Group and Internal Code values make up a single display rule. The display rule also includes the Business PROFILE\_CODE that defaults to INSTITUTION or is set to an institution defined value.

MGRSDAX is delivered with the following records that all have an Internal Group value of ADDRESS, and the business profile of INSTITUTION.

Internal Group: ADDRESS

Profile Code	Internal Code	Internal Code Sequence	External Code
INSTITUTION	ALUMMAIL	1	BUS
INSTITUTION	ALUMMAIL	2	ART
INSTITUTION	ALUMMAIL	3	RES
INSTITUTION	ALUMMAIL	4	CPS
INSTITUTION	RECRLETR	1	ACCEPT
INSTITUTION	RECRLETR	2	CHKL
INSTITUTION	RECRLETR	3	COLLEGE NIGHT
INSTITUTION	RECRLETR	4	DCSN
INSTITUTION	RECRLETR	5	INTERVIEW ONE

The first four records also share the same Internal Code value of ALUMMAIL. These four records make up the Display Rule that defines which Mail codes to retrieve for Advancement-related reporting views. The last five records share the Internal Code value RECRLETR. These five records make up the display rule that defines which MAIL internal codes to retrieve for the COMMUNICATION\_SLOT and Recruiting-related reporting views.

By editing the above values to reflect the Advancement and Recruiting Mail internal code values used by your institution, your users can then report on the desired data. Before your users begin creating reports, you need to review all of the delivered display rules, and edit them to reflect your institution's specific values.

 **Note**

After changing display rules for views that work from slotted database tables, the corresponding slotted tables must be reloaded before the updated values will display in the reporting views seen by your users. By default, this happens during the incremental refresh cycle, which typically occurs nightly. However, if you want to see more immediate results, reload the corresponding slotted table(s) manually via the Schedule a

Process page. See [“Schedule a Single Process” on page 2-63](#) ■

 **Note**

Also note that there are few reporting views, like the PERSON\_ADDRESS and ADDRESS\_BY\_RULE, that go directly against the rules in the MGRSDAX database table and do not need to be reloaded for you to view the changes. ■

## Display Rule Information in Published Meta Data

Meta Data includes a business definition for each reporting view. When the reporting view being defined uses display rule entries from Banner ODS MGRSDAX database table, the required rule code, INTERNAL\_GROUP and INTERNAL\_CODE values are explained as part of the business definition. Most reporting views that require MGRSDAX rules have a column labeled PROFILE\_CODE, and a column with the name of the view and XXXXXXXX\_RULE that are used as the INTERNAL\_GROUP for that set of display values.

When the reporting view has a column that uses the MGRSDAX database table, that is explained in the column business definition.

## Display Rule Cross-Reference Chart

Display rules are defined by a set of records stored in Banner ODS database table, MGRSDAX. You can use the Display Rule Cross-Reference Chart to identify display rule value combinations as they are delivered.

The Display Rule Cross-Reference Chart lists all views, tables, procedures or packages that use the MGRSDAX table. The chart enables you to see the rule values that are set up to retrieve the data, and how your solution is impacted if changes are made to the display rules on MGRSDAX. The codes on the chart followed by an asterisk (\*) indicate user defined rules that can be changed to fetch the EXTERNAL\_CODE or REPORTING\_DATE.

To open the Display Rule Cross-Reference Chart, access the displayRulesXREF.csv file delivered with the product documentation.

You can open the file in Microsoft Excel or a similar spreadsheet application. You can reorganize the columns as needed. A description of each column on the chart follows.

Column	Description
REPORTING_VIEWS	The view that is directly affected by a change to an MGRSDAX value in Banner ODS.
INTERNAL_GROUP	Value Banner ODS is using to connect the set of display rules with the reporting view and/or column that are to use them. These values are coded within Banner ODS and must be used for the purpose specified.
INTERNAL_CODE	Institutions may define any values as required to represent the business rules of the institution. Some values are extracted from Banner GTVSDAX rules for institutions that use the O:A views.
EXTERNAL_CODE	X identifies valid institutions values must be provided.
REPORTING_DATE	X indicates that the Reporting Date is used for sequence of display values.
TABLES	Banner ODS Composite table used as the basis for the selection of values based on the display rules defined by the institution on the MGRSDAX database table.

Column	Description
COMPARISON COLUMN	Column within Banner ODS composite table that is used to retrieve data based on the value in the either MGRSDAX_EXTERNAL_CODE or MGRSDAX_REPORTING_DATE.
	<p><b>Example:</b></p> <p>TEST rule: The MSKTEST package gets the MGRSDAX_EXTERNAL_CODE value from MGRSDAX based on the MGRSDAX_INTERNAL_GROUP = 'TEST' and the MGRSDAX_INTERNAL_CODE = 'STDNTEST'. This value is then used to retrieve records from the TEST column in MST_TEST to populate the MST_TEST_SLOT table and the TEST_SLOT view.</p>
REPORTING	<p>Package View Name of the package or view in which MGRSDAX is referenced.</p> <p>Procedure/Function Name of the process or function being used by MGRSDAX.</p>

Records with the same Profile Code, Internal Group and Internal Code combination make up one display rule. The display rules that are delivered have a default business profile code of INSTITUTION.

 **Note**

When more than one Internal Code is listed, there are multiple display rules for the value in the Internal Group. For example, there are several ADDRESS rules listed for different departments like Admissions (Internal Code = ADMSADDR), Faculty (Internal Code = FACLADDR), Recruiting (Internal Code = RECRADDR), etc. ■

 **Note**

When more than one Profile Code is listed, there are multiple display rules for the value in the Internal Group. ■

The example below will help you tie together one use of the chart with the Administrative UI.

### **Example:**

You want to see what display rules exist for (or are impacted by) the **VENDOR** reporting view because you want to change the external code for that reporting view. Follow the steps below:

1. The copy of the chart is already sorted in alphabetical order by Reporting View. Look in the Reporting View column (the first column) of the chart. Find **VENDOR**. It is near the end of the list.

You will find that the assigned Profile Code is **INSTITUTION**, the Internal Group is **ADDRESS** and the Internal Code is **VENDADDR** for **VENDOR**.

2. Open the Set Up a Display Rule web page in your Administrative UI.
3. Select the Profile Code (**INSTITUTION**), Internal Group (**ADDRESS**) and Internal Code (**VENDADDR**) from the drop-down lists.
4. Click **Search**. The Select an Existing Display Rule page opens. This page shows the display rule for the reporting view **VENDOR**.
5. To change the External Code, click **BU** under the External Code column.

The Update an Existing Display Rule page opens. You can change the external code from this page.

6. Click **Save**.

## **Set up a Display Rule**

You may want to create new display rules by adding new internal codes for a business purpose, or by adding additional external codes not currently defined.

### **Note**

You may want to set up your business profiles before you set up display rules. ■

To create a new rule, follow the steps below:

1. From the Administrative menu, Click **Preferences & Security**.
2. Click **Set Up Data Display Rules**.

### **Note**

If a **PROFILE\_CODE** is to be used in the display rule, it must be set up first. See [“Set up Fine-Grained Access Security” on page 2-17](#) for information on setting up business profiles. ■

3. Click **Create** from the Set Up a Display Rule page.

4. Enter the information for the new display rule, or click an existing code from one of the drop-down lists. Each field is described below:

Field	Description
Business Profile Code	<p>Business Profile for which you want to set up display rules.</p> <p>You can create additional Business Profiles from the Create a Banner ODS Business Profile web page. <i>INSTITUTION</i> is the default code for users for whom no other business profile is defined.</p>
Internal Group	<p>High-level group of rows of data (Internal Codes) that are categorized together to provide multiple entries for a single concept. The value is predefined in the system. It should <i>not</i> be changed, but new internal groups can be added for client specific processing. (Click the appropriate value from the Internal Group list.)</p>
Internal Code 1	<p>Specific code relationships for source system concepts. This field is used internally within PL/SQL functions and procedures to determine which row(s) to retrieve from the MGRSDAX table.</p> <p>You can add new internal codes to be used for business purposes, and then click the appropriate code when writing a report. (Click the appropriate value from the Internal Code list.)</p>

Field	Description
Internal Code Sequence Number	<p>Internal sequence number that provides either a hierarchy or positional identifier:</p> <p><b><i>Hierarchy identifier</i></b></p> <p>Order in which to retrieve display rule driven data (such as the Address Type for a designated mailing address in the PERSON_ADDRESS view). Each sequence number should be a single numeric value. Give the most important code value a sequence number of 1, and number each subsequent value consecutively (such as 2, 3, 4).</p> <p><b><i>Positional identifier</i></b></p> <p>Position within a slotted view where a repeating group should appear. The sequence number entered by the user must correspond to the slotting concept applicable to the slotted view for which the display rule is being created (such sequence numbers 1-7 for the seven available test score slots in the TEST_SLOT view).</p>
External Code	<p>Institution-specific values that usually come from rules and validation tables in the transactional or administrative source system. Enter the values used by your institution to define either the hierarchy or the positional value for the particular display rule.</p> <p><b>Note:</b> Change this value so that the external codes match your institution's code value for each display rule value.</p>

- Click **Save**. The Update an Existing Display Rule page opens.

 **Note**

After changing display rules the corresponding slotted tables must be reloaded for those changes to take effect. By default, this happens during the incremental refresh cycle, which typically occurs nightly. However, if you want to see more immediate results, reload the corresponding slotted table(s) manually via the Schedule a Process page. See [“Schedule a Single Process” on page 2-63](#) ■

## Update Display Rules

You may want to display different types of test scores, address information, etc. If the display rule already exists, then you can use the steps below to add, update, duplicate or delete display rules.

### Note

You can use these steps for every Profile Code, Internal Group and Internal Code combination listed in the table in the [“Display Rule Cross-Reference Chart”](#) section. ■

1. From the Administrative menu, click **Preferences & Security**.
2. Click **Set Up Data Display Rules**. The Set Up a Display Rule page opens.
3. Choose a Business Profile, Internal Group, and Internal Code combination from the drop-down lists on the Set Up a Display Rule page. Or, you can show all groups and codes.
4. Click **Search**. The Select an Existing Display Rule page opens.

### Note

Use the Meta Data reporting view business definition and the Display Rule Cross Reference chart, available from the **Help** button in the Administration UI, to identify Internal Group and Internal Code combinations that make up a Display Rule. Information about this chart is available in the [“Display Rule Cross-Reference Chart”](#) section. ■

5. Review all information for the selected combination. Determine the data on which your users want to report (it may be different from what is delivered). Create a list of the data you want to use in place of the data that was delivered.
6. Choose an External Code link from the External Code column to edit a record. The Update an Existing Display Rule page opens.
7. Make your change
8. Click **Save** to save the display rule. Click **Delete** to remove the display rule.

### Note

After changing display rules, the corresponding slotted tables must be reloaded for those changes to take effect. By default, this happens during the incremental refresh cycle, which typically occurs nightly. However, if you want to see more immediate results, reload the corresponding slotted table(s) manually via the Schedule a Process page. See [“Schedule a Single Process” on page 2-63](#). ■

## Duplicate Display Rules

To save time, you can copy the settings from an existing display rule and use it to create a new display rule.

1. From the Administrative menu, Click **Preferences & Security**.
2. Click **Set Up Data Display Rules**. The Set Up a Display Rule page opens.
3. Choose a Business Profile, Internal Group and Internal Code combination from the drop-down lists on the Set Up a Display Rule page. Or, choose to show all groups and codes.
4. Click **Search**. The Select an Existing Display Rule page opens.
5. Choose an external code link from the External Code column. The Update an Existing Display Rule page opens.
6. Enter the External Code information or select it from the drop-down list.
7. Click the **Duplicate**. The Create a New Display Rule page opens.
8. Replace the information for the existing display rule with the information for the new display rule.
9. Click **Save** to save your settings.

### **Note**

After changing display rules, the corresponding slotted tables must be reloaded for those changes to take effect. By default, this happens during the incremental refresh cycle, which typically occurs nightly. However, if you want to see more immediate results, reload the corresponding slotted table(s) manually via the Schedule a Process page. See [“Schedule a Single Process” on page 2-63](#). ■

## Reload using a Single Extract Transform and Load (ETL) Slot Process

Changes made to a display rule affect all associated slotted tables and reporting views. The ETL slot process must be rerun before any changes made to slotted tables or display rules can be viewed in the slotted reporting views. If only one slotted table was changed, then this process enables you to quickly run a single slot process. Use the following steps to schedule when you want to run a slot process job.

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**. The Select a Process page opens.

3. Click **Schedule Banner ODS Mappings**. The Select a Subprocess page opens.
4. Click **Run A Single ETL Slot Package** from the Select a Subprocess page. The Schedule a Process page opens.
5. Choose the table from the Slotted Table to Reload drop-down list.
6. Enter the required Scheduling Parameters information.
  - 6.1. Enter a Run Date (format dd-mon-yyyy) and Runtime (format hh24:mi:ss).
  - 6.2. If you want to run the process on a recurring basis, enter an Interval.

Click the link next to the Interval field. A sample Interval window opens. Click the link under the Interval Expression column for the interval in which you want to schedule a process. For example, to run a process every day at the same time click SYSDATE+1.
7. Click **Save** to save the information about this job. The job is entered into the job queue to run at the specified day and time.

## Set up Fine-Grained Access Security

---

Banner ODS includes two types of users:

- Oracle users who require an Oracle user account in your source system so they can access Banner ODS to build reports
- Administrative users who require a user account in the Administrative UI so they can use the UI to maintain Banner ODS.

This section explains how fine-grained access security applies to the first type of users - Oracle users when they access Banner ODS for reporting.

Fine-grained access security lets you selectively restrict an Oracle user's access to rows of Banner ODS data based on the value of a specific data element. For example, you might allow a user to see data only from their own department. After you set up security rules and assign them to Oracle users, the rules are applied when the user searches for information within Banner ODS.

### Note

This security applies to the rows of data returned, not the columns. To 'mask' columns of data for a given reporting view, create a copy of the view with those columns removed that contain sensitive data. ■

Secured access to data is controlled by Oracle Policies, in conjunction with the security rules set up in the Administrative UI. A policy is an Oracle construct that applies a

WHERE clause predicate to any queries made against a table. A security rule is simply data in Banner ODS security tables that determine what that WHERE predicate should look like for a given user.

By default, Banner ODS is delivered with no Policies (no security restrictions) on any tables. Therefore, you can set up data access values (security rules) for given users without affecting any user's ability to access Banner ODS data. However, once Policies are defined for the tables, users can only access data to which they have been granted permission.

Once a policy is set up on a Banner ODS table, Oracle calls the MGKSECR package to create a WHERE clause predicate any time that the database table is accessed, such as using a SELECT query. The MGKSECR package, in turn, uses the security rules data to generate the appropriate WHERE clause predicate for the current Oracle User ID. For users with access set to "all" (either All Banner ODS Data, All Data for that Area, or All Data for all columns and rules in the table), MGKSECR does not generate a predicate, thereby allowing those users full access to that data. For rules that list access to particular values, for example campus codes of A, B, or C, MGKSECR generates a corresponding WHERE clause code with the appropriate level of restriction.

 **Note**

Security rules are cumulative -- they are joined with an AND clause. Users must be granted access rights for each rule in a table in order to gain access. For example, if a table has three security rules defined, and two of the rules give *all* access, but the third rule gives the user access to *none*, that user will not have access to any data in that table. ■

You can manage users by grouping similar users together as business profiles. You can also manage Security and Display Rule assignments as a group rather than at the individual user account level.

Use Banner ODS menu selections in the order below to set up your security:

**1. Set up Organizational Areas.**

Set up one or many organizational areas by grouping similar areas together. See [“Set up and Maintain Organizational Areas” on page 2-19](#) for additional information.

**2. Set up and Maintain User ID Translations, and Set up Business Profiles**

These menu options can be completed in any order.

- User ID Translations

Bring Banner data into Banner ODS fine-grained access. See [“Banner User ID Translations” on page 2-21](#) for additional information.

- Business Profiles

Group similar users together. See [“Set up Business Profiles” on page 2-24](#) for additional information.

### 3. Security Rules

Defines the data that each user can access. See [“Set up and Maintain Security Rules” on page 2-27](#) for additional information.

### 4. (optional) Security Predicates

Review the code that generates the predicate in MGKSECR to determine if it is aligned correctly with your business rules. Also, verify the code that is generated for a security predicate. See [“Security Predicates” on page 2-47](#) for additional information.

### 5. Assign Security Rules

Enables security rules to work. Policies are either turned on or turned off. See [“Policy Management” on page 2-48](#) for additional information.

### 6. Transfer Banner Fine-Grained Access process.

This process transfers data for Finance Fund, Fund Type, and Organizations, and for Human Resources Organizations, and Employee Class from Banner to Banner ODS. To transfer additional data you need to set up additional rules. See [“Transfer Banner Fine-Grained Access” on page 2-99](#) for additional information.

## Set up and Maintain Organizational Areas

Organizational Areas are used to set up and group organizational areas together, and to help simplify the implementation of Banner ODS fine-grained access.

### *Example*

If you have users in the Human Resources area that should have access to all of the Human Resources tables. Instead of granting access for each user to each individual Human Resources table, you can define an Organizational Area called “HR” (the name is user-defined). Then, when you create your Banner ODS Security Rules for Human Resources tables, assign those rules to the “HR” Organizational Area. Once your Organizational Areas and Human Resources Security Rules are created, go to the Assign Security Rules page. Select your Human Resources users then, check the **Access All Data In This Area** check box. This gives the user access to all tables included in the “HR” Organizational Area.

Organizational Areas can be set up in any manner you wish. In the example above, an Organizational Area was created which included all Employee tables. However, you could also set up Organizational Areas that cross Banner product groups or you could set up Organizational Areas that are subsets of a product group. The idea is that you can set up Organizational Areas in any way that makes sense for the way you divide security among your reporting users.

Banner ODS is delivered with sample Organizational Areas and sample Security Rules that are assigned to them. The sample data gives an idea of how to go about setting up your own Organizational Areas and the Security Rules that apply to them.

## Create a Banner ODS Organizational Area

Use to create organizational areas.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set up and Maintain Organizational Areas**.
4. Click **Create a Banner ODS Organizational Area**.
5. Enter the code and description.
6. Click **Save**.

## Update a Banner ODS Organizational Area

Use to update organizational areas.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set up and Maintain Organizational Areas**.
4. Click an organizational area code description.
5. Select another organizational code, or change the current description.

### **Note**

The table at the bottom of the page indicates what rows in that table will be deleted if you delete the organizational area. ■

6. Click **Save**.

## Delete a Banner ODS Organizational Area

Use to delete organizational areas.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.

3. Click **Set up and Maintain Organizational Areas**.
4. Click an organizational area code description.
5. Select another organizational code, or change the current description.

 **Note**

The table at the bottom of the page indicates what rows in that table will be deleted if you delete the organizational area. ■

6. Click **Save**.

## Banner User ID Translations

Use Banner User ID Translations to match Banner security user IDs with Banner ODS security IDs if they are different and you plan to run the Transfer Banner Fine-Grained Access process.

The MGBXWLK table (owned by the IA\_ADMIN schema and set up through the Banner User ID Translations pages) is used to associate the two IDs. MGBXWLK contains two primary columns: the Banner User ID and the Banner ODS User ID. The Banner ODS User ID is not required, therefore you can transfer all Banner User IDs into the MGBXWLK table without triggering constraint errors. MGBXWLK has two primary purposes:

- Facilitate data transfer when user IDs are not the same
- Additional security. You may not want everyone with fine-grained access information in Banner to be able to access the data in Banner ODS. In that case, you would follow the instructions in [“- Restrict the Information Transferred to a Limited Group of Users” on page 2-22](#). Only those users whose user IDs were added to MGBXWLK are able to access Banner ODS data after all the fine-grained access policies are enabled.

The MGBXWLK table is populated based on the scenarios below.

**- Banner User IDs are the same as the Banner ODS User IDs**

- The MGBXWLK table does not need to be populated
- The delivered Administrative parameter record with internal group *BANNER TO ODS FGA TRANSFER* and internal code *ODS USER ID NOT FOUND* is used to tell the transfer job what to do when a given Banner ODS user ID is not found in MGBXWLK. As delivered the value of External Code is *USE BANNER USER ID*.

**- Some User IDs are the Same in Banner and Banner ODS, and Some are Not**

- Enter only users with different Banner and Banner ODS user IDs into the MGBXWLK table (using the Set Up and Maintain Banner User ID Translations pages). Users with the same user ID in Banner and Banner ODS can be omitted from the table.
- The delivered Administrative parameter record with internal group *BANNER TO ODS FGA TRANSFER* and internal code *ODS USER ID NOT FOUND* is used to tell the Transfer Banner Fine-Grained Access process what to do when a given Banner ODS user ID is not found in MGBXWLK. As delivered the value of external Code is *USE BANNER USER ID*.
- If you populated the MTVPARM record with an external code of *USE BANNER USER ID*, but populated MGBXWLK with only the Banner User IDs and the Banner ODS User IDs have not yet been populated, the process [“Transfer Banner Fine-Grained Access” on page 2-99](#), (MGKXFER.P\_TransferFGA), does not read the MGBXWLK table and the Banner User ID is used.

**- All Users are to have a record in MGBXWLK, regardless of whether the Banner and Banner ODS User IDs are the Same**

- Add all Banner user IDs (**Banner User ID** field) and Banner ODS user IDs (**Banner ODS User ID** field) to MGBXWLK. This includes users with the same Banner user ID as their Banner ODS user ID.
- Enter the same MTVPARM record as [“- Banner User IDs are the same as the Banner ODS User IDs”](#) and [“- Some User IDs are the Same in Banner and Banner ODS, and Some are Not”](#), but with an external code of *DENY ACCESS*.

**- Restrict the Information Transferred to a Limited Group of Users**

- Add the limited set of Banner user IDs to MGBXWLK. If the Banner ODS user IDs are different, enter them in the **Banner ODS User ID** field. If the Banner User IDs are the same in Banner ODS, enter the Banner User IDs in the **Banner User ID** field and the **Banner ODS User ID** field.
- Enter the same MTVPARM record as [“- Some User IDs are the Same in Banner and Banner ODS, and Some are Not”](#) and [“- All Users are to have a record in MGBXWLK, regardless of whether the Banner and Banner ODS User IDs are the Same”](#) but with an external code of *DENY ACCESS*.

## **Create Banner User ID Translations**

Use this to match a Banner user ID with a Banner ODS user ID.

## Prerequisites

It is recommended that the Banner ODS ID is set up so that it can be selected from the drop-down list that appears when you select **Select a Banner ODS User ID** on the translation Create and Update pages.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set up and Maintain Banner User ID Translations**.
4. Click **Create a New User ID Translation**.

If no User ID translations exist, you are taken directly to the Create a New User ID Translation page.

5. Enter the Banner user ID, or click the Select a **Banner User ID** link to choose it from the list.

The Banner User IDs are drawn from the Banner Finance tables FORUSFN, FORUSOR, and FOBPROF, and the Banner HR tables PSRORG, PTRUSER, and PSRECLS.

6. Enter the Banner ODS user ID, or click the **Banner ODS User ID** link to choose it from the list.

The Select a Banner ODS User ID list is drawn from the WAV\_ALL\_USERS view which contains a list of IDs for users most likely to run the reports. Your institution can change this view to include additional users (ODSMGR IA\_ADMIN, for example) so that additional user IDs will appear in the list.

7. Click **Save**.

## Update Banner User ID Translations

Use this to change the Banner ODS user ID matched with a Banner user ID.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set up and Maintain Banner User ID Translations**.
4. Select the Banner user ID you want to change.
5. Enter the Banner ODS user ID, or click the link to select if from a list.
6. Click **Save**.

## Delete Banner User ID Translations

Use this to delete the Banner ODS user ID matched with a Banner user ID.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set up and Maintain Banner User ID Translations**.
4. Select the Banner user ID you want to change.
5. Enter the Banner ODS user ID, or click the link to select it from a list.
6. Click **Delete**.

## Set up Business Profiles

Business Profiles enable you to easily manage groups of users by grouping similar users together. In turn, you can manage Security and Display Rule assignments as a group rather than at the individual user account level.

First you create a Business Profile, then associate one or more users with that Business Profile, or associate one or more Profiles with one or more users.

Multiple display rules can also be managed, or assigned, using business profiles.

### Note

Business profiles are only used when more than one Oracle user is used to access the data from your institution supported report writer. ■

If business profiles are used, then the system pulls the appropriate values for the profile with which the user is associated, if a rule exists for that profile.

### Note

If multiple profiles exist for that user, then the first profile with a matching display rule is used. If no display rules are found for any profiles assigned to the user, the display rule for the default profile (*INSTITUTION*) is used. ■

For reporting views such as the TEST\_SLOT view, use business profiles to designate unique sets of test score data and the positional order of that data within the view for different business offices and users at the institution.

For hierarchical reporting views such as the PERSON\_ADDRESS view, business profiles enable you to designate unique sets of (mailing) address type hierarchies for different business offices and users.

## Create a Business Profile

Perform the following steps to create a business profile.

### *Prerequisite*

Create an organizational area.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set Up Business Profiles**.
4. Click **Create a Banner ODS Business Profile**.
5. Enter a new profile code and description.
6. Click **Save**.

See [“View, Update or Delete a Business Profile” on page 2-26](#) for steps on updating and viewing Business Profiles.

## Associate Business Profiles with a User

Perform these steps to associate a Business Profile with a user or group of users. You can also link to the Set Up Banner ODS Security Rules page to set up security rule assignments for that profile or user.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Associate Users and Business Profiles**.
4. Choose the user to which you want to associate (or view existing) Business Profiles. If you selected the user from the user drop-down list, then click **Refresh Profile List** to redisplay the business profiles list for that user. Below the user drop-down list is an alphabetical list of all Business Profiles and the user name associated with them.
5. Check or uncheck the corresponding check boxes to associate or disassociate Business Profiles with the user.
6. To set up security rules for a user, click **Assign Security Rules**. See [“Set up Fine-Grained Access Security” on page 2-17](#) for additional information.
7. Click **Save** to update the user associations.

## Associate Users with a Business Profile

Use this option to associate a user or group of users with a Business Profile. You can also link to the **Set Up Banner ODS Security Rules** page to set up security rule assignments for that profile or user.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Associate Users and Business Profiles**.
4. Choose the Business Profile to which you want to associate (or view existing) users from the **Business Profile** column.



### Note

When you select the Business Profile column or Oracle User Name column, the table toggles between associating a Business Profile with a user and associating a user with a Business Profile. ■

5. Check the corresponding check boxes to associate or disassociate users with a Business Profile.
6. Click **Save** to submit your changes.
7. To set up security rules for a Business Profile, click **Assign Security Rules**.

See [“Set up Fine-Grained Access Security” on page 2-17](#) for instructions on assigning security rules.

## View, Update or Delete a Business Profile

Use this option to change or delete a Business Profile.

1. Click **Preferences and Security**.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set Up Business Profiles**.
4. Click the description of the Business Profile you want to change.

The Update a Banner ODS Business Profile page opens. From this page you can change the descriptions or delete the Business Profile.

5. Make your changes to the description.

6. Click **Save** to submit your changes.

*or*

Click **Delete** to remove the displayed profile.

 **Note**

The table at the bottom of the page indicates what rows in that table are also deleted if you delete the business profile. ■

## Set up and Maintain Security Rules

The following tables (in the IA\_ADMIN schema) are used to store the security rules information in Banner ODS.

Table	Functional Name	Security Rules Stored
MGBFGAA	Fine Grained Access User Areas	Indicates if the user has access to all of the elements and values within an area code
MGBFGAE	Fine Grained Access User Elements	Indicates if the user has access to all of the values within an element code
MGBFGAV	Fine Grained Access User Values	If the user does not have the MGBFGAV_ALL_IND or MGBFGAA_ALL_IND for an element, area, or all of FGA, indicates which values for the element the user may access.
MGBSECR	User Security Table.	Various user security related data.
MGBFGAR	Fine Grained Access Element Rule Table	The security rules that consist of Banner ODS tables and columns that have security applied to them.
MTVFGAA	Fine Grained Access Area Validation Table	The security rules that consist of Banner ODS area that have security applied to them.

Understanding the data relationships in these tables is best explained by reviewing the Administrative UI that maintains that data.

To set up security, you need to:

- Determine the data security requirements
- Set up and maintain the security rules

## Determine Data Security Requirements

Use this section to determine whether it's necessary to restrict some users' access to some of the data within Banner ODS and to determine the specific security restrictions that apply to each user.

### Warning

When deciding whether to apply fine-grained access, keep in mind that its use limits the accuracy and usefulness of data. The system does not inform users that the data they are seeing has been filtered by fine-grained access security. This can cause incorrect numerical results in some circumstances. ■

### Example

If a user queries across the entire institution, and that same user has been restricted from seeing data from some departments. Although the data appears to cover the whole institution, it does in fact sum data only from those departments which the user is allowed to access. The user may draw incorrect conclusions if he or she is unaware that the data is incomplete.

If you choose to use fine-grained access, you have the following options for the level of access you can give an individual Oracle user who accesses Banner ODS:

- Full access to all data in Banner ODS.
- Full access to all data at the level of the Organizational Dimension, for example, Academic, Course and Academic, Financial, or Workforce.
- Full access to all data at an element level, for example, college, department, major, organization, or fund level.
- Restricted access to data at the element level based on a list or range of values for a specific data element, for example., allow a user to access only data related to the user's department or a range of fund codes.

## Set up a Security Rule

If you want to secure data at a granular level, you need to create the security rules that define that level of security. A security rule consists of an Organization Dimension, Table, Rule Type, and Column (you may define one or two columns).

Setting up a rule involves entering and maintaining the data that comprises a rule in the MGBFGAR table. You can use the Administrative UI to create and maintain the list of security rules that can be applied to a given user account, and to assign particular values for a given rule to a given user account. (Another method available using the Administrative UI is to assign values for a given rule using [“Set up Fine-Grained Access Security” on page 2-17](#). The Administrative UI uses the MGKFGAC package to apply the security rules you define.

Use the [“Set up and Maintain Security Rules” on page 2-27](#) option within the Administrative UI to create, update, delete, and search for rules. (These processes are described in the next few sections.) Creating or updating rules is reflected in the MGBFGAR table. Deleting rules changes the MGBFGAR table, but in addition, any values related to a rule that are deleted are cascaded through the other fine-grained access tables. There is a list at the bottom of the security rules web pages that indicates what rows in the table are deleted if the security rule is deleted.

Sample security rules (generated from the ods\ia\_admin\dbscripts\mgbfgar\_data\_ods.sql script) are added to MGBFGAR when ODS is installed or upgraded. The delivered sample Finance and Human Resources security rules reflect the way that security rules should be set up if you plan to use Transfer Banner Fine-Grained Access. Since those rules are added to MGBFGAR by the install or upgrade, they can be viewed through the Administrative UI Set Up and Maintain Security Rules pages.

### **Prerequisites**

- Create organizational areas
  - Create business profiles
1. Determine a Banner ODS table and column value on which you want to secure information.
  2. Click **Preferences & Security** from the Administrative menu.
  3. Click **Set up Banner ODS Security Rules**.
  4. Click **Set Up and Maintain Banner ODS Security Rules**.
  5. Click **Create**. The Create a New Security Rule page opens.
  6. Enter the values for each field as described below.

Field	Description
Organizational Areas	This attribute enables you to group similar rules together for easier maintenance/assignment. You can grant access to entire sets of columns/tables at this level using a single check-box. Rules are delivered with four groupings. You can add more groupings using the <a href="#">“Set up and Maintain Organizational Areas” on page 2-19</a> .
Table	Banner ODS table on which you want to secure data, for example, the MST_TEST table, the MPT_EMPL_EARN_FY table, etc.

Field	Description
Rule Type	<p>The type of Security Rule. There are two possibilities:</p> <p><i>Range:</i> This type of rule pertains to limits, such as Financial amounts. Results in a WHERE clause predicate like: WHERE COLUMN1 &gt; [some value1] AND &lt; [some value2].</p> <p><i>List:</i> This type of rule pertains to lists of valid values. Results in a WHERE clause predicate that matches up the list of allowed values (from the MGBFGAV table) with the values in the source table itself.</p> <p><b>Note:</b> The Transfer Banner Fine-Grained Access process only uses security rules with rule type of "List".</p>
Column 1	Banner ODS table column to which the rule pertains.
Query for Column 1	<p>The PL/SQL SELECT statement used to populate the list of values in the Administrative UI for the specified Column 1 when assigning values to users. Click <b>Generate</b> to automatically create the PL/SQL statement.</p> <p>The base rules are delivered with simple SELECT DISTINCT queries for each of the columns on the various Banner ODS tables. However, if performance becomes an issue (for the SELECT DISTINCTs to return), you can create temporary tables (manually) from the results of a SELECT DISTINCT query, then change this query to have the rule point to the temporary table instead.</p> <p>For two-column rules, select distinct values for both columns into a temporary table and then include select distinct statements for both query for Column1 and query for Column 2.</p>

### Example

You have a two-column rule for MFT\_GENERAL\_LEDGER where Column 1 is FUND and Column 2 is CHART\_OF\_ACCOUNTS. First create a table: CREATE TABLE *temp\_table* as SELECT DISTINCT CHART\_OF\_ACCOUNTS, FUND FROM MFT\_GENERAL\_LEDGER. Then, in Query for Column 1 enter SELECT DISTINCT FUND FROM *temp\_table* and in Query for Column 2 enter SELECT DISTINCT CHART\_OF\_ACCOUNTS FROM *temp\_table*.

Field	Description
Column 2	<p>An optional second column on the Banner ODS table to which the rule pertains. This column can be used to join AND values together from two columns.</p> <p><b>Note:</b> If you are creating or modifying rules that deal with Finance such as Fund, Organization, Account, Location, or Program you <i>must</i> enter the table's Chart of Accounts column name in the <b>Column 2</b> field. This is required because Banner ODS Finance hierarchy tables check to see if there are additional permissions for a given user, and that lookup on the hierarchy table cannot occur without a value for Chart of Accounts.</p> <p><b>Example</b></p> <p>You want to set up a security rule for the FUND column on MFT_GENERAL_LEDGER. You enter <i>FUND</i> into <b>Column 1</b> and CHART_OF_ACCOUNTS into <b>Column 2</b>. If you create that rule without CHART_OF_ACCOUNTS in <b>Column 2</b>, a user's permissions for General Ledger Funds are incomplete because the Transfer Banner Fine-Grained Access process and the Fine-Grained Access Policy package, MGKSECR, are not able to read the Fund hierarchy table, MFT_FUND_HIERARCHY. If a user has access to Fund 0100 for Chart of Accounts A, the Transfer Banner Fine-Grained Access process and MGKSECR can look up the Fund hierarchy record and determine if there are additional Fund codes related to Fund 0100 that this user should also have access to. Those additional Fund codes would be stored on the hierarchy record in Fund Level 1, 2, 3, 4, and 5.</p>
Query for Column 2	<p>The PL/SQL SELECT statement used to populate the list of values in the Administrative UI for the optional Column 2. Click <b>Generate</b> to automatically create the PL/SQL statement.</p>

Field	Description
-------	-------------

FGA Transfer Type	Select a value for this field if you plan to use the Transfer Banner Fine-Grained Access process to transfer security information from Banner to Banner ODS.
-------------------	--

- Finance Organization
- Finance Fund
- HR Organization
- HR Employee Class

A rule is excluded from the Transfer Banner Fine-Grained Access process if this column is blank.

**Note:** A rule *must* contain a value in this field for the Transfer Banner Fine-Grained Access process to use the rule during the transfer.

***Example***

When the Transfer Banner Fine-Grained Access process transfers Finance Fund permissions into Banner ODS, it selects the rules from MGBFGAR that apply to the Finance Fund transfer. To include a rule in the Finance Fund part of the transfer process select *Finance Fund*.

Field	Description
Column 2 Type	When you create or modify a security rule that is used to apply security to an element of Finance, that rule must have a value for the Chart of Accounts in the <b>Column 2</b> field. In addition, Column 2 Type must contain the value <i>Chart of Accounts</i> which identifies the column 2 value as a Chart of Accounts value.

**Example**

If a rule is created to limit access to the **Fund** column on the General Ledger, you would select *FUND* as the Column 1 value, *CHART\_OF\_ACCOUNTS* as the column 2 value, and *Chart of Accounts* as the Column 2 Type.

**Note:** It is obvious that the column 2 value is a Chart of Accounts value because the name of the column is CHART\_OF\_ACCOUNTS. This is not obvious for all Chart of Accounts column names. Some appear as DESG\_CHART\_OF\_ACCOUNT on the MAT\_GIFT table and HOME\_ORGANIZATION\_CHART on MPT\_EMPLOYEE. The **Column 2 Type** field explicitly identifies a column 2 value as a Chart of Accounts column.

If the rule's column 2 value is a Chart of Accounts column name during the Transfer Banner Fine-Grained Access process, then the Chart of Accounts value is brought over from Banner when the data is written to MGBFGAV.

Field	Description
Predicate Code	Leave this field blank for all rules that pertain to Finance Fund, Finance Organization, and Human Resources Organization to transfer file access permissions using the Transfer Banner Fine-Grained Access process. When the field is blank, the Transfer Banner Fine-Grained Access process writes additional data to MGBFGAV from the appropriate Banner ODS Finance Hierarchy table.

### Example

If John Smith has access to Fund 0100, when the Predicate Code is blank on all of the Finance Fund security rules, the Transfer Banner Fine-Grained Access process reads the Banner ODS Finance Fund Hierarchy table, MFT\_FUND\_HIERARCHY, and determines if having access to fund 0100 also gives John access to additional Fund numbers. If the Fund Hierarchy entry for Fund 0100 includes a Fund Level 1 value of 0101 and a Fund Level 2 value of 0102, additional records are written to MGBFGAV giving John access to funds 0101 and 0102 as well as fund 0100.

However, if you preferred to have the Fund Hierarchy values for funds 0101 and 0102 added to the query predicate at query runtime by means of a join to the Fund Hierarchy, you can add a Predicate Code of *Fund* to each of the Finance Fund security rules.

**Note:** It is not recommended that you use the Predicate Code for Finance Fund, Finance Organization, or Human Resources Organization rules because adding the join to the Banner ODS applicable Hierarchy table at runtime can significantly impact query performance.

Keep in mind that Banner Finance permissions are transferred only for Fund and Organization (whether they apply to Finance or HR tables). If you want to add security rules for other portions of Finance, (namely Account, Location, or Program), those permissions are not transferred from Banner. You need to create those security rules with a predicate code of *Account*, *Location*, or *Program* so that the additional values from the appropriate hierarchy table are included in the query predicate at runtime.

However, the same potential performance warning applies for using predicate codes for those security rules. To resolve the performance issue you might consider adding records to MGBFGAV for rules pertaining to Account, Location, or Program.

---

## 7. Click Save.

## Update or Delete a Security Rule

Perform all of these steps for each Security Rule you want to set up. Use the following steps to update an existing Security Rule.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Set Up and Maintain Banner ODS Security Rules**.
4. From the drop-down list, choose the **organizational area**, **table**, and/or **column** for the rule you want to edit.
5. Click **Search**. The list of related Security Rules displays.
6. Click the link in the **Column 1** column for the rule you want to edit. The Update an Existing Rule page opens.
7. Edit **Query for Column 1**, **Query for Column 2**, **FGA Transfer Type**, **Column 2 Type**, and/or **Predicate Code**, then click **Save**.

or

Click **Delete** to remove the displayed security rule.

### **Note**

The table at the bottom of the page indicates what rows in that table will be deleted if you delete the security rule. ■

## Assign Security Rules

After security rules are created, you must determine what level of security each user requires. This is also where the rules are turned on and off.

Next, set up the security rules for users. You can use the Administrative UI to maintain the list of rules in the MGBFGAR table.

### **Note**

The administrator account that you use to set up fine-grained access control needs to have unrestricted access to all data, or the list of values the administrator can grant to others is limited to what the administrator can access. ■

Use any of the following methods to secure user access:

- user name
- organizational area
- business profile
- element

### Secure Access by User Name

Use the following steps to assign security by user name. This method also enables you to grant a user access to all data in the entire solution by checking a single checkbox.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Assign Security Rules**.

The list of User IDs is determined by the IA\_ADMIN.WAV\_ALL\_USERS view. This view contains a list of IDs for users most likely to run the reports. Your institution can change this view to include additional users (ODSMGR IA\_ADMIN, for example) so that additional user IDs will appear in the list.

4. Check the **Access to all Banner ODS Data** check box for each user in which you want to assign access to all data from the Secure Banner ODS Access by User Name page.

Each column is described below.

Click the individual user's name to restrict access to specific areas for that user. The link opens the Secure by Organizational Dimension page. See [“Secure Access by User ID” on page 2-38](#) for steps on restricting a user's access by Organizational Dimension.

Field	Description
Oracle User Name	<p>Grouping of similar rules for easier maintenance/assignment. Rules are delivered with four groupings, but more groupings can be added in the MTVFGAA validation table, and can be used for new or existing rules.</p> <p>To restrict access for a specific user, click that user's user name. Organizational Dimension restrictions are made on the Secure Banner ODS Access by Organizational Dimension page.</p> <p>See <a href="#">“Secure Access by User ID” on page 2-38</a> to restrict a user's access.</p>
Profiles	<p>Set up an existing business profiles on the Create a Business Profile page.</p> <p>Click <b>Assign Profiles</b> to open the View Business Profiles and User Association page.</p>
Access Level	<p>The current level of access the user has to areas of information. To grant full access, check the checkbox in the <b>Access to all Banner ODS Data</b> column.</p> <p>Possible values:</p> <p><i>All</i> Green. Full access.</p> <p><i>Partial</i> Yellow. Access to specified areas only.</p> <p><i>None</i> Red. No access.</p>
Access to all Banner ODS Data	<p>Check the checkbox to give the user unrestricted access to all areas and information.</p> <p>If the checkbox is checked, a Y is stored in the MGBSECR_FGA_ALL_IND column in the MGBSECR table. When the MGKSECR package is called from the policy, no predicate is returned. This allows access to all data.</p>

5. Click **Save** to update the Administrative UI.

## Secure Access by User ID

Use the following steps to assign security to an individual user.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Assign Security Rules**.
4. Check the **Access to all Banner ODS Data** check box to grant the user unrestricted access to all information.
5. Click the user name to which you want to assign access.

This page displays the security rules defined on the Set Up Banner ODS Security Rule page. The rules are grouped alphabetically by Organizational Dimension.

Each column is described below:

Column	Description
Oracle User Name	Click <b>Select Another User</b> to open the Secure Access by User Name page.
Profiles	Existing Business Profiles set up on the Create a Business Profile page.  Click <b>Assign Profiles</b> to open the View Banner ODS Business Profile and User Associations page
Access to All Banner ODS Data	Check the checkbox to give the user unrestricted access to all areas and information.  Click <b>Duplicate User</b> to open the Duplicate User Security Rules window.
Organizational Area	Area within the institution set up within the IA_ADMIN.MTVFGAA table.

Column	Description
Access All Data in this Area	<p>Select the checkbox to grant the user security access to information within the corresponding organizational area.</p> <p>The list of areas is stored in the <code>MTVFGAA</code> table. You may change this list as desired. Rules can be grouped differently, for example. The All Data indicator for an area is stored in the <code>MGBFGAA_ALL_IND</code> in the <code>MGBFGAA</code> table. If the indicator is <i>Y</i> for a given table you are accessing, no predicate is returned from <code>MGKSECR</code> and you have full access.</p>
Table	<p>Banner ODS table on which you want to secure data, for example, the <code>MST_TEST</code> table, the <code>MPT_EMPL_EARN_FY</code> table, etc.</p> <p>Click the link to enable or disable the security policies for that organizational area.</p>
Element	<p>Click an element to open the Secure Access by Element page.</p> <p>Elements can be set up as double or single column rule elements on the Create Security Rules page.</p> <p><b>Double Column Rules</b></p> <p>If a single rule was created that applies to two element columns, then both of the column names appear together in the <b>Element</b> column on the Secure Access by Organizational Dimension page, and are connected by an &amp; (ampersand). This may be done when the user needs to see both pieces of the data in order to accurately understand the data.</p> <p><b>Single Column Rules</b></p> <p>A single column rule is when an element column was set up with a single column and a single rule.</p>

Column	Description
Rule Type	<p>The type of Security Rule. There are two possibilities:</p> <p><i>Range:</i> This type of rule pertains to limits, such as Financial amounts. Results in a WHERE clause predicate like: WHERE COLUMN1 &gt; [some value1] AND &lt; [some value2]</p> <p><i>List:</i> This type of rule pertains to lists of valid values. Results in a WHERE clause predicate that matches up the list of allowed values (from the MGBFGAV table) with the values in the source table itself.</p>
Access Level	<p>The level of security access assigned to the user.</p> <p><i>All</i>     Green. Full access.</p> <p><i>Partial</i>   Yellow. Access to specified areas only.</p> <p><i>None</i>     Red. No access.</p>

6. To copy security access settings from one user or Business Profile to another, click **Duplicate User**. The Duplicate User Security Rules window opens.
  - 6.1. Choose the user(s) and Business Profiles(s) whose setting you want to merge, or duplicate. To choose more than one user or profile, hold down the Ctrl key while you continue to choose users or profiles.
  - 6.2. Use the radio buttons to indicate whether to merge current settings together, or replace one set of settings with another.
  - 6.3. Click **Duplicate** to save your settings, or **Cancel** to close the page.
7. Click **Save** at the bottom of the page to update the Administrative UI.

### Secure Access by Business Profile

Use the following steps to assign security by Business Profiles.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set up Banner ODS Security Rules**.

3. Click **Assign Security Rules**.

A description of each field on the page appears below:

Field	Description
Business Profile	Existing Business Profiles set up on the Create a Business Profile page.  Click a Business Profiles to open the Set Up Security Rules page.
Access Level	The level of security access assigned to the business profile.  <i>All</i> Green. Full access.  <i>Partial</i> Yellow. Access to specified areas only.  <i>None</i> Red. No access.
Access to All Banner ODS Data	Check the check box to give the business profile unrestricted access to all information.

4. Click the Business Profile to which you want to assign access from the Secure Banner ODS Access by Profile page. The Set Up Security Rules page opens.

This page displays the security rules defined on the Set Up Banner ODS Security Rule page. The rules are grouped alphabetically by Organizational Area. Each column is described below:

Column	Description
Profile	Click <b>Select Another Profile</b> to open the Secure Access by Profile page.
Users	The Users associated with this Business Profile.  Click <b>Assign Users</b> to open the View Banner ODS Business Profile and User Associations page.
Access to All Banner ODS Data	Check the checkbox to give the Business profile unrestricted access to all areas and information.  Click <b>Duplicate User</b> to open the Duplicate User Security Rules window.
Organizational Area	Area within the institution set up within the IA_ADMIN.MTVFGAA table.

Column	Description
Access All Data in this Area	<p>Select the checkbox to grant the Business Profile security access to information within the corresponding organizational area.</p> <p>The list of areas is stored in the <code>MTVFGAA</code> table. You may change this list as desired. Rules can be grouped differently, for example. The All Data indicator for an area is stored in the <code>MGBFGAA_ALL_IND</code> in the <code>MGBFGAA</code> table. If the indicator is <i>Y</i> for a given table you are accessing, no predicate is returned from <code>MGKSECR</code> and you have full access.</p>
Table	<p>Banner ODS table on which you want to secure data, for example, the <code>MST_TEST</code> table, the <code>MPT_EMPL_EARN_FY</code> table, etc.</p> <p>Click the link to enable or disable the security policies for that organizational area.</p>
Element	<p>Click an element to open the Secure Access by Element page. Elements can be set up as double or single column rule elements on the Create Security Rules page.</p>

#### ***Double Column Rules***

If a single rule was created that applies to two element columns, then both of the column names appear together in the **Element** column on the Secure Access by Organizational Dimension page, and are connected by an & (ampersand). This may be done when the user needs to see both pieces of the data in order to accurately understand the data.

#### ***Single Column Rules***

A single column rule is when an element column was set up with a single column and a single rule.

Column	Description
Rule Type	<p>The type of Security Rule. There are two possibilities:</p> <p><i>Range:</i> This type of rule pertains to limits, such as Financial amounts. Results in a WHERE clause predicate like: WHERE COLUMN1 &gt; [some value1] AND &lt; [some value2]</p> <p><i>List:</i> This type of rule pertains to lists of valid values. Results in a WHERE clause predicate that matches up the list of allowed values (from the MGBFGAV table) with the values in the source table itself</p>
Access Level	<p>The level of security access assigned to the user.</p> <p><i>All</i> Green. Full access.</p> <p><i>Partial</i> Yellow. Access to specified areas only.</p> <p><i>None</i> Red. No access.</p>

5. To copy security access settings from one user or Business Profile to another, click **Duplicate User**. The Duplicate User Security Rules window opens.
  - 5.1. Choose the user(s) and Business Profiles(s) whose setting you want to merge, or duplicate. To choose more than one user or profile, hold down the Ctrl key while you continue to choose users or profiles.
  - 5.2. Use the radio buttons to indicate whether to merge current settings together, or replace one set of settings with another.
  - 5.3. Click **Duplicate** to save your settings, or **Cancel** to close the page.
6. Click **Save** at the bottom of the page to update the Administrative UI.

### Secure Access by Element

Use the following steps to assign security by element.

1. Click **Preferences & Security** from the Administrative menu.
2. Click **Set up Banner ODS Security Rules**.
3. Click **Assign Security Rules**.

4. If you wish to secure by element for a Business Profile, select **Secure By Profile**.
5. Depending on whether you are securing by User ID or by Business Profile, choose a name for Oracle User Name or Business Profile column. The Set Up Security Rules page opens.

This page displays the security rules defined on the Set Up Banner ODS Security Rules page. The rules are grouped alphabetically by Organizational Dimension.

6. Elements can be set up as double or single column rule elements on the Create Security Rules page.

### **Double Column Rules**

If a single rule was created that applies to two element columns, then both of the column names appear together in the **Element** column on the Secure Access by Organizational Dimension page, and are connected by an & (ampersand). This is often done when the user needs to see both pieces of the data in order to accurately understand the data.

### **Single Column Rules**

A single column rule is when an element column is set up with a single column and a single rule.

7. Choose the element to which you want to assign security for the user.

From this page you can:

- choose another element
- assign profiles to the user/business profile to access all values for the element
- copy user access to another user

A description of each field appears below:

Field	Description
Oracle User Name	The user's Oracle User ID.  Grouping of similar rules for easier maintenance/assignment. Rules are delivered with four groupings, but more groupings can be added in the MTVFGAA validation table, and can be used for new or existing rules.
Organizational Area	Area within the institution set up within the IA_ADMIN.MTVFGAA table.
Table	Banner ODS table on which you want to secure data, for example, the MST_TEST table, the MPT_EMPL_EARN_FY table, etc.

Field	Description
Element	<p>Elements can be set up as double or single column rule elements on the Create Security Rules page.</p> <p><i>Double Column Rules</i> If a single rule was created that applies to two element columns, then both of the column names appear together in the <b>Element</b> column on the Secure Access by Organizational Dimension page, and are connected by an &amp; (ampersand). This is often done when the user needs to see both pieces of the data in order to accurately understand the data.</p> <p><i>Single Column Rules</i> A single column rule is when an element column is set up with a single column and a single rule.</p>
Rule Type	<p>The type of Security Rule. There are two possibilities:</p> <p><i>Range:</i> This type of rule pertains to limits, such as Financial amounts. Results in a WHERE clause predicate like: WHERE COLUMN1 &gt; [some value1] AND &lt; [some value2]</p> <p><i>List:</i> This type of rule pertains to lists of valid values. Results in a WHERE clause predicate that matches up the list of allowed values (from the MGBFGAV table) with the values in the source table itself.</p>
Allow this user/ profile Access to	<p>Click the appropriate button:</p> <p><i>All values:</i> The user is granted access to all values for this element, and is stored in the MGBFGAE_ALL_IND column as a Y. If new values are add, they will be considered accessible after the next refresh.</p> <p><i>Only the values specified below:</i> Specify which values the user can access. If new values are added then they will not be considered accessible after the next refresh. Each new value needs to be checked individually.</p> <p>You can click <b>All Values</b>, which is then stored in the MGBFGAE_ALL_IND column as a Y, then the user or business profile is granted access to all values for this element.</p> <p>If you can choose <b>Only the values specified below</b>, then you can choose the specific values to which the user will have access (a la carte style). Those selected values are then stored in the MGBFGAV table.</p>

8. Indicate whether you want to allow the user or business profile access to all values, or only the values that appear in the Values table below the **Allow this user (or profile) Access to** radio group.

8.1. If you selected a single column rule element, then refer to the sample screen for a single column rule element below:

Click the checkbox next to a value to give the user access to that value, then click Save.

**Oracle User Name:** DBSNMP  
**Organizational Area:** Academic Organization  
**Table:** MST\_ACADEMIC\_OUTCOME  
**Element:** CAMPUS [Select Another Element](#) [Duplicate Values](#)  
**Rule Type:** LIST  
**Allow this user access to**  All values  
 **Only** the values specified below

[Select All](#) | [Deselect All](#)

Values	Access
001	<input type="checkbox"/>
1	<input type="checkbox"/>
B	<input type="checkbox"/>
H	<input type="checkbox"/>
M	<input type="checkbox"/>
U	<input type="checkbox"/>
<b>NULL *</b>	<input type="checkbox"/>
Cypress College Campus	<input checked="" type="checkbox"/>

A description of each single column rule element column appears below:

Field	Description
Value	These values are set up in the validation tables in your source system. NULL indicates missing codes in your source system.
Access	Check the checkbox of the values to which you want to assign security for the selected user.

- 8.2. If you selected a double column rule element, then refer to the sample screen for a double column rule element below.

Click the checkbox next to a value to give the user access to that value, then click Save.

**Oracle User Name:** DBSNMP  
**Organizational Area:** Academic Organization  
**Table:** MFT\_GENERAL\_LEDGER  
**Element:** CHART\_OF\_ACCOUNTS [Select Another Element](#) [Duplicate Values](#)  
**Rule Type:** LIST  
**Allow this user access to**  All values  
 Only the values specified below

Values	Access to Values of FUND
B	3/53
I	0/37

A description of each column appears below

Column	Description
Value	These values are set up in the validation tables in your source system. NULL indicates missing codes in your source system.
Access to Values of (column name)	These values have one rule for two columns. The number to the right of the slash indicates the number of values in the column that have been assigned to the user. The second number indicates the total number of possible values available for that column. In the sample screen above, 2 out of 53 possible values have been assigned for the FUND column.

9. Click **Save** to keep your settings.

## Security Predicates

An important, but optional, step in your implementation of Banner ODS Fine-grained Access is to review the delivered code in the MGKSECR PL/SQL package. This way you can review the delivered business logic, and determine if it is appropriate for your institution. You can also determine if there is any business logic that you might want to add.

If you encounter issues using the Security system, you might examine the security predicates that are generated. Enter the following query:

```
select mgksecr.f_check_ODS_fga('ODSMGR', 'MST_TEST') from dual;
```

Returns:

```
exists (select 'x' from mgbfgav where mgbfgav_username =
sys_context( 'userenv','session_user') and
mgbfgav_fgaa_code='ACAORG'and mgbfgav_column_name = 'TEST' and
NVL(mgbfgav_value,1) = NVL(TEST,1)) and exists(select 'x' from
mgbfgav where mgbfgav_username = sys_context(
'userenv','session_user') and mgbfgav_fgaa_code = 'ACAORG' and
mgbfgav_column_name = 'TEST_TYPE' and NVL(mgbfgav_value,1) =
NVL(TEST_TYPE,1))
```

Oracle produces a JOIN to the security tables for any columns that do not have the **All Data** indicator set. This allows the Oracle query optimizer to determine the fastest way to retrieve the data.

## Policy Management

Typically, policies (and hence security) are either completely on or off. Two scripts are delivered with the Administrative UI to help manage the policies.

### Prerequisites

- Create organizational areas
- Create user ID translations
- Create business profiles
- Create security rules

### Policies for all Tables

To set up policies for all the tables that have security rules defined for them, run the following script:

```
sqlplus IA_ADMIN/<password> @create_all_fga_policies
```

#### Note

These scripts are delivered in the `dbscripts/utility_scripts` directory. ■

To remove all the policies from Banner ODS tables, run:

```
sqlplus IA_ADMIN/<password> @drop_all_fga_policies
```

#### Note

These scripts add or drop Policies only for those tables with defined security rules. However, by default, security rules are not defined for all Banner ODS tables. You should review the list of security rules in the Administrative UI to verify that all tables that you want to secure have

rules defined. Since you only set up Policies for the tables with rules, any other tables remain unsecured. Remember, however, you can always update the security rules later, and then rerun the “drop” and “create” scripts to establish Policies as well. ■

## Policies for a Single Table

Banner ODS is delivered with a script that can create a policy for a single table. This script enables you to independently test security access. Edit the script to supply the name of the table for which you want to create a policy, and then run the following:

```
sqlplus IA_ADMIN/<password> @create_fga_policy
```

Another way to enable a policy for a single table is available on the Assign Security Rules/ set Up Security Rules pages of the Administrative UI. In the **Table** column is a link that is either set to **Policy Enabled**, or **Policy NOT Enabled**. Click the link to toggle between enabling or disabling the policy for a single table.

### Example:

1. Create a new user to access Banner ODS - call the account BRUCE.
2. Use the MST\_TEST table, and add nine rows using the following commands:

```
TRUNCATE TABLE ODSMGR.MST_TEST;  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test1', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test1', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test1', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test2', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test2', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test2', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST,TEST_TYPE) VALUES  
( 'Test3', 'Type A' );  
INSERT INTO ODSMGR.MST_TEST (TEST) VALUES ( 'Test3' );  
INSERT INTO ODSMGR.MST_TEST (TEST) VALUES ( 'Test3' );  
COMMIT;
```

### Note

The last two rows have a NULL value for TEST\_TYPE. ■

Banner ODS does not have any policies in place when it is delivered. If the user BRUCE has been granted SELECT access to the MST\_TEST table, you can execute the following query:

```
SQL> select count(*) from odsmgr.mst_test;
COUNT(*)
-----
          9
```

- 3. Apply the policy to this table (from the IA\_ADMIN user account):

```
SQL> set serveroutput on size 50000;
SQL> exec mgkutil.p_createFGAPolicy('ODSMGR','MST_TEST',1);
Policy added to table: MST_TEST
PL/SQL procedure successfully completed.
```

- 4. Run the BRUCE query again. The following appears:

```
SQL> select count(*) from odsmgr.mst_test;
COUNT(*)
-----
          0
```

Look in the Administrative UI Security. The BRUCE account is displayed with no global access.

- 5. Select the **All Data** checkbox, and rerun the query. The following appears:

### Secure Access by User Name

 You have successfully updated this entry.

To give a user unrestricted access to all data, click the checkbox in user's name.

Oracle User Name	Access to All Data
<a href="#">ANONYMOUS</a>	<input type="checkbox"/>
<a href="#">BRUCE</a>	<input checked="" type="checkbox"/>
<a href="#">CTXSYS</a>	<input type="checkbox"/>

```
SQL> select count(*) from odsmgr.mst_test;
```

```

COUNT ( * )
-----
9

```

6. Clear the **All Values**.
7. Click the **Save**.
8. Choose the BRUCE account.

To duplicate these results check/uncheck the **Access All Data in This Area** checkbox for the Academic Organization. To continue to test this, choose a combination of values for the two columns in the MST\_TEST table, namely:

<u>MST_STUDENT_COURSE</u>	<u>COURSE_LEVEL</u>	<u>LIST</u>	<u>NOTE</u>
MST_TEST	<u>TEST</u>	LIST	None
MST_TEST	<u>TEST_TYPE</u>	LIST	None
<u>MST_COURSE_CATALOG</u>	<u>COURSE</u>	<u>LIST</u>	<u>NOTE</u>

9. Enable the first two values of the TEST element as follows:

**Oracle User Name:** BRUCE

**Organizational Dimension:** Academic Organization

**Table:** MST\_TEST

**Element:** TEST [Select Another Element](#)

**Rule Type:** LIST

**Allow this user access to**  All values  
 Only the values specified below

[Select All](#) | [Deselect All](#)

Values	Access
Test1	<input checked="" type="checkbox"/>
Test2	<input checked="" type="checkbox"/>
Test3	<input type="checkbox"/>

[Select All](#) | [Deselect All](#)

And yet:

```

SQL> select count(*) from odsmgr.mst_test;

COUNT ( * )
-----
0

```

Security rules are cumulative. Users must have access to values across *all* columns/ rules for a given table in order to access the data.

10. Update the TEST\_TYPE element as follows:

Oracle User Name: BRUCE  
Organizational Dimension: Academic Organization  
Table: MST\_TEST  
Element: TEST\_TYPE [Select Another Element](#)  
Rule Type: LIST  
Allow this user access to  All values  
 Only the values specified below

[Select All](#) | [Deselect All](#)

Values	Access
Type A	<input checked="" type="checkbox"/>
NULL *	<input type="checkbox"/>

(\* - refers to actual database NULL value)

[Select All](#) | [Deselect All](#)

The expected results are:

```
SQL> select count(*) from odsmgr.mst_test;  
  
COUNT(*)  
-----  
6
```

You can continue to test security using the Administrative UI, and see the results from queries that are run against the system.

## Administrative User Interface Data Access

Once policies are in place, you control all access to tables using the information in the security (MGBFGA\*) tables. You might wonder how can the Administrative UI issue the SELECT DISTINCT queries to retrieve the list of values? Shouldn't they need to be configured in the Security Tables also? Does the user account used by the web or application server have some kind of back door around the security system? The answer is, yes and no. As part of the Policy/FGA security system, Oracle provides a way to selectively bypass security using application context variables. You can create a context

that is associated with a particular package that has permission to set application context values. This can then be retrieved by other parts of the application.

In practice, this means you can create a context called IA\_FGA and associate it with the Administrative UI (MGKFGAC) package. In that package, you can set a context variable prior to making queries to the tables. Then, when Oracle calls the MGKSECR package to enforce the policy, it checks that the context variable exists, and sensing it, returns no predicate. This allows full access to the data in that table. The context variable only exists for the life of the package (in the application server memory) and can be accessed only by that package. So, no other attempts to access the context are allowed. This allows the Administrative UI to maintain complete access to administer security while keeping security in place for all other access attempts. (For more information on using Application Context for security, see the Oracle Database Security Guide)

## Set up and Synchronize Data

---

Maintaining current data in Banner ODS is key to producing accurate reports. Banner ODS uses programs—Oracle Warehouse Builder (OWB) mappings—to associate elements in the administrative system with their corresponding elements in Banner ODS. When you run a job (schedule a process via the Administrative UI), it calls the related mappings and loads or updates the data defined by them.

Banner ODS includes two main categories of mappings:

- **LOAD mappings**—load data from the administrative system into Banner ODS. These mapping names include a “LOAD\_” prefix.
- **REFRESH mappings**—update Banner ODS with data that has changed in the administrative system. Mappings in this category have an “UPDATE\_” or “DELETE\_” prefix. Typically, these mappings exist in pairs. To perform a complete refresh, you run the DELETE mapping followed by its associated UPDATE mapping.

Banner ODS is delivered with hundreds of mappings already defined. LOAD and REFRESH mappings exist for each composite table in Banner ODS. To make it easier to work with the mappings, they are organized into groups by product area. This gives you the ability to run one job that includes a group of mappings at one time. (For example, Finance-related mappings.) Or, you can run a single mapping.

Banner ODS exists in a self-contained environment separate from your source system. You synchronize data between the two systems using the processes that load and refresh data in Banner ODS. Even with daily synchronization, you can expect minor differences between the two systems. Three main reasons that differences exist are:

- Data currency in Banner ODS is dependent on the timing of a query against Banner ODS, and when Banner ODS was last refreshed. Changes that occur in the administrative system after the last refresh are not reflected until the next refresh

occurs. This causes a variance between the two systems until Banner ODS is refreshed again.

- Display rules may differ between the two systems. In Banner ODS, display rules defined on the MGRSDAX table drive Banner ODS views created to support existing Object:Access functionality. Differences may occur based on which rules are applied to each system.
- Security rules may also cause differences between the two systems. Your source system allows you to set up fine-grained access security at the element level as does Banner ODS. Rules in both systems are discrete, so there may be differences in the data a user can view based on the security rules defined within each system.

It is important to keep in mind these possible differences while reporting against Banner ODS.

- When you first install Banner ODS, populate it with data from your source system by running the “Load All Banner ODS Products” job
- Refresh data in Banner ODS on a regular basis by scheduling jobs that update Banner ODS each night
- Update specific areas of Banner ODS as needed by scheduling that job when data is changed in the source system

## Set up Parameters

---

Parameters that are delivered with your solution are stored in a table called MTVPARM. You can use the Administrative UI to view and modify the entries in MTVPARM, and to customize Banner ODS and the Administrative UI. (Example customizations: Schedule a process, define mappings that move data from the source system, define data cleansing, freeze data, publishing meta data, etc. See [“Set up Customized Scheduled Processes” on page 2-71](#) for additional information.)

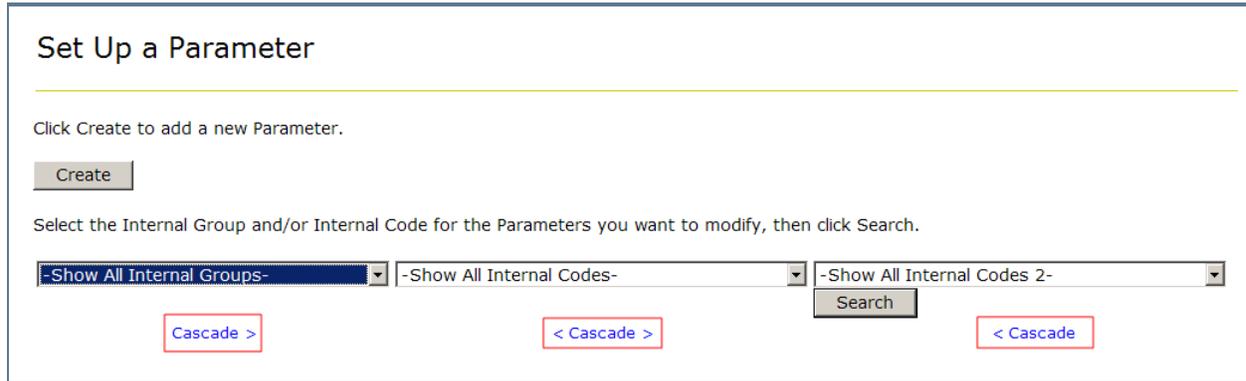
### Note

These parameters are different from the actual runtime parameters that you supply when you schedule a process (run the mappings). (See [“Schedule a Process Parameters” on page 2-75](#).) The parameters discussed in this section are internal parameters that are used in internal processing. ■

A parameter can include multiple values. The values for a single parameter all use the same Internal Code. You use the Internal Code to choose a parameter to edit. Parameters are edited on the Set Up a Parameter page of the Administrative UI.

## Cascade filter

The Cascade links on the Set Up a Parameter page let you filter the related field values when you select them. When you make a selection from any of the dropdown lists on the page then click the related Cascade link, the other lists will filter to display only related values.



The screenshot shows the 'Set Up a Parameter' page. At the top, it says 'Click Create to add a new Parameter.' with a 'Create' button. Below that, it says 'Select the Internal Group and/or Internal Code for the Parameters you want to modify, then click Search.' There are three dropdown menus: '- Show All Internal Groups -', '- Show All Internal Codes -', and '- Show All Internal Codes 2 -'. A 'Search' button is to the right of the third dropdown. Below each dropdown is a 'Cascade >' link. The first 'Cascade >' link is highlighted with a red box.

For example, select a value in the Internal Groups dropdown list, then click the [Cascade >](#) link to filter the Internal Code and Internal Code 2 dropdown lists to display only the values related to the select Internal Group.

## Create a parameter

Use the following steps to create a parameter entry.

1. Click **Options** from the Administrative UI menu. The Options menu opens.
2. Click **Set Up Parameters**. The Set Up a Parameter page opens.
3. Click **Create** from the Set Up a Parameter page, or click **Duplicate** from the Update an Existing Parameter page. The Create a New Parameter page opens.
4. Enter the information for the new parameter. A description of each field, followed by an example, appears below:

Field	Description
Internal Group	Rows of data with varying Internal Codes that are categorized together to provide multiple entries for one parameter.
Internal Code 1	Parameter values. Related values have the same Internal Code 1.
Internal Code 2	Used in combination with Internal Code 1 to further define the parameter values when the values in Internal Code 1 are not unique. Often this field is not used.

Field	Description
Internal Code Sequence Number	Order in which multiple rows of data appear within their parameter group. For parameters that are used to create a list, it specifies the order in which the values will appear in that list.
External Code	Short description of the parameter value for the related Internal Code. Also used as a Yes/No value indicator in some parameters.
Description	Long description of the parameter value for the related Internal Code.
System Required?	Yes or No. Indicates whether the field is required for production processing.

5. Click **Save** to create the new parameter.

**Example: Event parameter**

When you freeze data, you must specify an event so that the process knows where to load the new information. The Event parameter is used to define EVENT codes that are used for freezing data.

The Internal Group value is EVENT. It's used to identify all of the values for the Event parameter.

Internal Code 1 defines the various areas within Banner ODS that require different event definitions. It includes all the Subprocess values used to freeze data

Internal Code 2 defines each different event related to the areas defined by Internal Code 1. The values in this field are the valid values you can enter in the Event Code field.

The Internal Code Sequence is used to order parameter values that fall within the same area defined by Internal Code 1.

## Update or Delete a Parameter

Use the following steps to change or delete an existing parameter.

1. Click **Options** from the Administrative UI menu. The Options menu opens.
2. Click **Set Up Parameters**. The Set Up a Parameter page opens.

3. From the **Show All Internal Groups** drop-down list on the Set Up a Parameter page, choose the Internal Group and Internal Code name of the parameter you want to access. Or, keep the default setting to show all Internal Groups or Internal Codes.

 **Tip**

If you know the first letter of the Internal Group or Code you want to choose, open the **Show all Internal Groups** (or **Codes**) drop-down list then type the first letter of the group or code. Your cursor will move to the first group or code in the list that begins with that letter. This saves you from scrolling through the entire list. ■

4. Click **Search**. The Select an Existing Parameter page opens.
5. Click the description link that corresponds to the parameter entry you want to update or delete. The Update an Existing Parameter page opens.
6. Change the information as needed.

 **Note**

Only External Codes less than 80 characters in length display in the drop-down list. You can create entries that are longer than 80 characters, and they will exist in the system, but do not appear in the list. ■

7. Click **Save**, to save the parameter, or **Delete** to completely remove the parameter.

## System Parameters

Your solution is delivered with values that define aspects of your solution. Below are the delivered system parameters, and how they are used. Additional information can be found in the section [“Schedule a Process” on page 2-60](#).

 **Note**

The parameters listed below are delivered with Banner ODS. For a list of parameters used only to schedule a process, see [“Schedule a Process Parameters” on page 2-75](#). ■

## Parameters

This Parameter . . .	Used for this Task and Solution . . .	Does This . . .
ADMIN_PREFERENCES	Administrative UI for Banner ODS	Optional parameters. These are various settings used to control aspects of the Administrative UI. Currently can be used to control the number of Control Reports that are displayed on the main selection page.
BANNER TO ODS FGA TRANSFER	<a href="#">“Schedule a Process” on page 2-60</a>	Transfers security for Banner Finance Fund, Fund Type, and Organizations, and Banner Human Resources Organizations and Employee Class.
ETL CONTROL GROUP	<a href="#">“Schedule a Process” on page 2-60</a>	Groups together ETL MAP PACKAGE and/or ETL SLOT PACKAGE jobs as one job.
ETL MAP PACKAGE	<a href="#">“Schedule a Process” on page 2-60</a>	Groups related jobs (OWB mappings) as one job.
ETL MAP PACKAGE LOAD PURGE	<a href="#">“Schedule a Process” on page 2-60</a>	Identifies the required crosswalk DELETE mappings for the Load Purge Process.
ETL MAP PACKAGE LOGIC	<a href="#">“Schedule a Process” on page 2-60</a>	<p>Allows you to specify job termination logic for a mapping within a job stream. By default, all mappings in a job run in sequence regardless of whether they have errors or not. By defining an ETL Map Package record for a given mapping in a job, you can have the job stop if that mapping encounters errors.</p> <p>This parameter is used primarily with Banner EDW jobs as they have dependencies from one step (or mapping) to another, while Banner ODS mappings are independent of each other.</p>

This Parameter . . .	Used for this Task and Solution . . .	Does This . . .
ETL MAP PACKAGE RECONCILE LOGIC	<a href="#">“Reconcile a Single Table” on page 2-98</a> and <a href="#">“Reconcile Multiple Tables” on page 2-106</a>	Provides a list of mappings that are exceptions in the reconcile Banner ODS tables process. In this list are the mappings that are ignored in the reconcile process because of the complexity of the mapping or other factors outside the scope of reconciling that Banner ODS table. This list also includes mappings that require either multiple source composite views or mappings in order to reconcile a Banner ODS table.
ETL SLOT PACKAGE	<a href="#">“Schedule a Process” on page 2-60</a>	Groups together related slot jobs (SQL packages) as one job.
EVENT	<a href="#">“Freeze Data Maintenance” on page 2-125</a>	Defines EVENT codes used for freezing Banner ODS data.
EVENT-EDW	<a href="#">“Freeze Data Maintenance” on page 2-125</a>	Defines the Event parameter for freezing EDW business concepts.
INSTALLED PROCESS	<a href="#">“Schedule a Process” on page 2-60</a>	Populates a list of processes displayed on the Select a Process page.
JOB	<a href="#">“Schedule a Process” on page 2-60</a>	Defines the actual name of the job (program) to run when you schedule a process.
JOB INTERVAL	<a href="#">“Schedule a Process” on page 2-60</a>	Defines the list of sample Job Interval settings displayed in the Select an Interval window on the Schedule a Process page.
JOB_KILLER	<a href="#">“Kill a Running Job/ Process” on page 2-68</a>	Defines which administrative accounts have the ability to stop a process that is running.
JOB_NOTIFICATION	<a href="#">“Set up E-mail Notification” on page 2-124</a>	Defines a list of process parameters you need to set up e-mail notification.

This Parameter . . .	Used for this Task and Solution . . .	Does This . . .
METADATA	Meta Data publishing. <a href="#">“Meta Data” on page 2-137</a>	Defines meta data related settings. Currently there is one for where to publish Meta Data pages, and another for where to view them.
ODS FINANCE TEXT	Finance Reporting Text Views	Defines different types of text for Finance Reporting Text Views. For example, Encumbrance Text, Grant Text, Fund Text, Fixed Asset Text, etc.
OWB_SYSTEM_PARAMETER	<a href="#">“Schedule a Process” on page 2-60</a>	Defines the list of known OWB system – used when running mappings, to differentiate which mapping parameters are passed to OWB specifically, and which are passed to the mapping itself.
PARAMETER	<a href="#">“Schedule a Process” on page 2-60</a>	Defines a list of a job’s input parameters you need to supply when you schedule a process.
PUBL_CATE_CODE	<a href="#">“Schedule a Process” on page 2-60</a>	Used during meta data publishing to differentiate the source from the target types.
SSR CONFIGURATION (optional)	<a href="#">“Self-Service Reporting” on page 6-1</a>	Defines the location of the SSR help files and Banner ODS metadata used by SSR, if SSR is installed.
SUBPROCESS	<a href="#">“Schedule a Process” on page 2-60</a>	Populates a list of processes displayed on the Select a Subprocess page.

## Schedule a Process

You can schedule a job to run at a specific time. To run load and refresh (update) jobs, select the **Schedule a Process** option on the Options menu of the Administrative UI.

Before you schedule any jobs to run, you *must* review and set up parameters associated with scheduling a process. See [“Set up Parameters” on page 2-54](#) for more details.

## Process descriptions and details

Click the [Show Process Info] or [Show Subprocess Info] links to view a description of each job on the Select a Process or Select a Subprocess pages. Similarly, click the [Hide Process Info] or [Hide Subprocess Info] links to view only the job names.

Refer to the [“PROCESS INFO Parameter” on page 2-78](#) and [“SUBPROCESS INFO Parameter” on page 2-80](#) sections for details on how to define the descriptions that display for a process or subprocess.

If you choose to Show Process Info on the Select a Subprocess page, [Details] and [Edit] links display for each process in the list as illustrated in the following picture.

### Select a Subprocess

---

Select a subprocess to schedule. [\[ Select a Different Process \]](#) [\[ Hide Subprocess Info \]](#)

#### Schedule Banner ODS Mappings

[ODS 8.3 Reload Data](#) [\[Details\]](#)  
[\[Edit\]](#) Special job to be run once at time of upgrade. Use this process to reload the necessary ODS composite tables based on changes made in the ODS 8.3 release.

[Load All Banner ODS Products](#) [\[Details\]](#)  
[\[Edit\]](#) Use this process to load all ODS composite tables with extracted data from all Banner products.

[Refresh All Banner ODS Products](#) [\[Details\]](#)  
[\[Edit\]](#) Use this process to refresh all ODS composite tables with extracted data from all Banner products.

[Load Accounts Receivable](#) [\[Details\]](#)  
[\[Edit\]](#) Use this process to load data in all Accounts Receivable ODS composite tables with extracted Banner data.

## Details

Click the [Details] link next to a process name to view the job details including the title, code, description, list of all processes included in that job, and the Banner ODS tables or Banner EDW Fact tables that get loaded by the job.

The ADMIN\_PREFERENCES parameter lets you define whether the details of a process, when you select to view them on the Schedule a Subprocess page, display in:

- One popup window which refreshes each time you display the details of a process
- Multiple popup windows, one for each process whose details you want to display

The Parameter record with the following combination of values:

- **Internal Group** = ADMIN\_PREFERENCES

- **Internal Code 1** = SCHEDULE\_UI
- **Internal Code 2** = DETAILS\_DIALOG

controls whether to use one or multiple popup windows when displaying process details. Set the **External Code** = 0 for this Parameter record to allow multiple popup windows or set the **External Code** = 1 to use a single popup window.

## Edit

Click the [Edit] link next to a process description to go to the Update a Parameter page and edit the process description.

The ADMIN\_PREFERENCES parameter lets you define whether to display the [Edit] link next to each process or subprocess description.

The Parameter record with the following combination of values:

- **Internal Group** = ADMIN\_PREFERENCES
- **Internal Code 1** = SCHEDULE\_UI
- **Internal Code 2** = DISPLAY\_EDIT\_LINK

controls whether to display the [Edit] link. Set the **External Code** = 1 for this Parameter record to display the [Edit] link next to each description or set the **External Code** = 0 to not display the link.

## Banner ODS Processes

The following sections detail the Banner ODS processes that you can run.

### Schedule Banner ODS Mappings

Use the options on this menu to load or update the corresponding data into all Banner ODS composite and slotted tables.

### Banner ODS Utilities

Use the options on this menu to report source change table counts, reconcile tables, add comments to reporting views, and run checks and balances.

## Schedule a Single Process

Use the following steps to schedule when you want a single process to run:

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**. The Select a Process page opens.
3. Choose the type of process you want to schedule to run from the Select a Process page.

If you chose Schedule Banner ODS Mappings or Freeze Multiple Banner ODS Tables/Views, then the Select a Subprocess page opens. Continue to the next step.

All other selections open the Schedule a Process page. Skip to step #4.

4. Choose the subprocess you want to run. The Schedule a Process page opens.
5. If you selected the subprocess **Run a Single Banner ODS Mapping**, choose the mapping from the **Mapping to Run** drop-down list.
6. Enter values for other Process Parameters for the selected process, if any exist.
7. Enter the required Scheduling Parameters information.
  - 7.1. Enter a **Run Date** (format dd-mon-yyyy) and **Runtime** (format hh24:mi:ss).
  - 7.2. If you want to run the process on a recurring basis, enter an **Interval**.

Click the link next to the **Interval** field. A sample Interval window opens. Click the link under the **Interval Expression** column for the interval in which you want to schedule a process. For example, to run a process every day at the same time select *SYSDATE+1*.
8. Click **Save** to save the information about this job. The job is entered into the job queue to run at the specified day and time.

## Schedule Multiple Processes

You can schedule and list multiple processes with different parameters as a group. For example, if you want to run multiple Banner ODS Freeze Tables.

To create a multiple process schedule, you must export the definition of each desired single process (including all related parameters) to a comma separated values (.csv) file. You can then use that information to define/copy multiple job definitions in that file into a single master schedule which is then re-imported into the job queue.

To schedule multiple processes:

1. From the Administrative UI menu, click **Options**.
2. Click **Schedule a Process**. The Select a Process page opens.
3. From the Select a Process page, choose the type of process you want to schedule.

If you chose Schedule Banner ODS Mappings, Banner ODS Utilities, or Freeze Multiple Banner ODS Tables/Views, then the Select a Subprocess page opens. Continue to the next step below.

For all other selections, the Schedule a Process page opens. Skip to step 5.

4. Choose a subprocess. The Schedule a Process page opens.
5. If you selected the subprocess **Run a Single Banner ODS Mapping**, choose the mapping from the **Mapping to Run** drop-down list.
6. To open the .csv file, click **Export**.

You can either open the file directly, or save it to another directory and open it from there.

The columns names in the .csv file are described below:

Column	Description
JOBDEF	A constant for parsing the input data.
DATE	Date the job should run. Use MON-DD-YYYY format.
TIME	Time the job should run. Use H:MM:SS format.
PROCESS and SUBPROCESS	Internal identifiers for the job.
(Additional job specific parameters)	Any job-specific parameters such as Event, Source Institution, etc.  For job-specific parameters that use drop-down lists of allowable values, all possible values for those fields are provided in the export download so that they can be copied when setting up the job records to import.

 **Warning**

You *must* retain the formatting of each field in the .csv file. Each field is surrounded by single quotes. These must be retained for the import to parse the data correctly. Microsoft Excel sometimes strips a leading single quote from the contents of a cell, so you *must* be sure it is retained in the .csv output. You may want to use an alternate editing application, although Microsoft Excel works fine as long as you are careful. ■

7. Duplicate the **JOB** line once for each run desired.
8. Enter the date and time you want the process to run.
9. Enter the desired parameter values for each line.
10. Remove extra values in the additional lines. An example resulting .csv file is displayed below:

K JOBDEF									
	B	C	D	E	F	G	H	I	J
	DATE	TIME	PROCESS	SUBPROCESS	Source Institution	Event	Academic Period	Register	Event (Char)
	JAN-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG021	200910		Y
	FEB-29-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG020	200830		Y
	MAR-29-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG019	200820		Y
	APR-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG018	200811		Y
	MAY-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG017	200810		Y
	JUN-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG016	200808		Y
	JUL-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG015	200740		Y
	AUG-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG014	200730		Y
	SEP-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG013	200720		Y
	OCT-30-2007	23:00:00	MAPGROUP.EDW	LOAD_EDW_ENROLLMENT	SGHE	REG012	200710		Y

11. Click **Import** on the Schedule a Process page to re-import the .csv file into the Administrative UI.
12. Enter the name of the exported job into the subwindow, or search for it using **Browse**.
13. Click **Import Jobs**.

The Select and View Scheduled Processes window opens in the background listing the new jobs.

## View and Remove a Scheduled Process

You can schedule to run a process/job immediately, or at a future date/time. Processes scheduled to run at a future time remain in the job queue until runtime. Processes already in the queue can be edited as long as they have not run.

Use the steps below to access the queue and review which processes are scheduled, or to edit or delete a job from the queue.

1. Click **Options** from the Administrative menu.
2. Click **View and/or Remove Scheduled Processes**. The Select and View Scheduled Processes page opens.
3. Choose the date from which you would like to view scheduled processes from the Select and View Scheduled Processes page.

Click **Select a Date** to open a calendar window. The default date is *Today*. When you Choose a date on the calendar, that date appears in the date field.

4. Click **Display Jobs**. The processes scheduled for the selected date display.

To sort the columns in ascending or descending order, click the corresponding column header.

### **To Edit**

- 4.1. Click **Edit** next to the job number. The Schedule a Process page opens.
- 4.2. Make your changes.
- 4.3. Check the **Overwrite Existing Job in Queue** checkbox at the bottom of the page to overwrite the existing process.  
  
Or, leave the box unchecked to create a duplicate process with the information.
- 4.4. Click **Submit**.

### **To Delete**

- 4.1. To delete processes, check the checkbox in the **Delete** column for the process you want to delete.
- 4.2. Click **Delete Jobs**.

## **Configure an Account and Stop a Running Job/Process**

Sometimes jobs/processes run for too long, or are run by accident and you want to stop the job and maybe restart it later. A running job/process can be stopped from the job's control report if the user's account is configured to allow this feature.

### **Configure a User Account to Kill a Job/Process**

A user account name must be configured before that user has the ability to stop a job.

#### **Prerequisite**

Set up the Administrative user name accounts (See [“Set up Users and PINS” on page 2-3.](#))

1. Click **Options** from the Administrative UI menu.
2. Click **Set up Parameters**.
3. Click **Create**.
4. In the **Internal Group Code** field type *JOB KILLER*, or select it from the drop-down list.
5. In the **Internal Code 1** field type *ACCOUNT NAME*, or select it from the drop-down list.
6. In the **External Code** field, type the administrative user name (account log in name).

If the user name was entered as an External Code when the parameter was created, you can select the name from the drop-down list.

7. Enter a description into the description field. The description is usually the same as what appears in the External Code field.
8. Click **Save**.

## Kill a Running Job/Process

A running job/process can be stopped from within the job's/processes control report.

### *Prerequisite*

The administrative account user name must be set up with this ability. See [“Configure a User Account to Kill a Job/Process” on page 2-67](#)

1. Click **Options** from the Administrative UI menu.
2. Click **View Control Reports**.
3. Click the link in the **Process** column for the job/process you want to stop.

The Control Report for that process opens.

4. Click **Kill Job** located in the **Status** column.

### **Note**

This link only appears for jobs that are currently running, and if the user's account is properly configured to kill jobs. ■

The Process Termination Wizard window opens and displays the process attributes.

5. Choose to either kill the process (at the operating system level), or to have the wizard display a list of Oracle commands needed to kill the process manually from the command line outside Banner ODS.

Killing the process at the operating system level immediately stops the process, refreshes the Control Report, and displays a *Terminated* status for the process.

### **Note**

Killing a running process could leave the affected parts of Banner ODS in an undefined state, depending on the process that was stopped. Be sure to clean up data as necessary. Rerun the process to overwrite existing data. ■

## Run a Process from Outside the Administrative UI

All Banner ODS and Banner EDW processes can be run from outside the Administrative UI. The processes are defined in the database as PL/SQL packaged procedures, therefore they can be run from outside the Administrative UI using any application that executes Oracle commands (typically Oracle's sqlplus utility). The name of the (packaged) procedure to run for a given process is defined using the JOB parameter (See [“JOB Parameter” on page 2-82](#) in the Banner ODS and Banner EDW Handbook for additional information.). However, an easy way to determine the name of the procedure used to execute a process is to submit that process to run at a future date, then view the process definition in the Job Queue using the steps below.

1. Schedule the Load Student process to run from the Schedule Banner ODS Mappings menu.
2. Select a date or time in the future.
3. Click **Submit**.
4. Return to the Options menu.
5. Select **View and/or Remove Scheduled Processes**.
6. Enter the date you scheduled the process to run.
7. Click **Display Jobs**.

Below is an example page that might display:

Select and View Scheduled Processes

Current Date: 07-OCT-2008 13:04:24

Delete	Job	Next Run Date	Interval	Job To Run
<input type="checkbox"/>	811 <a href="#">Edit</a>	31-OCT-08 23:00:00	null	mgkmap.P_RunETLMapSlots(8_JOB,'LOAD_STUDENT',NULL, '');

This page indicates in the **Job To Run** field that the Load Student process calls the (IA\_ADMIN.) MGKMAP.p\_runETLMapSlots, and takes the following parameters:

```
PROCEDURE P_RunETLMapSlots(userID IN VARCHAR2 DEFAULT NULL,  
jobNumber IN BINARY_INTEGER DEFAULT NULL,  
process IN VARCHAR2 DEFAULT NULL,  
subProcess IN VARCHAR2 DEFAULT NULL,  
parms IN VARCHAR2 DEFAULT NULL);
```

Parameter	Description
userID	Name/ID of the user associated with the job.  The 8 in the example is the Administrative UI user account for “BILL” (for example, select mgbuser_id from mgbuser where mgbuser_pidm=&userID;)
jobNumber	Number of the job, and of the corresponding Control Report created when the job runs. When these jobs are run using the Administrative UI, using the DBMS_JOBS queue to run them, Oracle takes the <i>JOB</i> keyword and substitutes in the actual job number in the queue for this value. (811 in the example above).  When the job is run outside the Administrative UI, you can give the job any number you want to.  <i>Tip:</i> Do not use a number that is currently used by the Control Report or you’ll have duplicate numbers. Begin numbering with high numbers so that the jobs are easy to find.
process	Name of the PROCESS to schedule (see Administrative UI PROCESS parameter description). In the example it is LOAD_STUDENT.
subprocess	Name of the SUBPROCESS to schedule (see Administrative UI PROCESS parameter description). In the example it is NULL (or empty)
parms	Any process-specific parameters needed. In the example there are none. Typical jobs in the Banner ODS do not take parameters. When scheduling the job through the Administrative UI, you specify these parameters on the Submit page. See <a href="#">“PARAMETER Parameter” on page 2-92</a> in the Banner ODS and Banner EDW Handbook for additional information.)

**8.** Issue the following command to run LOAD\_STUDENT job:

```
EXEC mgkmap.P_RunETLMapSlots(8,811,'LOAD_STUDENT',NULL, '');
```

In the example, this would run the LOAD\_STUDENT job as the userID 8 and the job number 811.

 **Note**

This executes the job synchronously, outside of the DBMS\_JOBS queue, meaning the job actually runs to completion and the above call does not return until the job completes. This is usually desired when calling jobs outside the Administrative UI.

It is also possible to submit jobs to the DBMS\_JOBS queue externally as well to run jobs asynchronously. See the DBMS\_JOBS package documentation for more details. ■

9. Remove the job from the queue when you are finished.

You can also externally execute all other Administrative UI processes, for example, Metadata Publishing and the Utilities, following similar steps.

## Set up Customized Scheduled Processes

A scheduled process can be set up to run one or more customized mappings, and to have the new, customized process appear in the list of scheduled processes on the Select a Subprocess page.

For example, you want to bring in additional data and you don't want to modify an existing mapping. You can create your own mapping(s) then run it either as part of one of the existing processes, like LOAD\_STUDENT, REFRESH\_ALL, etc., or create your own process, like LOAD\_MY\_DATA, etc.

The way mappings are organized can also be changed. Delivered mappings are grouped into processes. LOAD\_STUDENT runs all the Student LOAD mappings, REFRESH\_HR runs all the HR REFRESH mappings, etc. However, you can combine the groups differently to improve performance, to run them simultaneously in separate job processes, etc.

To set up a scheduled mappings process, you need to:

- create a parameter record with an internal group code using the ETL MAP PACKAGE parameter set up for each new OWB mapping to be scheduled
- use the SUBPROCESS parameter to create a new group containing one or more customized mappings (MAPGROUP) to appear on the Select a Subprocess web page, and on the Schedule Banner ODS Mappings menu. It is also possible to add the new OWB mapping to an already existing group, by selecting one of the entries in the pull-down list.
- link the JOB parameter record to the process. This tells Banner ODS which item in the Schedule Banner ODS Mappings list (MAPGROUP) to run.

Follow the steps below. Examples appear after the steps.

1. Click **Set Up Parameters** from the Options menu. The Set Up a Parameter page opens.
2. Click **Create** from the Set Up a Parameter page. The Create a New Parameter page opens. Enter the information for the new process, or select it from the drop-down lists.

3. Click **Save**.

Repeat these steps once for each mapping in the group to set up the ETL MAP PACKAGE parameter, once to set up the SUBPROCESS (or PROCESS) parameter, and once to set up the JOB parameter. They can be set up in any order.

4. To run the newly created process, click **Schedule a Process** from the Options menu. The Select a Process page opens.

5. Click **Schedule Banner ODS Mappings**. The Select a Subprocess page opens.

6. Choose your new process.

**Banner ODS Example:**

The example below walks you through how to create a scheduled process called TEST\_LOAD\_STUDENT\_COURSE. This group will have one mapping called TEST\_LOAD\_STUDENT\_COURSE\_1

First, create an internal group record using the ETL MAP PACKAGE parameter.

1. Click **Set Up Parameters** from the Options menu. The Set up a Parameter page opens.
2. Open the Create a New Parameter page.
3. Enter the information below into the fields.

In This Field ...	Enter This ...	Here's Why ...
Internal Group	ETL MAP PACKAGE	<i>Must</i> be ETL MAP PACKAGE.
Internal Code 1	TEST_LOAD_STUDENT_COURSE	Mapping group name. Create your own name, or specify an existing group if you want to add this mapping to an existing group.
Internal Code 2	TEST_LOAD_STUDENT_COURSE_1	Mapping name in OWB and the package name in Banner ODS.
Internal Code Sequence Number	1	Order of the mappings within the Mapping group (Internal Code 1). Controls the order in which multiple mappings are executed within that group. If you add more mappings then the code should on number up such as 2, 3, 4, 5, etc.

In This Field ...	Enter This ...	Here's Why ...
External Code	ODS_TARGET_STUDENT	Location/project in the OWB repository. These locations pertain to the schema containing the target table(s).
Description	TEST_LOAD_STUDENT_COURSE_1	Actual name of the mapping. <i>Must</i> be the exact same entry as entered into the <b>Internal Code 2</b> field.
System Required	No	Parameter records entered through the Administrative UI are marked as <i>No</i> to differentiate those delivered with the product. Display only.

4. Click **Save**.

Second, set up the SUBPROCESS parameter so that you can create and name a new group of one or more customized mappings. This tells Banner ODS that you want this new process(es) to appear on the Select a Subprocess page, and on the Schedule Banner ODS Mappings menu (MAPGROUP) on that page.

1. Click **Create a New Parameter** at the bottom of the page. The fields on the page reset.
2. Enter the following information.

In This Field ...	Enter This ...	Here's Why ...
Internal Group	SUBPROCESS	<i>Must</i> be SUBPROCESS. This tells Banner ODS to display this group on the Select a Subprocess menu.
Internal Code 1	MAPGROUP	Must be MAPGROUP in order to display this group on the Schedule Banner ODS Mappings menu. You can enter a different SUBPROCESS name if you want to create or use additional process listings.
Internal Code 2		This field remains blank.

In This Field ...	Enter This ...	Here's Why ...
Internal Code Sequence Number	1	Order of the entries on the Select a Subprocess menu. Entries with the same number are sorted by group name. If you add more mappings then the code should on number up such as 2, 3, 4, 5, etc.
External Code	TEST_LOAD_MST_STUDENT	Group name. Must be the same as what was entered into the <b>Internal Code 1</b> field when you set up the ETL MAP PACKAGE parameter.
Description	TEST Load MST Student	Actual text you want to display on the Schedule Banner ODS Mappings list on the Schedule a Subprocess page.
System Required	No	Parameter records entered through the Administrative UI are marked as <i>No</i> to differentiate those delivered with the product. Display only.

**3. Click Save.**

Third, link the JOB parameter to the new group of mappings. This tells Banner ODS which item in the Schedule Banner ODS Mappings list (MAPGROUP) to run.

1. Click **Create a New Parameter** at the bottom of the page. The fields on the page reset.
2. Enter the following information.

In This Field ...	Enter This ...	Here's Why ...
Internal Group	JOB	JOB <i>must</i> be entered.
Internal Code 1	MAPGROUP	Must match the <b>Internal Code 1</b> field of the SUBPROCESS record.
Internal Code 2	TEST_LOAD_STUDENT	<i>Must</i> match the <b>Internal Code 1</b> field when you set up the ETL MAP PACKAGE.
Internal Code Sequence Number	1	Leave as <i>is</i> .
External Code	0	Leave as <i>0</i> .

In This Field ...	Enter This ...	Here's Why ...
Description	mgkmap.P_RunETLMapSlots	Name of the PL/SQL procedure executed by the process. For mapping scheduled processes, use the standard procedure P_RunETLMapSlots
System Required	No	Parameter records entered through the Administrative UI are marked as <i>No</i> to differentiate those delivered with the product. Display only.

3. Click **Save**.

## Schedule a Process Parameters

The Administrative UI uses several system parameters to create the web pages associated with scheduling a process (running the mappings). The next sections describe these parameters, their purpose, and their role in scheduling a process.

### Note

These runtime parameters are different from the set up parameters stored in MTVPARM (See [“Set up Parameters” on page 2-54.](#))

Parameters are maintained on the Set Up a Parameter page of the Administrative UI. See [“Update or Delete a Parameter” on page 2-56](#) for additional information on updating parameters. Each parameter and its purpose appear below:

Parameter	Definition
<a href="#">“INSTALLED PROCESS Parameter” on page 2-77</a>	Populates a list of processes displayed on the Select a Process page.
<a href="#">“PROCESS INFO Parameter” on page 2-78</a>	Defines process descriptions that optionally display on the Select a Process page.
<a href="#">“SUBPROCESS Parameter” on page 2-78</a>	Populates a list of processes displayed on the Select a Subprocess page.
<a href="#">“SUBPROCESS INFO Parameter” on page 2-80</a>	Defines subprocess descriptions that optionally display on the Select a Subprocess page and defines information specific to a subprocess that displays on the Schedule a Process page.
<a href="#">“JOB Parameter” on page 2-82</a>	Defines the actual name of job (program) to run when you schedule a process.

Parameter	Definition
<a href="#">“ETL MAP PACKAGE Parameter” on page 2-84</a>	Groups related jobs (OWB mappings) as one job.
<a href="#">“ETL MAP PACKAGE LOAD PURGE Parameter” on page 2-86</a>	Identifies DELETE mappings for the Load Purge Process.
<a href="#">“ETL MAP PACKAGE LOGIC Parameter” on page 2-88</a>	<p>Allows you to specify job termination logic for a mapping within a job stream. By default, all mappings in a job run in sequence regardless of whether they have errors or not. By defining an ETL Map Package record for a given mapping in a job, you can have the job stop if that mapping encounters errors.</p> <p>This parameter is used primarily with Banner EDW jobs as they have dependencies from one step (or mapping) to another, while Banner ODS mappings are independent of each other.</p>
<a href="#">“ETL MAP PACKAGE RECONCILE LOGIC Parameter” on page 2-88</a>	Provides a list of mappings that are exceptions in the reconcile Banner ODS tables process. In this list are the mappings that are ignored in the reconcile process because of the complexity of the mapping or other factors outside the scope of reconciling that Banner ODS table. This list also includes mappings that require either multiple source composite views or mappings in order to reconcile a Banner ODS table.
<a href="#">“ETL SLOT PACKAGE Parameter” on page 2-89</a>	Groups together related slot jobs (SQL packages) as one job.
<a href="#">“ETL CONTROL GROUP Parameter” on page 2-90</a>	Groups together ETL MAP PACKAGE and/or ETL SLOT PACKAGE jobs as one job.
<a href="#">“EVENT parameter” on page 2-92</a>	Defines Events for data being loaded in Banner ODS. These Events let you freeze or take a snapshot of data at a point in time.
<a href="#">“PARAMETER Parameter” on page 2-92</a>	Defines a list of a job’s input parameters you need to supply when you schedule a process.

## INSTALLED PROCESS Parameter

The Description field for this parameter defines the process names that display on the Select a Process page of the Administrative UI. You can choose from that list to schedule a process.

This parameter is delivered with one entry for each type of process (job) that you can run. The processes defined by this parameter have ‘children’ defined by the SUBPROCESS and JOB parameters. To designate the parent/child relationship, match the External Code of the INSTALLED PROCESS to the Internal Code 1 of the SUBPROCESS and the Internal Code 1 of the JOB.

The following table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

**Internal Group:** INSTALLED PROCESS

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
Can be a short description of the process. This field is not used in processing, but it requires a value.	N/A	Order for entries on Select a Process page.	Short description of the process. Map values of this field to the <b>Internal Code 1</b> values of Subprocesses and related Jobs to define them as its children.	Actual process name that appears on the Select a Process administrative page.
ADHOC_FREEZE	N/A	6	ADHOC_FREEZE	Freeze A Single Banner ODS Table/View
FREEZE_TABLE	N/A	5	FREEZE_TABLE	Freeze Multiple Banner ODS Tables/Views
MAPGROUP	N/A	1	MAPGROUP	Schedule Banner ODS Mappings

### *Setting up the INSTALLED PROCESS parameter*

The only field you should change for the delivered values of this parameter is the **Description** field. If you want to change the name of a process that appears on the Select a Process page, change its description.

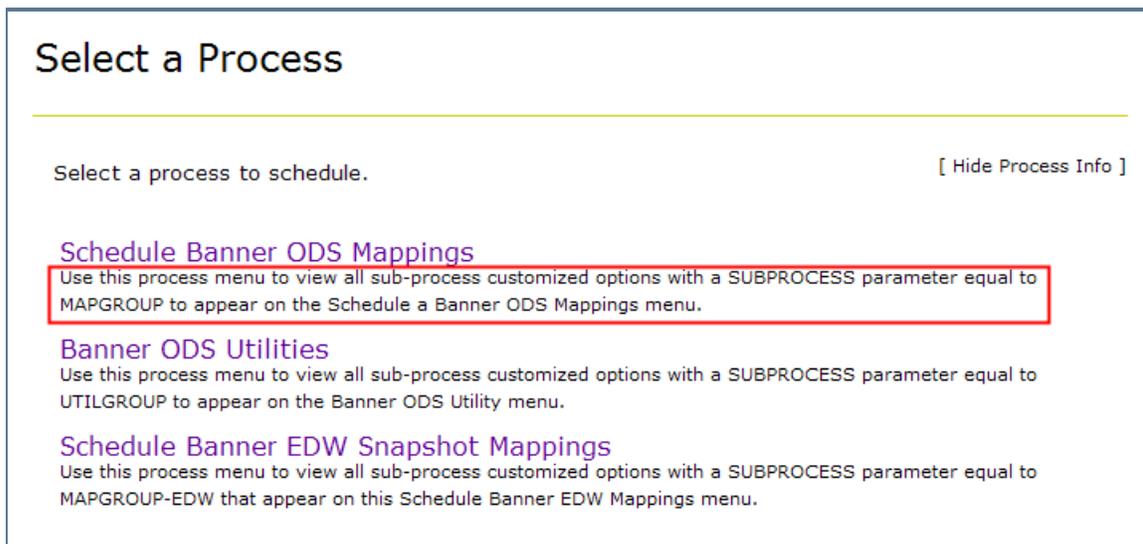
### *Create a new Installed Process parameter value*

If you want to add a process developed by your institution, create the process and add it as a new record for this parameter.

## PROCESS INFO Parameter

The Process Info Parameter records define process descriptions that optionally display on the Select a Process page. Process Info parameter records match to the related Installed Process parameter records based on sharing the same **Internal Code 1** and **External Code** values.

Parameter records with an **Internal Code Group** = *PROCESS INFO* and **Internal Code 2** = *PROCESS\_LIST* use the **Description** field to define the process descriptions that a user can optionally display when they select the [Show Process Info] link on the Select a Process page as illustrated in the following picture.



The screenshot shows a web interface titled "Select a Process". At the top, it says "Select a process to schedule." with a "[ Hide Process Info ]" link on the right. Below this, there are three process descriptions, each with a purple heading and a description text. The first description, "Schedule Banner ODS Mappings", is enclosed in a red rectangular box. The second is "Banner ODS Utilities" and the third is "Schedule Banner EDW Snapshot Mappings".

Descriptions are included for each process that is delivered with the warehouse. To define a processes description for a new job, create a Parameter record with values defined as follows:

## SUBPROCESS Parameter

The **Description** field of this parameter defines the subprocess names that display on the Select a Subprocess of the Administrative UI.

This parameter is delivered with one entry for each subprocess, which are processes grouped under one of the main processes—Schedule Banner ODS Mappings, Freeze Multiple Banner ODS Tables/Views, or Freeze A Single Banner ODS Table/View.

Subprocesses are related to JOB parameter values and both are “children” of one of the processes defined by the *INSTALLED PROCESS* parameter. To designate the parent/child relationship, match the **External Code** of the *INSTALLED PROCESS* to the **Internal Code 1** of the *SUBPROCESS* and the **Internal Code 1** of the *JOB*.

The following table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

**Internal Group:** SUBPROCESS

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
Map to <b>External Code</b> of the INSTALLED PROCESS that is parent to this subprocess.	N/A	Order for entries with same <b>Internal Code 1.</b>	Short description of the subprocess. Use values of this field in the <b>Internal Code 2</b> of its related Job.	Actual process name that appears on the Select a Subprocess administrative page.
MAPGROUP	N/A	1	LOAD_ALL	Load all Banner ODS products
MAPGROUP	N/A	2	LOAD_ALL_SLOTS	Load all Banner ODS slotted tables
MAPGROUP	N/A	3	LOAD_FINANCE	Load finance
MAPGROUP	N/A	8	REFRESH_ALL	Refresh all Banner ODS products
MAPGROUP	N/A	9	REFRESH_FINANCE	Refresh finance
MAPGROUP	N/A	14	RUN_SINGLE_MAP	Run a single mapping

**Set up the SUBPROCESS Parameter**

The only field you should change for the delivered values of this parameter is the Description. If you want to change the name of a subprocess that appears on the Select a Subprocess page, change its Description.

**Create a SUBPROCESS Parameter**

You can add to the subprocess list jobs developed by your institution that you can then run via the Administrative UI. Use the following steps to do this.

1. Create the job.
2. Add the job to the list of subprocesses you can schedule by creating a new Subprocess parameter with the following values:
  - 2.1. **Internal Group:** Subprocess
  - 2.2. **Internal Code 1:** The External Code value of the INSTALLED PROCESS you want the subprocess to be listed under. Existing values include:

- MAPGROUP to list under the Schedule OWB Mappings process.
- FREEZE\_TABLE to list under the Freeze Multiple Banner ODS Tables process.
- ADHOC\_FREEZE to list under the Freeze A Single Banner ODS Table process.

**2.3. Internal Code 2:** blank

**2.4. Internal Code Sequence Number:** Number indicating the order in which to run this subprocess.

**2.5. External Code:** The External Code value of the `INSTALLED PROCESS` you want the subprocess to be listed under. See existing values listed above with the **Internal Code 1** field.

**2.6. Description:** The name of the subprocess that will display on the Select a Subprocess page in the Administrative UI.

## **SUBPROCESS INFO Parameter**

The Subprocess Info Parameter records define one of the following pieces of information:

- [“PROCESS LIST records”](#) - Define subprocess descriptions that optionally display on the Select a Subprocess page
- [“SCHEDULING PAGE records”](#) - Define special instructions or information that will display for a process on the Schedule a Process page

Subprocess Info parameter records match to the related Subprocess parameter records based on sharing the same **Internal Code 1** and **External Code** values.

## PROCESS\_LIST records

Parameter records with an **Internal Code Group** = *SUBPROCESS INFO* and **Internal Code 2** = *PROCESS\_LIST* use the **Description** field to define the subprocess descriptions that a user can optionally display when they select the [Show Subprocess Info] link on the Select a Subprocess page illustrated in the following picture.

### Select a Subprocess

---

Select a subprocess to schedule. [\[ Select a Different Process \]](#) [\[ Hide Subprocess Info \]](#)

#### Schedule Banner ODS Mappings

[ODS 8.3 Reload Data \[Details\]](#)  
[Edit]Special job to be run once at time of upgrade. Use this process to reload the necessary ODS composite tables based on changes made in the ODS 8.3 release.

[Load All Banner ODS Products \[Details\]](#)  
[Edit]Use this process to load all ODS composite tables with extracted data from all Banner products.

[Refresh All Banner ODS Products \[Details\]](#)  
[Edit]Use this process to refresh all ODS composite tables with extracted data from all Banner products.

[Load Accounts Receivable \[Details\]](#)  
[Edit]Use this process to load data in all Accounts Receivable ODS composite tables with extracted Banner data.

## Create Subprocess Info/PROCESS\_LIST records

Descriptions are included for each subprocess that is delivered with the warehouse. To define a Subprocesses description for a new job, create a Parameter record with values defined as follows:

## SCHEDULING\_PAGE records

Parameter records with an **Internal Code Group** = *SUBPROCESS INFO* and **Internal Code 2** = *SCHEDULING\_PAGE* use the **Description** field to define information specific to the subprocess that will display in the Process Info section of the Schedule a Process page illustrated in the following picture.

## Schedule a Process

To submit this subprocess to the job queue, fill in the information below, then click Submit.

\* - indicates a required field.

### Process to Schedule

**Schedule Enrollment Management Mappings**  
**Load/Refresh Student Engagement Aggregate**

### Process Parameters

#### Process Info

Make sure that the Student Progress Aggregate is up to date before the Student Engagement Aggregate is loaded.

#### Scheduling Parameters

To schedule this job to run immediately, click [here](#) or type "NOW" into the Run Date and Run Time fields.

**Run Date (dd-mon-yyyy):\***  [Select a Date](#)

**Run Time (hh24:mi:ss):\***  [Select A Time](#)

## Create Subprocess Info/SCHEDULING\_PAGE records

To define information that will display for Subprocesses in the Process Info section of the Schedule a Process page, create a Parameter record with values defined as follows:

### JOB Parameter

This parameter defines the actual program name of a job that gets sent to the job queue via the **Schedule a Process** administrative page in the Options menu.

This parameter is delivered with one entry for each process (job) that you can schedule. A Job is related to a SUBPROCESS and a “child” of one of the processes defined by the INSTALLED PROCESS parameter. To designate the parent/child relationship, match the **External Code** of the INSTALLED PROCESS to the **Internal Code 1** of the SUBPROCESS and the **Internal Code 1** of the JOB.

The following table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

**Internal Group:** JOB

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
Map to <b>External Code</b> of the INSTALLED PROCESS that is parent to this job.	Map to <b>External Code</b> of the SUBPROCESS related to this job.	N/A	The number of parameters that get passed to the mapping.	Actual program name (package.procedure) for the job. Refer to the mgkproc package for more information about submitting jobs.
MAPGROUP	LOAD_ALL	1	0	mgkmap.P_RunETLMapSlots
MAPGROUP	LOAD_ALL_SLOTS	1	0	mgkmap.P_RunETLMapSlots
MAPGROUP	LOAD_AR	1	0	mgkmap.P_RunETLMapSlots
MAPGROUP	LOAD_FINANCE	1	0	mgkmap.P_RunETLMapSlots

### Set up the JOB Parameter

You should not edit any of these entries. If your institution doesn't maintain one of the areas of Banner ODS data, you can delete all of the entries for that area.

### Create a Job Parameter Value

You can add a program developed by your institution to the Schedule a Process page. Create the program and define it by adding a new record for this parameter with the program name in the **Description** field.

### OWB Mappings and Slot Packages

OWB mappings are executed from the Administrative UI via the MGKMAP package. This package provides routines for running both OWB mappings and slotted table LOAD and UPDATE jobs. Refer to the MGKMAP package for more details.

The primary APIs used in the MGKMAP package are:

#### *P\_RunETLMapSlots:*

When a process/subprocess pair is passed to the procedure, it runs all mappings and slot package records associated with that process/subprocess combination. Specifically, if any **ETL Control Group** records are defined for the process/subprocess pair and the **Description** value is *Y*, then the procedure runs all Mapping and Slot Package records associated with those Control Group areas. If there are no **ETL Control Group** records associated the process/subprocess pair but there are individual mapping records associated with the pair, the procedure runs those Mapping and Slot Package records.

### **Example: P\_RunETLMapSlots**

As delivered, the ETL Control Group parameter records for all baseline systems have an External Code value of *Y*. This means data for all systems is loaded into Banner ODS when you submit the Load All Banner ODS Products and Refresh All Banner ODS Products processes. If you want to load only Student and Finance data into Banner ODS, set the **External Code** field to *N* for the ETL Control Group record for each of the other systems. The Mappings and Slot Packages will only run for Student and Finance when you submit the Load All Banner ODS Products and Refresh All Banner ODS Products processes.

#### *P\_RunETLMaps:*

When a process/subprocess pair is passed to the procedure, it runs all mappings associated with that process/subprocess combination. This API follows the same processing rules as *P\_RunETLMapSlots*, except that it only runs mappings; it does *not* run Slot Packages.

#### *P\_RunETLSlots:*

When a process/subprocess pair is passed to the procedure, it runs all Slot Packages associated with that process/subprocess combination. This API follows the same processing rules as *P\_RunETLMapSlots*, except that it only runs Slot Packages; it does *not* run mappings.

#### *P\_RunSingleMap:*

When a process/subprocess pair and mapping name are passed to the procedure, it runs that single mapping.

## **ETL MAP PACKAGE Parameter**

Hundreds of OWB mappings are used to load and refresh Banner ODS data. The ETL Map Package parameter defines groups of related mappings as one job. This allows you to quickly run just one job that, for example, loads all of the AR mappings.

This parameter is delivered with one entry for each mapping. The actual program name for the mapping occupies the Internal Code 2 and Description fields and is associated with an ETL group name in the Internal Code 1 field.

### **Example**

When you run the `LOAD_AR` job using the Schedule a Process option in the Administrative UI, the mappings associated with each ETL Map Package entry that has an Internal Code 1 of `LOAD_AR` is run. The External Code field contains the Location value defined for the mappings in OWB. These values are defined at mapping deployment time (usually at install) and are generally not modified.

The following table shows the entries for ETL Map Package entries that have an Internal Code 1 value of `LOAD_AR`. The following table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

## Internal Group: ETL MAP PACKAGE

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
The ETL group to which the mapping is assigned.	Mapping name.	Order to run the mapping within the group.	OWB locations of the mapping.	Mapping name.
LOAD_AR	LOAD_MTT_ACCOUNT	1	ODS_TARGET_AR	LOAD_MTT_ACCOUNT
LOAD_AR	LOAD_MTT_ACCOUNT_DE TAIL	2	ODS_TARGET_AR	LOAD_MTT_ACCOUNT_DE TAIL
LOAD_AR	LOAD_MTT_APPLICATION_OF_PAYM	3	ODS_TARGET_AR	LOAD_MTT_APPLICATION_OF_PAYM
LOAD_AR	LOAD_MTT_AR_DEPOSIT S	4	ODS_TARGET_AR	LOAD_MTT_AR_DEPOSIT S
LOAD_AR	LOAD_MTT_CONTRACT	5	ODS_TARGET_AR	LOAD_MTT_CONTRACT
LOAD_AR	LOAD_MTT_EXEMPTION	6	ODS_TARGET_AR	LOAD_MTT_EXEMPTION
LOAD_AR	LOAD_MTT_INSTALLMEN T_PLAN	7	ODS_TARGET_AR	LOAD_MTT_INSTALLMEN T_PLAN
LOAD_AR	LOAD_MTT_LEDGER_ACC OUNTING	8	ODS_TARGET_AR	LOAD_MTT_LEDGER_ACC OUNTING

### **Set up the ETL MAP PACKAGE Parameter**

You shouldn't change any of the delivered values for this parameter.

### **Create an ETL Map Package**

You can define a new ETL group of mappings by creating a new set of related ETL Map Package parameter entries. Create one new entry for each mapping in the group using the following steps.

1. Create new entries with a new ETL Group name in the **Internal Code 1** field.
2. Specify in the **Internal Code 2** and **Description** fields for the mappings you want to include in the group.
3. Specify the location of each mapping in the **External Code** field.

4. Specify the order in which to run the mappings in the **Internal Code Sequence Number** field.
5. Add the new ETL group to the list of subprocesses you can schedule by creating a new Subprocess parameter with the following values:
  - 5.1. **Internal Group:** Subprocess
  - 5.2. **Internal Code 1:** MAPGROUP
  - 5.3. **Internal Code 2:** blank
  - 5.4. **Internal Code Sequence Number:** Number indicating the order in which to run this subprocess.
  - 5.5. **External Code:** the new ETL Group name you created. This is the value in **Internal Code 1** in the ETL Map Package entries created above.
  - 5.6. **Description:** Name of the subprocess (ETL Group job) as it appears on the Select a Subprocess administrative page.

## ETL MAP PACKAGE LOAD PURGE Parameter

As part of the LOAD mapping Change Table Purge process, use this parameter to define the appropriate DELETE mapping for those LOAD\_x mappings that do not have an equivalent DELETE\_x counterpart, or where no Change table purge is required.

The MGKMAP package in Banner ODS (in the IA\_ADMIN schema which executes the various OWB mappings that make up a job) automatically runs the Purge process for each change table that is related to a particular Load mapping. The name of the change table and the PROCESS\_ID (a key field in the change table that identifies which records relate to a given mapping) are retrieved from the corresponding Delete mapping of the same name where LOAD\_x = DELETE\_x. For example, for the LOAD mapping LOAD\_MST\_STUDENT, the DELETE\_MST\_STUDENT mapping is used to identify the change table and process ID. However, occasionally there is no direct equivalent DELETE mapping for the LOAD mapping in context, or no change table purge is required. For example:

- Sometimes the mapping names do not match exactly (for example, LOAD\_MAT\_ORGANIZATION\_CONTACT and DELETE\_MAT\_ORGANIZATION\_CONT).
- LOAD mappings that require multiple DELETE mappings.
- LOAD mappings where change tables do not exist (such as the VALIDATION mappings) and subsequently no purge process is required.
- LOAD mappings are broken up across several sequential mappings (such as LOAD\_MFT\_TRANS\_HISTORY\_1, LOAD\_MFT\_TRANS\_HISTORY\_2,

LOAD\_MFT\_TRANS\_HISTORY\_3, and so on) and the change table purge process is only required to run once (DELETE\_MFT\_TRANS\_HISTORY).

In these cases, a Load Purge parameter is required to provide the MGKMAP package with the appropriate crosswalk information to designate what DELETE mapping(s) are required to run the Change Table Purge process, or when the Change Table Purge process should be ignored.

Any errors encountered when running the purge appear in the Load Control Report.

Use the following codes:

- Group Code: ETL\_MAP\_PACKAGE\_LOAD\_PURGE
- Internal Code: Enter the designated LOAD mapping
- External Code: Enter the designated DELETE mapping(s) or, enter NA to disable the Purge process for a given LOAD mapping.

The following table illustrates a sample of the values as delivered. This is just a sample. The first row gives a definition of each field.

**Internal Group:** ETL MAP PACKAGE LOAD PURGE

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
Name of LOAD mapping		Ignored	Name of DELETE mapping(s – note comma-separated if >1) or “NA”	Simple text to explain record’s purpose.
LOAD_MAT_ORGANIZATION_CONTACT		1	DELETE_MAT_ORGANIZATION_CONTACT	Load Purge Record
LOAD_MGT_VALIDATION_GENERAL		1	NA	Load Purge Record
LOAD_MFT_TRANS_HISTORY_1		1	DELETE_MFT_TRANS_HISTORY	Load Purge Record
LOAD_MFT_TRANS_HISTORY_2		1	NA	Load Purge Record
LOAD_MFT_TRANS_HISTORY_3		1	NA	Load Purge Record

## ETL MAP PACKAGE LOGIC Parameter

This parameter controls job processing if an error occurs during one of the mappings. By default, the MGKMAP package, which executes mappings for a job, runs all mappings for the job, regardless of whether they complete successfully. This assumes that there are no dependencies between mappings. Use this parameter to override processing logic. Specifically, if a parameter record exists with the ETL Map Package Logic group code, and the same Internal Code 1 (the job name) and Internal Sequence Number as the ETL Map Package record for the mapping for the job in question, and the External Code is set to “Terminate Job,” then the job stops if there is an error in that particular mapping.

## ETL MAP PACKAGE RECONCILE LOGIC Parameter

This parameter controls how the reconciliation process identifies LOAD mappings which do not follow the standard pattern (of one source Composite view equating to one Banner ODS Composite table). Those exceptions are notes by the External Code, being either:

- **IGNORE:** used to identify mappings *not* to try to reconcile
- **IGNORE COLUMN:** used to identify specific columns (1 parameter record per column) not to try to reconcile, where Int Code2 stores the column name
- **UNION:** used to identify Composite tables populated by multiple Composite views, in which case the name(s) of the related mappings are stored in the **Description** field.

**Internal Group:** ETL MAP PACKAGE RECONCILE LOGIC

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
Name of LOAD mapping	Ignored	1	Action to take	Either simple description or name of related LOAD mappings
LOAD_MAT_CONSTIT_ STAFF_ASSIGN	Ignored	1	IGNORE	Do not reconcile this mapping
LOAD_MST_STDNT_CR SE_ATT_STEP_1	Ignored	1	UNION	LOAD_MST_STDNT_ CRSE_ATT_STEP_2
LOAD_MTT_ACCOUNT_ DETAIL	OPERATING _DATE	1	IGNORE COLUMN	Do not reconcile this column

## ETL SLOT PACKAGE Parameter

The ETL Slot Package parameter is similar to the ETL Map Package parameter; it defines groups of related Slot Packages as one job. The difference is that the groups defined by the Slot Package parameter use the slot packages to load data into the slotted tables within Banner ODS. The groups of jobs defined by this parameter let you easily run one job that, for example, loads all of the Financial Aid slot slotted tables.

This parameter is delivered with one entry for each package that loads or updates data in a slotted table in Banner ODS. The actual program name for the slot package occupies the **Description** field and is associated with an ETL group name in the **Internal Code 1** field. For example, when you run the `LOAD_FINANCIAL_AID` job from the Schedule a Process option in the Administrative UI, the slot packages associated with each `ETL_SLOT PACKAGE` entry that has an **Internal Code 1** of `LOAD_FINANCIAL_AID` is run.

The following table shows the entries for ETL Slot Package entries that have an **Internal Code 1** value of `LOAD_FINANCIAL_AID`. The table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

**Internal Group:** ETL\_SLOT\_PACKAGE

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
The ETL group to which the mapping is assigned.	This field is not used.	Order to run the mapping within job group.	Package.procedure	Slot Package name.
LOAD_ FINANCIAL_ AID		1	MRKBCMP.P_POPULATE('L')	MRT_FINAID_BUDGET_ COMP_SLOT
LOAD_ FINANCIAL_ AID		2	MRKTREQ.P_POPULATE('L')	MRT_TRACKING_ REQUIREMENT_SLOT

### Set up the ETL SLOT PACKAGE Parameter

You should not change any delivered values for this parameter. If you want to define a new ETL group of slot packages, you can create new entries with a new ETL group name in the **Internal Code 1** field. Then specify the slot packages that you want to include in the group. Create one new entry for each package in the group.

## ETL CONTROL GROUP Parameter

This parameter gives you the ability to load or refresh all the data in your Banner ODS by running one job. The parameter is used in conjunction with the ETL Map Package parameter to further combine groups of jobs into one job.

As delivered, the ETL Control Group parameter defines which groups of job mappings, defined by the ETL Map Package parameter, to run when you run the `LOAD_ALL` and `REFRESH_ALL` jobs.

This parameter is delivered with one entry for each ETL group defined by the ETL Map Package parameter. The actual ETL group name (e.g., `LOAD_AR`, `LOAD_FINANCE`, `LOAD_GENERAL`, etc.) occupies the Internal Code 2 field. Each entry is associated with either the `LOAD_ALL` or `REFRESH_ALL` control group job in the **Internal Code 1** field. The External Code field for each record has the value `Y`, which means that all jobs (mappings) defined by the group are run when you run the `LOAD_ALL` job.

The following table shows the entries for the ETL Control Group when the value of **Internal Code 1** is `LOAD_ALL`. The table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

### Internal Group: ETL CONTROL GROUP

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
Control Group to which the ETL group is assigned.	ETL Group Name Map to <b>Internal Code 1</b> of the ETL MAP PACKAGE related to this ETL group.	Order to run the ETL group within the control group.	Controls whether to run the group. <code>Y</code> = run ETL group <code>N</code> = do not run ETL group	A description of the record. This values is not used in processing.
<code>LOAD_ALL</code>	<code>LOAD_ADVANCEMENT</code>	1	<code>Y</code>	Advancement Load ETL Control Record
<code>LOAD_ALL</code>	<code>LOAD_AR</code>	2	<code>Y</code>	AR Load ETL Control Record
<code>LOAD_ALL</code>	<code>LOAD_FINANCIAL_A ID</code>	3	<code>Y</code>	FinAid Load ETL Control Record
<code>LOAD_ALL</code>	<code>LOAD_FINANCE</code>	4	<code>Y</code>	Finance Load ETL Control Record
<code>LOAD_ALL</code>	<code>LOAD_GENERAL</code>	5	<code>Y</code>	General Load ETL Control Record

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
LOAD_ALL	LOAD_HUMAN_RESOU RCES	6	Y	HR Load ETL Control Record
LOAD_ALL	LOAD_STUDENT	7	Y	Student Load ETL Control Record

### ***Set up the ETL CONTROL GROUP Parameter***

Review all of the entries delivered for this parameter. If your institution doesn't maintain some of the areas of Banner ODS data, change the **External Code** value to *N* for those areas. For example, if your institution doesn't use Advancement and Human Resources, change the **External Code** value to *N* for entries that have **Internal Code 2** values of LOAD\_ADVANCEMENT or LOAD\_HUMAN\_RESOURCES.

### ***Create an ETL Control Group***

You can define a new ETL Control Group by creating a new set of related ETL Control Group parameter entries. Create one new entry for each ETL Group you want to include in the Control Group using the following steps.

1. Create new entries with a new Control Group name in the **Internal Code 1** field.
2. Specify in the **Internal Code 2** field the ETL Groups that you want to include in the Control Group.
3. Specify that you want to run each ETL Group by entering a *Y* in the **External Code** field.
4. Specify the order in which to run the ETL Groups in the **Internal Code Sequence Number** field.
5. Enter a **description** for the new Control Group entry.
6. Add the new ETL Control Group to the list of subprocesses you can schedule by creating a new Subprocess parameter with the following values:
  - 6.1. **Internal Group:** Subprocess
  - 6.2. **Internal Code 1:** MAPGROUP
  - 6.3. **Internal Code 2:** blank
  - 6.4. **Internal Code Sequence Number:** Number indicating the order in which to run this subprocess.

- 6.5. **External Code:** the new ETL Control Group name you created. This is the value in **Internal Code 1** in the ETL Control Group entries created above.
- 6.6. **Description:** Name of the subprocess (ETL Control Group job) as it will appear on the Select a Subprocess administrative page.

## EVENT parameter

An Event is a logical point in time when you extract information from the source system and load it into the Banner ODS, essentially freezing the data and giving you a snapshot of the data at that point in time. You can then use these data snapshots for reporting purposes. A logical point in time refers to a conceptual time, not an actual calendar date. For example, a logical time to extract financial information may be at the end of each month or for student registration data on the census date for the academic period.

Before you run the processes that load data into Banner ODS, you need to define events that are relevant for your institution's business needs. This parameter will create the Event parameter list when you Freeze Multiple Banner ODS Tables/Views.

## PARAMETER Parameter

The Parameter parameter is a processing parameter named "Parameter." This parameter defines the parameters that you must enter at runtime when you Schedule a Process. Basically, all values set up with the Internal Group of "Parameter" and the same Internal Code 2, display on the Schedule a Process page as the runtime parameters for the job defined by that Internal Code 2 value. The values of this parameter are stored in the MTVPARM table.

For example, when you freeze data in a Banner ODS table, you need to specify which table to freeze and the name you want to give the frozen table. Those two parameters are defined by the first two rows in the table below. This parameter is delivered with one entry for every process parameter. The following table illustrates a sample of the values as delivered. This is just a sample. The second row gives a definition of each field.

**Internal Group:** PARAMETER

Internal Code 1	Internal Code 2	Internal Code Seq.	External Code	Description
<b>External Code</b> of the process that is parent to this subprocess.	<b>External Code</b> of subprocess for this parameter.	Order for entries with same <b>Internal Code 2</b> .	Short description of parameter. This value is used in the processing code.  Short description of parameter. This value is used in the processing code.	Actual parameter field prompt that appears on <b>Schedule a Process</b> administrative page.
ADHOC_FREEZE		1	TABLE_NAME	Enter Table to Freeze
ADHOC_FREEZE		2	TABLE_HISTORY	Enter Table Name to Freeze to

 **Note**

If the Internal Code 2 field is left blank, the parameter appears for all subprocesses under the parent process in Internal Code 1. For example, the Enter Table to Freeze parameter does not have an entry for Internal Code 2. This parameter value appears for all subprocesses under the ADHOC\_FREEZE (Internal Code 1) area ■

**Set up the PARAMETER Parameter**

The only existing values you should change for this parameter are the descriptions. If you want to change the name of a parameter that appears on the Schedule a Process page, change its description.

If you want to add a process to the Schedule a Process page and it requires input parameters, you need to define the parameters by adding new values for this Parameter parameter.

**Create Runtime Parameters to Scheduled Processes**

You may add new processes to the Administrative UI that require runtime parameters, or you may want to add runtime parameters to existing processes, for example, a defined Freeze Data list. Create a new record for this Parameter to define a runtime parameter.

The easiest way to understand how to create a new runtime parameter is to review the existing ones. In the previous table, there are two records whose Internal Code 1 = FREEZE\_TABLE and the Internal Code 2 field is blank. Each of these records defines a runtime parameter that appears on the Schedule a Process page when Freeze Multiple Banner ODS Tables/Views is selected. The Internal Code 1 field of FREEZE\_TABLE on

the Parameter record here matches to the External Code of FREEZE\_TABLE on the INSTALLED PROCESS parameter.

### Example

If you add a runtime parameter to a freeze data list called TEST1, the following steps show the field values needed to create this new parameter.

1. Enter **Internal Group** = *PARAMETER*.
2. Enter **Internal Code 1** = *FREEZE\_TABLE*. The parent process for the TEST1 freeze data list.
3. Enter **Internal Code 2** = *TEST1*. The actual name of the freeze data list to associate the parameter.
4. Enter **Internal Code Sequence Number** = 2. The order that parameters are listed at runtime. You can add up to two parameters to a freeze data list.
5. Enter **External Code** = *ACADEMIC\_PERIOD*. The actual field value that you want the user to supply at runtime.
6. Enter **Description** = *Enter Term Code*. The prompt that a user needs to supply at runtime.
7. Choose **PARAMETER Type** = *SELECT*. Identifies how the user enters the runtime parameter. The field accepts four values:
  - **SELECT** = User must supply a valid PL/SQL statement.
  - **DATE** = User must supply a valid date.
  - **EDIT** = User can supply a text string.
  - **CHECKBOX** = User must check or uncheck an option.
8. Enter **PARAMETER SQL**. This field is only required when the **PARAMETER Type** is *SELECT*. Enter a valid PL/SQL statement, which is used to populate the valid field values to display in the drop-down list of the runtime prompt.
9. Enter **PARAMETER SQL Delimiter**. This field is only required when the **PARAMETER Type** is *SELECT* and you use a delimiter in the **PARAMETER SQL** field. Specify the delimiter used in the **PARAMETER SQL** field.

## Banner ODS Utilities

---

The Utilities process contains utility jobs or processes that perform various administrative tasks, and provide ongoing maintenance of the Banner ODS. For example, the Utilities

option enables you to compare the number of rows in one table or multiple tables in the source system with the number of rows in the composite tables in Banner ODS. You can also check for potential problems that may cause performance issues.

Once a job is completed, a control report is created. When discrepancies are found, the control report indicates the number of records found in each object, as well as the key values for the records that are not synchronized.

See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule processes.

## Report Banner ODS Source Change Table Counts

The Report Banner ODS Source Change Table Counts produces a control report that calculates how many rows are in each of the source system Change tables for each Banner ODS Composite table. This enables you to monitor the accumulation of Change records for a particular Composite table.

Depending on how many rows are in a Change table for a given Banner ODS Composite table, it may be more efficient to run a Load process instead of the Refresh process. Determining which process to run is a matter of individual experience with times for various Composite Tables.

### Change Tables and Control Reports

The row count totals that are reported in a Control Report for a given mapping reflect all rows that OWB accesses during the processing of that mapping. So, for a given DELETE mapping, OWB may SELECT 500 records, and then DELETE 500 records. Alternatively, if there is a filter condition applied in the mapping, the number SELECTED and DELETED (and/or UPDATED) may vary, based on the filter condition. Typically, these filter conditions occur to “filter” out change records associated with other processes. As an example, the SPRPCHG table processes many PIDM based tables, including SPRIDEN, SPBPERS, SPRADDR and SPRTELE. If the DELETE and UPDATE mappings for the Address process were running, and the SPRPCHG table had 100 records in it for different processes, the DELETE\_MST\_ADDRESS mapping would “select” all 100 records, but then it would only filter out and pass along those records where the field “SPRPCHG\_TABLE\_NAME” was equal to “ADDRESS”. Therefore, the delete would only “delete” those records, which would be the subset of the overall “selected” count.

The Change Tables contain the identifying data (sometimes the primary key value) about the records that are modified in the Banner system. Hence, if a record in a Banner table is edited 5 times, only one record will appear in the Change Table for it, which the Refresh process uses to know to bring the entire row of data over to the ODS for that record. Further, some of the Change Tables are used by multiple mappings, so the mappings will typically apply a filter when accessing the Change Table. This means that the row count totals in the Control Reports will usually not match up directly with the Change Table

counts, though there are some cases of Changes Tables that are used only by a single mapping and that mapping also doesn't apply filter conditions.

### Example

Addresses are managed through the ETL process by PIDM. If a person has 10 addresses and 3 of them change, then there is one record in the SPRPCHG table with the PIDM of the changed record, it has in it the most recent date/time of the last DML activity, and it also reflects the most recent DML action (C- change, D- Delete, or U- update). The delete mapping deletes all 10 addresses for that PIDM from the ODS even though there is only one record in the change table. The update mapping adds all the address records using the AS\_ADDRESS composite view where the PERSON\_UID field in the view matches the PIDM in the SPRPCHG table (ultimately re-adding all 10 addresses). Thus, a single change table entry results in 10 records being deleted and 10 records being inserted. The Change Table Counter process attempts to handle the change tables that are used by multiple mappings. That is, for a given DELETE mapping, it parses the mapping (PL/SQL) code for the default value of the second parameter (P\_TABLE) which it then uses to construct the following SQL code:

```
= 'SELECT COUNT(*) FROM ' || parm1Value || '@' || linkName || ' WHERE ' ||  
parm1Value || '_TABLE_NAME=' || parm2Value || '''; where parm1Value is the  
CHG_TABLE value and parm2Value is the P_TABLE value.
```

See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule processes.

## Synchronize Comments for Multiple Reporting Views

Run this process to generate comments on multiple reporting views. The meta data Business Definitions for each reporting view and the meta data business definitions for each of the columns is copied from the meta data into the database **Comments** field.

This process is scheduled from Banner ODS Utilities menu. (See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule a process.) The same functionality is available by selecting the **Sync Comments** link on the View Target Report page. (See [“Synchronize Meta Data Comments with Reporting Views” on page 2-147](#) for instructions.)

## Synchronize Comments for a Single Reporting View

Run this process to generate comments on a reporting view. The meta data Business Definition for the reporting view and the meta data business definition for each of the columns is copied from the meta data into the database **Comments** field.

This process is scheduled from Banner ODS Utilities menu. (See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule a process.) The same functionality is also available by selecting the **Sync Comments** link on the View Target Report page. (See [“Synchronize Meta Data Comments with Reporting Views” on page 2-147](#) for

instructions.)

## Banner ODS Checks and Balances

Banner ODS Checks and Balances utility processes can be run after an upgrade or intermittently to perform the following information checks.

- Check Mappings and Parameters  
Verifies that all Banner ODS Mapping packages have been created in the database and are valid. This process also confirms that all ETL MAP PACKAGE parameters have a corresponding DELETE\*, LOAD\* and UPDATE\* package (for example, LOAD\_MAT\_GIFT, UPDATE\_MAT\_GIFT, DELETE\_MAT\_GIFT).
- Check Metadata  
Compares the defined total of Banner ODS baseline delivered meta data records to a count of the records in Banner ODS to determine if the meta data records have loaded successfully
- Miscellaneous Checks  
Verifies that the database link to the source system exists and is working.
- Check Indexes  
Identifies the following:
  - Baseline Banner ODS indexes that are missing from the staged tables replicated from the source database.
  - Baseline Banner ODS indexes that are missing from change tables used to incrementally update the ODS composite tables.
  - Baseline Banner ODS indexes that are missing from the target database composite tables.

If there are local indexes you would like verified by this process, insert the appropriate data into the MGBINDX table.

- Freeze table changes  
As new versions of Banner ODS are released, Reporting views may have new columns added and, in some cases, existing column names changed. Therefore, if you have created freeze table data in earlier versions of Banner ODS, those table structures may become out of sync with newer versions of Reporting views, causing subsequent freeze processes to fail.

This process compares the table structure of any existing freeze table data against the current Reporting view, and any column discrepancies are reported. In addition, the appropriate Oracle 'ALTER TABLE' statement is also provided in the control report so you can resynchronize your freeze tables with the Reporting views.

- Check Triggers

Identifies any baseline table triggers that are missing from the staged tables replicated from the source database. The purpose of these triggers is to track changes that are needed to incrementally refresh the target database composite tables

See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule processes.

## Reconcile a Single Table

Reconcile a Single Banner ODS Table compares the number of records in a single Banner ODS composite table with the corresponding composite view in the source database. To display the .sql statement used in the process, check the **Show SQL** check box.

You can run this process at any time to verify that the source system and Banner ODS are synchronized. However, it is recommended that it is run directly after a LOAD or REFRESH, otherwise the counts will be off by the number of records in the change tables. You could also run the process during evening or non-processing hours to ensure that processing on the source system is not producing discrepancies in the reconciliation process.

The reconciliation process checks the source and Banner ODS objects dynamically. The process pulls SQL from the load mappings that are created and deployed from Oracle Warehouse Builder. Occasionally, Banner ODS tables were omitted from the reconciliation process because of the complexity of multiple sources of the mappings. These exceptions can be found in the Administrative UI, Set Up Parameters, under the Internal Group parameter ETL MAP PACKAGE RECONCILE LOGIC. The search displays a list of mappings that have been identified to ignore, or mappings that have multiple sources composite views.

1. Click **Options**.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Reconcile Single Tables**.
5. To display the .sql statement used in the process, check the **Show SQL** check box.
6. Select the **Reconciliation Type** drop-down list to choose whether to run the reconcile process by row counts or data.

### *Rowcounts*

Data is compared between the Banner ODS composite table and Banner composite view by counting the number of rows in each, based on the primary keys (which are

determined from the indexes on the composite views). If the row counts do not agree, a warning message displays in the Control Report indicating the row counts in both systems and the primary key values.

### **Data**

Every value in every column is compared between the two systems based on the primary key. Discrepancies and primary key information are displayed in the Control Report. The data reconcile process can require a lot more processing power and hence can run longer than row count.

7. Check the **Retain Output Table** check box to keep the temporary output table.

The data option uses a temporary output table to store results. This table is normally deleted when the process completes, but it can be retained (primarily for troubleshooting).

8. Enter the remaining required fields.
9. Click **Submit** to schedule the process to run.

See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule processes.

## **Transfer Banner Fine-Grained Access**

Use Transfer Banner Fine-Grained Access to transfer data for Banner Finance Fund, Fund Type, and Organizations, and for Banner Human Resources Organizations and Employee Class from Banner to Banner ODS.

### **Prerequisites**

- [“Set up and Maintain Organizational Areas” on page 2-19](#)
- [“Banner User ID Translations” on page 2-21](#)
- [“Set up Business Profiles” on page 2-24](#)
- [“Set up and Maintain Security Rules” on page 2-27](#)
- [“Policy Management” on page 2-48](#)
- [“Transfer Banner Fine-Grained Access” on page 2-99](#)

1. Click **Options**.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Transfer Banner Fine-Grained Access**.

5. Check the boxes that correspond to the fine-grained access security permissions you want to transfer.
6. Choose the **Transfer Mode** to use for the transfer.  
You can choose from the following modes:
  - REPLACE - Replaces all FGA rules of Banner in Banner ODS with new Banner FGA rules. Existing FGA rules in Banner ODS, created using Administrative UI, are not affected.
  - REPLACE\_ALL - Replaces FGA rules of both Banner and Banner ODS data with new Banner FGA rules. Existing FGA rules in Banner ODS that were not part of the current Banner data load are not affected.
  - TRUNCATE - Replaces all FGA rules, both Banner data and Banner ODS data, and refreshes Banner ODS with new Banner FGA rules.
7. Click **Submit** to schedule the process to run.

The transfer checks GUBINST to see if Banner Finance or Banner Human Resources is installed. If it is not, then a warning message displays.

Also, if Banner Finance is not installed, then the Banner Finance Fund/Organization transfer is bypassed. If Banner Human Resources is not installed, then both the Human Resources Organizations and Employee Class transfers are bypassed.

The data is transferred using the ODSMGR XXXX@SOURCE\_DB database link. Whether the data transfers to Banner ODS or not is based on whether security is turned on in Banner in the following areas:

- Finance: FOBSYSC\_FUND\_ORG\_SECURITY\_IND
- Human Resources Organization: PTRINST\_ORGN\_SECURITY\_IND
- Human Resources Employee Class: PTRINST\_ECLS\_SECURITY\_IND

In the Banner ODS Administrative UI (**Options** tab, **Set up Parameters** link) there are three process/job parameters under the Internal Group of BANNER TO ODS FGA TRANSFER which indicate whether or not Banner security settings should affect the job. By default, all three parameters are set to *N* (maintained on the Update a Parameter Administrative UI web page).

In Banner, FOBSYSC\_FUND\_ORG\_SECURITY\_IND, from the Banner FOBSYSC table, indicates whether or not Banner Finance Fund and Organizations security is active. For the Banner Finance Fund and Organizations transfer, the BANNER TO ODS FGA TRANSFER parameter with a value for Internal Code 1 of FINANCE FUND/ORG SECURITY ACTIVE determines whether not to consider the value of FOBSYSC\_FUND\_ORG\_SECURITY\_IND. If the job parameter has an external Code of *Y*, then this indicates that Banner Finance Fund and Organizations security must be turned on for the Fund/Org transfer to occur. If it is not turned on, a warning message is displayed

and the Banner Finance Fund and Organizations transfer is bypassed. If the external Code is *N*, then this indicates to go ahead and run the Finance Fund/Org transfer, regardless of whether Banner Fund/Org security is active.

The BANNER TO ODS FGA TRANSFER parameters with internal Codes of HR ORG SECURITY MUST BE ACTIVE and HR ECLS SECURITY MUST BE ACTIVE perform the same function against PTRINST\_ORGN\_SECURITY\_IND and PTRINST\_ECLS\_SECURITY\_IND respectively. After these parameters have been evaluated, the transfer begins.

At this point the data transfer begins and MGBXWLK is checked. The way that MGBXWLK has been configured, i.e., which of the four set up options you have chosen, determines how value-level data is written to MGBFGAV and column-level data is written to MGBFGAE.

The Finance Fund/Org transfer reads data from FORUSFN and FORUSOR and transfers user permissions for individual Funds and Organizations to IA\_ADMIN.MGBFGAV based on the fine-grained access rules in IA\_ADMIN.MGBFGAR. For users who have access to either all Finance Funds or all Finance Organizations, data is read from FOBPROF and is written to IA\_ADMIN.MGBFGAE. For users who have Fund Type permissions on FORUSFN, the fund numbers associated with each Fund permission, as listed by fund in the Banner ODS Fund Hierarchy table MFT\_FUND\_HIERACHY, are also written to MGBFGAV.

The Banner Human Resources Organizations transfer reads data from PSRORGN and transfers user permissions for individual Organizations to IA\_ADMIN.MGBFGAV based on the fine-grained access rules in IA\_ADMIN.MGBFGAR. For users who have access to all Banner Human Resources Organizations, data is read from PTRUSER and written to IA\_ADMIN.MGBFGAE.

The Banner Human Resources Employee Class transfer reads data from PSRECLS and transfers user permissions for individual Employee Classes to IA\_ADMIN.MGBFGAV based on the fine-grained access rules in IA\_ADMIN.MGBFGAR. For users who have access to all Banner Human Resources Organizations, data is read from PTRUSER and written to IA\_ADMIN.MGBFGAE.

See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule processes.

## Cleanup Reconcile Tables

You can clean up the reconcile tables that exist in your database by running the **Cleanup Reconcile Tables** job. This job allows you to identify and remove any extraneous

reconcile temporary tables that are no longer needed. Perform the following steps to clean up reconcile tables.

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Cleanup Reconcile Tables**.
5. Select the **Reconcile Tables to Clean Up** from the list of tables. Use Shift-click to select a contiguous list of table or Ctrl-click to select noncontiguous tables.

For your reference, the table listing displays the number of records in each temp table and the date the table was created.

6. Enter a **Run Date** (format dd-mon-yyyy) and **Run Time** (format hh24:mi:ss). Enter NOW in each field to run the job immediately.
7. Click **Submit** to run the job.

## Reconcile a Group of Tables

You can create a Reconcile Group, which defines a group of tables that you can reconcile by running one job via the **Reconcile a Group of Tables** option. This gives you the flexibility to reconcile a specific set of tables that you choose rather than all of the tables in a particular module, which is available by running the **Reconcile Multiple Tables** job. For example, you might create a Reconcile Group to reconcile only the Student Admissions related tables.

### Create a Reconcile Group

You can create as many Reconcile Groups as you want. You create a Reconcile Group by defining a Parameter record for each table that you want to include in the group. Once you create a group, it will display for selection on the Schedule a Process page when you select **Options>Schedule a Process>Banner ODS Utilities>Reconcile a Group of Tables**.

Define a Reconcile Group by creating Parameter records as follows:

1. Click **Options** from the Administrative UI menu.
2. Click **Set Up Parameters**.
3. Click **Create**.
4. Enter the information for a Group Reconcile parameter as follows:

Field	Value
Internal Group	RECONCILE_GROUP  The Internal Group value must always be RECONCILE_GROUP for every Reconcile group parameter you define. This value identifies the reconcile group jobs to display on the Schedule a Process page when you run the <b>Reconcile a Group of Tables</b> job.
Internal Code 1	<code for your group name>  Use the same value here for all records in one group. This shared value ties the records together as one Reconcile Group. (example uses REC_STU_GRP1)
External Code	<the table name to reconcile>  One of the tables that you want to include in the Reconcile Group you are defining.
Description	<a descriptive name for the reconcile group>  Use the same value here for all records in one Reconcile Group. This shared value displays in the Reconcile Group list on the Schedule a Process page.

5. Click **Duplicate** to add another record for the Reconcile Group.
6. Change the **External Code** value to another table that you want to include in the Reconcile Group you are defining.
7. Click **Save**.
8. Repeat steps [5](#) - [7](#) to add as many tables to the Reconcile Group as you need.

### Example

The following picture illustrates the values for a Reconcile Group named REC\_STU\_GRP1.

### Select a Parameter

---

Click a Description in the table below to select the Parameter you want to update or delete, or change the search criteria and click Search.

RECONCILE\_GROUP    -Show All Internal Codes-    -Show All Internal Codes 2-    Search

[Cascade >](#)                      < [Cascade](#) >                      < [Cascade](#)

Internal Group	Internal Code 1	Internal Code 2	Internal Code Sequence	External Code	Description
RECONCILE_GROUP	REC_STU_GRP_1			MST_SPORT	<a href="#">Student Reconcile Group 1</a>
RECONCILE_GROUP	REC_STU_GRP_1			MST_TAX	<a href="#">Student Reconcile Group 1</a>
RECONCILE_GROUP	REC_STU_GRP_1			MST_TEST	<a href="#">Student Reconcile Group 1</a>

[ [Create a New Parameter](#) | [Set Up a Parameter](#) ]

## Run a Reconcile Group job

After you define a Reconcile Group by creating the necessary Parameter records, the job will display in the Reconcile Group dropdown list on the on the Schedule a Process page when you select.

### Schedule a Process

---

To submit this subprocess to the job queue, fill in the information below, then click Submit.

\* - indicates a required field.

#### Process to Schedule

**Banner ODS Utilities**  
**Reconcile a Group of Tables**

#### Process Parameters

<b>Reconciliation Type*</b>	ROWCOUNTS - Compare Row Counts
<b>Reconcile Group*</b>	REC_STU_GRP_1-Student Reconcile Group 1

#### Scheduling Parameters

To schedule this job to run immediately, click [here](#) or type "NOW" into the Run Date and Run Time fields.

<b>Run Date (dd-mon-yyyy):*</b>	<input type="text"/>	Select a Date
<b>Run Time (hh24:mi:ss):*</b>	<input type="text"/>	Select A Time
<b>Interval:</b>	<input type="text"/>	Select An Interval
<input type="button" value="Submit"/>	<input type="button" value="Reset All Fields"/>	<input type="button" value="Export"/> <input type="button" value="Import"/>

Perform the following steps to run the Reconcile Group job.

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Reconcile a Group of Tables**.
5. Choose the **Reconcile Group** that you want to run.
6. Choose the **Reconciliation Type**.

You can reconcile a group in either ROWCOUNT or DATA mode. The DATA mode reconcile process does a complete row-by-row, column-by-column compare of data.

Depending on the amount of data involved, the reconcile process can take a long time to complete in this mode.

7. Enter a **Run Date** (format dd-mon-yyyy) and **Run Time** (format hh24:mi:ss). Enter NOW in each field to run the job immediately.
8. Click **Submit** to run the job.

## Reconcile temporary table names

When you reconcile tables in DATA mode, the Reconcile process uses one unique temporary (temp) table for each composite table that it is reconciling. The reconcile process creates temp table names by combining the composite table name with a leading “CP\_<module code character>\_” prefix. For example, the temp table created when reconciling the MST\_PERSON table would be named CP\_S\_PERSON. As a result, when running Group reconciles, if you select the job option to “Retain Temp Table”, one “CP\_” table will exist in the database (under the ODSMGR schema) for every table reconciled in that group process. If there are no discrepancies found during the reconcile process (i.e., the temp table is empty) then the empty temp table is deleted when the reconcile process completes.

In the past, the temp table was specific to the job instance (i.e., the table name included the job number). This meant that when doing a Group Reconcile the same temp tablename was reused, which limited the ability to track each reconcile item within the job.

## Reconcile Multiple Tables

Reconcile Multiple Banner ODS Tables compares the number of records for all Banner ODS composite tables (by subject area) with the corresponding Banner source composite views.

You can run this process at any time to verify that the source system and Banner ODS are synchronized. However, it is recommended that it you run it directly after a LOAD or REFRESH, otherwise the counts will be off by the number of records in the change tables. You could also run the process during evening or non-processing hours to ensure that processing on the source system is not producing discrepancies in the reconciliation process.

The reconciliation process checks the source and Banner ODS objects dynamically. The process pulls SQL from the load mappings that are created and deployed from Oracle Warehouse Builder. Occasionally, Banner ODS tables are omitted from the reconciliation process because of the complexity of multiple sources of the mappings. These exceptions can be found in the Administrative UI, Set Up Parameters, under the Internal Group parameter [“ETL MAP PACKAGE RECONCILE LOGIC Parameter” on page 2-88](#). The

search displays a list of mappings that have been identified to ignore, or mappings that have multiple sources composite views.

1. From the main menu, click **Options**.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Reconcile Multiple Tables**.
5. To display the .sql statement used in the process, check the **Show SQL** check box.
6. Select the **Reconciliation Type** drop-down list to choose whether to run the reconcile process by row counts or data.

### **Rowcounts**

Data is compared between the Banner ODS composite table and Banner composite view by counting the number of rows in each, based on the primary keys (which are determined from the indexes on the composite views). If the row counts do not agree, a warning message displays in the Control Report indicating the row counts in both systems and the primary key values.

### **Data**

Every value in every column is compared between the two systems based on the primary key. Discrepancies and primary key information are displayed in the Control Report. The data reconcile process can require a lot more processing power and hence can run longer than rowcount mode.

7. Enter the remaining required fields.
8. Click **Submit** to schedule the process to run.

See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule processes.

# Materialized Views

---

A materialized view (MVIEW) is the actual physical implementation of the logic within a reporting view. Within the Administrative User Interface, you can create a materialized view based on its underlying reporting view. The system performs all the calculations and business logic of the reporting view and stores the resulting data in a Banner ODS database table.

A materialized view is a snapshot of the data from the source database at the time the view was materialized or later refreshed. The nightly refresh process will update the data in a materialized view to make it current as of that last build.

The benefit of creating materialized views is that queries run against a materialized view can perform significantly faster than queries run against the underlying reporting view because the calculations required by the base reporting view have already been performed and stored in the database.

By contrast, some reporting views are very complex. They access multiple tables joining data from other views and using business logic like database functions that require many computations. Each time you execute a query against a complex reporting view the system needs to perform the joins and the business logic of that view, which can result in longer run and response times for the query.

Each table associated with a materialized view is a simple table like one you would create from performing a “CREATE TABLE AS SELECT \* FROM <view>” operation. However, the MVIEW table has various internal database attributes, such as indexes, that are hidden. The MVIEW also allows you to index the columns most likely used for reporting and to use query rewrite in the database.

## Materialize a View

Materializing a reporting view simply means creating a materialized version of the view and replacing the existing reporting view with the materialized view. The materialized view keeps the same name as the reporting view so that all existing reporting accesses, such as reporting tool meta data and reports, continue to function correctly regardless of whether or not the view is materialized.

The reporting view and its associated materialized view cannot exist in the database at the same time. For this reason, when you create a materialized view, the source code for the associated reporting view is stored in the IA\_ADMIN.MGBMVEW table. This table also stores specific information, such as elapsed time and row count, about the materialized view. This information is used to create and refresh the materialized version of a view. When you delete a materialized view, the system uses the information about that view stored in the MGBMVEW table to recreate the reporting view. Then the row related to that view is deleted from the MGBMVEW table.

You perform the tasks of creating, refreshing, and deleting materialized views using the Administrative User Interface (UI). It is possible to use the API calls defined in the MGKODSU package to manipulate the materialized views outside of the Administrative UI. However, any changes you make directly to the package or the APIs are not supported.

## Materialized View Considerations

Though you can create a materialized version of any existing reporting view, there are a number of factors to consider when deciding whether to convert a view from a reporting view to its materialized version. Converting a large or complex reporting view that performs slowly to a materialized view can improve reporting performance. However, the improved performance will introduce additional processing time to create and refresh the materialized view and it will require extra disk space to store the materialized data.

You'll want to consider these factors when determining which reporting views to materialize. In addition, you need to weigh these considerations against other standard performance tuning techniques (like creating additional indexes on underlying tables, specifying Oracle database parameter settings, and performing hardware upgrades) when deciding which reporting views to materialize.

## Materialized Views and Patch Releases

There will be times when you need to apply a patch release which includes changes that affect the reporting views. If a reporting view has been materialized in the system and you apply a patch release, application of the patch may fail since Oracle won't overwrite an existing materialized view with a reporting view of the same name. In these cases, you will need to delete the materialized view before you apply the patch release then recreate the materialized view.

Refer to the documentation delivered with a particular patch release for information about dependent objects. That documentation will direct you on how to determine which materialized views you need to delete and then recreate before and after applying the patch.

## Create an Index of Materialized View

The ODS Materialized View Create process can create indices on the materialized view. The value set for the Recommended Search Columns in the ODS Meta Data determines whether to create a materialized view index. Generally, the Recommended Search Column value(s) are set in the meta data for that reporting view before it gets materialized, so that when it get materialized (during the CREATE process), indexes are added appropriately.

If you want to create an index of a materialized view, where no Recommended Search Columns are defined, you can add them prior to creating the materialized view.

To create an index of the materialized view, perform the following steps:

1. Log on to the Administrative User Interface.
2. Click the **Meta Data** tab.
3. Click **Maintain Banner ODS Meta Data** link. The View Target Report List page opens.
4. Choose the reporting view for which you want to create an Index.
  - 4.1. Click the **Select** link next to the Subject Area.
  - 4.2. Choose a Report Type and Subject Area.
  - 4.3. Select the view for which you want to add an index.

For example, to choose the ACADEMIC\_STUDY reporting view, click the **Select** link next to the Subject Area, choose Student, and select the ACADEMIC\_STUDY reporting view.

5. Click the **Properties** link.
6. In the **Recommended Search Columns** field, enter the columns you want to be in the index. The columns list must be separated by a comma and then a space. If you want more than one index, you can do that too, by entering <BR> and then the next column list.
7. Click **Save Changes**. This will save a local version of the metadata and not affect any baseline data. The Admin UI will show you have local changes by displaying the headers in a different color. (You can refer to other definitions to see examples.)
8. Create the Materialized View. It will generate an index for the columns that you specified. For more information on creating materialized view, see [“Create a Materialized View”](#).

## Create a Materialized View

A materialized view (MVIEW) is the actual physical implementation of the logic within a reporting view. Within the Administrative User Interface, you can create a materialized view based on its underlying reporting view. The system performs all the calculations and business logic of the reporting view and stores the resulting data in a Banner ODS database table. You may want to create a materialized view because queries typically run faster against a materialized view. Perform the following steps to create one or more materialized views using the Administrative User Interface (UI).

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Create Materialized Views**.
5. Select the **Reporting Views** that you want to materialize. Use Shift-click to select a contiguous range of views or Ctrl-click to select noncontiguous views.

There is a 3800 character limit on the values you can select. As you select values, you can see the total “Selected size” in the status bar at the bottom of the window. If you want to choose values that total more than 3800 characters, break the selections into multiple groups and run the jobs separately.

6. Enter a **Run Date** (format dd-mon-yyyy) and **Run Time** (format hh24:mi:ss). Enter NOW in each field to run the job immediately.
7. Click **Submit**.

When the job completes running, you can review its activity by clicking **View Control Reports** at the bottom of the Select a Process page.

## Maintain a Materialized View

Typically, your institution implements a process each night that updates the Banner ODS data to synchronize it with the current source Banner data. This process runs the refresh jobs that update any existing materialized views that have been created, changed or deleted since the last refresh. You may want to refresh a materialized view on a more timely basis. Refreshing the view repopulates the associated database table with current data. Perform the following steps to refresh a materialized view on an as needed basis using the Administrative User Interface (UI).

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Maintain Materialized Views**.
5. Select the **Materialized Views** that you want to refresh. Use Shift-click to select a contiguous range of views or Ctrl-click to select noncontiguous views.

There is a 3800 character limit on the values you can select. As you select values, you can see the total “Selected size” in the status bar at the bottom of the window. If you

want to choose values that total more than 3800 characters, break the selections into multiple groups and run the jobs separately.

6. Select Refresh from the **Materialized Views Action** drop-down list.
7. Enter a **Run Date** (format dd-mon-yyyy) and **Run Time** (format hh24:mi:ss). Enter NOW in each field to run the job immediately.
8. Click **Submit**.

When the job completes running, you can review its activity by clicking **View Control Reports** at the bottom of the Select a Process page.

## Delete a Materialized View

Deleting a materialized view removes it from the database and replaces it with the original reporting view. Perform the following steps to delete a materialized view using the Administrative User Interface (UI).

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Maintain Materialized Views**.
5. Select the **Materialized Views** that you want to delete. Use Shift-click to select a contiguous range of views or Ctrl-click to select noncontiguous views.

There is a 3800 character limit on the values you can select. As you select values, you can see the total “Selected size” in the status bar at the bottom of the window. If you want to choose values that total more than 3800 characters, break the selections into multiple groups and run the jobs separately.

6. Select Drop from the **Materialized Views Action** drop-down list.
7. Enter a **Run Date** (format dd-mon-yyyy) and **Run Time** (format hh24:mi:ss). Enter NOW in each field to run the job immediately.
8. Click **Submit**.

When the job completes running, you can review its activity by clicking **View Control Reports** at the bottom of the Select a Process page.

## Materialized View Status Report

The Materialized View Status report (REPORT\_MVIEW\_STATUS) is a process that you can run to view basic information about each materialized view currently defined in the system. The status report includes the following information about each materialized view: creation date and time, elapsed creation time, and the number of rows created. If the view was refreshed, the report also includes the latest refresh date time, elapsed refresh time, and the number of rows refreshed.

### Report Materialized View Status

The Materialized View Status report includes information about each materialized view currently defined in the system. Run the report if you want to know when a materialized view was created or refreshed, how long it took to create or refresh a view, or how many rows were created or refreshed in a view. Perform the following steps to run the Materialized View Status report.

1. Click **Options** from the Administrative UI menu.
2. Click **Schedule a Process**.
3. Click **Banner ODS Utilities**.
4. Click **Report Materialized View Status**.
5. Enter *NOW* in the **Run Date** and **Run Time** fields.
6. Click **Submit**.

When the job completes running, you can review its activity by clicking **View Control Reports** at the bottom of the Select a Process page.

## Materialized View Control Reports

Each time you create, refresh, or delete a materialized view, the CREATE\_MVIEWS or MAINTAIN\_MVIEWS process creates a control report. The report includes information about creating, refreshing or dropping the materialized view and creating or dropping the related reporting view as appropriate. The report also includes the date and time that each action within the job completed.

You can save the control report as a .csv (comma separated values) file. You can open and review the .csv file in a spreadsheet application like Microsoft Excel. This option is especially useful for viewing large control reports. Click the CSV Summary button on the Display a Control Report page to view or save the report in .csv format.

For most reporting views, the following ORACLE command is used to create the associated materialized view:

```
create materialized view <view> BUILD IMMEDIATE REFRESH
COMPLETE ON DEMAND;
```

Some reporting views use different Oracle syntax to create the associated materialized view. The control report notes these view exceptions when the materialized view is created. Although a view may be created using alternate syntax, the resulting materialized view is functionally the same as the reporting view.

## View Control Reports

---

When a process runs, it creates a control report that details the progress, status, and errors in the process. Each control report highlights items like run time errors, record counts, and the job status for the process submitted.

Follow the steps below to review the control reports to determine whether a process ran successfully and to view errors.

1. From the Administrative menu, click **Options**.
2. Click **View Control Reports**. The Select a Control Report page opens.
3. On the Select a Control Report page, find the process you want to review in the list. Check the Status column to see if the process ran successfully. If the status is **ERROR**, there was a problem with the process.
  - To sort the list of control reports click one of the column headings - Run Date, Job Number, Process, User ID, or Status.
  - To filter the list of control reports, select the filter button next to one of the column headings (Run Date, Process, User ID, or Status), select the filter values and click **Select**. You can only apply one filter at a time.
4. Click **Refresh Job Status Codes** to see the most current job status. Often a job status will change from Running to Completed.

The **Refresh Job Status Codes** button is helpful with jobs that have been terminated in the database (due to a shutdown, or other error, etc.). If a job is terminated in the database, it locks the status as Running on the View Control Reports page. Therefore, if you click this button you not only refresh all status codes, but also ensure that any Terminated status codes display correctly.

To delete a control report, select the corresponding checkbox in the Delete column. To select or clear all the control reports, click **Select All** or **Deselect All**.

5. To review additional information on how a process ran, click the link for that process from the Process column. The Display a Control Report page opens.

**5.1.** Click **View error message(s)** to view the first error message.

**5.2.** Click **Next error** to browse all errors for the job.

A description of each button on the Display a Control Report page appears below:

Button	Description
Filter Report	<p>To view only selected lines of a report.</p> <p>The Select Report Filters window opens. Check the box next to the filter phrases that you want to see in the report. (Select All and Deselect All links select or clear all lines on the report.) Click <b>Filter Report</b>.</p> <p>To redisplay the information, click <b>Show Detail</b>.</p> <p>To filter the report for certain leading words/phrases, enter one or more delimiter characters and click <b>Scan</b>. (The default delimiter is a colon.) The page then displays all the unique occurrences of text up to the first occurrence of that delimiter, which you can use to filter the Control Report.</p>
CSV Summary	<p>Click to save a control report to a .csv (comma separated values) file that you can open and review in a spreadsheet application like Microsoft Excel. This option may be especially helpful for viewing large control reports.</p> <p><b>Note:</b> This option is primarily intended to use with output from LOAD and REFRESH jobs, reporting the number of rows processed for each mapping. There are also CSV outputs specific to the Reconcile utilities, as well as the Change Table Counts utility process.</p>
Reschedule Process	<p>Click to open the Schedule a Process page and reschedule a process.</p>

## Error Messages

This section lists some of the error messages you may encounter on the control report for any process. Not all error messages are documented, so this is not a complete list.

## Banner ODS Checks and Balances Process

### **Warning: *Obsolete sequence numbers in MGBPSQL***

Reason: Each row in the IA\_ADMIN.MGBPSQL table should have a corresponding row in the IA\_ADMIN.MTVPARAM table matching on the sequence number. Any unmatched rows in MGBPSQL are reported

Action: Sequence numbers that exist in MGBPSQL but not in MTVPARAM should be deleted from the table

### **Error: *<mapname> is INVALID in the database***

Reason: A delivered ETL mapping (PL/SQL) package currently has an INVALID status, and will not run during any of the jobs.

Action: Recreate the mapping package in the ODSMGR schema of the database.

### **Warning: *<mapname> parameter does not have corresponding MAPPING***

Reason: Baseline ETL mapping packages, that have been created with Oracle Warehouse Builder, exist in the ODSMGR schema with a name starting with "LOAD\_", "DELETE\_", "UPDATE\_". Each package has a corresponding Parameter record with the same name. This warning indicates that a parameter exists for the specified mapping, but the actual package does not exist in the ODSMGR schema of the database.

Action: For baseline packages, create the ETL mapping package in the ODSMGR schema. If the mapping package does not exist in the database, use the Administrative UI to remove the parameter.

### **Warning: *<mapname> mapping package does not have corresponding parameter record***

Reason: The baseline Banner ODS ETL mapping packages, that have been created with Oracle Warehouse Builder, exist in the ODSMGR schema with a name starting with "LOAD\_", "DELETE\_", "UPDATE\_". Each package has a corresponding parameter with the same name. Without this parameter record, the mapping will not be run during any of the jobs.

Action: Create the parameter record for the <mapname> package, similar to the other ETL MAP PACKAGE parameter records. Note: If the mapping is a locally developed package, consider using a different naming standard (ex: 'MY\_LOAD\_%', 'MY\_DELETE\_%'), OR create a different schema for local modifications.

### **ERROR: *Parameters not loaded for Banner ODS mappings (ETL MAP PACKAGE)***

Reason: The mapping parameters for Banner ODS have not been created in MTVPARAM.

Action: Check with technical staff to create the missing entries.

**Warning: ---> <view name> is documented but does not exist**

Reason: This check will verify that all reporting views documented in the metadata actually exist in the database. The warning message reports views that do not exist in the database.

Action: Check with technical staff to create the missing view in the ODSMGR schema of the database.

**Warning for REPORTING View: <view name> WARNING: ---> MetaData column missing in view: <column name>**

Reason: Baseline reporting views are delivered with corresponding metadata for each view column. The column that is documented does not exist within the view.

Action: Check with technical staff to determine why the column is missing from the view, and recreate the view if necessary.

 **Note**

Client developed reporting views can be imported into the metadata using the Administrative User Interface. If the column should not be documented for a locally developed view, use the Administrative UI to remove the metadata. ■

**Warning for REPORTING View: <view name> WARNING: ---> View column missing in MetaData: <column name>**

Reason: Baseline reporting views are delivered with corresponding metadata for each view column. The column exists in the view, but is not documented in the metadata

Action: Check with technical staff to determine why the column is not documented in the metadata. Document the missing column with the Administrative UI.

 **Note**

Client developed reporting views can be imported into the metadata using the Administrative User Interface. If the column should not be documented for a locally developed view, use the Administrative UI to create the metadata. ■

**Baseline index <index\_name> is missing from table**

Reason: Delivered index names are stored in the IA\_ADMIN.MGBINDX table. Any missing indexes may impact Banner ODS performance and are reported

Action: Create the missing index to ensure optimum system performance.

**Additional index (index\_name) found for table**

Reason: Local indexes that do not exist in the IA\_ADMIN.MGBINDX table are reported.

Action: To eliminate the warning message from the control report, insert the index information into MGBINDX with local = YES.

**Warning: More than one database link found as source location for OWB**

Reason: Verify that only one source database is identified for the OWB.

Action: Remove or rename incorrect database links from Banner ODS database. (Search DBA\_DB\_LINKS where LINK\_NAME like '%SOURCE\_DB%' to identify these).

**WARNING: Use the Freeze Data Maintenance page to remove these columns from the freeze table <freeze\_table>**

Reason: It is possible to select columns to include in the freeze data. If a column that has been used in a freeze table is no longer valid in the source, a warning message is provided

Action: Use the Freeze Data Maintenance page in the Administrative UI to locate the freeze table and review the selected columns. Remove the obsolete columns from the selected columns list.

**Freeze Table <freeze\_table> does not exist. Used in Freeze Data List <freeze\_data\_list>**

Reason: Freeze data lists are created to freeze multiple tables

Action: Review tables in the Freeze Data Lists reported to determine why the freeze data has not been generated.

**ERROR: AR dblink test failed**

Reason: A query from AT\_AR\_DEPOSIT view in Banner database failed.

Action: If the database link is valid, verify that the listed view exists in Banner database

**ERROR: ADVANCEMENT dblink test failed**

Reason: A query from a single Advancement view in Banner database (AA\_CONSTITUENT) is done as a check that the system configuration is correct for Advancement ETL mapping packages to execute. The error message would indicate that the database link is incorrect or the view is not valid.

Action: Verify that Advancement views have been created in Banner database

**ERROR: FINANCE dblink test failed**

Reason: A query from a single Finance view in Banner database (AF\_PURCHASE\_ORDER\_ACCOUNTING) is done as a check that the system configuration is correct for Finance ETL mappings to execute. The error message would indicate that the database link is incorrect or the view is not valid.

Action: Verify that Finance views have been created in Banner database.

#### ***FINANCIAL AID dblink test failed***

Reason: A query from a single Financial Aid view in Banner database (AR\_AWARD\_BY\_PERSON) is done as a check that the system configuration is correct for Financial Aid ETL mappings to execute. The error message would indicate that the database link is incorrect or the view is not valid.

Action: Verify that Financial Aid views have been created in Banner database.

#### ***ERROR: COMMON dblink test failed***

Reason: A query from a single view in Banner database (AS\_PERSON) is done as a check that the system configuration is correct for Common ETL mappings to execute. The error message would indicate that the database link is incorrect or the view is not valid.

Action: Verify that this view and other General views have been created in Banner database.

#### ***ERROR: HR dblink test failed***

Reason: A query from a single Human Resources view in Banner database (AP\_REVIEW) is done as a check that the system configuration is correct for Human Resources ETL mappings to execute. The error message would indicate that the database link is incorrect or the view is not valid.

Action: Verify that Human Resources views have been created in Banner database.

#### ***ERROR: STUDENT dblink test failed***

Reason: A query from a single Student view in Banner database (AS\_COURSE\_CATALOG) is done as a check that the system configuration is correct for Student ETL mappings to execute. The error message would indicate that the database link is incorrect or the view is not valid.

Action: Verify that Student views have been created in Banner database.

## **Materialized View Control Report Messages**

Following are explanations of messages that you might see in a control report related to creating or maintaining a materialized view.

#### ***Warning: Creating materialized view: <viewName> with REFRESH WITH ROWID option due to lack of primary key constraint.***

Oracle is unable to determine a primary key when joining the data used in the view. The system must use the ROWID construct to uniquely identify rows of data.

**Note: Reporting View: <viewName> contains sub-query so temporary view used**

Oracle doesn't support creating a materialized view from a view that contains a nested sub-query. The system is creating a temporary view on top of the reporting view. It will then materialize the temporary view

**Note: Reporting View: <viewName> contains too many key columns - USING NO INDEX mode enabled.**

Oracle uses the columns from the view that it determines to be the "primary key" columns when building its internal index for data access. The number of natural key columns in some reporting views exceed the Oracle limit. When this happens, the system uses the NO INDEX syntax.

## Freeze Process

**Process failed, no mgbfrez records for this list**

The selected freeze table list does not have entries in MGBFREZ table. Click the **Freeze Data Maintenance** menu to review the tables included in the freeze list.

**Multiple owners for inputted table/view to freeze. Please precede table name with owner.**

The original table or view name exists in more than one schema. Verify which table the data should be selected from, and precede the table name with the owner name.

**Source table not found**

The original table was not found in the database.

**Warning, no data found to Freeze**

There are no rows in the original table, or the where condition caused no rows to be selected.

**\*\*\*Warning--Replace parameter is N and EVENT exists!!!- did not replace data**

Data has been previously frozen to the new table with the same event code. If the data should truly be replaced, submit the process with the **Replace parameter** checkbox checked. If the existing data should remain intact, use a different event name to freeze additional data into the new table.

## **Publish Meta Data (PUBLISH\_META\_DATA)**

### ***Configuration error: No script found for COPY\_SCRIPT parameter***

The location of the ftp script used to transfer the html files was not found in the MTVPARM table. Click the menu options of **Options** and **Set Up Parameters**, with the internal group = METADATA and the internal code = PUBLISH, to store the copy script.

### ***P\_MakeAllTarget - E\_NoTablesFound***

There were entries found in metadata tables

### ***P\_MakeAllTarget - E\_NoMetafileLoc***

The parameter record in MTVPARM does not exist. To create this records, click the menu options of **Options** and **Set Up Parameters** with the internal group = METADATA and the internal code2 = PUBLISH\_LOCATION.

### ***P\_MakeAllTarget - E\_NoUTLfileLoc***

The file location supplied in the parameter is not valid. Click the menu options **Options** and **Set Up Parameters** with the internal group = METADATA and the internal code2 = PUBLISH\_LOCATION to verify the correct location for the creation of the meta data files.

## **Reconcile (RECONCILE\_JOB, RECONCILE\_SINGLE\_JOB)**

If there are zero discrepancies, the number of rows in the source view match the number of rows extracted to Banner ODS table. Run a refresh (or load) for the mapping that has the discrepancies, then rerun the reconcile job.

### ***mapName has 'n' discrepancies***

There are 'n' differences between Banner and Banner ODS. (The message below provides additional details.)

### ***Banner ODS has 'n' rows while the source has 'n' rows. Key values are:***

### ***'n' rows while Banner has 'n' rows. Key values are:***

This indicates the key values for the rows in either Banner ODS or Banner that do not match to the other system. Use these key values to further diagnose the discrepancy.

### **Note**

If you run this reconcile process after the refresh process is run, records that have been updated (with changes noted in the change tables) may have caused the discrepancies - you can use the key values to confirm this. ■

**Mapping processes (DELETE\_mapping, UPDATE\_mapping, LOAD\_mapping, REFRESH\_mapping)**

OWB Runtime not running - waited for 'n' minutes...

**ETL Mapping Package record not found for mapping: <map name>**

Run Banner ODS Utilities - 'Checks and Balances' job to ensure that all parameter records exist and mapping packages are valid.

**Mapping not found - Please check the mapping name and location.**

Run Banner ODS Utilities - 'Checks and Balances' job to ensure that all parameter records exist and mapping packages are valid.

**No ETL CONTROL GROUP or ETL MAP PACKAGES found for this job.**

Check that records exist in MTVPARM table  
where mtvparm\_internal\_code\_group = 'ETL MAP PACKAGE'.

**No ETL SLOT PACKAGE entry found for this table: <table name>**

Check that records exist in MTVPARM table  
where mtvparm\_internal\_code\_group = 'ETL SLOT PACKAGE'.

## Oracle Warehouse Builder Runtime Audit Browser Integration

Oracle Warehouse Builder (OWB) provides a utility called the Runtime Audit Browser (RAB) that displays status information for mappings that have been run. You can use RAB to view in depth statistics and job analysis. (For more information on setting up RAB, refer to the OWB Installation documentation).

### Integration Setup

The Administrative UI can be configured to automatically link to the RAB for mappings that have been run. All you'll need to do is click a hyperlink from the control report to view RAB mapping information. A new browser window opens displaying the RAB information for that mapping. Follow the steps below to set up a parameter RAB\_URL:

1. Click **Options** from the Administrative UI menu. The Options page opens.
2. Click **Set Up Parameters**. The Set Up a Parameter page opens.

3. Select Internal Group *METADATA* and Internal Code *RAB\_URL* from the drop-down lists on the Set Up a Parameter page.

 **Note**

(If Internal Code *RAB\_URL* does not appear in the drop-down list, then click **Create** to create the parameter. See [“Set up Parameters” on page 2-54](#) for instruction on how to create this optional parameter.) ■

4. Click **Search**. The Select an Existing Parameter page opens.

The External Code on the Select an Existing Parameter page can be any value. (It is required, but ignored. You can enter a hyphen, for example.) The key is the **Description** column. It must be the URL for the RAB that you have installed and set up. It will be similar to the URL below:

```
http://<machine_name>/owbb/  
RABMapExecution.uix?event=navigate&p_type=PLSQLMap&repos=RUNREP
```

To access your URL, continue to the next step.

5. Open the RAB in another browser window. Copy your URL from the address bar in that window, and paste it into the **Description** column on the Select an Existing Parameter page.

 **Note**

The particular RAB address (“RABMapExecution.uix”) and the associated parameters need to match the above address, with the exception of the “repos” parameter, which should reflect the repository owner in your system (if it isn’t the default RUNREP schema/user). ■

## RAB Authentication

The integration is not complete in the sense of typical web-based “single-signon.” You must first sign into the RAB in that separate browser window before you can browse any of the mapping execution information. Once signed in, your RAB credentials are stored locally (in a cookie) in your browser so you can close the RAB window (after logging in).

 **Note**

Those cookie credentials are persistent, so future attempts to view RAB reports will succeed until you Log Out of your RAB session explicitly (via the Log Out link in the RAB window). ■

Subsequent links from the control report should take you directly to the mapping information for that report. Click the link after the OWB Audit Execution ID on the control report. The Runtime Repository page opens.

# Set up E-mail Notification

---

You can configure the Administrative UI to send an e-mail when a process (job) is completed. To do this, set up the following system parameters (MTVPARM records).

These parameters are not delivered. (You must create them. See the [“Set up Parameters” on page 2-54.](#)) E-mails are only sent if all parameters (except the Administration URL) are set up. No e-mail notification takes place until you set these parameters.

Column	Description
EMAIL_ADMIN _ URL	<p>The complete URL to connect to the Administration system. If this parameter is defined, the URL is included in the e-mail message which makes it easier for the recipient to log into the system. The message contents are:</p> <p><b>Subject:</b> &lt;Job Name&gt; Job Completion &lt;with Errors&gt; (where the Job Name is the job that ran, and “with Errors” is appended only for jobs that had errors.)</p> <p><b>Message:</b> This job has completed. Check the Administrative UI for more details.</p> <p>Job Name: &lt;Job Name&gt;</p> <p>Job User: &lt;Admin Username of account that ran the job&gt;</p> <p>Job Number: &lt;job number&gt;</p> <p>&lt;For jobs that run mappings:&gt;</p> <p>Job Execution Time:</p> <p>    Start of Mapping at (start time)</p> <p>    Process completed at (end time)</p> <p>&lt;If any errors occurred during the job, they are listed next as follows:&gt;</p> <p>Error Details:</p> <p>    Error ...</p>
EMAIL_FROM_ ADDRESS	<p>E-mail address in the <b>From</b> section of the e-mail. This is typically a server address. Required.</p>

Column	Description
EMAIL_LIST	E-mail address to receive a job notification message for all Administration jobs that complete. Create one parameter for each recipient address. By default, you only receive e-mail notification for jobs submitted by that account. If the Administrative UI user name that ran the job matches (not case sensitive) the name in the <b>Description</b> field for this parameter. Or, you can set up an email address to receive notifications for all jobs that are run by setting the <b>INTERNAL_CODE_2</b> field to <i>GET_ALL_JOBS</i> . Required
EMAIL_SERVER	The machine name of your SMTP server machine. Required

## Freeze Data Maintenance

Freezing data enables you to take snapshots of related data at any point in time and keep a static copy of that data. You may want to run data comparison reports at the same point-in-time (example: each month, semester, or year) To do this you will need to ‘freeze’ the data at each point-in-time. As you save these data slices over time, you will create a history (freeze) of the data on which to report. You can also associate that point-in-time with an event name, for example, YearEnd, MonthEnd, or SummerSession.

Banner ODS freezes data from a single table/view or from multiple tables/views. When the freeze data has been defined, the freeze process must be scheduled to run (refer to [“Freeze a Single Banner ODS Table/View” on page 2-130](#) and [“Freeze Multiple Banner ODS Tables/Views at the Same Time” on page 2-131](#)).

Use the Freeze Data Maintenance Option to:

- Set up Freeze Data list for Banner ODS tables/views
- Add additional Banner ODS tables to existing freeze lists
- Review events in existing Banner ODS freeze tables

## Set up Banner ODS Freeze Data Lists

A Freeze List is what Banner ODS calls for one or more tables/views that have related data to be frozen at the same time. The freeze process selects data from the source table/view, creates a table with the ‘history’ name supplied, and copies (freeze) the selected source data into the history table. By default, all the columns from the source table are copied to the freeze table. Click **Select Columns** to specify if only specific columns are required for the freeze.

### **Example**

During a student registration cycle it may be important to capture student courses weekly. First, you would create a freeze list called `STUDENT_COURSE_REGDATA`. The source data would then be selected from `STUDENT_COURSE`.

The data from the source is stored in a freeze table which could be named `STUDENT_COURSE_STATIC`, for example. The new table is created the first time the freeze is run. Any successive freezes for this freeze list reuses the static table.

### **Note**

It is recommended that your institution have a naming convention in place for freeze lists and freeze tables ■

There is an optional `WHERE` condition that allows you to qualify the data to be frozen from each source table. The condition is `ACADEMIC_PERIOD = '200510'`.

### **Note**

Do *not* include the actual word `WHERE` in the condition; it is assumed. ■

1. Click **Options**.
2. Click **Freeze Data Maintenance**. The Set Up Freeze Data Lists page opens.

3. Click **Create** from the Set Up Freeze Data Lists page. The Create a Freeze Data Table page opens.

The links on this page are described below:

Link	Description
View Current Lists	Opens a window of the current freeze lists.
Copy Table Name	<p>Copies the source name to the <b>Freeze Table Name</b> field.</p> <p><b>Note:</b> The freeze table name must be different than the source name.</p> <p>The data is frozen with this table.</p>
Select Columns	<p>Opens a window of existing freeze columns. Choose the column(s) to freeze or not freeze, then click the corresponding arrows to move them to the appropriate box.</p> <ul style="list-style-type: none"> <li>• Click a single arrow to move one column.</li> <li>• Click a multi-arrow to move all columns.</li> <li>• Hold down the Ctrl key while selecting to move a few columns.</li> </ul> <p>The number of columns selected out of the total number of columns appears on the page and in the window. For example, (178/181) indicates that 178 columns out of 181 will be frozen.</p>

4. Enter the new list name, source name and freeze table name.
5. (Optional) Enter a valid PL/SQL WHERE condition. Use fields from the table or view being frozen and exclude the word “where”, which is added by the system.

**Example**

ACADEMIC\_PERIOD= '200510' and COURSE\_LEVEL = '01'

6. Click **Save**.

## Add a Table/View to a Banner ODS Freeze Data List

Maintaining freeze lists may require that additional tables be included in specific freeze lists, that a freeze list be deleted, that a freeze list be renamed or duplicated. It is also useful to review which events exist in which freeze tables.

In the example above, it is decided to capture data from `STUDENT_COHORT_SLOT` and so an additional table should be added to the `STUDENT_COURSE_REGDATA` freeze list.

- Click the freeze list called **STUDENT\_COURSE\_REGDATA** from the drop down list.
- Click **Add another Table**.
- Click **STUDENT\_COHORT\_SLOT** as the source table.
- For this example, the freeze table will be `STUDENT_COHORT_SLOT_STATIC`

There is an optional `WHERE` condition that will allow you to qualify the data to be frozen from each source table (`ACADEMIC_PERIOD = '200510'`). **NOTE:** Do *not* include the actual word `WHERE` in the condition. It is assumed.

1. Click **Options**.
2. Click **Freeze Data Maintenance**. The Set Up Freeze Data Lists page opens.
3. Choose the freeze list you want to modify from the drop-down list on the Set Up Freeze Data Lists page.
4. Click **Search**. The Select a Freeze Data Table page opens displaying the freeze tables associated with the displayed freeze list.
5. Click the link in the **Source Name** column for the tables/views you want to add. The Update an Existing Freeze Data Table page opens.
6. Click **Add Another Table**. The Create a Freeze Data Table page opens.

The links on this page are described below:

Link	Description
Select Another Lists	Returns to the Select a Freeze Data Table page.

Link	Description
Add Another Table	Opens the Create a Freeze Data Table page where you can enter freeze tables to add.
Select Columns	<p>Opens a window of existing freeze columns. Click the column(s) to freeze or not freeze, then click the corresponding arrows to move them to the appropriate box.</p> <ul style="list-style-type: none"> <li>• Click a single arrow to move one column.</li> <li>• Click a multi-arrow to move all columns.</li> <li>• Hold down the Ctrl key while selecting/deselecting to move a multiple columns.</li> </ul> <p>The number of columns selected out of the total number of columns appears on the page and in the window. For example, (178/181) indicates that 178 columns out of 181 will be frozen.</p>

7. Enter the new list name, source name and freeze table name.
8. (Optional) Enter a valid PL/SQL **Where Condition**. Use fields from the table or view being frozen and exclude the word “where”, which is added by the system.  
  
For example: *academic\_period = '200510'*.
9. Click **Save**.
10. Click **Add Another Table** to add another table to your list.

## Delete, Rename or Duplicate Banner ODS Freeze Data

Follow the steps below to delete, rename, or duplicate freeze data list.

1. Click **Options** from the Administrative menu.
2. Click **Freeze Data Maintenance**. The Set Up Freeze Data Lists page opens.
3. Choose the Freeze List you want to modify from the drop-down list on the Set Up Freeze Data Lists page.
4. Click **Search**. The Select a Freeze Data Table page opens. The list of tables currently included in the list displays.

Use the links on this page to delete, rename or duplicate a freeze list. Each link is described below:

Link	Description
Delete Freeze List	Confirms that you want to delete the displayed freeze list. Tables of frozen data will not be deleted.
Rename Freeze List	Displays the Rename a Freeze List window. Enter the new freeze list name, then click <b>Rename</b> . Tables of frozen data will not be renamed.
Duplicate Freeze List	Displays the Duplicate a Freeze List window. Enter the new name, then click <b>Duplicate</b> . None of the history tables are duplicated.

- (Optional) Check the **Show Event Names** checkbox to indicate whether to display the event within each table. An extra column of names displays.

 **Note**

You choose how these events are handled when scheduling a job by choosing to either to insert, delete or replace the events from the **Event Handling** field on the Schedule a Process page. ■

## Freeze a Single Banner ODS Table/View

You can freeze a single table using the Schedule a Process>Freeze a Single Banner ODS Table option. Follow the steps below:

- From the Administrative menu, click **Options**.
- Click **Schedule a Process**. The Select a Process page opens.
- Click **Freeze A Single Banner ODS Table/View** from the Select a Process page.
- Enter the required process parameters.
  - Type the name of a table/view into the **Enter Table to Freeze** field.
  - Type the new (history) table name into the **Enter Table Name to Freeze to** field. (Follow your history table naming conventions.)

5. Enter the required scheduling parameters.
  - 5.1. Enter a **Run Date** (format dd-mon-yyyy) and **Runtime** (format hh24:mi:ss).
  - 5.2. If you want to run the process on a recurring basis, enter an **Interval**. For example, to run a process every day at the same time enter *SYSDATE+1* in the **Interval** scheduling parameter.

See [“Update or Freeze Recurring Banner ODS Data” on page 2-135](#) for more details on setting the **Interval**.
6. Click **Save** to save the information about this freeze job. The job is entered into the job queue to run at the specified day and time.

## Freeze Multiple Banner ODS Tables/Views at the Same Time

If the freeze is going to occur repeatedly, it may be useful to create a Freeze List. The Freeze List is a name/label/title for one or more tables/views with data to be frozen at the same time. See [“Freeze Data Maintenance” on page 2-125](#) for instructions on how to define a list of freeze tables.

Follow the steps below to freeze multiple tables/views:

1. From the Administrative menu, click **Options**.
2. Click **Schedule a Process**. The Select a Subprocess page opens.
3. Click **Freeze Multiple Banner ODS Tables/Views** from the Select a Process page.

All freeze data lists defined within Freeze Data Maintenance display.
4. Click the freeze data list. The Schedule a Process page opens.
5. From the **Event Handling** drop-down list, indicate whether you want to replace, insert (add), or delete existing events from the tables in the freeze data list.
6. Choose an event to capture. The system tags the information extracted during this process with the event code you choose.

### Note

You have to choose an event name when you submit the freeze job to run (refer to the “System Parameters” section). Once that freeze job is run, the data exists in the freeze tables with an 'event' name attached. There could be multiple event names in a single freeze table. ■

7. Enter the **Run Date** (format dd-mon-yyyy) and **Runtime** (format hh24:mi:ss).

If you want to run the process on a recurring basis, enter an Interval. For example, to run a process every day at the same time enter *SYSDATE+1* in the Interval scheduling parameter. See [“Update or Freeze Recurring Banner ODS Data” on page 2-135](#) for more details on setting the Interval.

8. Click **Save** to save the information about this freeze job. The job is entered into the job queue to run at the specified day and time.

## Create a dynamic Freeze List parameter

You can create dynamic parameters and dropdown lists of valid values for selection parameters associated with a Freeze List. When you run a Freeze List, you can then use the parameter to reduce the amount of data that is frozen. The value selected and the associated column name are then appended to the “where” clause for each table in the freeze list.

Setting up this functionality requires the following conditions:

- Only tables or views owned by the ODSMGR schema can be used as source for a Freeze List
- A column that will be used as the “target” for a parameter value must be present in all source objects in the Freeze List

You can create a new view to meet this condition (For example, you could create a new view joining STUDENT to PERSON\_DETAIL, adding ACADEMIC\_PERIOD to the new view (STUDENT\_PERSON) so only the relevant “person” rows are frozen.)

Use the following steps to set up a parameter for a Freeze List.

1. Create a Freeze List.
2. Select **Options>Set Up Parameters**.
3. From the **Internal Groups** dropdown list, select PARAMETER.
4. Click **Create** to create a new entry for the Freeze List.
5. Enter values to create a new Parameter record. Use the descriptions in the following table as a guide to define the new Parameter record.

Field	Value
Internal Group	PARAMETER
Internal Code 1	FREEZE_TABLE

Field	Value
Internal Code 2	<Freeze List Name>  Enter the name of the Freeze List you created in the step <a href="#">1</a> of this procedure.
Internal Code Sequence Number	Null or the sequence in which you want the parameter to display on the Schedule a Process page
External Code	<Source COLUMN_NAME to be tested>
Description	<Description of Process>  Enter the description of the process that will display on the Schedule a Process page.
System Required	No

The following image shows the page for the “Parameter” parameter after it was created, and then updated to choose Parameter Type and input the Parameter Value.

Update an Existing Parameter

To update this Parameter, make changes to the information below, then click Save.

\* - indicates a required field.

Internal Group: PARAMETER [Search More](#)

Internal Code 1: FREEZE\_TABLE

Internal Code 2: PAUL\_FRZ

Internal Code Sequence Number: 1

External Code: \*  -OR-

Description: \*

System Required?

PARAMETER Type:

PARAMETER Value(or SQL):

PARAMETER SQL Delimiter:

[Test SQL](#)

For SELECT parameter types the Parameter Value is the SELECT statement used to populate the dropdown list. In this example the statement was:

```
SELECT VALUE FROM VALIDATION WHERE TABLE_NAME = 'STVTERM' ORDER BY  
VALUE DESC
```

You can test the correctness of the SELECT statement by clicking the **Test SQL** link in the lower right corner of the page, resulting (when the arrow is clicked) in the image to the right.

The value stored as the External Code is added to the “Where” clause for each source in the Freeze list and tested for equality to the parameter value chosen from the list as the job is submitted.

### **Example - Freeze List parameter**

The easiest way to understand how to create a new runtime parameter is to review the existing ones. In the previous table, there are two records whose Internal Code 1 = FREEZE\_TABLE and the Internal Code 2 field is blank. Each of these records defines a runtime parameter that appears on the Schedule a Process page when Freeze Multiple Banner ODS Tables/Views is selected. The Internal Code 1 field of FREEZE\_TABLE on the Parameter record here matches to the External Code of FREEZE\_TABLE on the INSTALLED PROCESS parameter.

If you add a runtime parameter to a freeze data list called TEST1, the following steps show the field values needed to create this new parameter.

1. Enter **Internal Group** = *PARAMETER*.
2. Enter **Internal Code 1** = *FREEZE\_TABLE*. The parent process for the TEST1 freeze data list.
3. Enter **Internal Code 2** = *TEST1*. The actual name of the freeze data list to associate the parameter.
4. Enter **Internal Code Sequence Number** = 2. The order that parameters are listed at runtime. You can add up to two parameters to a freeze data list.
5. Enter **External Code** = *ACADEMIC\_PERIOD*. The actual field value that you want the user to supply at runtime.
6. Enter **Description** = *Enter Term Code*. The prompt that a user needs to supply at runtime.
7. Choose **PARAMETER Type** = *SELECT*. Identifies how the user enters the runtime parameter. The field accepts four values:
  - SELECT = User must supply a valid PL/SQL statement.
  - DATE = User must supply a valid date.
  - EDIT = User can supply a text string.
  - CHECKBOX = User must check or uncheck an option.

8. Enter **PARAMETER SQL**. This field is only required when the **PARAMETER Type** is *SELECT*. Enter a valid PL/SQL statement, which is used to populate the valid field values to display in the drop-down list of the runtime prompt.
9. Enter **PARAMETER SQL Delimiter**. This field is only required when the **PARAMETER Type** is *SELECT* and you use a delimiter in the **PARAMETER SQL** field. Specify the delimiter used in the **PARAMETER SQL** field.

## Update or Freeze Recurring Banner ODS Data

You'll need to refresh the data in your Banner ODS on a regular basis to keep it synchronized with data in your administrative system. You may also want to freeze portions of Banner ODS data on a regular basis so that your users can create data comparison reports.

To automate the refresh or freeze processes, use the **Schedule a Process** option to define processes that run on a recurring basis. Specify that a job run on a recurring basis by entering a valid PL/SQL value in the **Interval** field. This field accepts a data expression value, which defines the length of time between processing runs. The key to setting the interval correctly is determining whether you need to run a job so that:

- Each execution of the job follows the previous run by a specific time interval.
- or**
- The job executes on specific dates and times.

The first thing you need to do is determine when and/or how often your institution needs to update Banner ODS data.

### Update Banner ODS Daily

It is recommended that Banner ODS is updated daily. Use the **Schedule a Process** option to define processes that run on a recurring basis. Specify that a job run on a recurring basis by entering a valid PL/SQL value in the **Interval** field. This field accepts a data expression value, which defines the length of time between processing runs. The key to setting the interval correctly is determining whether you need to run a job so that:

- Each execution of the job follows the previous run by a specific time interval.
- or**
- The job executes on specific dates and times.

The first thing you need to do is determine when and/or how often your institution needs to update Banner ODS data.

In this case, the interval value is a date arithmetic expression like *SYSDATE+N*, where *N* represents the time interval expressed in days. So, an interval of *SYSDATE+1* runs the job on a daily basis.

Job intervals set using date expressions do not guarantee that the next execution happens at a specific day or time, only that the spacing between executions is at least what was specified.

**Example**

If a job is first executed at 12:00 p.m. with an interval of *SYSDATE + 1*, it will be scheduled to execute the next day at 12:00 p.m. However, the job is executed manually at 4:00 p.m. using *DBMS\_JOB.RUN*, then it is rescheduled for execution at 4:00 p.m. the next day. Another example is when the database is down or the job queue is so busy that the job cannot be executed exactly at the time scheduled. In this case, the job runs as soon as it can, but the execution time then moves away from the original submission time due to the later execution.

**Update Banner ODS on Specific Dates and Times**

You can set the Interval to execute jobs on a specific date and time. This type of interval involves more complex interval date expressions. Specifying intervals like these can get tricky, so be sure that your date arithmetic expression is correct. The following table provides samples of both simple and more complex types of job intervals.

 **Note**

Refer to your Oracle documentation for more information on setting job intervals. ■

Run job	Interval Value
Daily	<i>SYSDATE+1</i>
Hourly	<i>SYSDATE + 1/24</i>
Weekly (every 7 days)	<i>SYSDATE + 7</i>
Every day at 12:00 midnight	<i>TRUNC(SYSDATE + 1)</i>
Every day at 8:00 a.m.	<i>TRUNC(SYSDATE + 1) + 8/24</i>
Every Tuesday at 12:00 noon	<i>NEXT_DAY(TRUNC(SYSDATE ), TUESDAY) + 12/24</i>
First day of the month at midnight	<i>TRUNC(LAST_DAY(SYSDATE ) + 1)</i>

Run job	Interval Value
Last day of the quarter at 11:00 P.M.	<i>TRUNC(ADD_MONTHS(SYSDATE + 2/24, 3), 'Q') - 1/24</i>
Every Monday, Wednesday, and Friday at 9:00 a.m.	<i>TRUNC(LEAST(NEXT_DAY(SYSDATE, MONDAY), NEXT_DAY(SYSDATE, WEDNESDAY), NEXT_DAY(SYSDATE, FRIDAY))) + 9/24</i>

## Meta Data

Meta data is “data about data” or information, or characteristics, about data entities such as a column name, description, format, length, origin and destination.

Meta data in Banner ODS tells what data columns are in Banner ODS , a definition of their business use, the type of data (number, character, date, etc.), how long they are, where they come from (in the source system) and their destination (in the target system.)

The Administrative UI meta data pages include reports that show the relationship between the data stored in Banner ODS and the source from which it is extracted.

### Note

The meta data includes Banner ODS reporting views and source composite views, both with the original source tables and source column names. Banner ODS recreated Object:Access views are *not* delivered in the meta data. They are additional reporting views to be used for clients migrating from Datamart 1.0, or clients who used the source Object:Access views for custom reporting. Newly developed Banner ODS reporting should *not* use the Object:Access views. ■

The following navigation links and buttons display throughout the Administrative UI meta data pages:

This Link/Button ...	Does this ...
<-	Moves through the subject areas in alphabetical order.
->	Moves alphabetically through the views within a subject area.
	Moves through the columns within a view in column ID order.
Select	<p><b><i>From the Subject Area:</i></b></p> <p>Opens the Select A Subject Area window.</p> <p>Click the <b>Target</b> or <b>Source</b> radio group, and click <b>Reporting View</b> or <b>Composite View</b> radio group to indicate the report type with which you want to work.</p> <p>Choose the new subject area with which you want to work. The window closes automatically.</p> <p><b><i>From the Reporting or Composite View:</i></b></p> <p>Opens the Select A Target window.</p> <p>Choose the reporting or composite view with which you wish to work. The window closes automatically.</p> <p><b><i>From the EDW star Target report List</i></b></p> <p>Opens the Select a Report Type and a Star window. Choose the target or source view and select the star. The window closes automatically.</p> <p><b><i>From the Reporting or Composite View Column:</i></b></p> <p>Opens the Select a Target Column or Select a Source Column window.</p> <p>To have the columns listed alphabetically, click the <b>Sort By: Column Name</b> radio group. To have the columns listed in column order, click the <b>Sort By: Column Order</b> radio group. Click the column with which you wish to work. The window closes automatically.</p>
Add Target	Adds a target view or table. The Target window opens.
Add Source	Adds a source view. The Source window opens.

This Link/Button ...	Does this ...
Add Target Column	Add a target column. The Add a New Column window opens
Add Source Column	Add a source column. The Add a New Column window opens
List Composite Views	Displays the composite views for the selected subject area.
List Banner ODS Reporting Views	Displays the reporting views for the selected subject area.
Preview	Save, then click <b>Preview</b> to review your changes. Do not click this link then click the back button. Or, click <b>Preview</b> to review the list of: <ul style="list-style-type: none"> <li>• all reporting or composite views within a subject area</li> <li>• a list of all source, or target columns within a view</li> <li>• a list of fact/dimension tables within a star</li> </ul>
Properties	Works in conjunction with the <b>Columns</b> link to toggle between the Edit Target Columns and the Edit Target Properties pages.
Columns	Works in conjunction with the <b>Properties</b> link to toggle between the Edit Target Properties and the Edit Target Columns pages.
Preferences	Opens the Institutional Preferences window.
Publish	Publishes the meta data.
CSV Export	Exports meta data into a .csv file that you can open in Microsoft Excel.
Import	Enables you to choose a view to import into the meta data. The view must exist in the ODSMGR schema. All the columns in the view are created as LOCAL meta data. Click the button to display a list of views that do not exist in the meta data for that subject area.
Show Baseline/Hide	Toggles between displaying baseline information versus local information.

This Link/Button ...	Does this ...
Save Changes	Records your changes.
Select Another Maintenance Function	Returns to the Maintain Banner ODS Meta Data page.

## Baseline and Local Meta Data

Baseline meta data is the meta data delivered with your solution. When you change the baseline meta data, a local copy is created and the edited version becomes your local meta data. Local meta data appears on the Administrative UI page in the color specified in your Institutional Preferences. The **Local Record** field on the Edit Target (or Source) Columns pages indicates whether the displayed meta data is the baseline or local version.

If both local and baseline meta data exist for the column meta data, only the local meta data displays and can be edited. Only local meta data can be changed or deleted.

## Create Meta Data

When Banner ODS is installed, the baseline meta data is installed as well. The sections “Set Up Meta Data Publish Preferences” and “Meta Data Parameter Set Up for Publishing Reports,” describe procedures that were completed during installation. They are included here for completeness, but you do not have to perform them to create meta data. The maintaining target or source meta data sections describe how to update the meta data repository with your own meta data.

## Set up Institutional Meta Data Publish Preferences

The Meta Data Publish Preferences option controls which pieces of meta data can be previewed on the screen and saved (published) in a report. Meta data is considered ‘published’ after you save the selected source or target information as an HTML file using the Administrative UI. Before you publish meta data, follow the steps below to set the preferences.

1. Click **Preferences & Security** from the Administrative menu. The Preferences & Security menu opens.
2. Click **Institutional Preferences**. The Set Up Banner ODS Publishing Options menu opens.
3. Click **Banner ODS Publishing Preferences**. The Set Up Meta Data Publish Preference page opens.

4. In the Banner ODS Meta Data Target Report Preferences area on the Set Up Meta Data Publish Preference page, check the checkbox to indicate the meta data you want to display in your meta data target or source reports. Your solution is delivered with the default check boxes selected.
5. Choose the color in which you want your report rows and local meta data information to appear.

 **Note**

Colors appear institution-wide. They are not personal colors. ■

6. Indicate whether reports should appear in column or name order.
7. Click **Save** to keep your changes.

## Meta Data Parameter Set up for Publishing Reports

The meta data reports are created as static HTML pages from the Administrative UI or from the command line. This process is called ‘publishing’. (See [“Publish Meta Data from the Administrative UI” on page 2-157](#) for additional information on publishing meta data reports from the command line.)

There are system parameters that must be configured when Banner ODS is installed. The PUBLISH\_LOCATION parameter provides the directory location on the database server where the HTML pages are written when using the Publish buttons from the Administrative UI, or by running the PUBLISH.BAT script in batch mode.

There are two supported techniques for specifying the location on the file system where the HTML pages are created:

- Use an Oracle DIRECTORY object

This is the preferred method as it does not require you to restart the database for changes to take effect, and also offers greater control over security. DIRECTORY objects are like any other objects in the Oracle database and offer the same levels of security control (grants by schema/user) while the UTL\_FILE\_DIR parameter setting is a global resource that does not offer tighter security control.

- Use the Oracle initialization UTL\_FILE\_DIR parameter

This technique has been replaced by the DIRECTORY usage but will be supported for backward compatibility.

When using an Oracle DIRECTORY, use the following syntax to create the directory object in the IA\_ADMIN schema:

```
CREATE DIRECTORY <DIRNAM> FOR '<PATH>';
```

where <DIRNAME> is a string, like *METADATA\_DIR* and <PATH> is the actual path to the folder/directory location where the files are created.

The DIRECTORY object should be created and owned by the IA\_ADMIN schema. The value of the PUBLISH\_LOCATION parameter would then be set to the DIRECTORY name (in the above example, the value: *METADATA\_DIR*).

You need to specify the initialization parameter UTL\_FILE\_DIR within the init.ora file for Banner ODS instance. This UTL\_FILE\_DIR parameter must contain the name of the directory where the Admin pl/sql package (MGKPUBL) generates the meta data files on the database server.

Once this directory is known and the UTL\_FILE\_DIR parameter is set, then configure the PUBLISH\_LOCATION parameter through the Administrative UI. (Follow the directions in the section [“Configure Publishing Parameters and Create Meta Data Web Directory”](#).)

The VIEW\_URL parameter provides the Web server location where the published files are hosted. It is recommended that you use the delivered /meta data folder to store the generated reports for viewing. This is a subdirectory beneath the “document root” for the Web server instance.

- Specify the VIEW\_URL parameter as a relative path to the document root.
- If the Oracle http server (Web server) is on a different computer from Banner ODS database server, then newly published reports must be copied from the PUBLISH\_LOCATION to the /metadata subdirectory before they can be viewed from the Operational Data Store Meta Data Reports page.

The COPY\_SCRIPT parameter allows you to specify a script to accomplish the moving HTML files from the application server to the web server.

The sample script delivered (ia\_admin\dbscripts\utility\_scripts\copyMetaData.sh) demonstrates how to do this using FTP, but the script can be replaced with any technique (such as SFTP, copying files directly using a mapped drive, even just copying them from one directory to another if the application server and web server are on the same machine, etc.). It is recommended that you examine and customize this script as needed to comply with your institutional security requirements and policies.

## Configure Publishing Parameters and Create Meta Data Web Directory

1. Login to the Administrative UI.

### Example

```
http://machine.sct.com:port/pls/ods/  
twbkwbis.P_GenMenu?name=bmenu.P_MainMnu
```

2. Click **Options** from the Administrative UI menu.

3. Click **Set Up Parameters**.
4. From the **Show All Internal Groups** drop-down list, select *METADATA*.
5. Click **Search**.
6. Look for the PUBLISH\_LOCATION, VIEW\_URL, or COPY\_SCRIPT parameter in the **Internal Code 2** column.
7. Click the corresponding link in the **Description** column.
8. Each link for the selected parameter appears in the **Description** field of the Update a Parameter page.

For parameter ...	Change this Directory or Server Location ...
-------------------	--

PUBLISH_LOCATION	UTL_FILE_DIR location or DIRECTORY name. Select this parameter to set up the location to which the meta data is published.
------------------	--

If the Web server is running on a Banner ODS machine, set up the UTL\_FILE\_DIR location (for output of generated pages) to be the same as the meta data subdirectory path under the Web server document root. PUBLISH\_LOCATION would be set to the same thing.

*Example:*

D:\ORACLE\UTL\_FILE

PUBLISH\_LOCATION may be case sensitive. The directory name on the Banner ODS database server should be in the same case as the UTL\_FILE\_DIR entry. If the case does not match, you may receive the error “Unknown Status: ERR\_UTL\_FILE” when attempting to Publish.

The description should correspond to the UTL\_FILE\_DIRECTORY setting.

*Example*

The `initSID.ora` file may contain this line:

```
utl_file_dir = D:\ORACLE\UTL_FILE
```

The database init parameter file (`initSID.ora`) typically resides in the Oracle Home\database directory (Windows) or the Oracle Home/dbfs directory (Unix).

VIEW_URL	The VIEW_URL parameter is set to the meta data subdirectory of the document root. This saves you from copying files each time they are published.
----------	---

*Example:*

\metadata

If the Oracle http server (Web server) is on a different computer from the ODS database server, then newly published reports must be copied from the PUBLISH\_LOCATION to the /metadata subdirectory before they can be viewed from the Banner Operational Data Store Meta Data Reports page.

COPY_SCRIPT	Script used to move HTML pages from the database server to the application Web server.
-------------	--



**Tip** On the Update a Parameter page, you can only change the **External Code** and **Description** fields. But, if you click **Duplicate** you can change

any of the fields. For example, to update the Internal Code you could duplicate the current one and change the Internal Code. Then, go back and delete the original parameter (to clean up). Click the back button (twice), then click **Delete**. ■

9. Click **Save** from the Update a Parameter page to save the new settings.
10. Return to the Select a Parameter page to set up a different parameter.

## Edit Target Meta Data Properties

Follow the steps below to change the properties of your target meta data.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** The View Target Report List page opens.

### Note

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group. Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

### Note

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. Click the reporting or composite view whose properties you want to change. The View Target Report page opens.
7. Click **Properties** located to the right of the reporting/composite view name. The Edit Target Properties page opens.
8. Make your changes.
9. Click **Save Changes** at the bottom of the page to keep your new information. The page refreshes automatically.

After the page refreshes, the **Local Record** field changes from *No* to *Yes* to indicate that this is now local meta data. The field names also display in the color that was set up in your Institutional Preferences page to indicate local meta data.

The **Show Baseline** and **Delete Local** links appear to the right of the **Local Record** field after you save.

## Add Target Views and Target Columns

Follow the steps below to add target reporting or composite views and target columns to a subject area.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

### Note

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.

### Note

Check the **Show locally modified targets only** checkbox to display local target views only. ■

5. Click **Add Target**. The Add Target window opens.
6. Enter the new target name.
7. Click **Add Target** to save the new view. The View Target Report page opens displaying the new target reporting or composite name.
8. Click **Add Target Column** to add columns to the view. The Add a New Column window opens.
9. Enter the new information, then click **Add Column** to save. The View Target Report page refreshes and displays the new target column information.

## Edit Target Views and Target Columns

Follow the steps below to change the information for target and reporting or composite views.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. Click the reporting or composite view you want to change. The View Target Report page opens.
7. Click the Target Column you want to change.  
The Edit Target Columns page opens.
8. Enter your changes. Click **Save Changes** to keep your changes.

## Synchronize Meta Data Comments with Reporting Views

Use this option to generate multiple comments on a reporting view. The meta data business definitions for the reporting view and the meta data business definitions for each of the columns is copied from the meta data into the database **Comments** field. Any existing comment will be overwritten.

This process (for a single or for multiple business definitions) can also be scheduled from Banner ODS Utilities menu. (See [“Schedule a Process” on page 2-60](#) for instructions on how to schedule a process.)

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data**. The View Target Report List page opens.
3. Click **Select** to choose the subject area, or click <- or -> to move through the subject areas in alphabetical order.

 **Note**

If you click **Select**, which opens the Select a Subject Area window, keep the default Reporting View and Target radio groups. The window closes automatically after you select a subject area. ■

4. Click the reporting view to which you want to add comments.
5. Click the **Sync Comments** link.

The business definitions are copied to the database comments.

## Delete Local Target Properties

Follow the steps below to delete local target properties:

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. Click the reporting or composite view whose target properties you want to delete. The View Target Report page opens.
7. Click **Delete**. A message window appears.
8. Click **OK** to delete the target, or **Cancel** to keep the target. If you delete the target, the View Target Report List page opens. If you keep the target, you remain on the View Target Report page.

## Delete Local Target Columns

Follow the steps below to delete local target columns.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data a**. The View Target Report List page opens.



### Note

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group. Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.



### Note

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. Click the reporting or composite view whose columns you want to delete. The View Target Report page opens.
7. Click the target column you want to delete. The Edit Target Column page opens.
8. Click **Delete Local**. A message window appears.
9. Click **OK** to delete the target, or **Cancel** to keep the target. If you delete the target, you return to the View Target Report page. If you keep the target, you remain on the Edit Definitions page.

## Edit Source Meta Data Properties

Follow the steps below to change the properties of your source meta data.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified sources only** checkbox to display local target views only. ■

6. Click the Source Name whose properties you want to change. The View Source Report page opens.
7. Click **Properties** located to the right of the source name. The Edit Source Properties page opens.
8. Make your changes.
9. Click **Save Changes** at the bottom of the page to keep your new information. The page refreshes automatically.

After the page refreshes, the **Local Record** field changes from *NO* to *YES* to indicate that this is now local meta data. The field names also display in the color that was set up in your Institutional Preferences page to indicate local meta data.

The **Show Baseline** and **Delete Local** links appear to the right of the **Local Record** field after you save.

## Add Source Names and Source Columns

Follow the steps below to add source names and source columns to a subject area for reporting and composite views

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. Click **Add Source**. The Add Source window opens.
7. Enter the new source name. Click **Add Source** to save the new name. The View Source Report page opens displaying the new source name.
8. Click **Add Source Column** to add columns to the source. The Add a New Column window opens.
9. Enter the new column information, then click **Add Column** to save. The Edit Source Columns page opens and displays the new source column information.

## Edit Source Names and Source Columns

Follow the steps below to change the properties of your source meta data for reporting and composite views.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

### Note

All subject areas on the View Target Report List page display in alphabetical order by default.

To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

### Note

Check the **Show locally modified Sources only** checkbox to display local source views only. ■

6. Click the source name you want to change. The View Source Report page opens.
7. Choose the source column you want to change. The Edit Source Columns page opens.
8. Enter your changes. Click **Save Changes** to keep your changes.

## Delete Local Source Properties

Follow the steps below to delete local source properties:

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

### Note

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.

4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified sources only** checkbox to display local target views only. ■

6. Click the source name whose source properties you want to delete. The View Source Report page opens.
7. Click the **Delete**. A message window appears.
8. Click **OK** to delete the source, or **Cancel** to keep the source. If you delete the source, the View Source Report List page opens. If you keep the source, you remain on the View Source Report page.

## Delete Local Source Columns

Follow the steps below to change the properties of your source meta data.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified sources only** checkbox to display local target views only. ■

6. Click the source name whose columns you want to delete. The View Source Report page opens.
7. Click the source column you want to delete. The Edit Source Column page opens.

8. Click the **Delete Local**. A message window appears.
9. Click **OK** to delete the source, or **Cancel** to keep the source. If you delete the source, you return to the Edit Source Column page. If you keep the source, you remain on the Edit Source Column page.

## Add and Delete Source to Target Meta Data Local Mappings

Meta data contains information about which source column in the source system contained the information that is in the target column. You can create your own local source to target meta data mappings.

Follow the steps below to add or delete local source to target mappings to the meta data:

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. Click the reporting view to map. The View Target Report page opens.
7. Click the target column you want to map. The Edit Target Columns page opens.
8. *To Add:*  
Click **Add Local Mapping** at the bottom of the web page. The Add a Source Mapping window opens. (Continue to the next step below.)

*To Delete:*

Click the **Delete Local Mapping** link at the bottom of the web page.

Click **OK** to delete the local mapping.

9. Enter the source subject area, table and column (required fields). Or, search for them using the corresponding links. Choose the table or column from the drop-down list associated with that link.
10. Click **Add Mapping** to save the newly mapped meta data.

## Import Target and Source Meta Data

The Import option enables you to import an entire view into the meta data. The view must exist in the ODSMGR schema. All the columns in the view are created as local meta data.

Follow the steps below to change the properties of your source meta data.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

### Note

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Click the subject area you want to view. The window closes automatically.

### Note

Check the **Show locally modified targets (or sources) only** checkbox to display local target (or source) views only. ■

6. Click **Import** located at the top right side of the web page. The Select a View window opens.
7. Click one or more views to import.

To choose more than one view, click the first view, the hold down the Ctrl key while selecting the remaining views.

8. Click **Import**.

## CSV Export

The Export option enables you to export target and source meta data into a .csv file that you can open in Microsoft Excel, or similar application.

Follow the steps below to change the properties of your source meta data.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data** . The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

3. Click **Select** on the View Target Report List page to choose the target subject area you want to edit. The Select A Subject Area window opens.
4. Click either the **Reporting View(s)** or **Composite View(s)** radio group . Then, click the **Target(s)** radio group.
5. Choose the subject area you want to view. The window closes automatically.

 **Note**

Check the **Show locally modified targets only** checkbox to display local target views only. ■

6. To export all reporting or composite views in a subject area, click **CSV Export** located at the top right side of the View Target (or Source) Report List page.
7. A window opens either to warn you that the operation will take a long time, or to indicate whether you want to save or open the file. Click **Cancel** to stop.

# Publish Meta Data from the Administrative UI

---

Meta data is considered ‘published’ after the selected source or target information is saved as an HTML file and a meta data report is created. And, it can be published for some or all sources and targets. Meta data enables users to easily view the relationships between Banner ODS columns and their sources. Meta data can be published from the Administrative UI, or from the command line outside the Administrative UI. Once a meta data report is published, it can be stored on a server that is accessible to reporting users.

## Note

If the Web server is not on the Banner ODS machine, the files need to be copied to the Web server after publishing. ■

## Publish Meta Data for an Entire Subject Area

Follow the steps below to publish meta data for an entire subject area (Student, Finance, etc.).

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data**. The View Target Report List page opens.

## Note

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

## Note

Check the **Show locally modified targets only** checkbox to display local target views only. ■

3. Click **Select** on the View Target Report List page. The Select A Subject Area window opens.
4. Click the **Target(s)** radio group.
5. Click the subject you want to view. The window closes automatically.
6. Click **Publish** located at the top right side of the web page.
7. Click **Ok** to confirm that you want to publish all reports for the subject area.

## Publish Meta Data for One Source or Target

Follow the steps below to preview and publish the meta data for one source or target.

1. Click **Meta Data** from the Administrative UI menu.
2. Click **Maintain Banner ODS Meta Data**. The View Target Report List page opens.

 **Note**

All subject areas on the View Target Report List page display in alphabetical order by default. To move through the subject areas in alphabetical order, click <- or ->. ■

 **Note**

Check the **Show locally modified targets only** checkbox to display local target views only. ■

3. Click **Select** on the View Target Report List page. The Select A Subject Area window opens.
4. Click the **Target(s)** radio group.
5. Click the subject you want to view. The window closes automatically.
6. Click the reporting view whose meta data you want to preview or publish.
7. Click **Preview** to open the View Target Report List page, and preview the report. The meta data is not permanently published until you complete the following step.
8. Click **Publish** at the top of the web page. An HTML file is published (saved as a report). The file is saved to the location specified by the parameters with an internal group *METADATA* and *internal\_code\_2= PUBLISH\_LOCATION*.

## Publish Meta Data Reports

---

Meta data can be published using three methods.

- Publish all meta data by scheduling a process. See [“Publish Meta Data by Scheduling a Process” on page 2-159](#)
- Publish for an entire subject area. See [“Publish Meta Data for an Entire Subject Area” on page 2-157](#)

- Publish for one source or target. See [“Publish Meta Data for One Source or Target” on page 2-158](#)

 **Note**

Baseline meta data reports are provided when your solution is installed. Therefore, you should *not* need to perform the publishing step initially. ■

## Publish Meta Data by Scheduling a Process

You can schedule meta data to publish at a predetermined day and time. Follow the steps in the [“Schedule a Process” on page 2-60](#) section. You should click the **Publish Meta Data** process.

## Publish Meta Data from the Command Line

You can publish all meta data reports using the MGKPUBL.P\_MakeAllReports procedure. A sample script, PUBLISH.SQL, is provided in the dbscripts/utility\_scripts for publish.sql. To generate all the meta data reports, use the following command:

```
SQLPLUS IA_ADMIN/<password> @PUBLISH.SQL
```

The following PUBLISH.BAT script (in the web\_files/metadata directory) can be customized to perform the entire process (generating the files, and then using FTP to put them on a remote server):

```
if "%1" == "move" goto movem
echo Publishing...
echo SET SERVEROUTPUT ON SIZE 500000 > doit.sql
echo EXEC MGKPUBL.P_MakeAllReports >> doit.sql
echo QUIT >> doit.sql
type doit.sql
sqlplus ia_admin/<password>@<Oracle database> @doit.sql
:movem
echo Moving...
if exist *.html del *.html
ftp -n -s:getfiles.dat <ODS machine>
ftp -n -s:putfiles.dat <web server machine>
```

## View Published Meta Data

Meta data is considered ‘published’ after the selected source or target information is saved as an HTML file and, as a result, a meta data report is created. There are two kinds of reports for reporting view and composite view meta data. They are target reports and source reports.

### *Target Reports:*

Show the relationship between the columns in Banner ODS reporting views (or composite views) and the columns to which they are mapped in the source system tables.

### *Source Reports:*

Show the relationship between columns in the source system tables and the columns to which they are mapped in Banner ODS reporting view (or composite view).

## Reporting View Meta Data

Use the following steps to view a published reporting view meta data report.

1. Click **Meta Data** from the Administrative menu.
2. Click Banner Operational Data Store. Banner Operational Data Store Reporting View Meta Data Reports page opens.
3. Choose a subject area from Banner Operational Data Store Reporting View Meta Data Reports page. The Reporting View Meta Data Reports page opens.
4. Choose a reporting view. The selected report displays.

### **Note**

Sometimes the number of targets in the source report can exceed a 30,000 character limit. If this happens the output for the source is cut off, and a message “(More Targets...)” displays. ■

## Composite View Meta Data

Banner ODS composite view meta data is also available as published meta data. Use the following steps to view published composite view meta data reports.

1. Click **Meta Data** from the Administrative menu.
2. Click **Banner Operational Data Store**. The Reporting View Meta Data Reports page opens.

3. Click **Banner ODS Composite View Meta Data Reports** located in the top right-hand corner of Banner Operational Data Store Reporting View Meta Data Reports page. The Banner Operational Data Store Composite View Meta Data Reports page opens.
4. Choose the subject area. The Composite View Meta Data Reports page opens listing the view name and description.
5. To view the column details associated with the selected composite view, choose one of the composite views. A report opens listing the Local Target, Target Column, Business Definition, Database Data Type, Source Name and Source Column.

## Metamodel

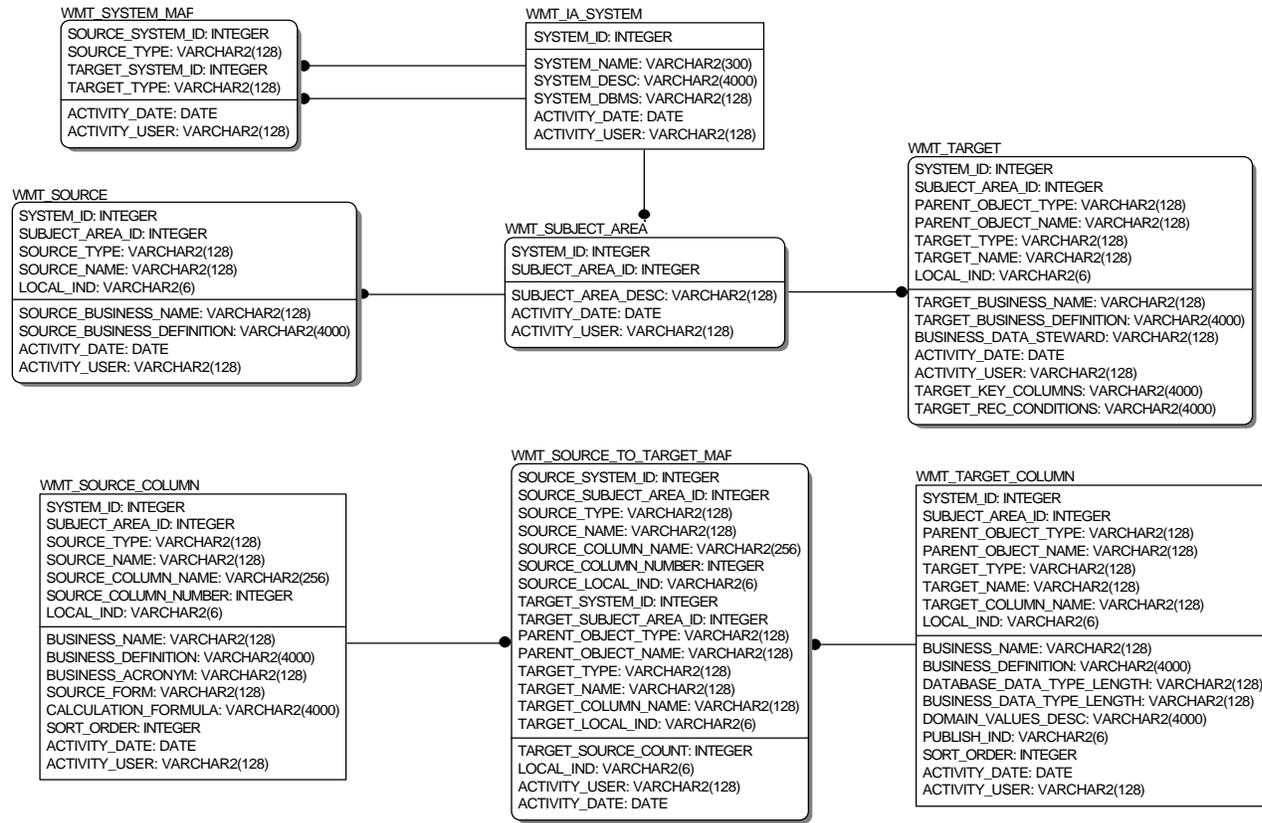
The delivered metamodel is the physical relational data model that stores the meta data. This should not be confused with the meta data repository, which refers to the physical database tables that contain the meta data.

Meta data tables are stored in a repository that is owned by the user - IA\_ADMIN. Each table in the meta data repository begins with a “WMT\_” prefix to identify it as a Banner ODS “Warehouse Meta Data Table.” In addition, there is a public synonym for each table that simply removes the “WMT\_” prefix.

The meta data tables and views that make up the metamodel illustrate the different pieces of meta data available, and how they relate to each object type. The object types are the reporting views and the source tables.

Meta Data Table Name	Synonym
WMT_IA_SYSTEM	IA_SYSTEM
WMT_SOURCE	SOURCE
WMT_SOURCE_COLUMN	SOURCE_COLUMN
WMT_SOURCE_TO_TARGET_MAP	SOURCE_TO_TARGET_MAP
WMT_SUBJECT_AREA	SUBJECT_AREA
WMT_SYSTEM_MAP	SYSTEM_MAP
WMT_TARGET	TARGET
WMT_TARGET_COLUMN	TARGET_COLUMN

A diagram of the metamodel follows:



These meta data tables that store information about the meta data are further described in the [“Banner ODS Meta Data Object Types”](#) section.

## Banner ODS Meta Data Object Types

Information exists in the meta data layer for the following types of objects:

Object	Description
Target View	Banner ODS reporting views that join related information from Banner ODS tables. Use these views to build reports.  <i>Example:</i> CONSTITUENT reporting view is the Advancement constituent data.
Source Table	Database tables in your source system used as the source for the data in Banner ODS.  <i>Example</i> APBCONS is the Constituent Base Table.
Source Function	Functions that use data from the source system's source tables to create new data to be stored in Banner ODS.

## Source Meta Data Tables

The following meta data tables store information about the source of Banner ODS data. In Banner ODS, this is meta data about the source systems.

### *Source Table (WMT\_SOURCE)*

Columns	Descriptions
System ID	Unique ID for the source system.
Subject Area ID	Unique ID for the subject area.
Source Type	Identifies whether the source is a table, view, or function. Sample source types are TABLE, REPORTING VIEW and FUNCTION.
Source Name	Source table, view, or function name.
Source Business Name	Source table, view, or function descriptive name.
Source Business Definition	Table or view business purpose.
Local Ind	Indicates whether the row is a local or baseline version.

Columns	Descriptions
Activity Date	Date the meta data was changed.
Activity User	User who changed the meta data.

**Source Column Table (WMT\_SOURCE\_COLUMN)**

Columns	Descriptions
System ID	Unique ID for the source system.
Subject Area ID	Unique ID for the subject area.
Source Type	Identifies whether the source is a table, view or function. Sample source types are TABLE, REPORTING VIEW, and FUNCTION.
Source Name	Source table, view or function name.
Source Column Name	Source table/view column name. If the source name is FUNCTION, the function name is entered. If the source name is CONSTANT, the value of the constant is entered. If the source name is CALCULATION, the calculation is entered.
Source Column Number	Distinguishes between source columns that have the same names.
Local Ind	Indicates whether the row is a local or baseline version.
Business Name	Descriptive name for the column in the source.
Business Definition	Source column defined in business terms.
Business Acronym	Acronym for the source column, if it has one.
Source Form	Source system form name from which the data was captured.
Calculation Formula	Any calculations that are applied to create the data in the target column.
Sort Order	Column order in the table or view. It is determined by numbering the columns in alphabetical order.
Activity Date	Date the meta data was changed.
Activity User	User who changed the meta data.

## Target Meta Data Tables

The following meta data tables store information about the target of Banner ODS data, Banner ODS reporting views.

### *Target Table (WMT\_TARGET)*

Columns	Descriptions
System ID	Unique ID for Banner ODS.
Subject Area ID	Unique ID for the subject area.
Parent Object Type	This column is used in Banner EDW only.  In the case of Banner EDW, the parent object type is STAR. Not used in Banner ODS.
Parent Object Name	This column is used in Banner EDW only.  In Banner EDW, this identifies the star to which the target belongs.
Target Type	Stores whether this is a Banner ODS table or view. Currently, reporting and composite view information is available.  Sample values for Banner ODS are REPORTING VIEW and COMPOSITE VIEW.  Sample values for the EDW are DIMENSION TABLE, FACT TABLE, and STAR.
Target Name	Table or view name.
Target Business Name	Target descriptive name.
Target Business Definition	Target business purpose.
Business Data Steward	Person or department responsible for the data in the target.
Local Ind	Indicates whether the row is a local or baseline versions.
Activity Date	Date the meta data was changed.
Activity User	User who changed the meta data.

Columns	Descriptions
Target Key Column	Describes how the data is to be returned when extracted, with any information and/or comments specific to this particular set of data.
Target Rec Conditions	Columns used in report filters and queries that return the best performance for the specified reporting view. These conditions are not mandatory, but recommended for performance. You may retrieve data from the reporting views using different criteria.

**Target Column Table (WMT\_TARGET\_COLUMN)**

Columns	Descriptions
System ID	Unique ID for Banner ODS.
Subject Area ID	Unique ID for the subject area.
Parent Object Type	This column is used in Banner EDW only. In Banner EDW, the parent object type is STAR.
Parent Object Name	This column is used in Banner EDW only. In Banner EDW, this identifies the star to which the target belongs.
Target Type	Stores whether this is a Banner ODS table or view. Currently, reporting view information is available. Sample values for Banner ODS are REPORTING VIEW and COMPOSITE VIEW.
Target Name	Table or view name.
Target Column Name	Target column name.
Local Ind	Indicates whether the row is a local or baseline version.
Business Name	Descriptive name for the column in the target.
Business Definition	Defines the target column in business terms. This is the comment on column in the relational database data dictionary in your target system.
Database Data Type Length	Comes from the relational database data dictionary in Banner ODS. This is stored in the meta data tables, not just the relational database data dictionary, so that it is easily available in one place with the rest of the meta data.

Columns	Descriptions
Business Data Type Length	Used when writing reports for formatting purposes. The business data type may be character, integer, float, etc. It also contains the length of the data.
<b>Example</b>	
The relational database data type and length for the internal ID may be varchar(63), but the business data type and length is 8-digits. Even though the database allows for a width up to 63 characters, the column will never be more than eight.	
Domain Values Desc	Description of the valid values that a column can contain. It could be a list of codes and code descriptions.
Publish Ind	Indicates whether to publish the column information to meta data reports so users can use the meta data for reporting purposes. It may not be published because the column contains sensitive information. The column may also contain technical information like a key that would not be used in a report.
Sort Order	Physical order of the columns in the table or view from the relational database data dictionary.
Activity Date	Date the meta data was changed.
Activity User	User who changed the meta data.

## Source and Target Meta Data Tables

The following meta data tables store information about the source and target of the data. This includes meta data about the source systems and Banner ODS.

### **System Table (WMT\_IA\_SYSTEM)**

Columns	Descriptions
System Id	Unique ID for a system.
System Name	Administrative source or Banner ODS solution system name.
System Desc	Administrative source or Banner ODS solution system description.
System DBMS	Database management system software, Oracle for example, used to implement the source or target system.

Columns	Descriptions
Activity Date	Date the meta data was changed.
Activity User	User who changed the meta data.

**Subject Area Table (WMT\_SUBJECT\_AREA)**

Columns	Descriptions
System ID	Unique ID for the system.
Subject Area ID	Unique ID for the subject area.
Subject Area Desc	Advancement, Student or Human Resources, for example.

**System Map Table (WMT\_SYSTEM\_MAP)**

Columns	Descriptions
Source System ID	Source system unique ID.
Source Type	Identifies whether the source is a table, view, or function. Sample source types are TABLE, REPORTING VIEW, and FUNCTION.
Target System Id	Banner ODS.
Target Type	Stores whether this is a Banner ODS table or view. Currently, reporting and composite view information is available. Sample values for Banner ODS are REPORTING VIEW and COMPOSITE VIEW.
Activity Date	Date the meta data was changed.
Activity User	User who changed the meta data.

**Source to Target Map Table (WMT\_SOURCE\_TO\_TARGET\_MAP)**

Columns	Descriptions
Source System ID	Source system unique ID.
Source Subject Area ID	Subject area unique ID.

Columns	Descriptions
Source Type	Identifies whether the source is a table, view or function. Sample source types are TABLE, REPORTING VIEW and FUNCTION.
Source Name	Source table, view or PL/SQL function name.
Source Column Name	Source column name from the source table or view, if the source is a table or view. If the source name is FUNCTION, the function name is entered. If the source name is CONSTANT, the value of the constant is entered. If the source name is CALCULATION, the calculation is entered.
Source Column Number	Distinguishes between source columns that have the same names.
Source Local Ind	Indicates whether the row is a local or baseline version.
Target System Id	Banner ODS unique ID.
Target Subject Area Id	Subject area unique ID.
Parent Object Type	This column is used in Banner EDW only. In Banner EDW, the parent object type is STAR.
Parent Object Name	This column is used in Banner EDW only. In Banner EDW, this identifies the star to which the target belongs.
Target Type	Stores whether this is a Banner ODS table or view. Currently, reporting and composite view information is available.  Sample values for Banner ODS are REPORTING VIEW and COMPOSITE VIEW.  Sample values for Banner EDW are DIMENSION TABLE, FACT TABLE, and STAR
Target Name	Table or view name.
Target Column Name	Column name in the target reporting view.
Target Local Ind	Indicates whether the row is a local or baseline version.
Target Source Count	Count indicates how many sources there are for a target.

Columns	Descriptions
Local Ind	Indicates whether the row is a local or baseline version.
Activity User	User who changed the meta data.
Activity Date	Date the meta data was changed.

## Reporting Meta Data Views

The following views exist in the meta data repository, and are owned by the user IA\_ADMIN.

Views	Descriptions
WMV Source	Lists all information associated with sources and source columns.
WMV Source To Target Map	Lists all information associated with sources, targets, and source and target columns.
WMV Target	Lists all information associated with targets and target columns.

Each view joins a specific combination of the data stored within the meta data tables. You can use these views to query and report the meta data information. They provide easier access to the meta data in the same way that Banner ODS reporting views provide access to the data in Banner ODS tables.

### **Source Meta Data View (WMV\_SOURCE)**

Views	Descriptions
System Name	Administrative source or Banner ODS solution system name.
System Desc	Administrative source or Banner ODS solution system description.
Subject Area Desc	Advancement, Student or Human Resources, for example.
Source Type	Identifies whether the source is a table, view or function. Sample source types are TABLE, REPORTING VIEW and FUNCTION.
Source Name	Source table, view or PL/SQL function name.
Source Business Name	Source descriptive name.

Views	Descriptions
Source Business Definition	Source business purpose description.
Source Column Name	Source column name from the source, if the source is a table or view. Function name if the source is a function.
Business Definition	Target column description in business terms.
Calculation Formula	Any calculations that are applied to create the data in the target column.
Sort Order	Column order in the table or view. It is determined by numbering the columns in alphabetical order.
Business Name	Column name in the source.
Business Acronym	Source column acronym, if it has one.
Source Form	Source system form name from which the data was captured.
Local Ind	Indicates whether the row is a local or baseline version.

**Source to Target Map Meta Data View (WMV\_SOURCE\_TO\_TARGET\_MAP)**

Views	Descriptions
Target System Name	Solution system name.
Target System Desc	Solution system description.
Target Subject Area Desc	Advancement, Student or Human Resources, for example.
Parent Object Type	This column is used in Banner EDW only. In the case of Banner EDW, the parent object type is STAR.
Parent Object Name	This column is used in Banner EDW only. In the case of Banner EDW this identifies the star to which the target belongs.

Views	Descriptions
Target Type	<p>Stores whether this is a Banner ODS table or view. Currently Reporting Views information is available.</p> <p>A sample value for Banner ODS is REPORTING VIEW.</p> <p>Sample values for Banner EDW are DIMENSION TABLE, FACT TABLE and STAR.</p>
Target Name	Table or view name.
Target Business Name	Target descriptive name.
Target Business Definition	Target business purpose.
Business Data Steward	Person or department responsible for the data in the target.
Target Column Name	Target column name.
Target Column Business Name	Target column descriptive name.
Target Column Business Def	Target column description in business terms. This is the comment on column in the relational database data dictionary in your target system.
Database Data Type Length	Comes from the relational database data dictionary in Banner ODS. This is stored in the meta data tables, not just the relational database data dictionary, so that it is easily available, in one place with the rest of the meta data, for meta data users.
Business Data Type Length	Used when writing reports for formatting purposes. The business data type may be character, integer, float, etc. It also contains the length of the data.
<p><b>Example:</b></p> <p>The relational database data type and length for an internal ID may be varchar(63), but the business data type and length is eight digits. Even though the database allows for a width up to 63-characters, the column will never be more than eight.</p>	
Domain Values Desc	Description of the valid values that a column can contain. It could be a list of codes and code descriptions.

Views	Descriptions
Publish Ind	A flag that indicates whether to publish the column information to meta data reports so users can use the meta data for reporting purposes. It may not be published because the column contains sensitive information. The column may also contain technical information like a key that would not be used in a report.
Target Sort Order	Columns physical order in the table or view from the relational database data dictionary.
Target Local Ind	Indicates whether the row is a local or baseline version.
Source System Name	Solution system name.
Source System Desc	Solution system description.
Source Subject Area Desc	Advancement, Student or Human Resources, for example.
Source Type	Identifies whether the source is a table, view or function. Sample source types are TABLE, REPORTING VIEW and FUNCTION.
Source Name	Source table, view or function name.
Source Business Name	Source descriptive name.
Source Business Definition	Business purpose of the source.
Source Column Name	Source column name from the source, if the source is a table or view. Function name if the source is a function.
Source Column Business Name	Column in the source descriptive name.
Source Column Business Def	Source column described in business terms.
Business Acronym	Source column acronym, if it has one.
Calculation Formula	Any calculations that are applied to create the data in the target column.

Views	Descriptions
Source Sort Order	Column order in the table or view. It is determined by numbering the columns in alphabetic order.
Source Form	Source system form name from which the data was captured.
Source Local Ind	Indicates whether the row is a local or baseline version.

**Target Meta Data View (WMV\_ TARGET)**

Views	Descriptions
System Name	Administrative source or Banner ODS solution system name.
System Desc	Administrative source or Banner ODS solution system description.
Subject Area Desc	Advancement, Student or Human Resources, for example.
Parent Object Type	This column is used in Banner EDW only. In the case of Banner EDW, the parent object type is STAR.
Parent Object Name	This column is used in Banner EDW only. In the case of Banner EDW this identifies the star to which the target belongs.
Target Type	Stores whether this is a Banner ODS table or view. Currently, Reporting and Composite View information is available.  Sample values for Banner ODS are REPORTING VIEW and COMPOSITE VIEW.  Sample values for Banner EDW are DIMENSION TABLE, FACT TABLE, and STAR.
Target Name	Table or view name.
Target Business Name	Target descriptive name.
Target Business Definition	Target business purpose.
Business Data Steward	Person or department responsible for the data in the target.

Views	Descriptions
Target Column Name	Column name in the target.
Business Name	Descriptive name for the column in the target.
Business Definition	Target column in business terms. This is the comment on column in the relational database data dictionary in your target system.
Database Data Type Length	Comes from the relational database data dictionary in Banner ODS. This is stored in the meta data tables, not just the relational database data dictionary, so that it is easily available in one place with the rest of the meta data, for meta data users.
Business Data Type Length	Used when writing reports for formatting purposes. The business data type may be character, integer, float, etc. It also contains the length of the data.
<b>Example:</b>	
<p>The relational database data type and length for an internal ID may be varchar(63), but the business data type and length is 8-digits. Even though the database allows for a width up to 63 characters, the column can never be more than 8.</p>	
Domain Values Desc	Description of the valid values that a column can contain. It could be a list of codes and code descriptions.
Publish Ind	Indicates whether to publish the column information to meta data reports so users can use the meta data for reporting purposes. It may not be published because the column contains sensitive information. The column may also contain technical information like a key that would not be used in a report.
Sort Order	Columns physical order in the table or view from the relational database data dictionary.
Local Ind	Indicates whether the row is a local or baseline version.

# Staging and data replication

You have the option to use either the Oracle Streams or Materialized Views architecture framework for staging data in Banner ODS. You chose which option to implement when you installed or upgraded Banner ODS. Regardless of which framework you use, you can maintain the staging environment using the options on the Staging menu of the Administrative UI.



## Staging

### Maintain Stage Tables

Add or remove tables from the Banner ODS staging area.

### Report Staging Area Status

View a list of staged tables, the status of processes, and the details of any staging errors or issues.

### Refresh Staging Collections

Refresh a list of staging collections.

Available only in MViews framework

### Refresh Staging Tables

Refresh a list of staging tables.

### Reconcile Stage Tables

Reconcile Stage tables by schema or table name.

You must assign an Administrative UI user the “BPRA Staging” role in WebTailor to allow that user access to the Staging tab. Refer to the section [“Update User Roles”](#) for instructions on how to do this.

### Note

Refer to [“Extract, Transform, and Load process \(ETL\)”](#) section of the [“Architecture”](#) chapter for more information about the Materialized Views architecture and how it is used by the target database. ■

## Staging options

The following options on the Staging menu let you manage the staging environment. These options are available no matter which framework (Streams or MViews) you implement. The jobs allow you to do the following tasks.

- Maintain Stage Tables - add or delete non-baseline staging tables and schemas to or from the Banner ODS staging area. (Refer to the [“Maintain stage tables”](#) section for more information.)
- Report Staging Area Status - view a list of staged tables and perform checks on the status of various staging area items that may require user action, for example, it can list unknown mviews or missing baseline staged mviews. (Refer to the [“Staging](#)

[Area Status](#)” section for more information.)

- Reconcile Stage Tables - compare target database tables (materialized views in the MViews framework) to source tables and restage staging tables in the target database. (Refer to the [“Reconcile and restage stage tables”](#) section for more information.)

## Materialized Views staging options

The following options on the Staging menu let you manage how and when to refresh and reconcile data in the target database *when you implement the Materialized Views framework as your approach to replicate data in the Banner ODS. These options only display when you implement the MViews framework.* The jobs allow you to do the following:

- Refresh Staging Collections - refresh a collection of materialized views
- Refresh Staging Tables - refresh selected materialized views

The actual jobs associated with these menu options are defined in the SUBPROCESS parameter and named as follows:

- RECONCILE\_STAGE\_SCHEMA
- RECONCILE\_STAGE\_TABLE

## Maintain stage tables

If you want to include additional data from the source database that isn't included in your baseline target database, you need to create stage tables for the new data in the target database. Depending on your data, you may also need to create new schemas associated with the new tables.

You have the ability to add or remove non-baseline stage tables and add a schema using the **Maintain Stage Tables** page available from the **Staging** menu in the Administrative UI.

### Add a non-baseline staging table to the Banner ODS

You may want to replicate data from source Banner tables that are not part of the baseline Banner ODS. Perform the following steps to add non-baseline stage tables to the Banner ODS.

1. Click **Staging** from the Administrative UI menu.
2. Click **Maintain Stage Tables**.

3. Choose a **Source Database**. You will only need to select a source database if your institution uses multiple source databases.
4. Click the table owner for the area of tables that you want to add.
5. Select tables from the **Tables to Add** list. Use Shift-click to select a contiguous range of tables or Ctrl-click to select noncontiguous tables.
6. Enter a **Run Date** (format dd-mon-yyyy) and **Runtime** (format hh24:mi:ss) to schedule when to run the job that will add the stage tables to the Banner ODS. Enter *NOW* in each field to run the job immediately.
7. Click **Submit** to schedule the job to run.

The selected tables are added to the Banner ODS stage environment. A local record for each table is also created in the MGBSTGE table if a record doesn't already exist in the table.

## Remove a non-baseline staging table from the Banner ODS

Perform the following steps to remove local stage tables from the Banner ODS. You can only remove stage tables that are not part of baseline Banner ODS. These are the stage tables that your institution added locally.

1. Click **Staging** from the Administrative UI menu.
2. Click **Maintain Stage Tables**.
3. Choose a **Source Database**. You will only need to select a source database if your institution uses multiple source databases.
4. Click **Remove Stage Tables**.
5. Select tables from the **Tables to remove** list. Use Shift-click to select a contiguous range of tables or Ctrl-click to select noncontiguous tables.
6. Enter a **Run Date** (format dd-mon-yyyy) and **Runtime** (format hh24:mi:ss) to schedule when to run the job that will remove the stage tables from the Banner ODS. Enter *NOW* in each field to run the job immediately.
7. Click **Submit** to schedule the job to run.

The selected tables are removed from the Banner ODS stage environment. The local record for each table is also removed from the MGBSTGE table.

## Add a schema

Depending on which data you want to add to the Banner ODS stage tables, you may need to add a user schema within the Administrative UI to obtain access to the additional table data. Use the following steps to add a schema to the Maintain Stage Tables Administrative UI page.

The following considerations apply when adding a schema.

- A schema must exist in both the Banner and Banner ODS databases before you can add it to this menu.
  - The Default Tablespace name for the schema in Banner ODS must match the Default Tablespace name for the schema in Banner.
  - The Default Tablespace name cannot be either SYSTEM or SYSAUX.
1. Click **Staging** from the Administrative UI menu.
  2. Click **Maintain Stage Tables**.
  3. Choose a **Source Database**. You will only need to select a source database if your institution uses multiple source databases.
  4. Click **Add Another Schema to this List**.
  5. Select a schema from the **Schema to Add** list.
  6. Click **Submit**.

The selected schema is added to the list of tables on the Maintain Stage Tables page.

## Remove a schema

Run the following command as IA\_ADMIN on the Banner ODS to remove a schema from the list of available staging schemas.

```
SET SERVEROUTPUT ON
EXEC mgksstg.P_DelOwnerRecs(source alias, schema to remove);
```

## Staging Area Status

You can get information about the state of the Banner ODS staging tables by running the Staging Area Status (STAGE\_AREA\_STATUS) process from the Administrative UI. When you run this job, the information included in the report differs depending on which staging framework your Banner ODS uses. Refer to the following section for the framework you implement.

## Streams framework

When you run the Staging Area Status job against the Oracle Streams framework, you can get the following information.

- The source archive log files.
- The status of various Oracle Streams processes.
- Errors encountered by the apply process while making changes to the stage area.
- List of source tables currently in the staging area.
- The status of various staging area items.

## Materialized Views framework

When you run the Staging Area Status job against the Materialized Views framework, you can get the following information.

- List all the staged tables in Banner ODS (including any non-baseline tables added to the staging area)
- or*
- List staged tables that have at least a specified number of changes in them

## Run Staging Area Status report

Perform the following steps from the Banner ODS Administrative UI to run the Staging Area Status process.

1. Click **Staging** from the Administrative UI menu.
2. Click **Report Staging Area Status**.
3. Select the **Source Database** to identify which source database to run the report against. If your institution includes information from multiple sources in the Banner ODS, there will be one entry for each database in the Source Database drop-down list.
4. Check the items that you want to include in the report.
  - Check **Display Process Status** to include in the report status the relevant information about various Oracle Streams components. (Available only in Oracle Streams framework.)
  - Check **Display Apply Errors** to include in the report any errors the apply process encountered while making changes to the staging area. (Available only in Oracle Streams framework.)

- Check **Display Staged Tables** to include in the report a list of all source tables that are currently in the stage area.

If you select the value *Yes, with Change Counts* for the **Display Staged Tables** field, you need to enter a **Change Count Limit** value as well. This value defines the minimum number of changes required on a stage table for it to get listed in the Staged Tables list.

- Check **Perform Staging Checks** to include in the report the status of various staging area items that may require user action.

 **Note**

You don't need to perform the staging checks every time you run the Staging Area Status job. However, you will want to periodically run the report with the option to perform staging checks turned on so that you can ensure that the staging environment isn't encountering any of the issues flagged by the checks. ■

- Check the **Check Staging Triggers** option to include in the report any baseline table triggers that are missing from the staged tables replicated from the source database.
5. Enter a **Run Date** and **Run Time** to schedule when to run the job. Enter NOW in each field to run the job immediately.
  6. Click **Submit**.
  7. Click **View Control Reports**.
  8. Select the **STAGE\_AREA\_STATUS** process associated with your User ID to view the status report.

Report Status Information	Report Includes
Required Source Archived Logs (Oracle Streams framework only)	Any source archived log files that are required by the Oracle Streams capture processes. The report includes the directory path where archived log files are saved. Refer to the <a href="#">“Required Source Archived Logs”</a> later in this chapter for more information.
Process Status (Oracle Streams framework only)	Status and relevant information about the following Oracle Streams components: <ul style="list-style-type: none"> <li>• Capture process - Status, State, First SCN, Last captured SCN, and Last applied SCN</li> <li>• Capture queue - Enqueue, Dequeue, Number of messages, and Spill messages</li> <li>• Propagation schedule - Status</li> <li>• Apply queue - Enqueue, Dequeue, Number of messages, and Spill messages</li> <li>• Apply process name - Status, Reader state, Coordinator state, Server state, and Apply tag</li> </ul>
Apply Errors (Oracle Streams framework only)	Any errors the apply process encountered while making changes to the stage area are entered into the error queue.
Staged Tables List	All source tables that are currently in the stage area.

Report Status Information	Report Includes
Staging Checks (Oracle Streams framework)	<p>When the <b>Perform Staging Checks</b> option is selected in the Oracle Streams framework, the report includes information on the following items.</p> <ul style="list-style-type: none"> <li>• Source table columns that are missing from the stage tables.</li> <li>• Banner ODS stage table triggers that are set to fire once.</li> <li>• Captured source tables that are not instantiated in the destination.</li> <li>• Instantiated destination tables that are not captured at the source.</li> </ul>
Staging Checks (Materialized Views framework)	<p>When the <b>Perform Staging Checks</b> option is selected in the Materialized Views framework, the report includes information on the following items.</p> <ul style="list-style-type: none"> <li>• Unknown (Non-Baseline) staging materialized views - lists any materialized views in one of the staging schemas that have not been created using the baseline process and are not recorded in the MGBSTGE table.</li> <li>• Missing baseline staging materialized views - lists any missing materialized views that should be in the baseline target database.</li> <li>• Staging materialized views that are not in a Refresh Group or Staging Collection - lists any Staging materialized views that are not in a Refresh Group or Refresh Staging Collection and are not getting refreshed by one of the Banner ODS processes. (This will only be an issue if you made a mistake while adding a new Refresh Group or changing a delivered Refresh Group.)</li> <li>• Refresh Groups not associated with an ODS Refresh process. (This will only be an issue if you reorganized Refresh Groups and didn't create the necessary Staging Collection record ETL PACKAGE parameter.)</li> <li>• Missing baseline warehouse staging indexes - lists any missing indexes that are specific to the baseline target database and are used to improve performance of the ETL mappings.</li> </ul>

## Reconcile and restage stage tables

It is possible that after upgrading the source database, the structure of the tables in the staging area of the target database may no longer match the source tables. This could happen when new columns or indexes are added to the source database during an upgrade

and are then missing in the target database. When this happens, the affected stage tables must be restaged to make sure the table structure matches the source system. If structural changes are found in the restaged tables, a Recently Restaged Tables job gets created that you can use to reload the affected composite tables.

You have the option to reconcile stage tables by choosing an entire schema of tables to reconcile at once or by selecting specific tables to reconcile. The reconcile job allows you to tailor the actions performed depending on your need. You have the following options when you run the job.

- Compare the structure of stage tables with the associated source table.
- Compare and restage only those stage tables with a structure that does not match the source system.
- Restage the stage tables regardless of whether the structure matches the source table or not.

### **Compare**

When running a compare as part of the reconcile process, the following items are compared between the stage tables and the associated source tables:

- Table columns
- Table column data types
- Indexes
- Indexed columns
- Primary keys
- Primary key columns

If you want to verify the row counts or that the data within the staged tables match the source, refer to the steps in the section [“Run stage table reconcile”](#).

### **Restage**

When running a restage as part of the reconcile process, the following actions are performed.

- A new job is created in the Administrative UI that will load just the composite tables that are based on restaged tables. A link to run the job named “Load for Recently Restaged Tables” is created under Options>Schedule a Process>Schedule Banner ODS Mappings. Running this job will bring the data in the Banner ODS up-to-date with the restaged data.
- Run a refresh for all mview refresh groups. This option is available only in the Materialized Views framework. It is an important step that will guarantee staged data is current with the restaged materialized views. This ensures a consistent data set that you can use for ad hoc reporting or as a starting point for the ETL refresh.

## Run stage table reconcile

Use the following steps to run the Reconcile Stage Tables job.

1. Click **Staging** from the Administrative UI menu.
2. Click **Reconcile Stage Tables**.
3. Click either **Reconcile by Schema** or **Reconcile by Table Name**.
4. Select the **Source Database** to identify which source database to run the report against. If your institution includes information from multiple sources in the Banner ODS, there will be one entry for each database in the Source Database dropdown list.
5. Choose **Which Schema(s)** or **Which Table(s)** to reconcile. This field will vary depending on whether you originally chose to reconcile by schema or table name.

Use Shift-click to select a contiguous range of schemas or tables or Ctrl-click to select noncontiguous schemas or tables.

6. Select the **Action** to perform.
  - Compare structure of all selected schemas (or tables)  
**Note:** Running a compare by itself will not restage any tables.
  - Restage only selected schemas (or tables) that differ
  - Restage all selected schemas (or tables)
7. Choose whether to **Refresh All MViews**. (Available only in Materialized Views framework.)

If any table is restaged and this checkbox is selected, then all mview refresh groups will be refreshed as part of this job.

8. Enter a **Run Date** and **Run Time** to schedule when to run the job. Enter NOW in each field to run the job immediately.
9. Click **Submit**.
10. Click **View Control Reports** at bottom of the page to check the control report, which lists the tables reconciled and the mappings that are affected by the changes to the stage tables. If structural changes are found in the restaged tables, a Recently Restaged Tables job gets created that you should also run to reload the affected composite tables.

11. Run the Load for Recently Restaged Tables job as follows:
  - 11.1. Select **Options**.
  - 11.2. Select **Schedule a Process**.
  - 11.3. Select **Schedule Banner ODS Mappings**.
  - 11.4. Select **Load for Recently Restaged Tables**.
  - 11.5. Enter a **Run Date** and **Run Time** to schedule when to run the job that will remove the stage tables from the Banner ODS. Enter NOW in each field to run the job immediately.
  - 11.6. Click **Submit** to schedule the job to run.

## Maintain Oracle Streams framework

There are a number of tasks you may want to perform to maintain the Oracle Streams framework after it is implemented at your institution. For example, you can remove the Streams framework, remove baseline stage tables not used at your institution, or stop and start various Streams components. You perform these tasks outside the Administrative User Interface

Refer to the following topics for more information about maintaining the Streams architecture.

- [“Create Streams Framework”](#)
- [“Remove Streams Framework”](#)
- [“Configure Streams Replication for Baseline Tables”](#)
- [“Remove a Baseline Staging Table from the Banner ODS”](#)
- [“Start or Stop the Streams Capture Process”](#)
- [“Start or Stop the Streams Propagation Schedule”](#)
- [“Start or Stop the Streams Apply Process”](#)

### Note

The “source alias” specified in the following sections is the parameter that identifies the source database if you load data from multiple sources into the Banner ODS. Refer to the *Source Alias* section of the *Architecture* chapter for more information. ■

## Create Streams Framework

The Streams framework includes the queues, propagation schedule, and capture and apply processes. The Streams framework is created during the install or upgrade process. If you need to recreate the Streams framework, perform the following steps:

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
EXEC MGKSTRC.P_CREATE_LOCAL_ENV (database link, source
alias);
```

where you enter your institution's values for the parameters in parentheses.

## Remove Streams Framework

The Streams framework includes the queues, propagation schedule, and capture and apply processes. Perform the following steps to remove the Streams framework.

### Warning

Before you remove the Streams framework, you need to remove all staging tables from the Streams replication environment. ■

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
EXEC MGKSTRC.P_DROP_LOCAL_ENV (source alias);
```

where you enter your institution's values for the parameter in parentheses.

## Configure Streams Replication for Baseline Tables

The MGBSTGE table stores the schemas and tables from Banner that will be included in the baseline Banner ODS Streams staging environment. Replication of these tables is performed during the install or upgrade process. If you need to configure the replication of Streams baseline Banner ODS tables at some point after installation, perform the following steps:

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
EXEC MGKSTRM.P_ADD_TO_LOCAL_ENV(source alias, schema, table
name) ;
```

where you enter your institution's values for the parameters in parentheses.

 **Note**

You can add all of the tables in the MGBSTGE table at once by using the '%' value for the 'table name' parameter. ■

## Remove a Baseline Staging Table from the Banner ODS

The MGBSTGE table stores the schemas and tables from Banner that will be included in the baseline Banner ODS Streams environment. If you need to remove a baseline Banner ODS staging table from the Streams replication process, perform the following steps:

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
EXEC MGKSTRM.P_REMOVE_FROM_LOCAL_ENV(source alias, schema,
table name) ;
```

where you enter your institution's values for the parameters in parentheses.

 **Note**

You can remove all of the tables in the MGBSTGE table at once by using the '%' value for the 'schema' and 'table name' parameters. ■

## Start or Stop the Streams Capture Process

The Streams capture process identifies relevant changes in the Banner database redo log, converts them into logical change records, and puts them in a queue to be applied in the Banner ODS.

*Perform the following steps to **start** the Streams capture process.*

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
```

```
EXEC MGKSTRM.P_START_CAPTURE(source alias);
```

where you enter your institution's values for the parameter in parentheses.

*Perform the following steps to **stop** the Streams capture process.*

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON  
EXEC MGKSTRM.P_STOP_CAPTURE(source alias);
```

where you enter your institution's values for the parameter in parentheses.

## **Start or Stop the Streams Propagation Schedule**

The Streams propagation schedule moves the change messages identified by the capture process from the source Banner database queue to a queue on the Banner ODS.

*Perform the following steps to **start** the Streams propagation schedule.*

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON  
EXEC MGKSTRM.P_START_PROPAGATION(source alias);
```

where you enter your institution's values for the parameter in parentheses.

*Perform the following steps to **stop** the Streams propagation schedule.*

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON  
EXEC MGKSTRM.P_STOP_PROPAGATION(source alias);
```

where you enter your institution's values for the parameter in parentheses.

## **Start or Stop the Streams Apply Process**

The Streams apply process in the Banner ODS removes change messages from the queue and applies them directly to the destination tables in the Banner ODS.

Perform the following steps to **start** the Streams apply process.

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
EXEC MGKSTRM.P_START_APPLY(source alias);
```

where you enter your institution's values for the parameter in parentheses.

Perform the following steps to **stop** the Streams apply process.

1. Log in to the Banner ODS.
2. Issue the following commands:

```
SET SERVEROUTPUT ON
EXEC MGKSTRM.P_STOP_APPLY(source alias);
```

where you enter your institution's values for the parameter in parentheses.

## Required Source Archived Logs

Archived log files on the source Banner system may be required by the Oracle Streams capture processes. These log files track any changes to the source Banner system. The capture process uses information in the archived log files to synchronize the Banner ODS staging tables with the source Banner tables.

Be sure to keep the archived log files in the archive directory on the source Banner database server. If the capture process is unable to locate a required archived log file, then the capture will abort. When this happens, restore the archived log file to its expected location so the capture process can continue replicating changes to the Banner ODS staging area.

The Staging Area Status process control report includes the directory path where archived log files are saved. The following query displays the archive redo log files required by the Streams capture process:

```
SELECT R.CONSUMER_NAME "Capture Process Name",
       R.SOURCE_DATABASE "Source Database",
       R.SEQUENCE# "Sequence Number",
       R.NAME "Required Archived Log Name"
FROM DBA_REGISTERED_ARCHIVED_LOG R, DBA_CAPTURE C
WHERE R.CONSUMER_NAME = C.CAPTURE_NAME AND
       R.NEXT_SCN >= C.REQUIRED_CHECKPOINT_SCN;
```

## Monitor Streams for Apply Errors

The Streams environment includes an error queue. Any errors encountered when applying logical change record (LCR) messages to the Banner ODS database are placed in this queue. Use the following steps to monitor the errors in the queue.

1. Query the `DBA_APPLY_ERROR` data dictionary view to see any errors in the error queue.

Alternately, you can run the Staging Area Status process from the Banner ODS Administrative UI, select to display apply errors, and then view the process control report.

- 1.1. Click **Staging** from the Administrative UI menu.
  - 1.2. Click **Report Staging Area Status**.
  - 1.3. Check the **Display Apply Errors?** process parameter and any other parameters that you want to display in the status report.
  - 1.4. Enter *NOW* in the a **Run Date** and **Runtime** fields.
  - 1.5. Click **Submit**.
  - 1.6. Click **View Control Reports**.
2. Issue the following commands to see detailed information about error messages in the destination:

```
SQL> SET SERVEROUTPUT ON
SQL> EXEC MGKSTRE.P_PRINT_ERRORS;
```

3. Fix the errors.
4. Reapply or delete the messages in the error queue.
  - Issue the following command to reapply all error messages for the “apply1” process in the error queue:

```
BEGIN
  DBMS_APPLY_ADM.EXECUTE_ALL_ERRORS(
    apply_name      => 'apply1',
    execute_as_user => false);
END;
/
```

- Issue the following command to delete all error messages for the “apply1” process in the error queue:

```
BEGIN
    DBMS_APPLY_ADM.DELETE_ALL_ERRORS(
        apply_name => 'apply1');
END;
/
```

## Monitor Source Capture Queue for Growth

If captured messages don't get propagated to the destination database, the number of messages will grow in the capture queue. You need to periodically monitor the capture queue to determine how full it is.

Use the following query to display the number of messages in memory and spilled to disk for each queue.

```
SELECT QUEUE_SCHEMA "Schema",
       QUEUE_NAME "Queue Name",
       (NUM_MSGS - SPILL_MSGS) "Messages in Memory",
       SPILL_MSGS "Messages Spilled",
       NUM_MSGS "Total Messages"
FROM V$BUFFERED_QUEUES;
```

## Monitor the Status of Source Propagation Jobs

If the Streams propagation schedule becomes disabled, the capture queue on the source Banner database may fill up. Perform the following steps to monitor the propagation jobs.

1. Issue the following query to display the status of each propagation job.

```
SELECT DISTINCT P.PROPAGATION_NAME "Propagation",
               P.SOURCE_QUEUE_NAME "Source Queue",
               P.DESTINATION_QUEUE_NAME "Destination Queue",
               S.LATENCY "Latency",
               DECODE(S.SCHEDULE_DISABLED,
                     'Y', 'DISABLED',
                     'N', 'ENABLED') "Propagation Status",
               S.PROCESS_NAME "Process Name",
               S.FAILURES "Failures"
FROM DBA_QUEUE_SCHEDULES S, DBA_PROPAGATION P
WHERE P.DESTINATION_DBLINK = S.DESTINATION
```

```
AND S.SCHEMA = P.SOURCE_QUEUE_OWNER
AND S.QNAME = P.SOURCE_QUEUE_NAME;
```

2. Determine whether any propagation jobs are disabled.
3. Stop and restart any disabled propagation jobs using the steps provided in the [“Start or Stop the Streams Propagation Schedule”](#) section.

## Set the Checkpoint Frequency and Retention Time (optional)

The Oracle Streams capture process takes a checkpoint periodically and stores it in the Banner SYSAUX tablespace. Checkpoints record information on scanned system change numbers (SCN) and allow for a quicker restart of the capture process. By default, a checkpoint is taken every 10MB of scanned redo.

The required checkpoint SCN is associated with the lowest SCN at which the capture process requires redo data. The checkpoint retention time defines how long to keep checkpoints prior to the required checkpoint SCN. The default retention time is 60 days. The first SCN for a capture process defines the lowest SCN at which the capture process can begin capturing changes. Once a checkpoint is purged, the first SCN is updated to match the next stored checkpoint. You may need to modify these values based on the amount of redo generated and size of the SYSAUX tablespace in your Banner database.

Run the following in the Banner database to modify the checkpoint frequency and retention time:

```
BEGIN
  DBMS_CAPTURE_ADMIN.SET_PARAMETER(
    capture_name => capture process name,
    parameter    => '_checkpoint_frequency',
    value        => size of scanned redo in MB);
END;
/

BEGIN
  DBMS_CAPTURE_ADMIN.ALTER_CAPTURE(
    capture_name           => capture process name,
    checkpoint_retention_time => number of days);
END;
/
```

Refer to Oracle Streams Concepts and Administration Guide for more information about capture process checkpoints.

## Avoid NOLOGGING and UNRECOVERABLE Keywords in Source

Avoid using the keywords `NOLOGGING` and `UNRECOVERABLE` in the Banner source code. If you use these keywords in the source code, related source changes are not written to the redo logs. Because the Streams capture process mines the redo logs for change information, any changes using these keywords will be lost. Streams will not be able to replicate these changes in the Banner ODS stage tables. As a result, the source Banner and destination Banner ODS tables will not be synchronized leading to future Streams errors.

## Use Streams Tags with Batch Processes

All changes made to the Banner database are written to the redo log in the Banner system. By default, the Banner ODS Streams environment is set up to capture only changes in the redo log that do not include a tag. You can set a Streams tag on specified log entries, which basically includes an additional column with each entry in the redo log. A Streams capture process can then examine the extra column for each tagged entry in the redo log.

You can set a Streams tag with a non-NULL value before running a large batch file in a Banner session. Any inserts, updates, or deletes made during that tagged session will not be caught by the capture process and the changes will not be replicated to the Banner ODS.

Be aware that after the batch process has completed some destination tables in the Banner ODS may no longer be synchronized with the source Banner tables. When this happens, the affected destination tables need to be dropped from Streams and re-added after running the batch process.

Issue the following command to set the Streams tag to a value of “00” for a specific Banner session:

```
BEGIN
    DBMS_STREAMS.SET_TAG(TAG => HEXTORAW('00'));
END;
/
```

## Use Data Dictionary Views to Display Streams Information

Refer to the “Monitoring a Streams Environment” chapter of Oracle's *Streams Concepts and Administration Guide* for a complete list of the static data dictionary views and dynamic performance views related to Streams.

## Maintain Materialized Views framework

Keeping the target database materialized views synchronized with the source database tables is the key to maintaining the Materialized Views framework. You will need to refresh data in the materialized views on a regular basis (timing to be determined by your institution) so that, in turn, the staging tables are up to date with data before running the ETL processes to load or reload the warehouse. In addition, upgrades to source products may modify table definitions in the source database. When this happens, it will be necessary to restage the affected materialized views in the target database so they match the source tables.

To ensure that the materialized views are refreshed before data is loaded into the warehouse, materialized views refresh processing has been added to run at the beginning of the “Refresh” ETL jobs that refresh the warehouse data. For example, the REFRESH\_STUDENT job in the Banner ODS, which refreshes all of the MST composite tables, now includes a job at the beginning to refresh the associated SATURN (student-related) materialized views prior to refreshing the composite tables. This addition to the Refresh jobs ensures that all materialized views related to the data being updated in the warehouse will get refreshed before the ETL loads new data into the warehouse.

There are staging refresh jobs that refresh only the stage tables and materialized views. You can use these jobs to synchronize data on a more frequent basis. This will reduce the amount of work that the integrated materialized views refresh has to do as part of the nightly Banner ODS refresh process.

### Refresh groups

To help you manage the process of refreshing materialized views, Oracle includes a component called a Refresh Group, which lets you group together materialized views. Once materialized views are grouped in a Refresh Group, you gain the ability to refresh all of the materialized views within a Refresh Group at the same point in time.

Banner ODS uses Refresh Groups to make it easier to manage related groups of materialized views. When delivered, baseline Banner ODS includes defined Refresh Groups that group together sets of materialized views by schema and table type, for example, grouping validation versus non-validation tables. For each schema staged in Banner ODS, two refresh groups are associated with that schema:

```
ODS_REFGROUP_<schema>ODS_REFGROUP_<schema>_VAL
```

For example, there is a Student Refresh Group that includes all Student related materialized views, and a Student Validation Refresh Group that includes just the Validation tables (STV\*) from the Student product

By default, any table staged using the APIs, `mgkmview.P_stage_mview()`, will add that materialized view to one of the two refresh groups based on the Banner table naming standard. If a table has a "V" character in the 3rd position of the table name, that table is included in the validation refresh group.

This mechanism is used to group all source tables together by Subject Area when delivered. You can reorganize these groupings using command-line API calls as delivered in baseline scripts. Also, the Oracle DBMS\_REFRESH package provides APIs to create or delete Refresh Groups, and to associate materialized views with them that can be used to reorganize refresh processing.

## Staging Refresh Collections

A materialized view can only exist in a single Refresh Group at a time. However, you can include a materialized view in more than one functional area at a time. The STAGING\_REFRESH\_COLLECTION parameter stored in the MTVPARM table lets you create a “group of refresh groups”, which is called a “collection” in BPRA. You define a collection by creating multiple MTVPARM records that link multiple Refresh Groups to one ODS Refresh Job.

STAGING\_REFRESH\_COLLECTION entries are delivered for all baseline schemas. The delivered records associate all the Refresh Groups with both the subject area specific Refresh jobs (for example, STUDENT, GENERAL, ALUMNI) as well as the REFRESH\_ALL job and the REFRESH\_VALIDATION jobs, so that by default the materialized views will get refreshed as part of the nightly Banner ODS Refresh jobs.

## Associate Staging Collections with Refresh Jobs

You associate a staging collection with the actual Banner ODS Refresh jobs using the ETL\_PACKAGE parameter as a “PRE” step, which means it runs before any mappings or slotted packages. The ETL\_PACKAGE parameter records link a collection of Refresh Groups to the Banner ODS Refresh job.

## Refresh materialized views

When delivered, the system is set up to refresh the materialized views whenever you run any of the ETL Refresh jobs. It is recommended that you run the ETL Refresh jobs at least once a day typically during the nightly build. You will probably want to refresh the materialized views more often than once a day.

You have two options for refreshing the materialized views outside of the ETL Refresh jobs:

- Refresh a collection of materialized views (staging collection)
- Refresh selected materialized views (staging tables)

These two refresh options are jobs that you run from the Staging menu in the Administrative UI. They are the option that are only present on the Staging menu if you implement the Materialized Views framework.

## Staging

### Maintain Stage Tables

Add or remove tables from the Banner ODS staging area.

### Report Staging Area Status

View a list of staged tables, the status of processes, and the details of any staging errors or issues.

### Refresh Staging Collections

Refresh a list of staging collections.

Available only in MViews framework

### Refresh Staging Tables

Refresh a list of staging tables.

### Reconcile Stage Tables

Reconcile Stage tables by schema or table name.

## Refresh a Staging Collection

A “collection” of materialized views is a group of Refresh Groups that have been associated together in the STAGING REFRESH COLLECTION parameter.

The “Refresh Staging Collections” job allows you to select one or more Collections to refresh. Use the following steps to refresh a collection of tables.

1. Click **Staging** from the Administrative UI menu.
2. Click **Refresh Staging Collections**.
3. Choose the **Staging Groups to Refresh**. Use Shift-click to select a contiguous range of tables or Ctrl-click to select noncontiguous tables.
4. Choose the **Logging Mode** which defines the level of detail to include in the control report.
  - Display summary stats by Refresh Group - displays summary of information for each Refresh Group selected for staging (default selection)
  - Display detail about each MView Refresh - displays information about each materialized view in all Refresh groups selected for staging
5. Enter a **Run Date** and **Run Time** to schedule when to run the job that will remove the stage tables from the Banner ODS. Enter NOW in each field to run the job immediately.
6. Click **Submit** to schedule the job to run.

## Refresh selected materialized views

The “Refresh Staging Tables” job allows you to select one or more materialized views to refresh. Use the following steps to refresh a collection of tables.

1. Click **Staging** from the Administrative UI menu.
2. Click **Refresh Staging Tables**.
3. Choose the **Staging Tables to Refresh**. Use Shift-click to select a contiguous range of tables or Ctrl-click to select noncontiguous tables.

The list of materialized views includes the Refresh Group name in parentheses to help organize the listing and allow easier selection of a group of materialized views related by Refresh Group.

4. Enter a **Run Date** and **Run Time** to schedule when to run the job that will remove the stage tables from the Banner ODS. Enter NOW in each field to run the job immediately.
5. Click **Submit** to schedule the job to run.

## Refresh Staging Collection job control reports

When you submit a job to Refresh a Staging Collection or Refresh Staging Tables, each job generates a standard control report listing details about the Refresh Groups or materialized views processed by the job. In addition, because the standard ETL Refresh jobs also refresh Staging Collections at the beginning of each job, similar output is displayed in the ETL Refresh job control report.

The following figure illustrates the control report for the Refresh Staging Collections job.

Run Date	Job Number	Process	Status
Apr 03, 2012 02:57 pm	902	REFRESH_STAGING_COLLECTIONS	Complete

[Filter Report](#) | [Delete Report](#) | [CSV Summary](#)  
[<- Previous Report](#) | [Select a Report](#) | [Next Report->](#)

Procedure	Message
	Start of Process at 03-APR-2012 14:57:44
	Session set to Resumable, timeout = 3 days (259200)
	Logging Mode: Display detail about each MView Refresh with 1 or more changes. <span style="color: red; font-weight: bold;">1</span>
	Refreshing Staging Refresh Group (1/2) ODS_REFGROUP_GENERAL at: 03-APR-2012 14:57:45 <span style="color: green; font-weight: bold;">2</span>
ODS_REFGROUP_GENERAL	MView Refreshed: GEBFUNC -- Changes: 394 -- Seconds: 1 -- Rate: 394 (changes/sec) <span style="color: blue; font-weight: bold;">3</span>
ODS_REFGROUP_GENERAL	MView Refreshed: GERATTD -- Changes: 4102 -- Seconds: 2 -- Rate: 2051 (changes/sec)
ODS_REFGROUP_GENERAL	MView Refreshed: GOBINTL -- Changes: 141688 -- Seconds: 96 -- Rate: 1476 (changes/sec)
ODS_REFGROUP_GENERAL	Stage Group Refreshed -- Tables: 22 -- Changes: 146184 -- Seconds: 99 <span style="color: purple; font-weight: bold;">4</span>
	Refreshing Staging Refresh Group (2/2) ODS_REFGROUP_SATURN at: 03-APR-2012 14:59:35
ODS_REFGROUP_SATURN	MView Refreshed: SPRTELE -- Changes: 29998 -- Seconds: 68 -- Rate: 441 (changes/sec)
ODS_REFGROUP_SATURN	Stage Group Refreshed -- Tables: 123 -- Changes: 29998 -- Seconds: 69
	Process completed successfully at 03-APR-2012 15:01:01
	End of Report

The control report includes the following information identified by the callout numbers in the sample control report above.

- 1 The report identifies the Logging Mode that was defined when the report ran.
- 2 Each Refresh Group associated with a selected Refresh Collection is refreshed with the start time noted.
- 3 Each MView Refreshed in a Refresh Group is identified including the number of Changes, how many Seconds it took to refresh that Materialized view, and the Rate of the refresh for that.
- 4 The report identifies the overall status of the Stage Group Refreshed including the number of Tables, Changes, and Seconds to complete. For example, in the sample report above the General Refresh Group (ODS\_REFGROUP\_GENERAL) refreshed 22 tables with 141688 changes in 99 seconds.

If an exception occurs when refreshing a single materialized view within a refresh group, the entire refresh for that group will not be processed. This is illustrated in the following sample error message for the Financial Aid group ODS\_REFGROUP\_FAISMGR, which could display in a control report.

```
ORA-12048: error encountered while refreshing materialized view
"FAISMGR"."RPATRM" ORA-04098: trigger
'FAISMGR.RT_RPRATRM_INSERT_ODS_CHANGE' is invalid and failed
re-validation
```

When this kind of error occurs, you can do one of the following to address it:

- Fix the underlying problem (in the above case, an invalid trigger on one of the materialized views)
- Use the Refresh Staging Tables option to manually select and refresh all of the materialized views except the one that caused the error. When you use the Refresh Staging Tables job, the materialized views are all listed, grouped by Refresh Group, so that you can easily identify which materialized views belong in which Refresh Group.

## Web Tailor Administration

---

The Enterprise Administrative application uses the Web Tailor application to build its look and feel. Web Tailor delivers customizable global Web rule definitions and procedures, customizable menus, menu items, graphics and text definitions.

From the Administrative Tool, use the Web Tailor Administration menu item to access the Web Tailor options. The tasks under this menu item allow you to customize various aspects of the Administrative Tool. Other sections include references to the various Web Tailor options that you may want to customize. To learn more about Web Tailor, refer to the *Web Tailor User Guide*.

## Functions

Web Tailor lets you build the look, feel, and unique personality of all your institution's web applications, so you can personalize your institution's interface to the world. Web Tailor delivers customizable global web rule definitions and procedures, customizable menus, menu items, graphics and text definitions.

The following Web Tailor functions are available from the Web Tailor Administration Menu.

- [“Customize a Web Menu or Procedure”](#)
- [“Customize a Graphic Element”](#)
- [“Customize a Set of Information Text”](#)
- [“Customize a Set of Menu Items”](#)
- [“Update User Roles”](#)

- [“Customize a Web Module”](#)
- [“Customize Web Rules”](#)
- [“Customize Web Tailor Parameters”](#)
- [“Customize a Login Return Location”](#)
- [“Customize Web Tailor Overrides”](#)
- [“Customize Global User Interface Settings”](#)

## Customize a Web Menu or Procedure

The **Customize a Web Menu or Procedure** option allows you to define the menus that will appear on your institution’s web pages for the different self-service applications, and to specify the procedures behind them.

Refer to the Web Tailor Online Help for more information about creating or customizing a web menu or procedure.

## Customize a Graphic Element

The **Customize a Graphic Element** option allows you to specify the images that will be available for use on your web pages. For each image, you can specify its name, the directory where it is located, its height, its width, and various other aspects.

Refer to the Web Tailor Online Help for more information about creating or customizing a graphic element.

## Customize a Set of Information Text

The **Customize a Set of Information Text** option allows you to add or customize Information Text (Info Text). Info Text can be:

- Instructions on how to use a page
- Help for the page
- Error messages

Refer to the Web Tailor Online Help for more information about customizing Information Text.

## Customize a Set of Menu Items

The **Customize a Set of Menu Items** option allows you to define the items that will appear on the menus on your institution’s web pages.

Refer to the Web Tailor Online Help for more information about customizing a set of menu items.

## Update User Roles

The **Update User Roles** option allows you to change the roles to which a person has been assigned. The User Roles define a high level of security and allow you to give users access to selected components of the Administrative User Interface. A users assigned roles determine which areas of the Administrative User Interface the user can access and make changes within the Banner ODS.

Refer to the [“Update User Roles”](#) section for more information about updating user roles.

## Customize a Web Module

This function allows you to modify a specific application or module that uses WebTailor, such as Accounts Receivable, Student Self-Service, or Banner Performance Reporting and Analytics.

Refer to the Web Tailor Online Help for more information about customizing a web module.

## Customize Web Rules

This function allows you to define certain rules for your institution’s web pages. For example, you can identify the number of minutes a person can be inactive before they are timed out, or specify the format for the date and time information that appears on your pages.

Refer to the Web Tailor Online Help for more information about customizing a web rules.

## Customize Web Tailor Parameters

This function allows you to customize parameters used in Web Tailor processing, such as the maximum length of PINs. You must exercise great care when modifying these parameters.

Refer to the Web Tailor Online Help for more information about customizing a WebTailor parameters.

## Customize a Login Return Location

Use this function to specify the page you would like to be displayed when a user is timed out, then logs back in.

Refer to the Web Tailor Online Help for more information about customizing a login return location.

## Customize Web Tailor Overrides

This page allows you to replace certain procedures and functions with your own under certain circumstances. This is necessary because you may have a stand-alone product you would like to use with the self-service products, and you need to use some of the procedures and functions in the other system. If an override is defined, that code will be run instead of the Web Tailor code.

Refer to the Web Tailor Online Help for more information about customizing WebTailor overrides.

## Customize Global User Interface Settings

This function allows you to set up rules that will apply to your institution's web pages as a whole. You can specify:

- Header information
- The location URL of CSS that control the pages' look-and-feel
- The location URL of CSS that control the look-and-feel of your Help text
- The location URL of where your Help text files are stored

### Note

It is recommended that you use Info Text as your Help text. ■

- Images that represent errors and warnings
- An image that indicates that a field is required

Refer to the Web Tailor Online Help for more information about customizing global user interface settings.



# 3 Banner ODS Business Concepts

---

Business concepts are used to organize the data available for different reporting requirements. A business concept shows the relationships between the data supporting a set of business processes. Because different business processes often require different perspectives on data, the relationships among the supporting database objects need to change based on the analysis being performed.

The Banner ODS is designed to take advantage of Cognos Framework Manager's ability to use database objects in multiple models. Each model is referred to as a namespace. In a Framework Manager namespace, database objects are defined as Cognos metadata query subjects. In that namespace the relationships between the different query subjects focus around a central or primary fact table query subject. All other query subjects are related to each other through the central or primary fact table. All data analysis and reporting completed using the business concept uses the central fact table to filter and determine what data to retrieve.

# Banner ODS Business Concepts

The following table lists the business concept and primary reporting view or database table for each business concept within the Banner ODS. These concepts are listed by subject areas. A subject area loosely corresponds to a SunGard Higher Education Banner product. When you write a report, use filters on the primary reporting view rather than the other reporting views whenever possible.

Subject Area	Business Concept	Primary Fact Table
Accounts Receivable	<a href="#">“Receivable Customer”</a>	RECEIVABLE_ACCOUNT
	<a href="#">“Receivable Revenue”</a>	RECEIVABLE_ACCOUNT_DETAIL
Advancement	<a href="#">“Advancement Prospect”</a>	PROSPECT_INFO
	<a href="#">“Advancement Rating”</a>	ADVANCEMENT_RATING
	<a href="#">“Annual Giving”</a>	ANNUAL_GIVING
	<a href="#">“Annual Giving Comparison”</a>	ANNUAL_GIVING
	<a href="#">“Campaign Giving History”</a>	CAMPAIGN_GIVING_HISTORY
	<a href="#">“Campaign Management”</a>	CAMPAIGN
	<a href="#">“Constituent”</a>	CONSTITUENT
	<a href="#">“Constituent Entity”</a>	CONSTITUENT_ENTITY
	<a href="#">“Designation”</a>	DESIGNATION
	<a href="#">“Designation Giving History”</a>	DESIGNATION_GIVING_HISTORY
	<a href="#">“Gift”</a>	GIFT_TRANSACTION
<a href="#">“Organizational Constituent”</a>	ORGANIZATIONAL_CONSTITUENT	
<a href="#">“Pledge”</a>	PLEDGE_TRANSACTION	

Subject Area	Business Concept	Primary Fact Table
	<a href="#">“Solicitation Effort”</a>	SOLICITOR_ORGANIZATION
<b>Common</b>	<a href="#">“Event”</a>	EVENT
	<a href="#">“Institution”</a>	INSTITUTION
	<a href="#">“Organization Entity”</a>	ORGANIZATION_ENTITY
	<a href="#">“Person Demographic”</a>	PERSON_DETAIL
	<a href="#">“Person Role”</a>	PERSON_DETAIL
	<a href="#">“Person Supplemental”</a>	PERSON_DETAIL
	<a href="#">“Relationship”</a>	RELATIONSHIP
<b>Finance</b>	<a href="#">“Account Index Audit”</a>	ACCOUNT_INDEX
	<a href="#">“Budget Availability Comparison”</a>	BUDGET_AVAILABILITY_LEDGER
	<a href="#">“Budget Availability Ledger”</a>	BUDGET_AVAILABILITY_LEDGER
	<a href="#">“Budget Detail”</a>	BUDGET_DETAIL
	<a href="#">“Cashier Session Analysis”</a>	RECEIVABLE_ACCOUNT_DETAIL
	<a href="#">“Encumbrance”</a>	ENCUMBRANCE_ACCOUNTING
	<a href="#">“Endowment Distribution”</a>	ENDOWMENT_DISTRIBUTION
	<a href="#">“Endowment Units”</a>	ENDOWMENT_UNIT
	<a href="#">“Fixed Asset”</a>	FIXED_ASSET_ITEM
	<a href="#">“General Ledger”</a>	GENERAL_LEDGER
	<a href="#">“Grant and Project”</a>	GRANT_VIEW
	<a href="#">“Grant Ledger”</a>	GRANT_LEDGER

Subject Area	Business Concept	Primary Fact Table
	<a href="#"><u>“Grant-Contract and Proposal”</u></a>	PROPOSAL
	<a href="#"><u>“Invoice Payable”</u></a>	INVOICE_ITEM
	<a href="#"><u>“Operating Ledger”</u></a>	OPERATING_LEDGER
	<a href="#"><u>“Operating Ledger Comparison”</u></a>	OPERATING_LEDGER
	<a href="#"><u>“Purchasing Payable”</u></a>	PURCHASE_ORDER_ITEM
	<a href="#"><u>“Transaction History”</u></a>	TRANSACTION_HISTORY
<b>Financial Aid</b>	<a href="#"><u>“Financial Aid Application”</u></a>	FINAID_APPLICANT_STATUS
	<a href="#"><u>“Financial Aid Award and Disbursement”</u></a>	AWARD_BY_PERSON
	<a href="#"><u>“Financial Aid Fund”</u></a>	AWARD_BY_FUND
	<a href="#"><u>“Loan Disbursement”</u></a>	STUDENT
<b>Human Resources</b>	<a href="#"><u>“Employee”</u></a>	EMPLOYEE
	<a href="#"><u>“Employee and Position”</u></a>	EMPLOYEE
	<a href="#"><u>“Human Resource Application”</u></a>	HR_APPLICATION
	<a href="#"><u>“Human Resource Faculty”</u></a>	FACULTY
	<a href="#"><u>“Payroll”</u></a>	PAYROLL_DOCUMENT
	<a href="#"><u>“Position”</u></a>	POSITION_DEFINITION
<b>Student</b>	<a href="#"><u>“Active Registration”</u></a>	ENROLLMENT
	<a href="#"><u>“Admissions Application”</u></a>	ADMISSIONS_APPLICATION
	<a href="#"><u>“Advisor Student List”</u></a>	STUDENT
	<a href="#"><u>“Course Catalog”</u></a>	COURSE_CATALOG

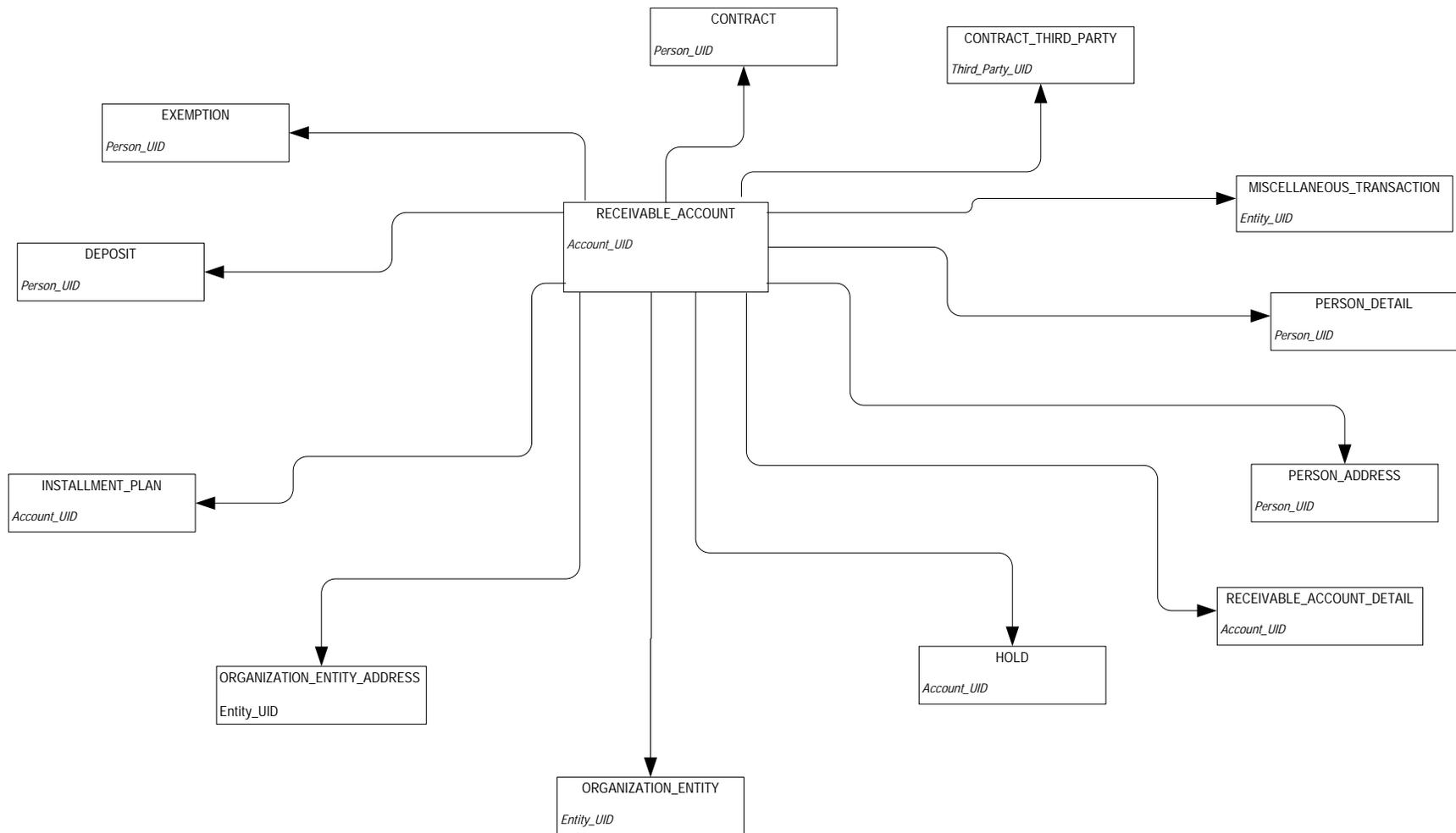
Subject Area	Business Concept	Primary Fact Table
	<a href="#">“Enrollment Management”</a>	ENROLLMENT
	<a href="#">“Enrollment Management Subset”</a>	ENROLLMENT
	<a href="#">“Faculty Assignment”</a>	FACULTY
	<a href="#">“Faculty Subset”</a>	FACULTY
	<a href="#">“Government Reporting”</a>	GOVERNMENT_STUDENT, GOVERNMENT_FINANCIAL_AID, GOVERNMENT_ADMISSIONS
	<a href="#">“Recruitment Information”</a>	RECRUITMENT_INFORMATION
	<a href="#">“Residential Life”</a>	PERSON_DETAIL
	<a href="#">“Schedule Offering”</a>	SCHEDULE_OFFERING
	<a href="#">“Student Detail”</a>	STUDENT
	<a href="#">“Student Detail Subset”</a>	STUDENT
<b>Travel and Expense</b>	<a href="#">“Authorization”</a>	AUTHORIZATION
	<a href="#">“Reimbursement”</a>	REIMBURSEMENT

## Diagrams

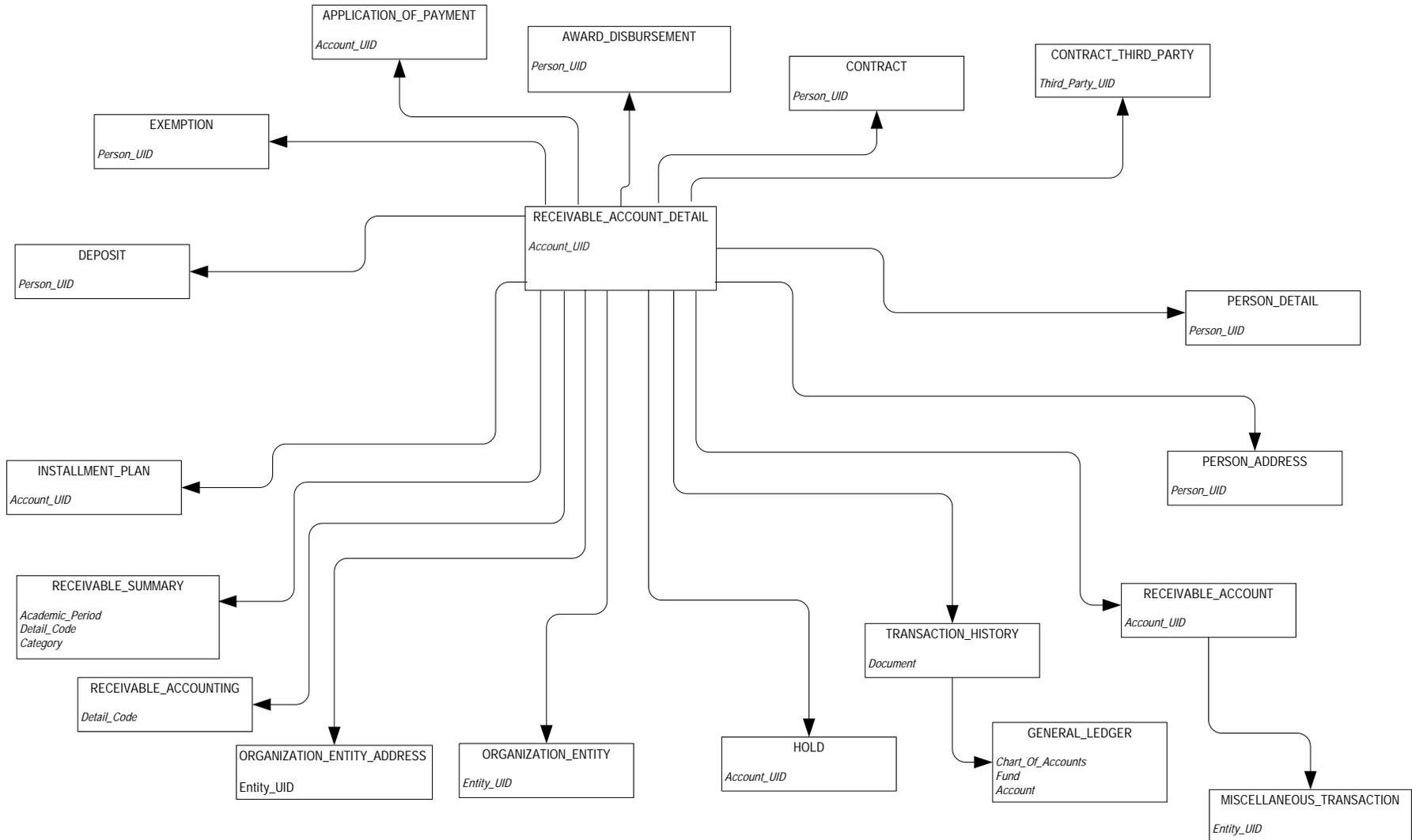
The relationships in the reporting tool meta data for Cognos 8 Business Intelligence and Oracle Business Intelligence Discoverer are the same. Following are the diagrams that show the relationships for the business concepts defined in the Banner ODS. There is one diagram for each business concept. The diagrams are grouped by subject areas such as Accounts Receivable and Advancement.

# Accounts Receivable

## Receivable Customer

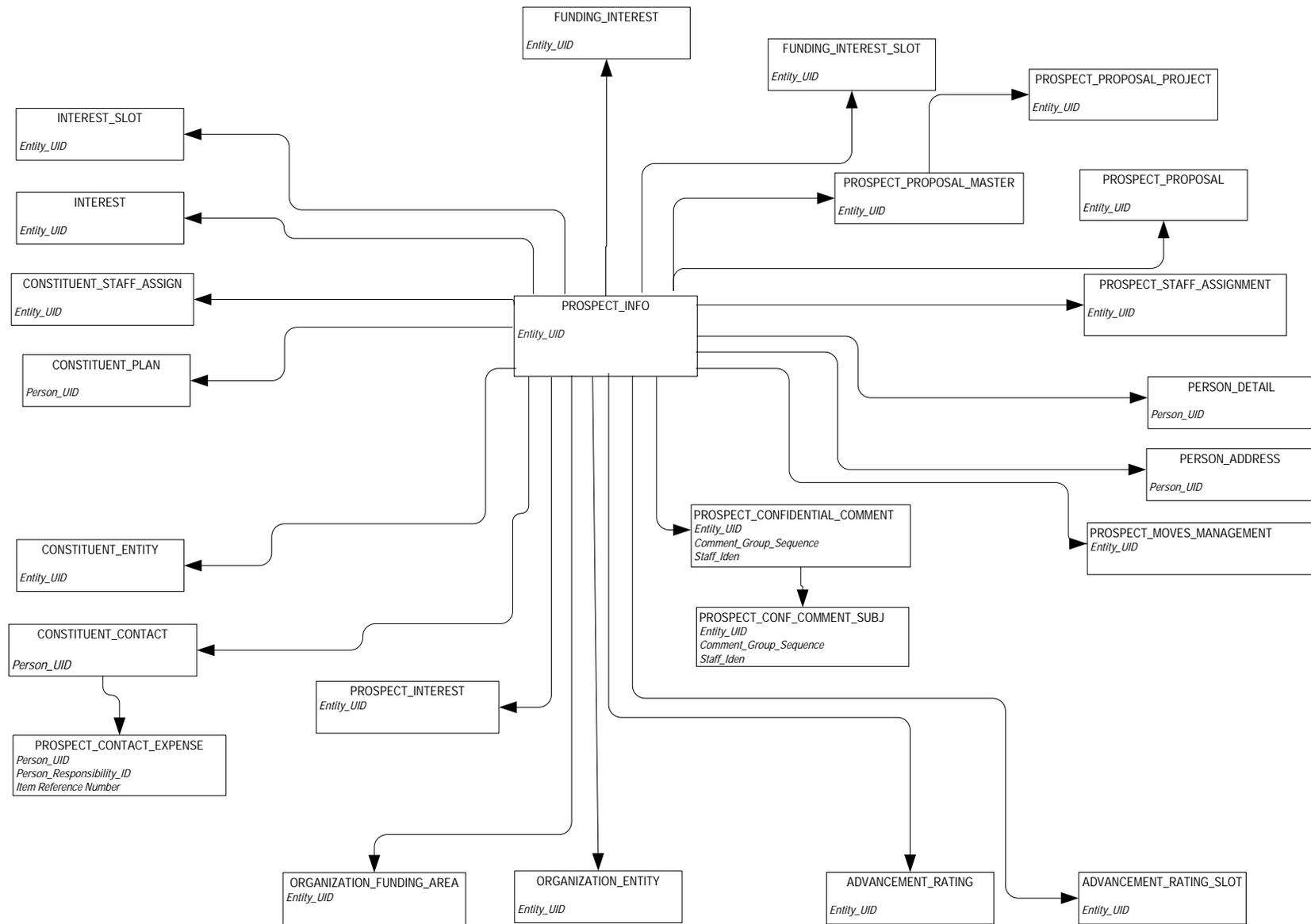


# Receivable Revenue

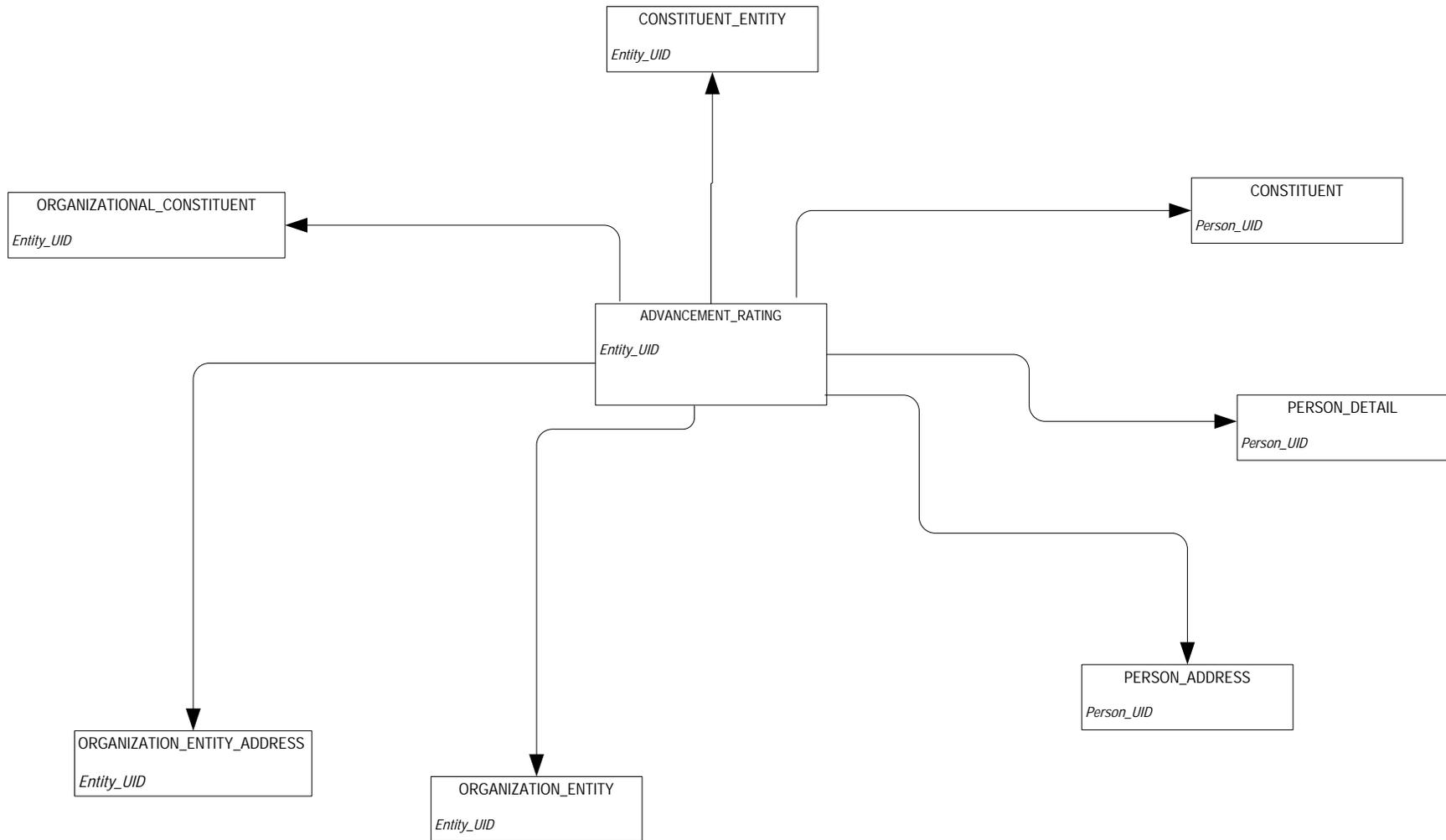


# Advancement

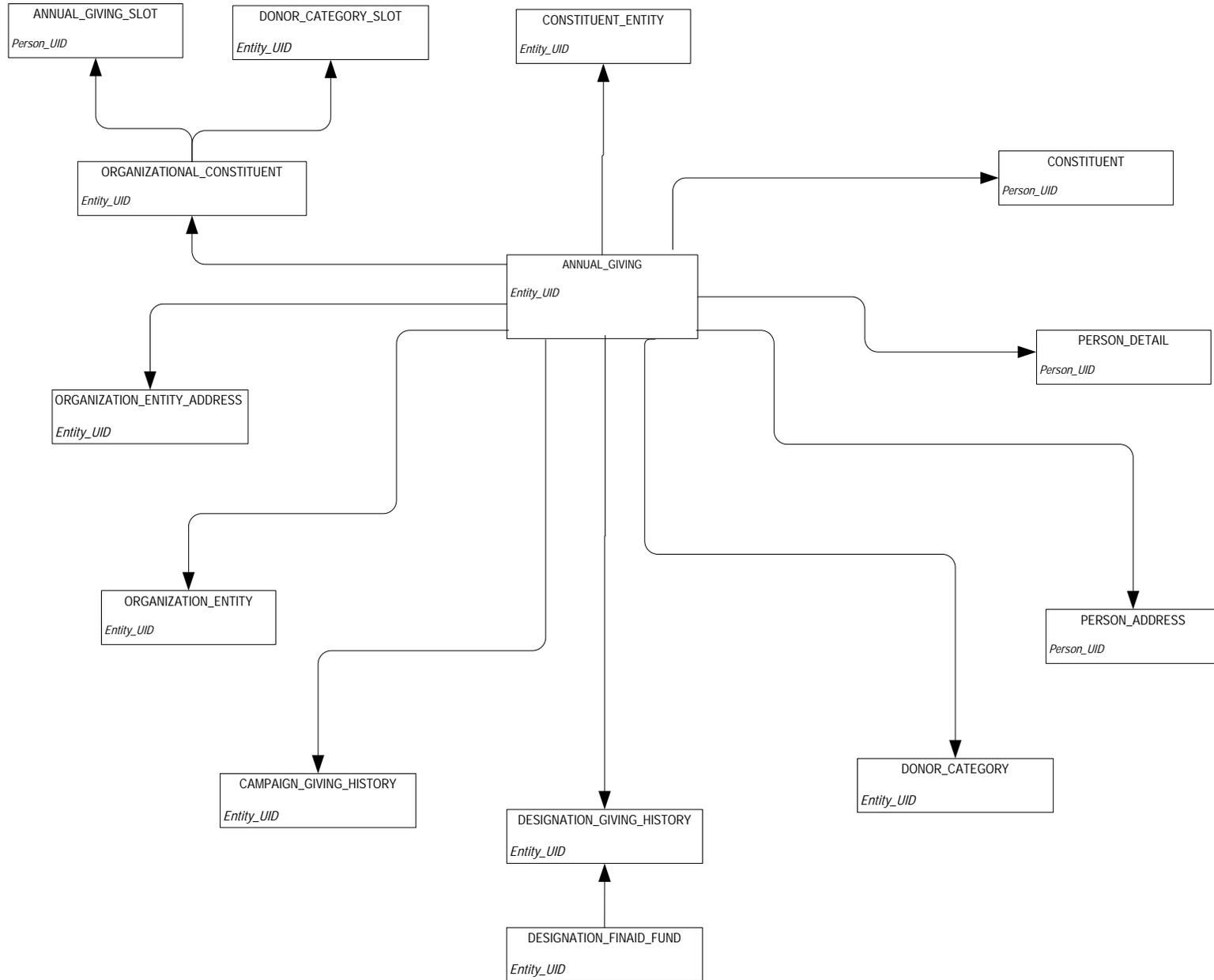
## Advancement Prospect



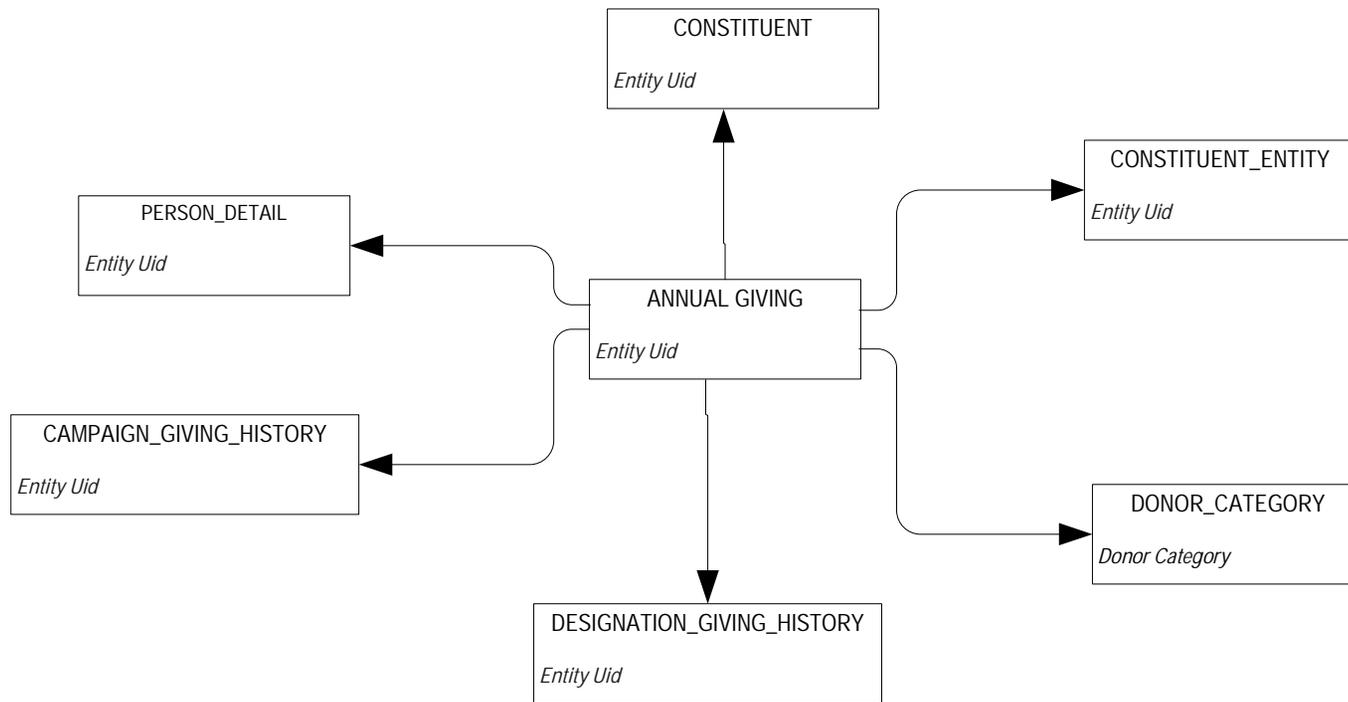
## Advancement Rating



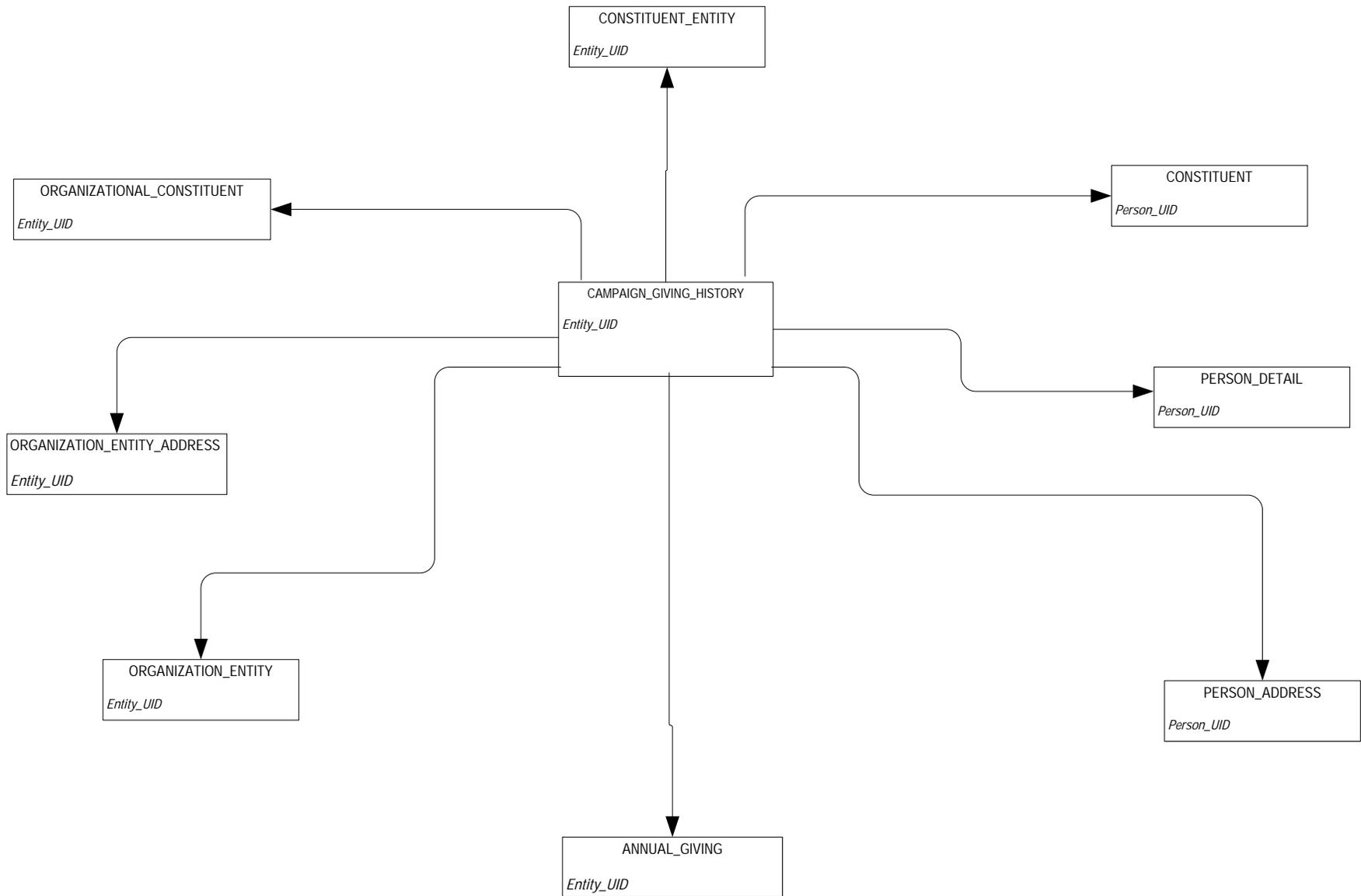
# Annual Giving



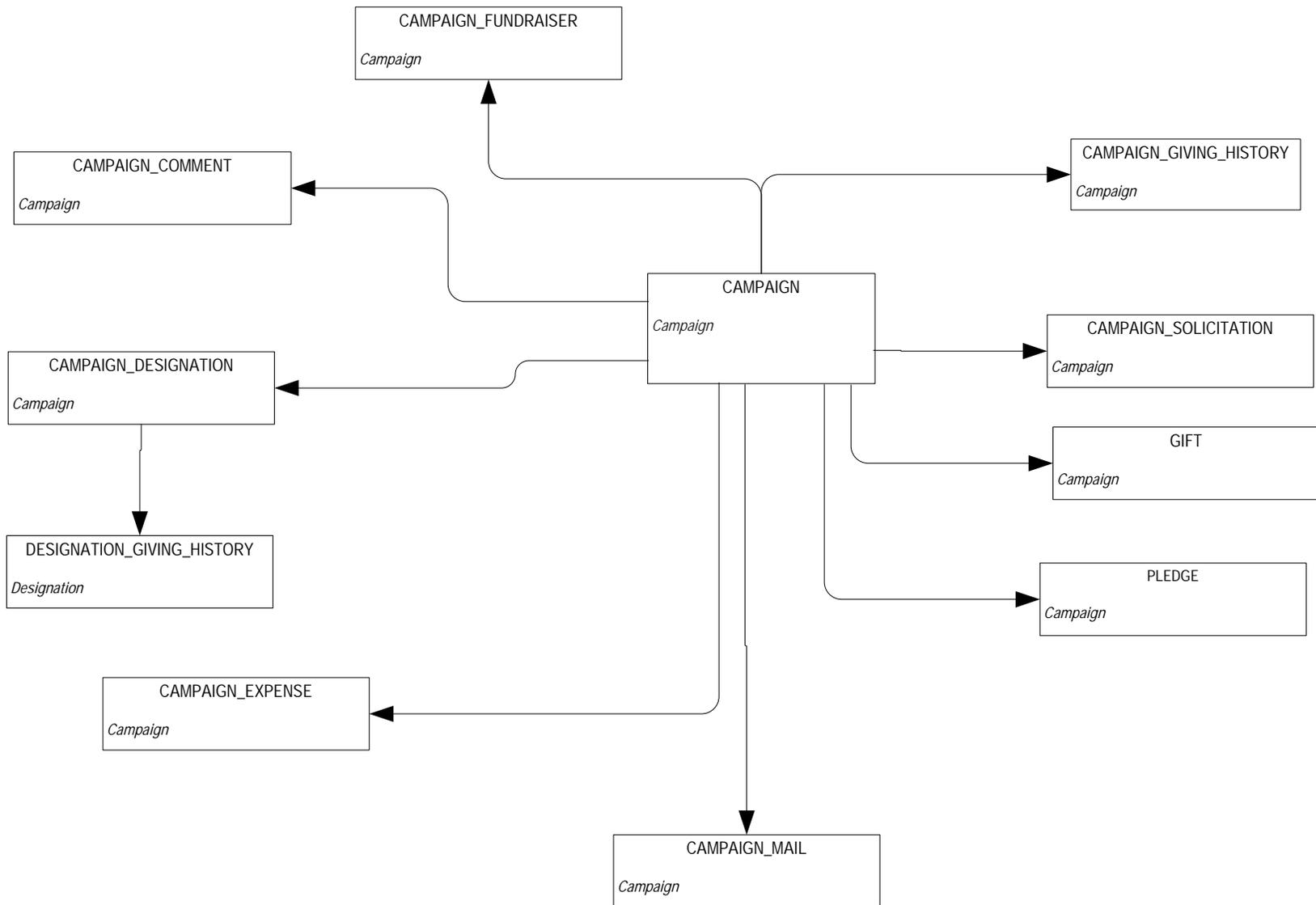
## Annual Giving Comparison



# Campaign Giving History

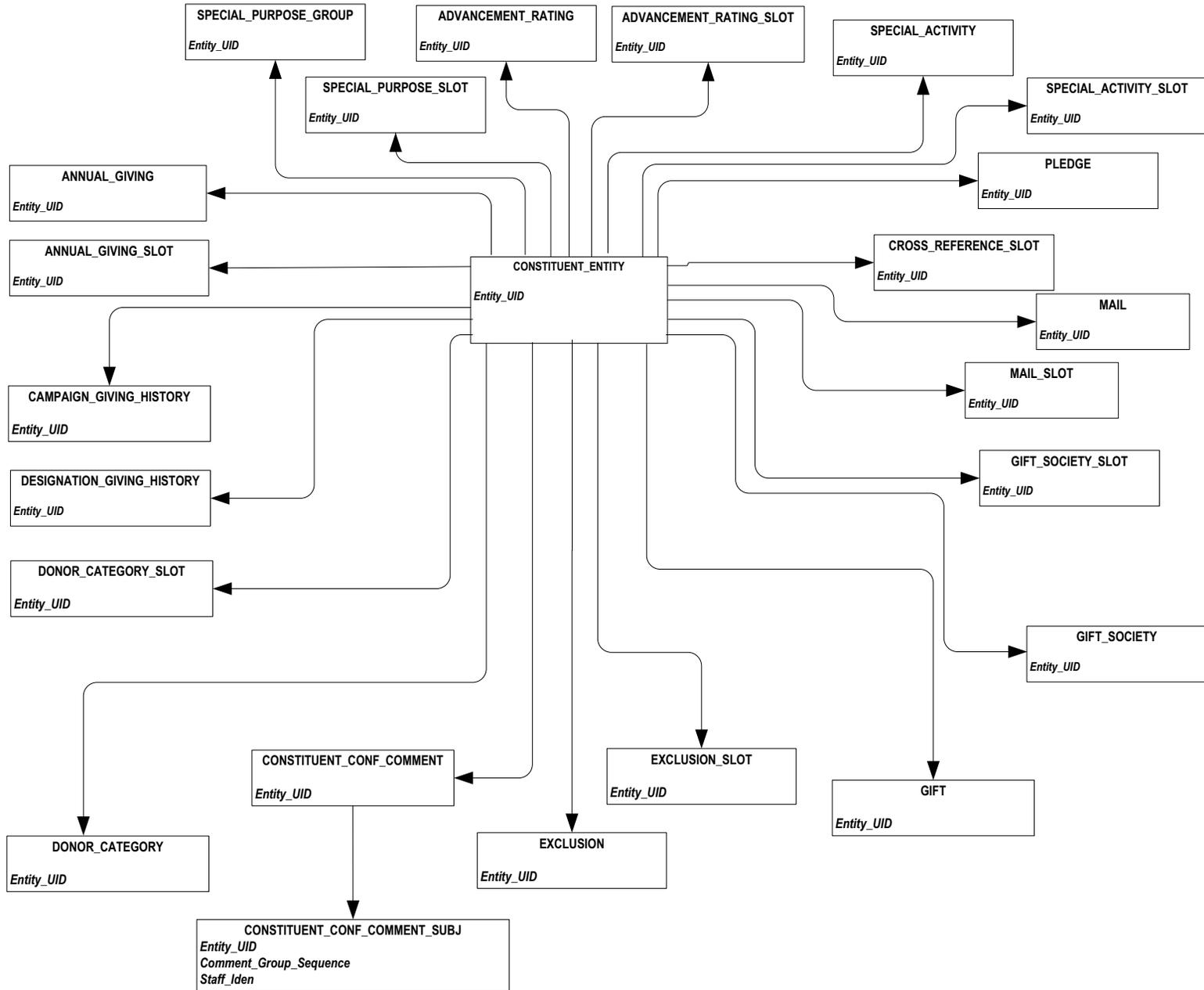


# Campaign Management

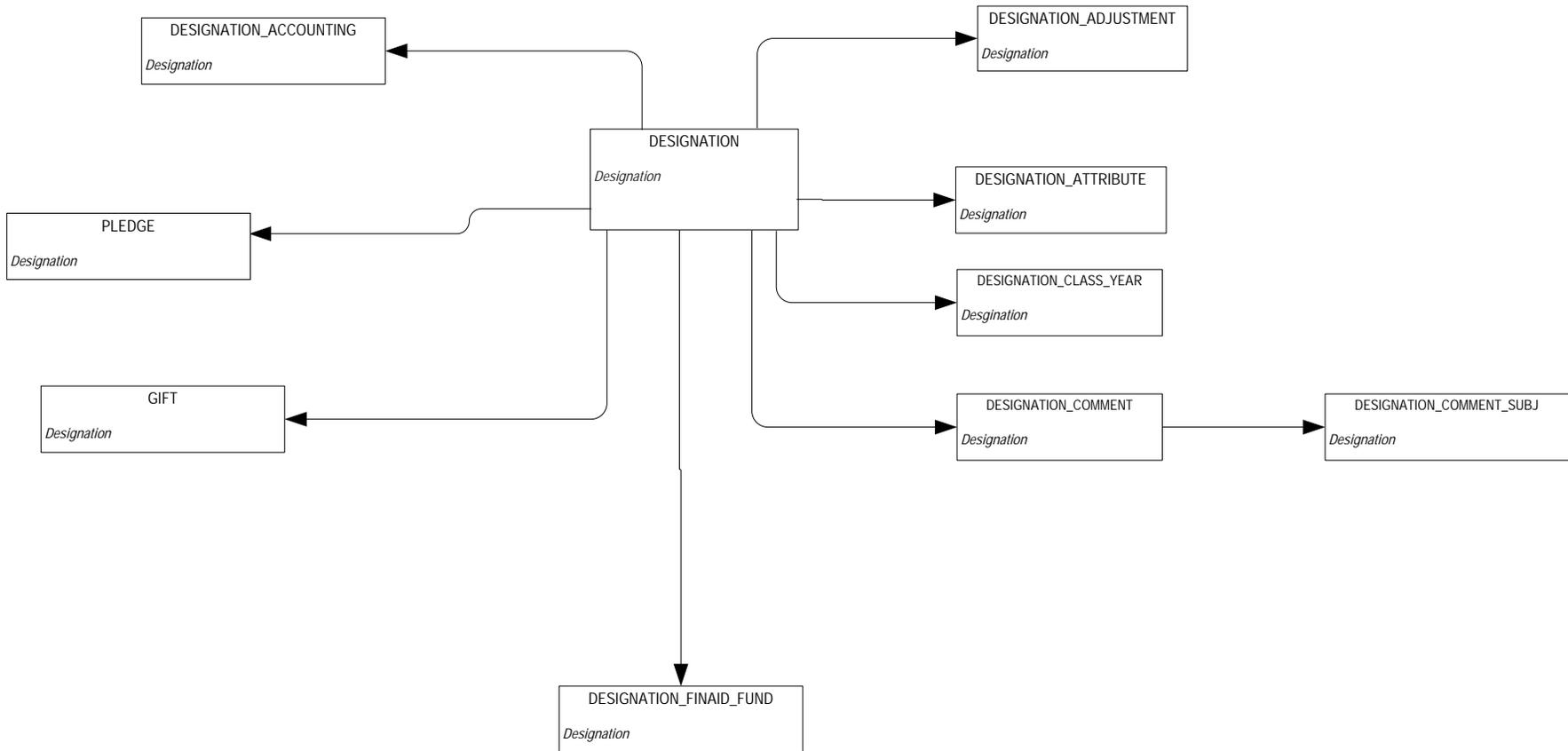




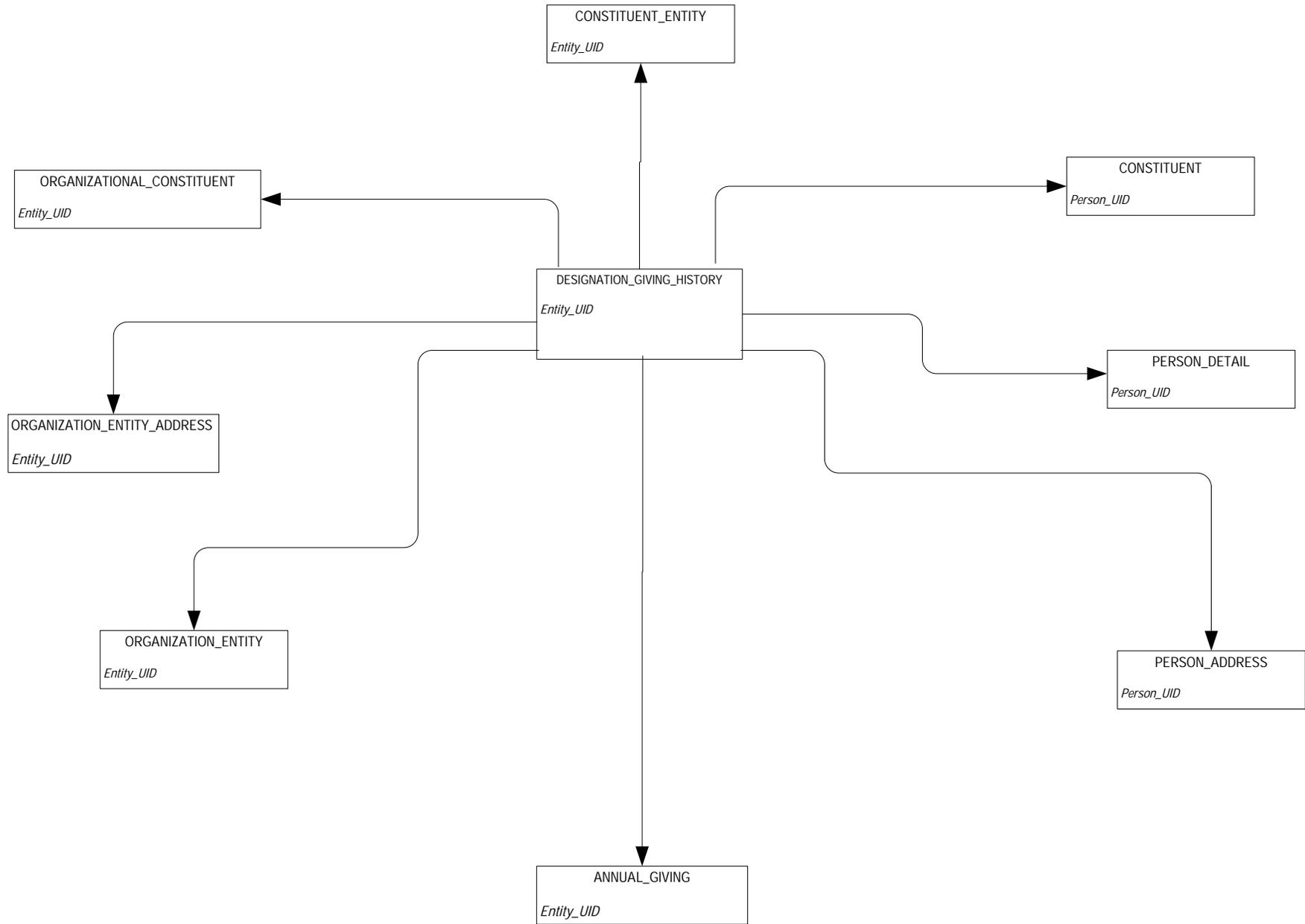
# Constituent Entity



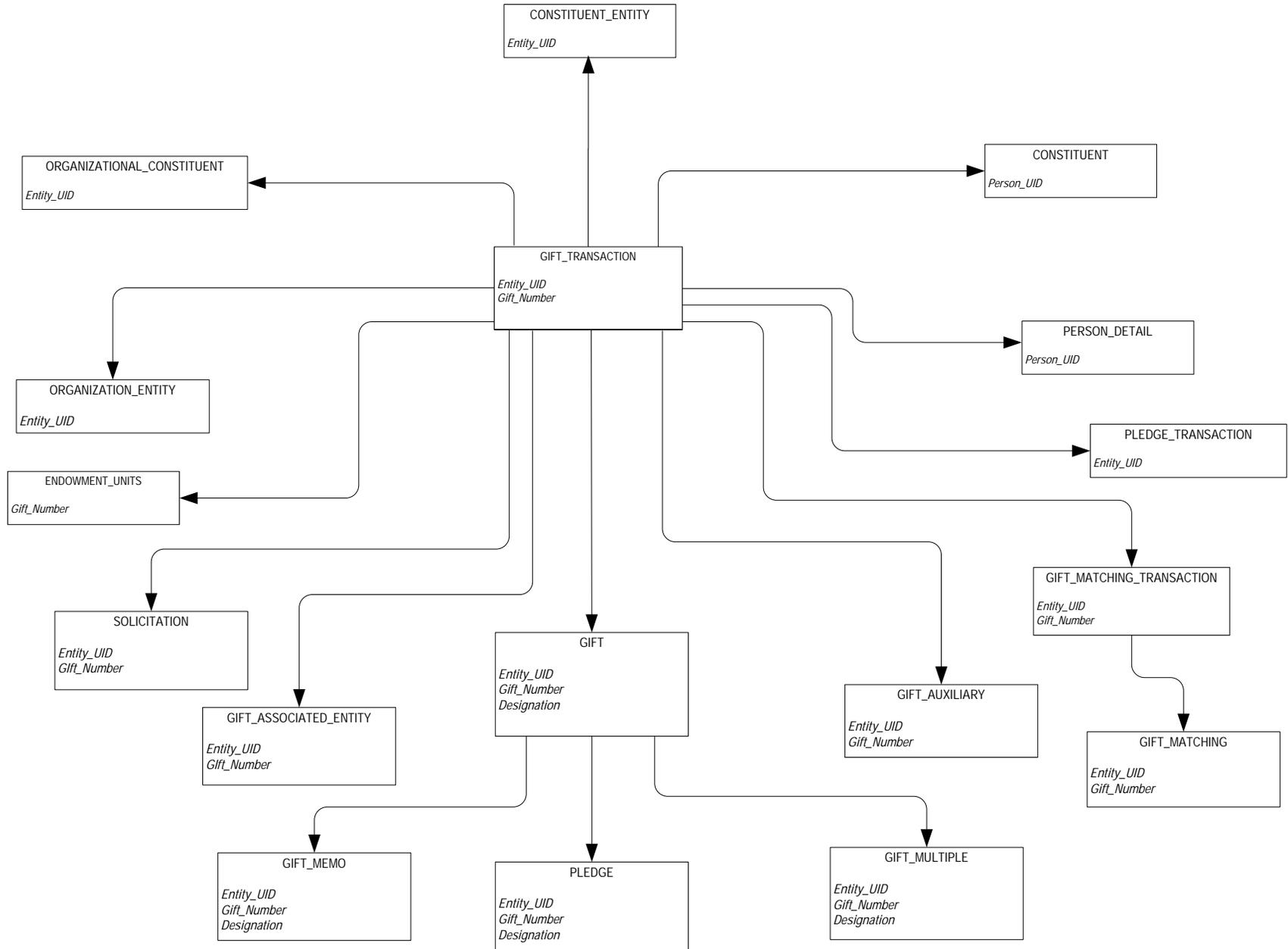
# Designation



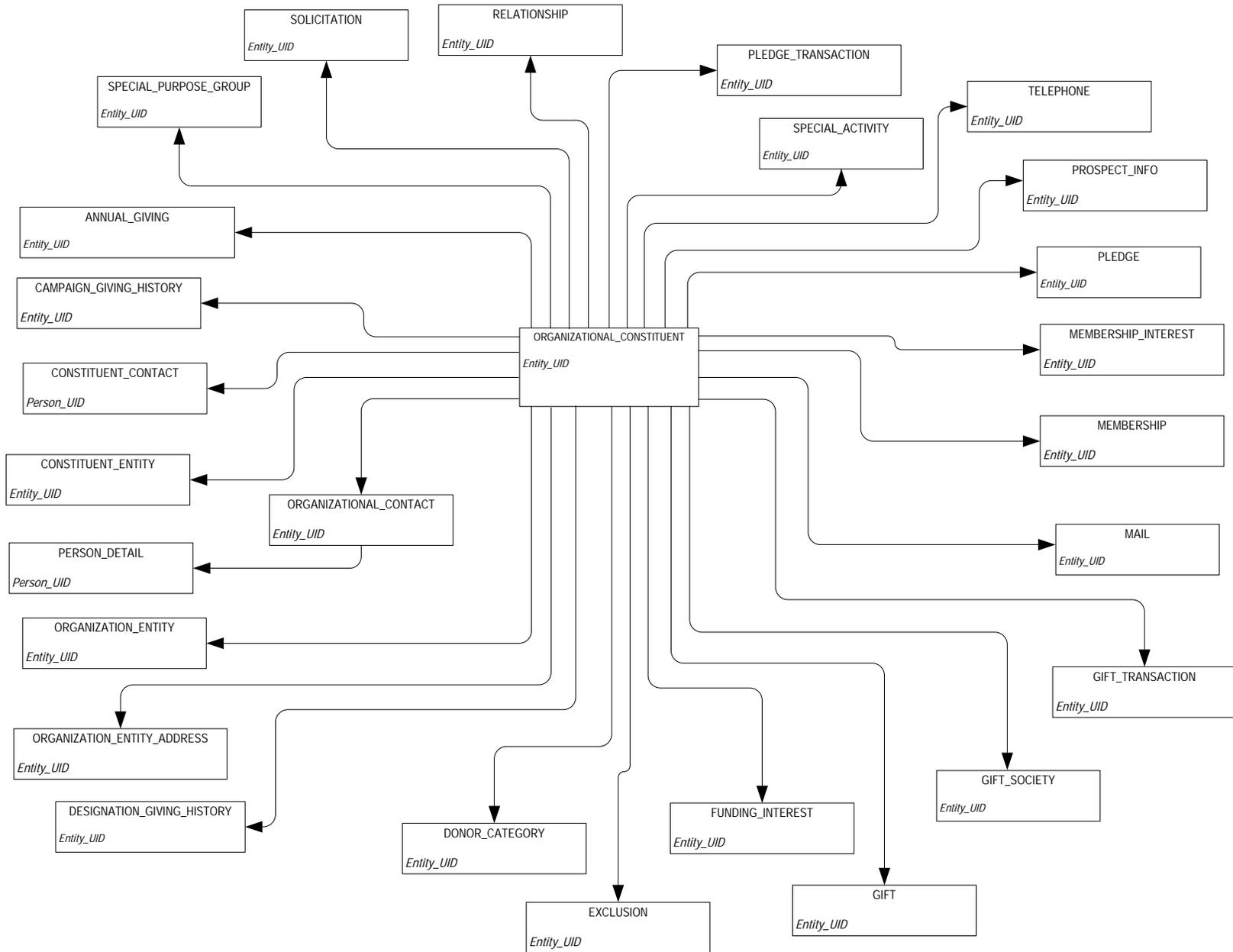
## Designation Giving History



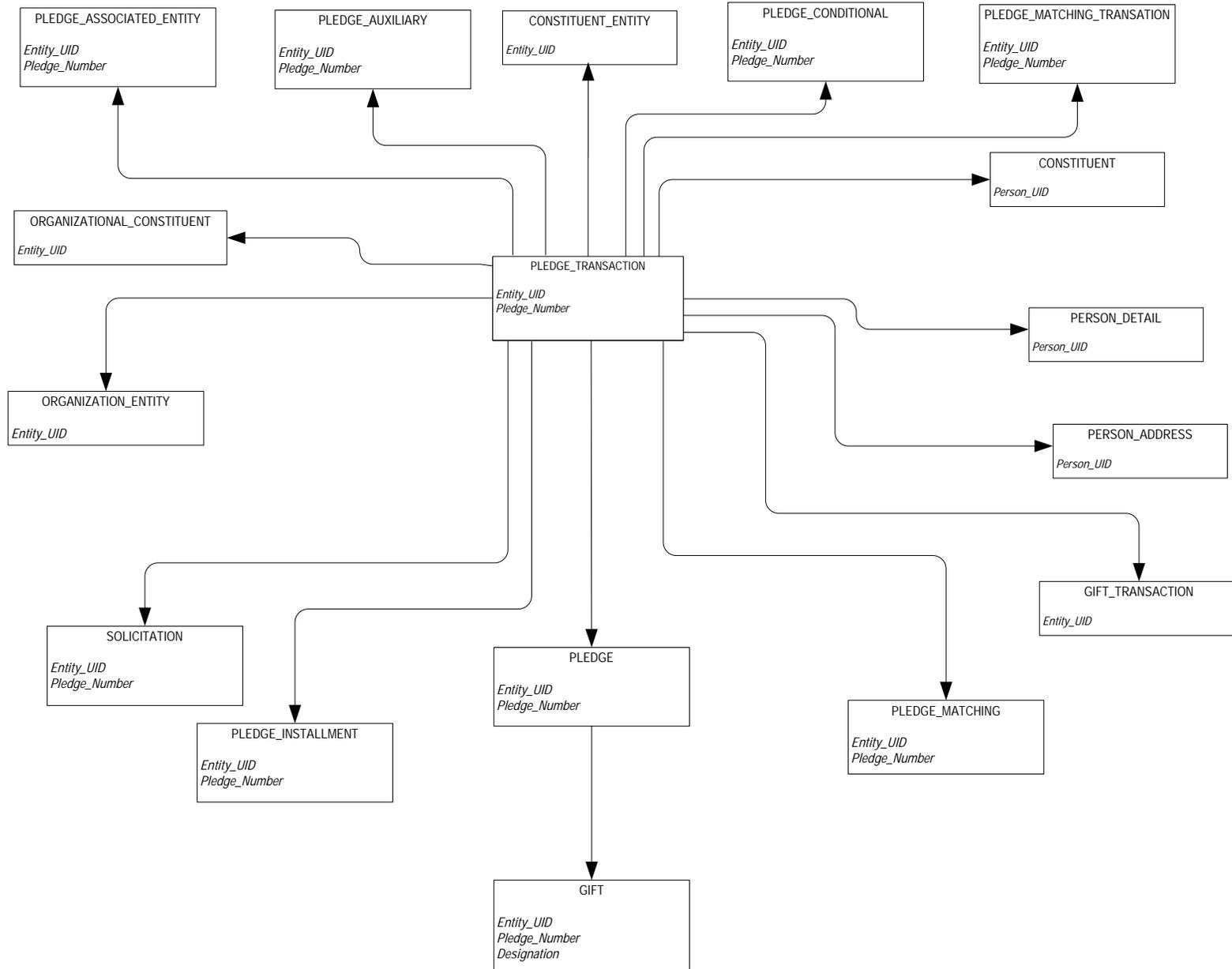
# Gift



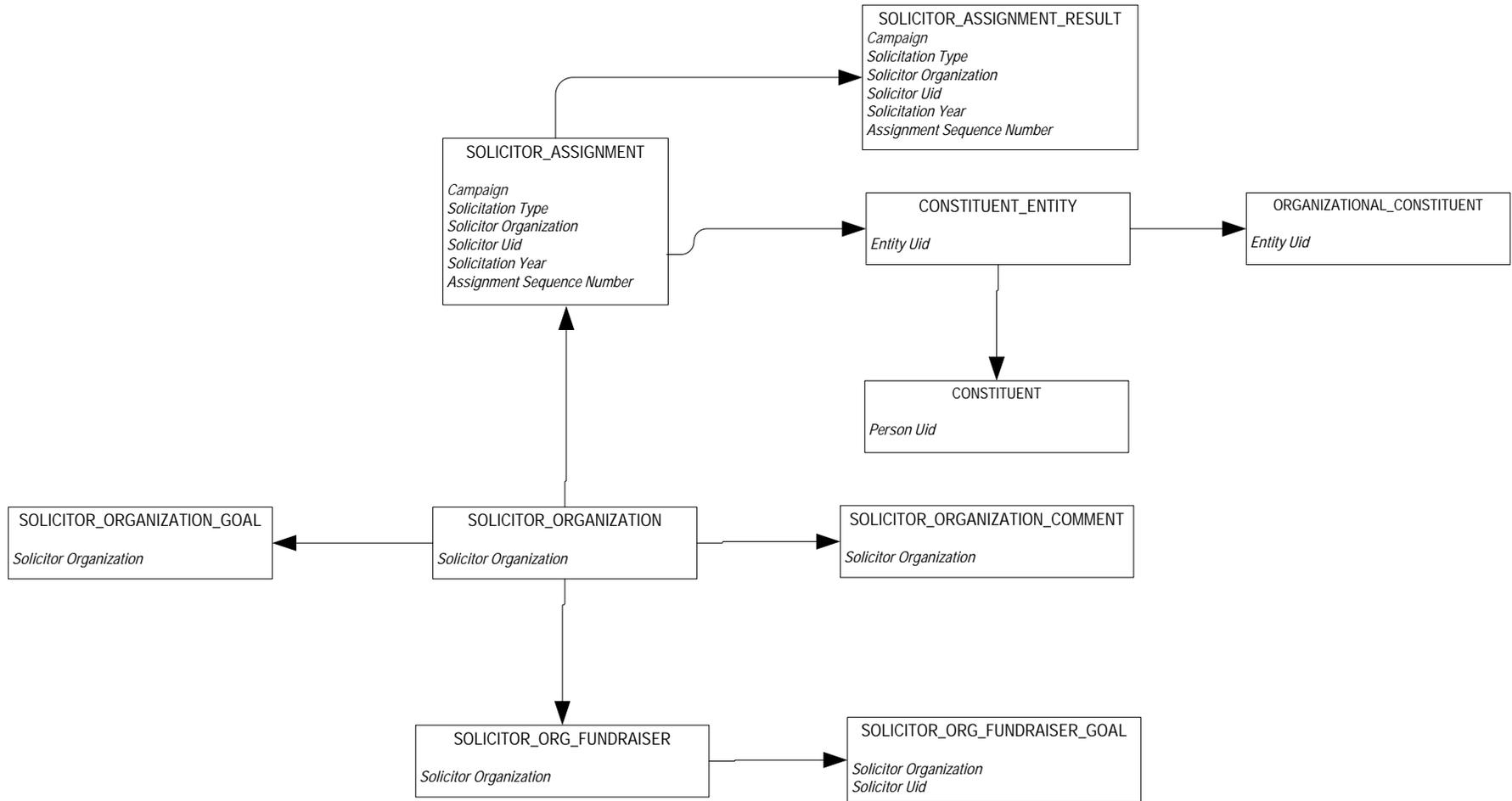
# Organizational Constituent



# Pledge

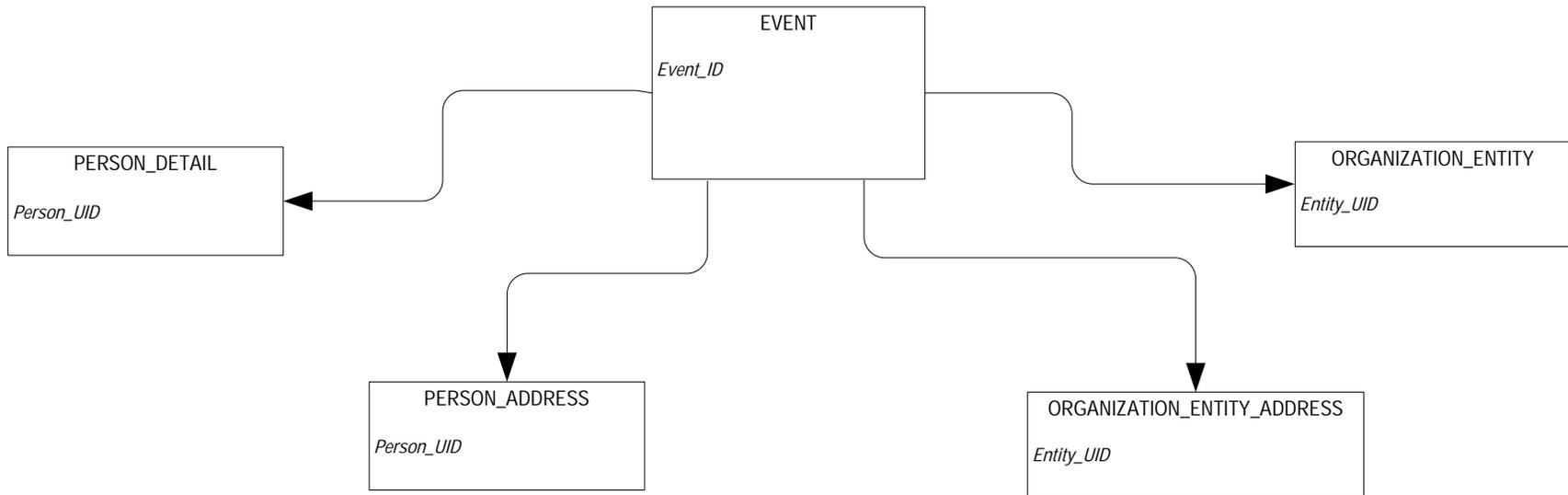


# Solicitation Effort

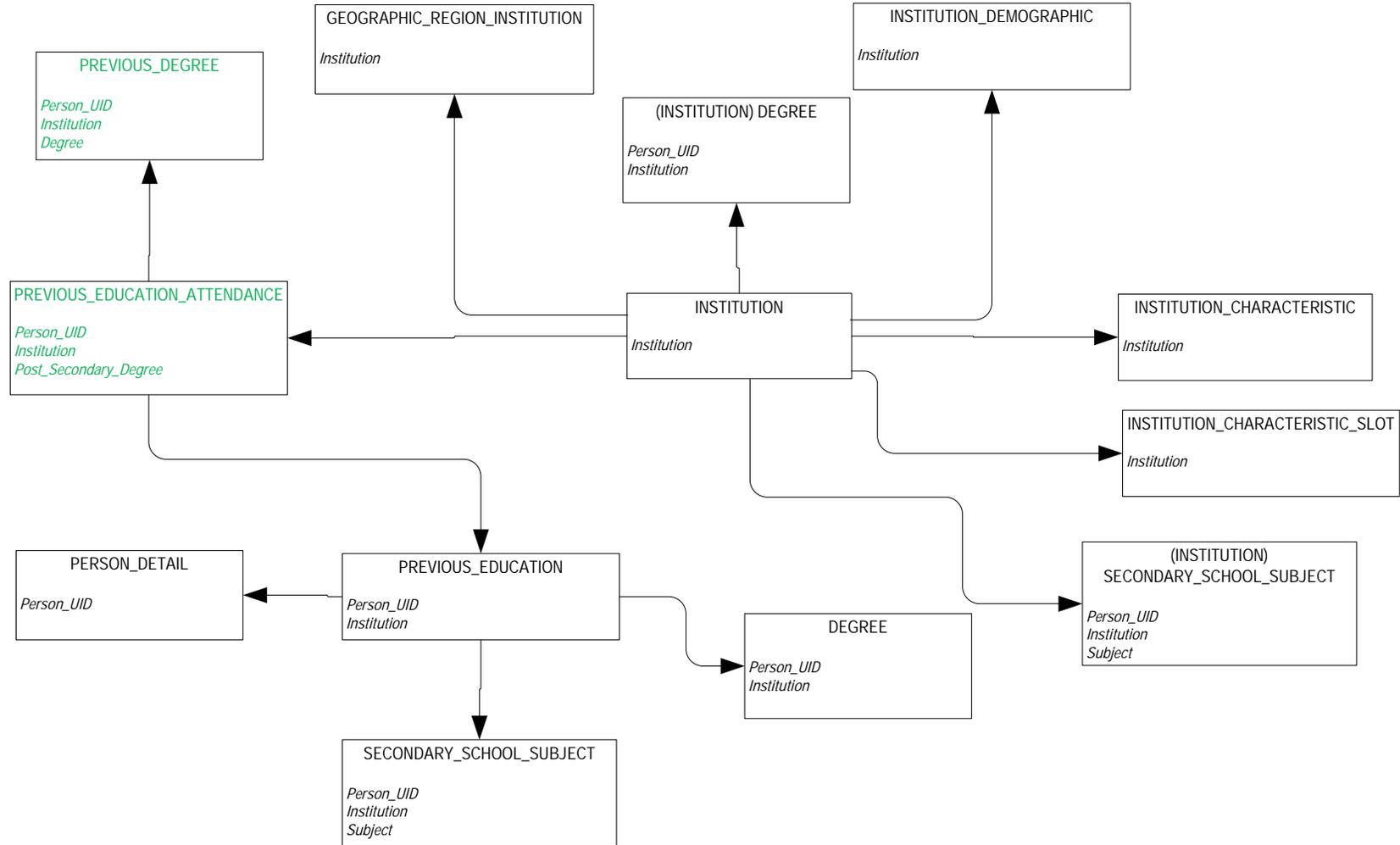


# Common

## Event

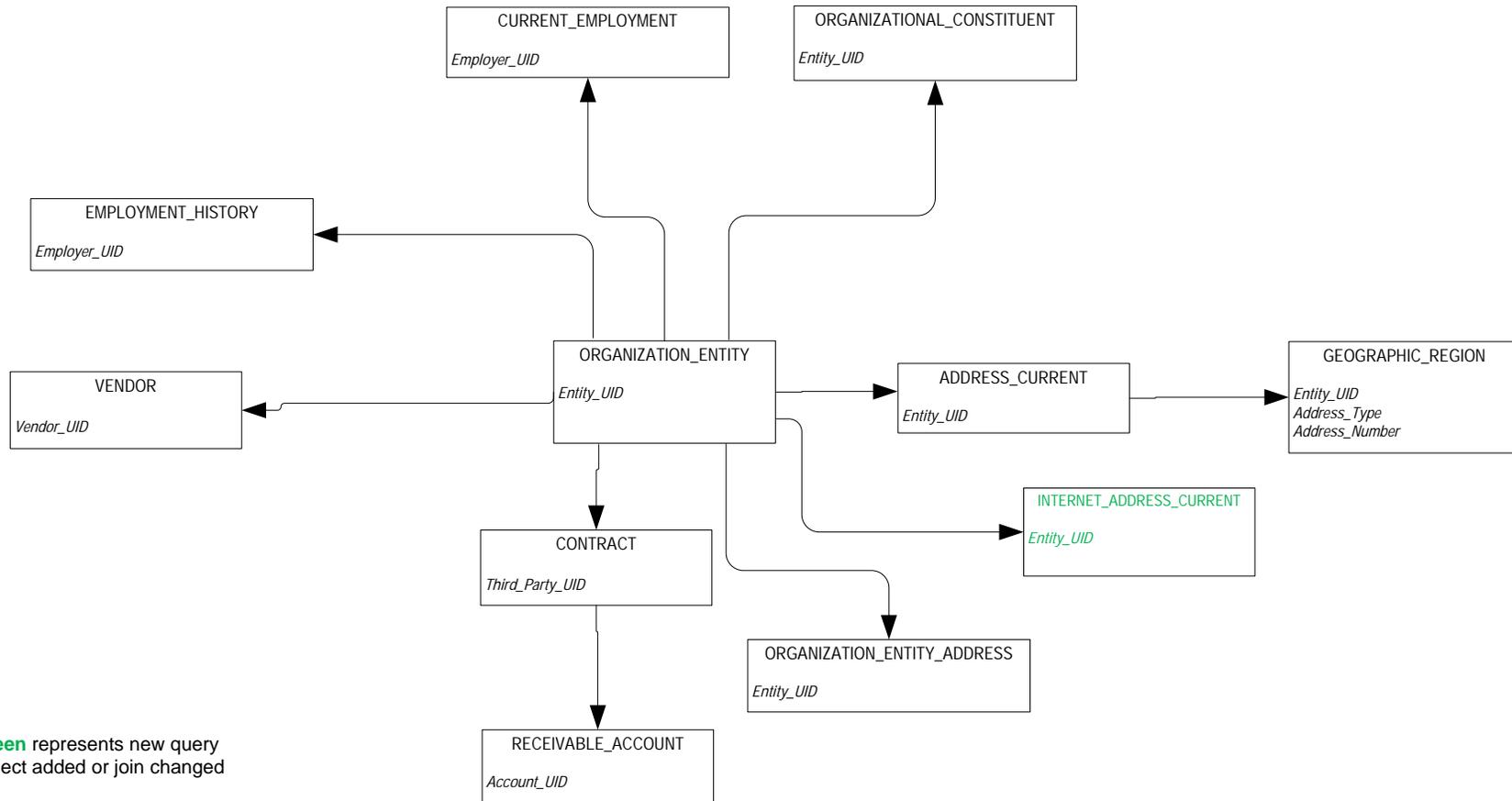


# Institution

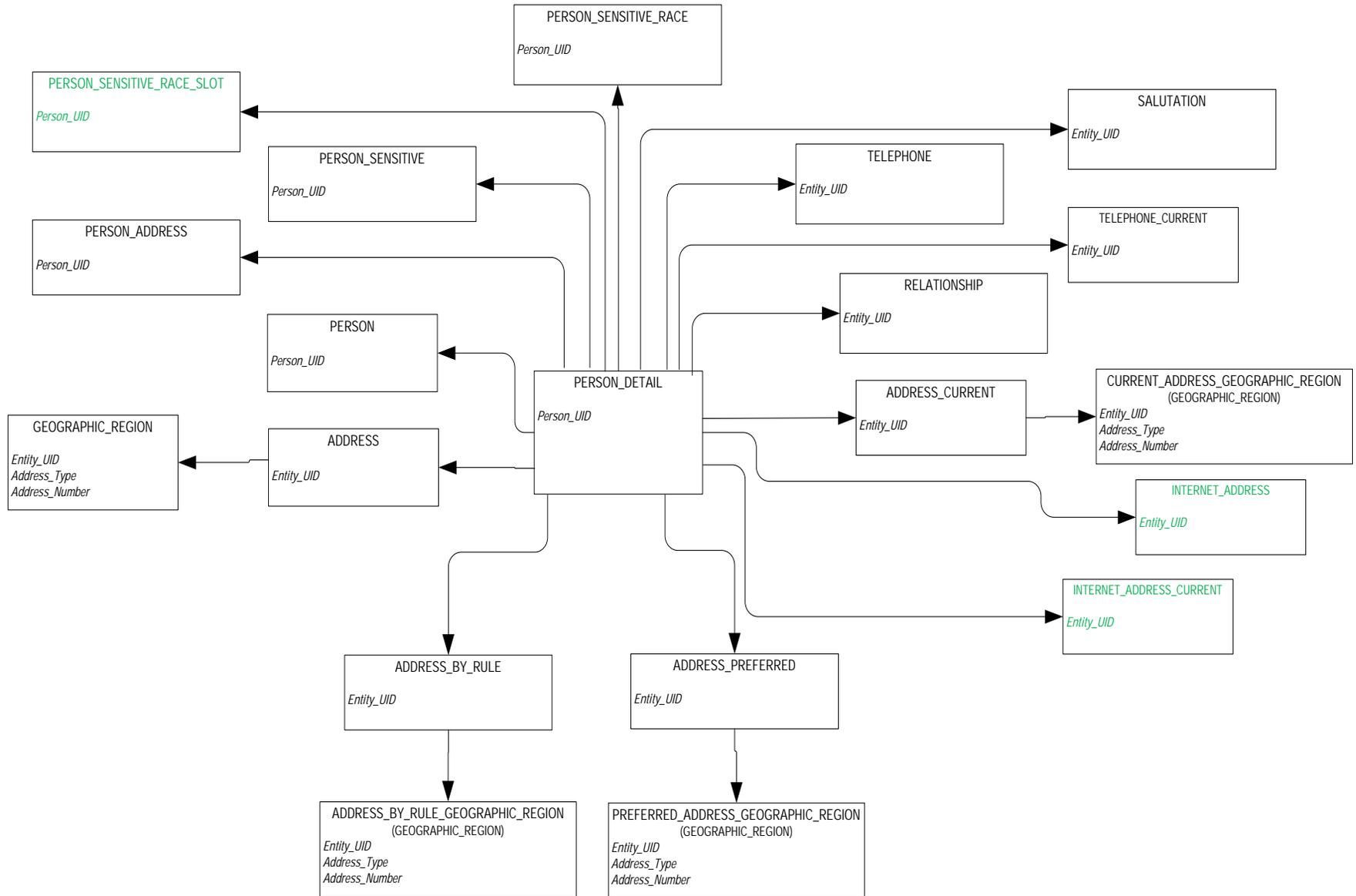


Green represents new query subject added or join changed

# Organization Entity

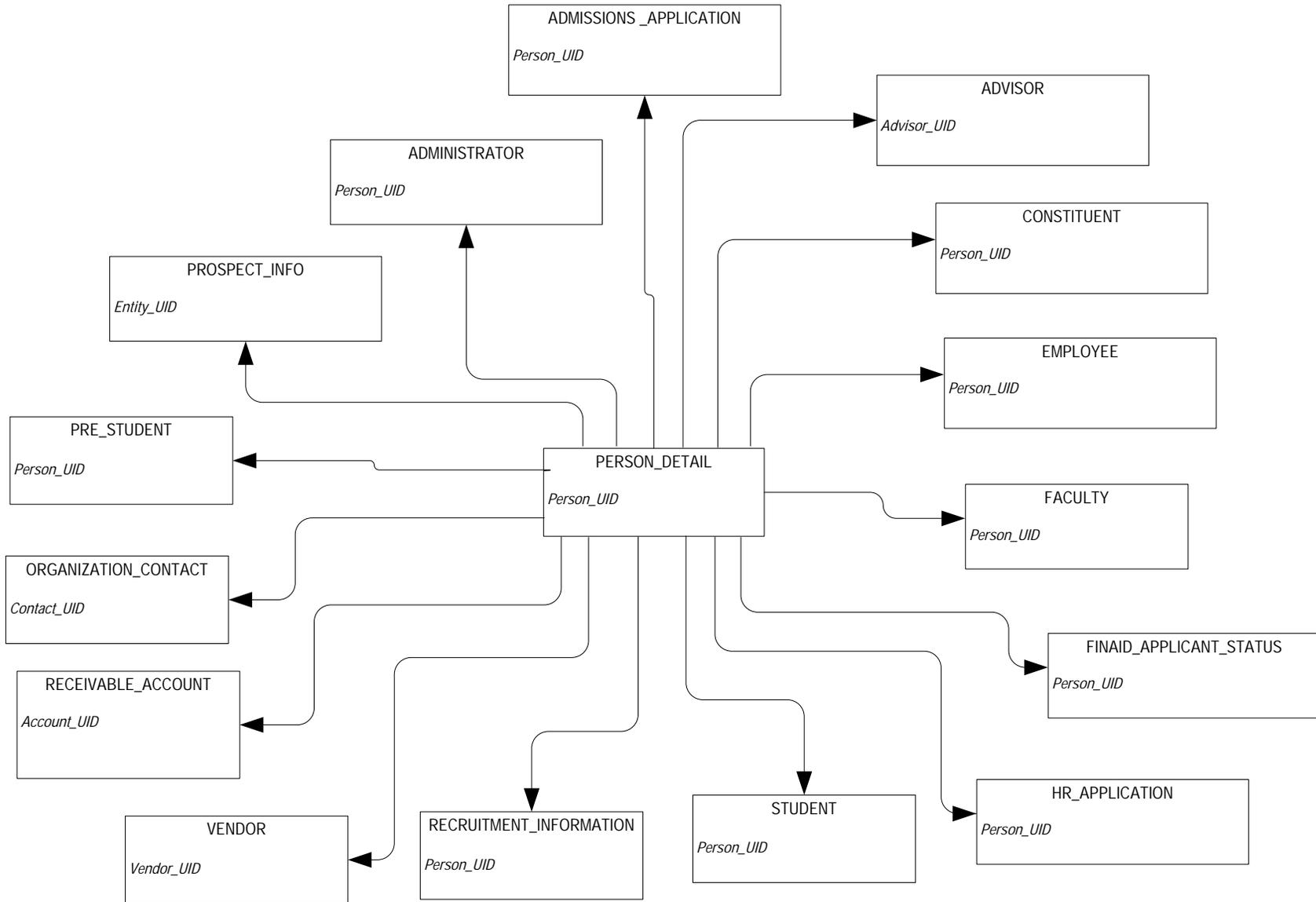


# Person Demographic

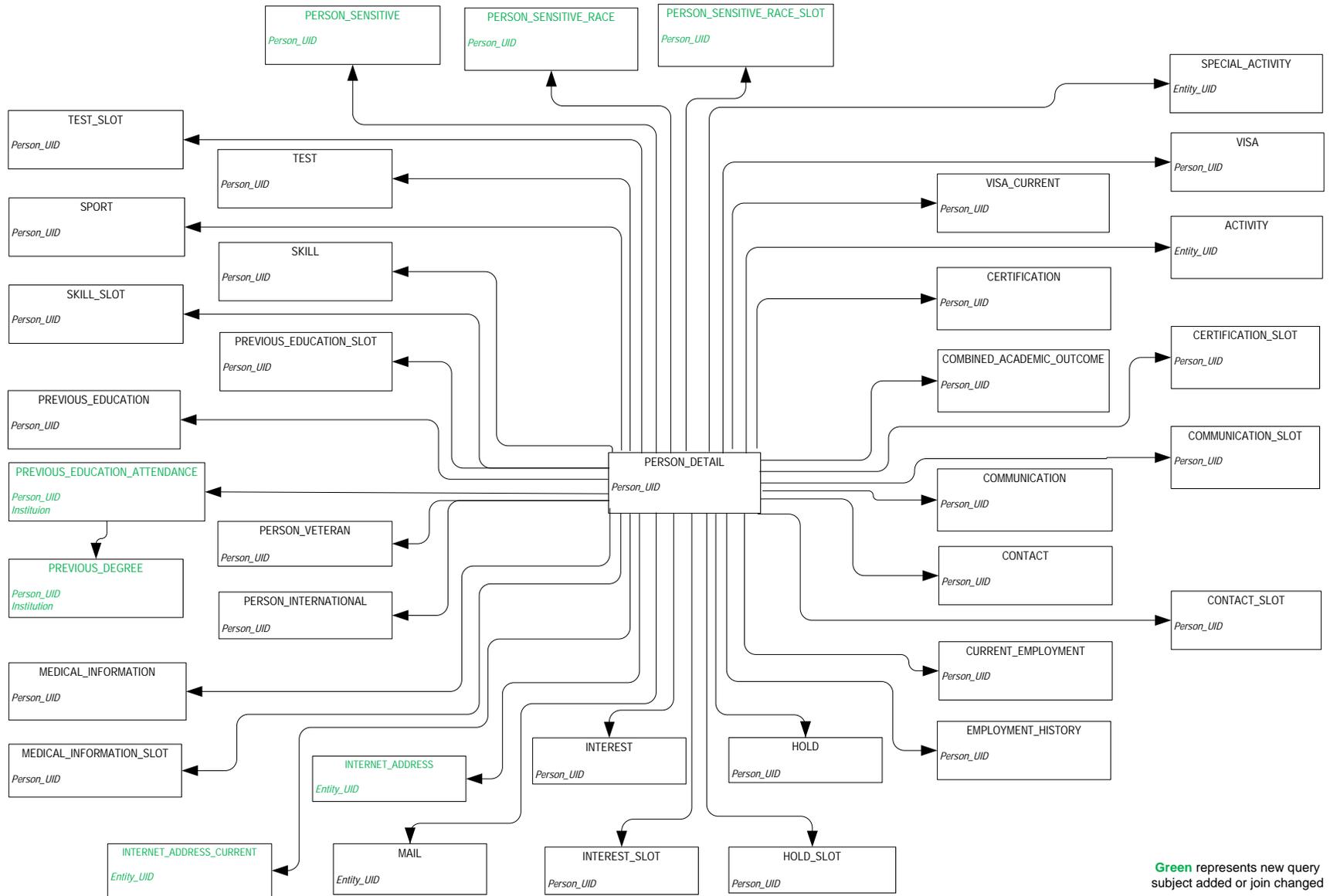


Green represents new query  
subject added or join changed

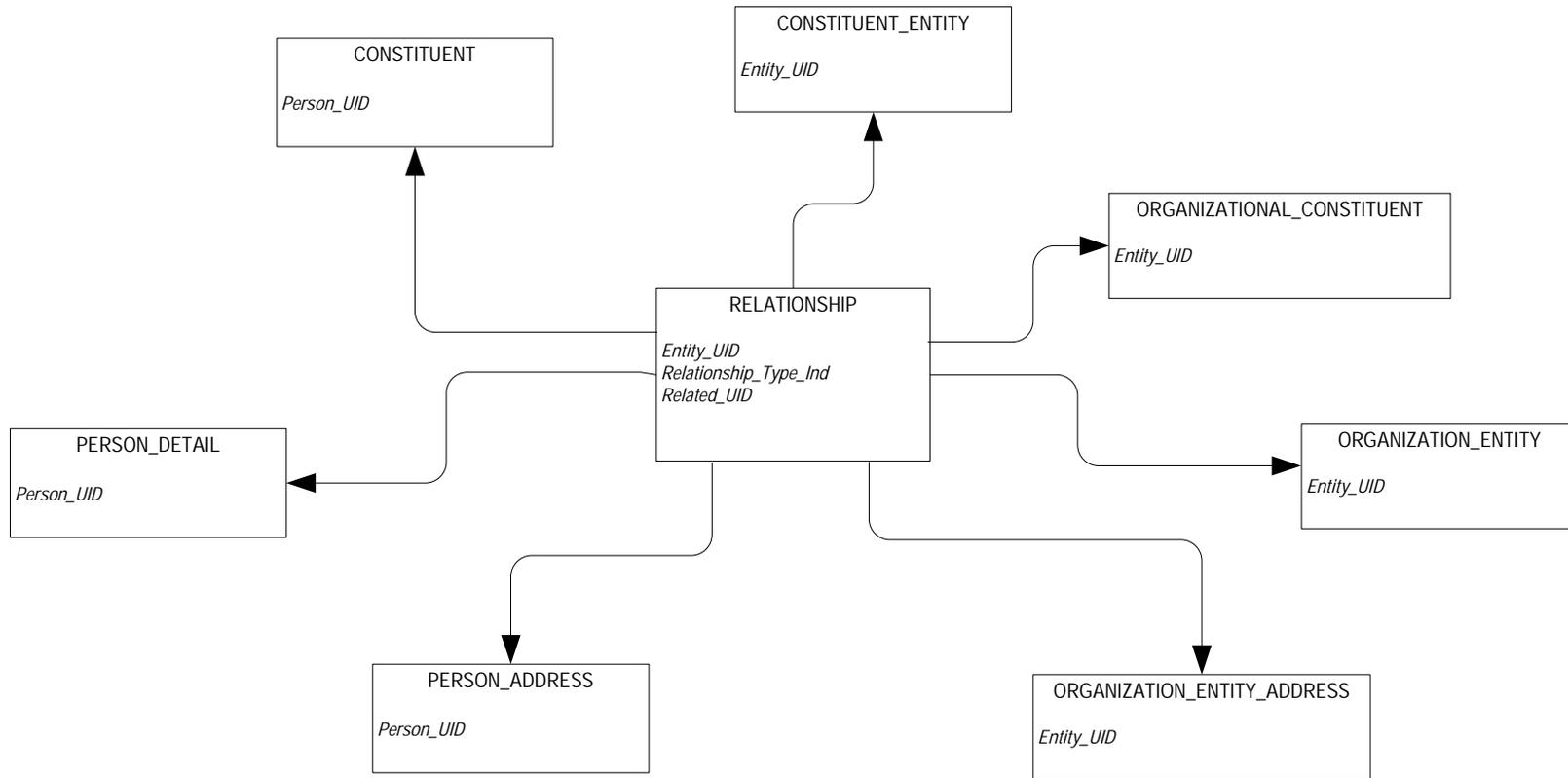
## Person Role



# Person Supplemental

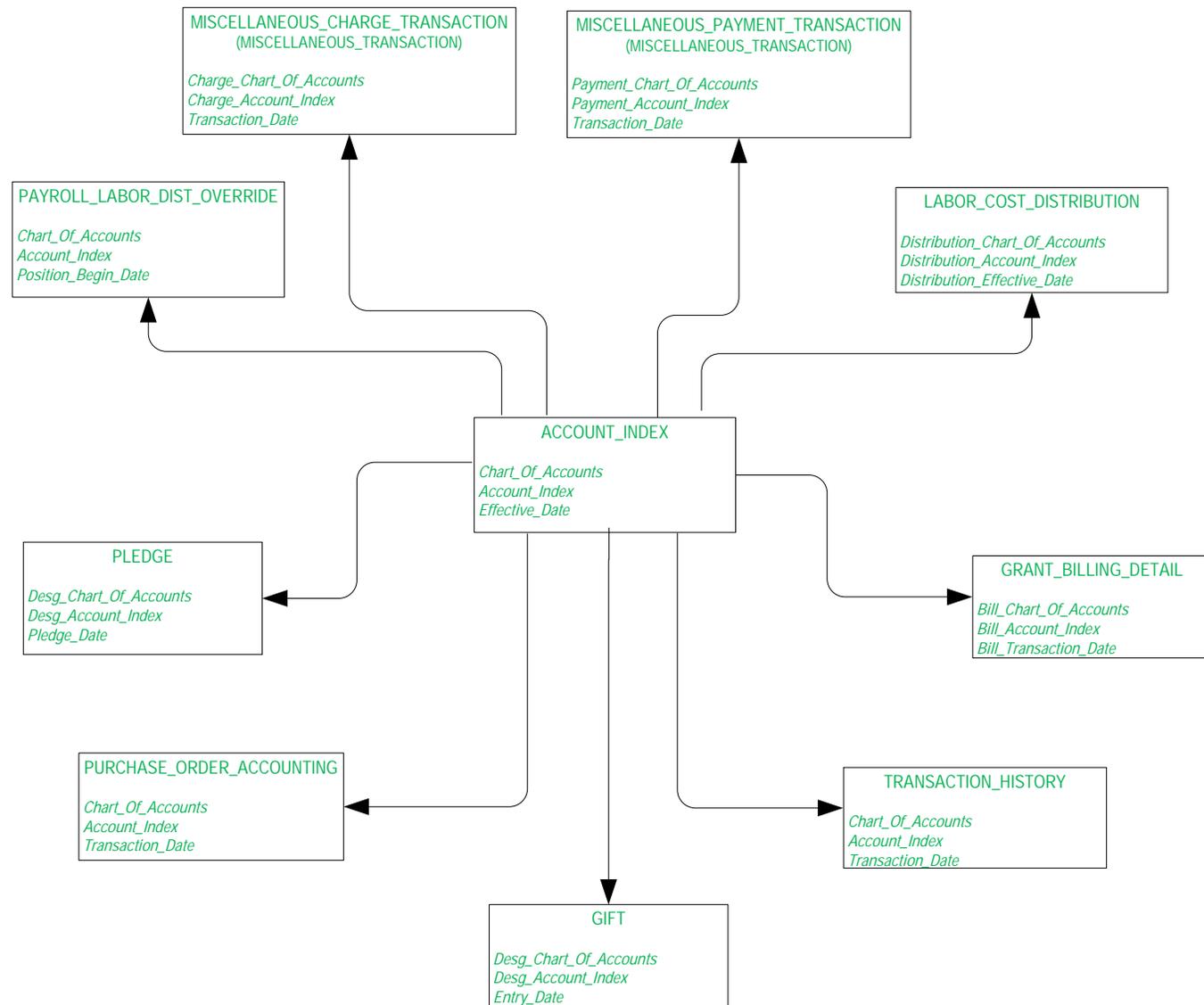


## Relationship



# Finance

## Account Index Audit



## Budget Availability Comparison

BUDGET\_AVAILABILITY\_LEDGER

*Chart Of Accounts*

*Fiscal Year*

*Fiscal Period*

*Fund*

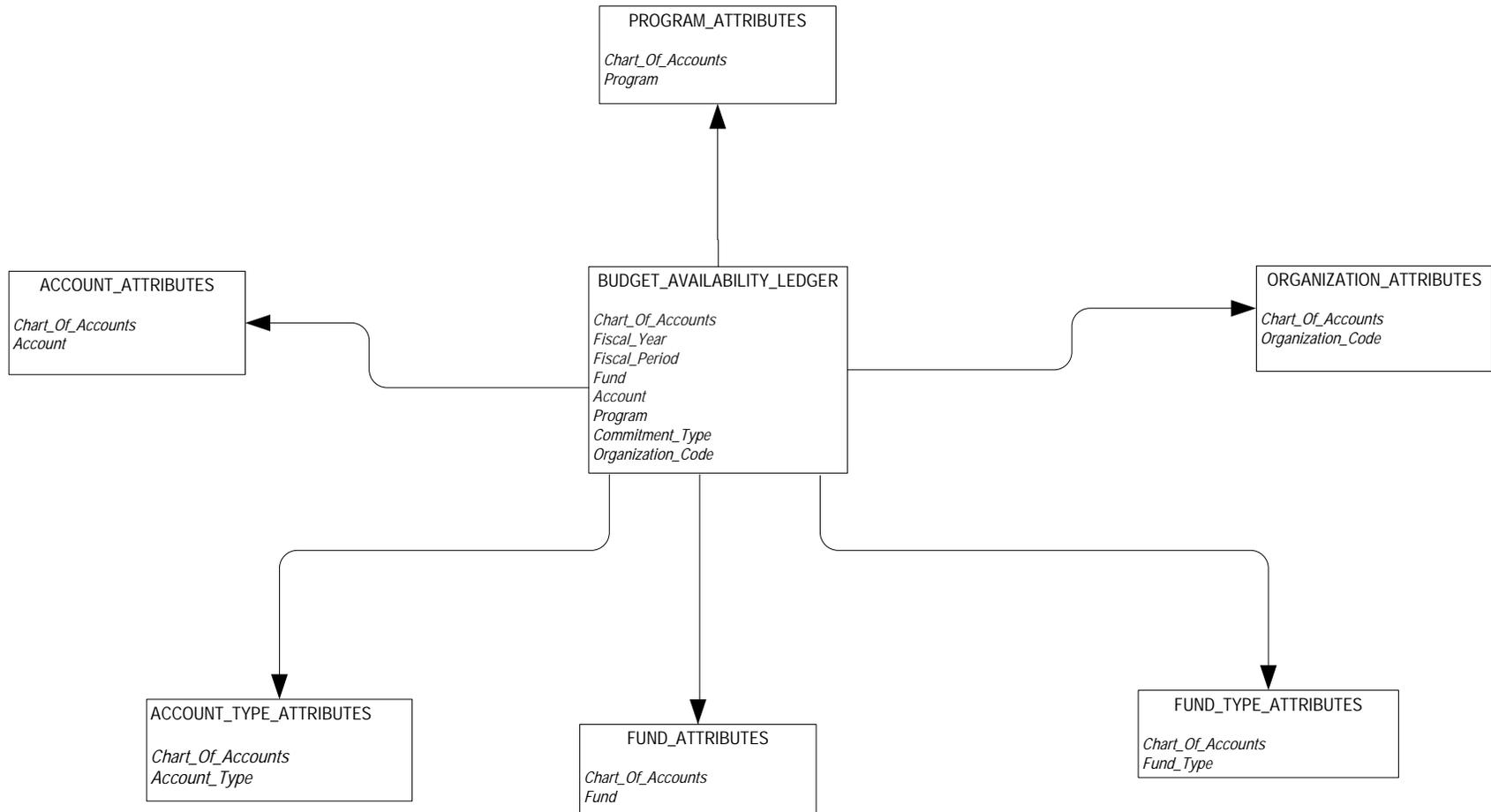
*Organization Code*

*Account*

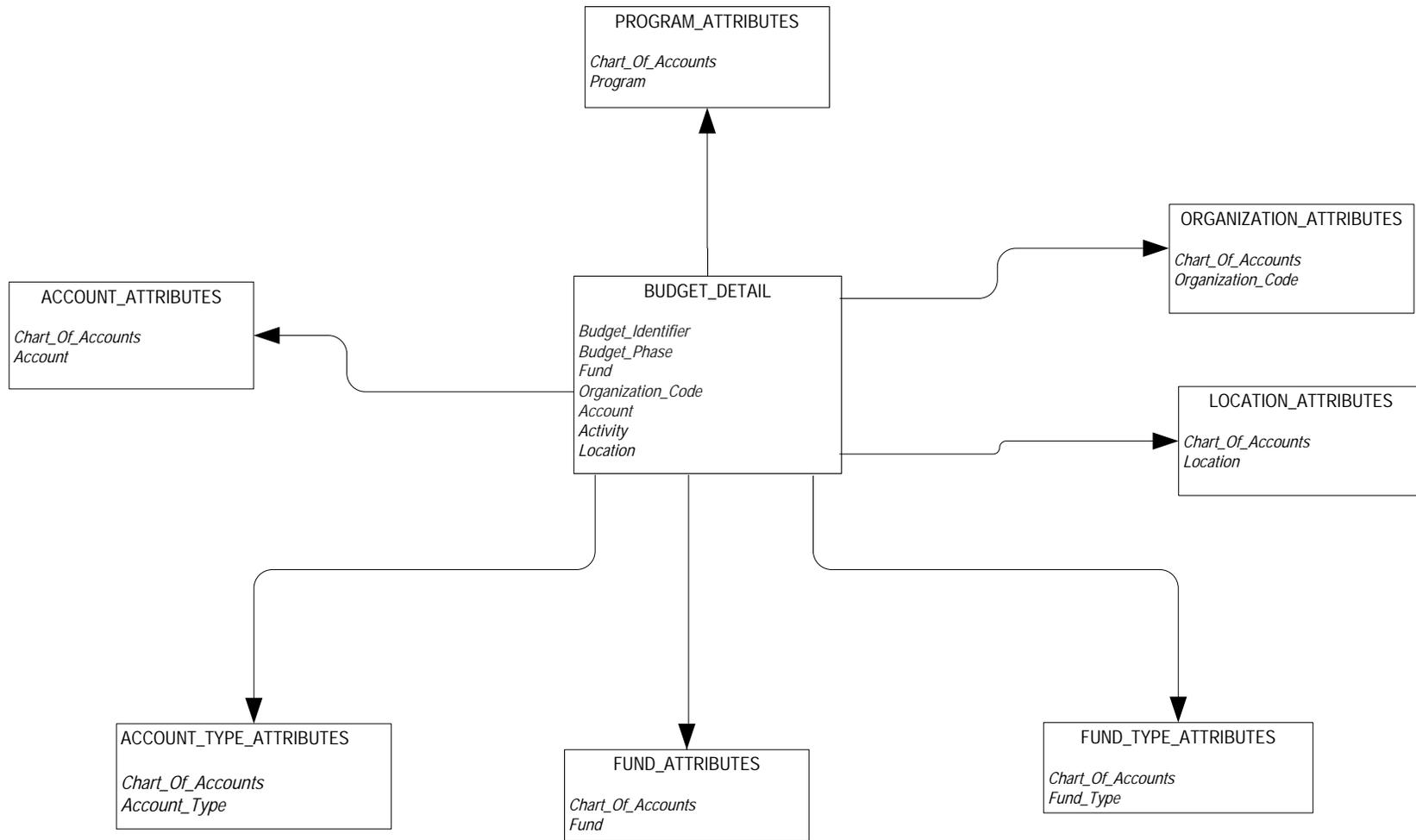
*Program*

*Commitment Type*

## Budget Availability Ledger



## Budget Detail

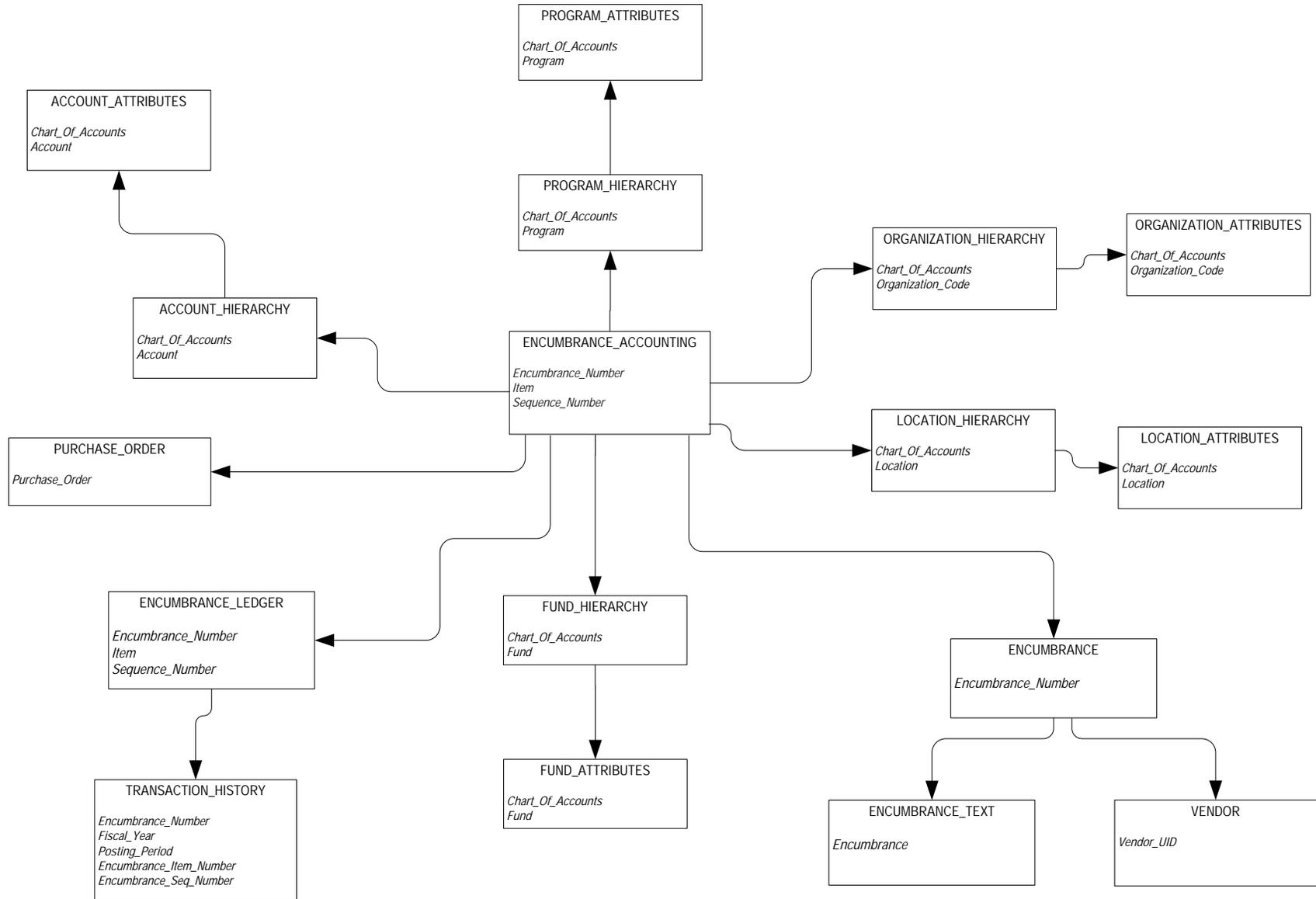


## Cashier Session Analysis

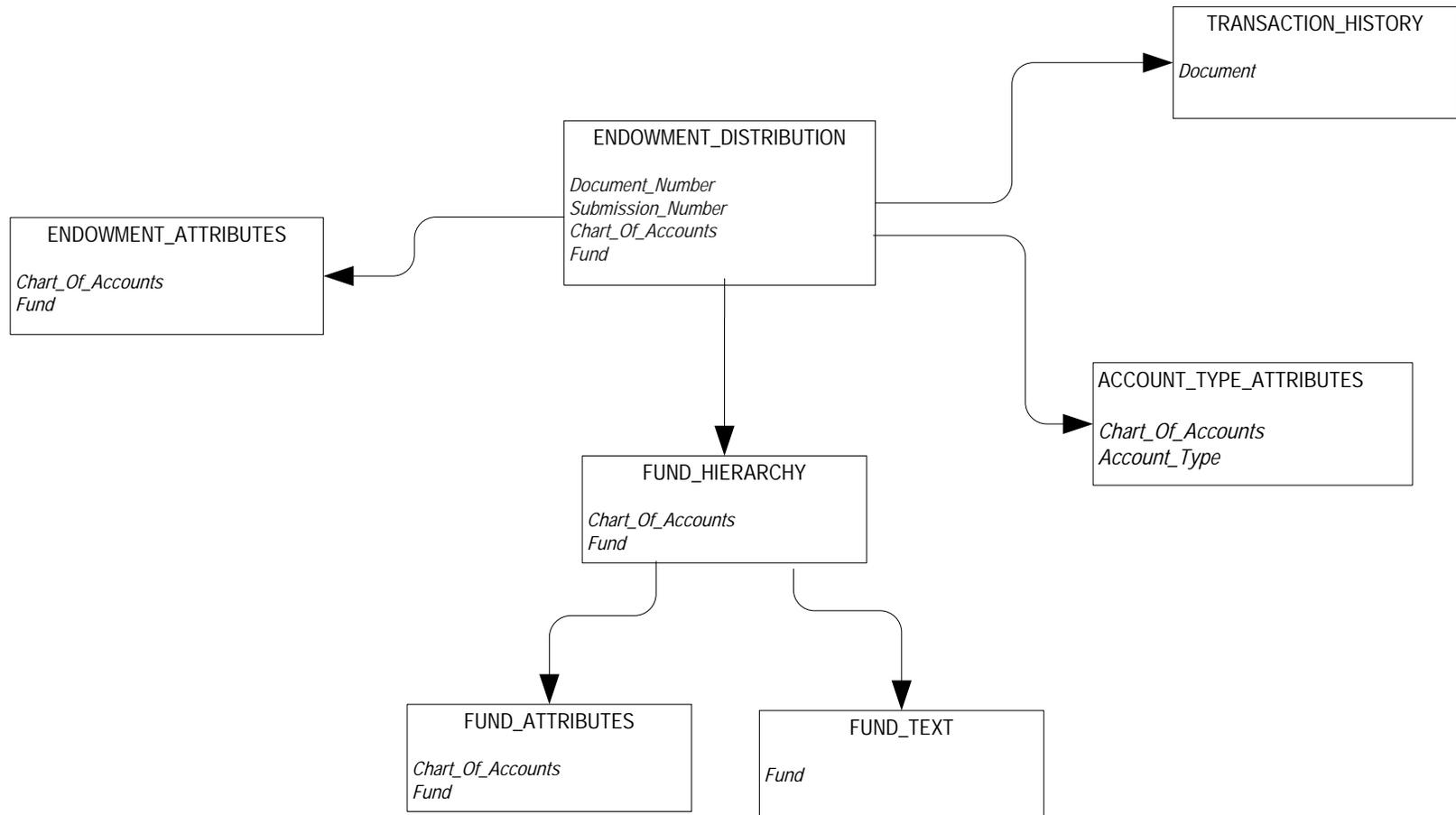
RECEIVABLE\_ACCOUNT\_DETAIL

*Account Uid*

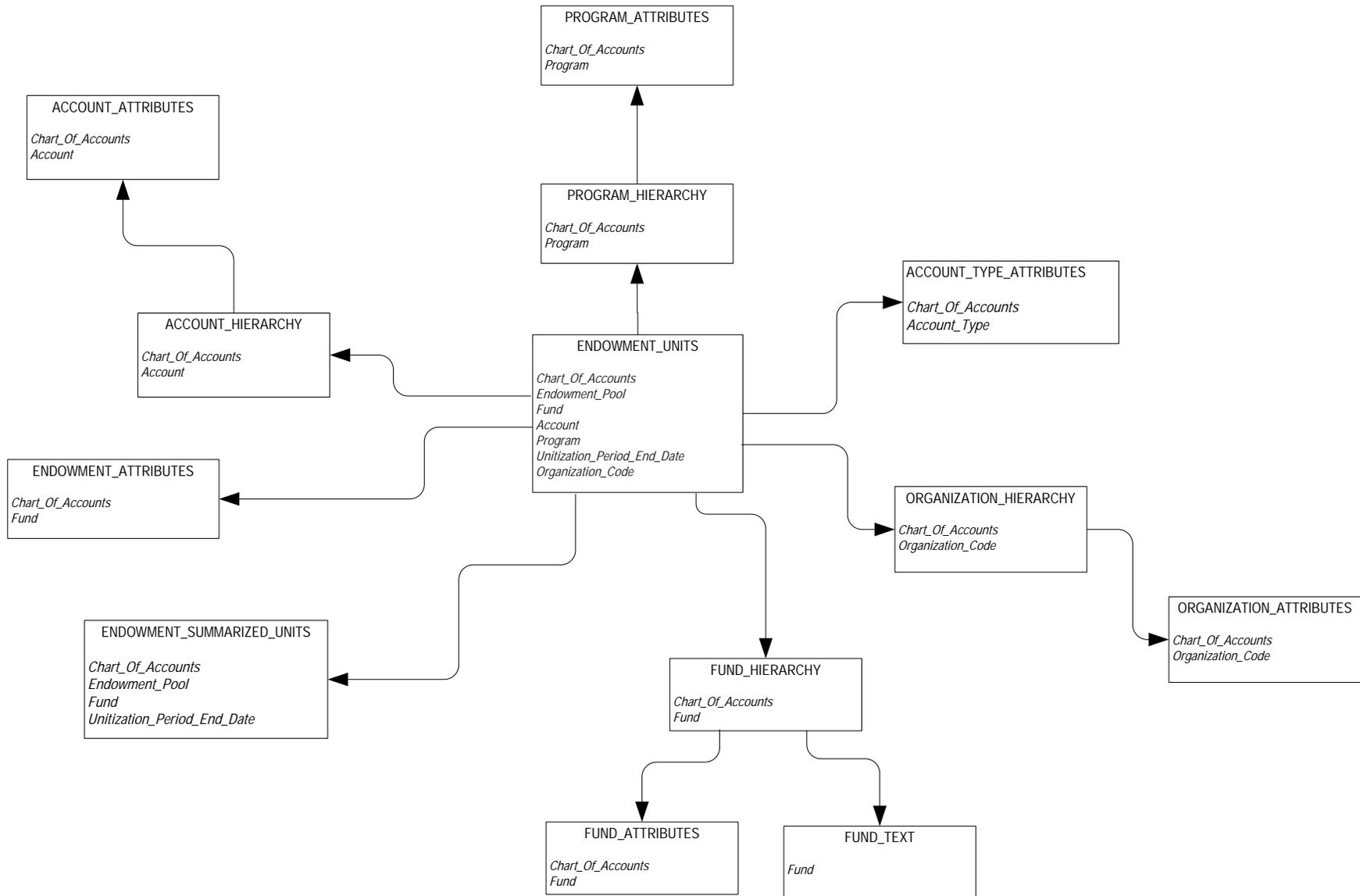
# Encumbrance



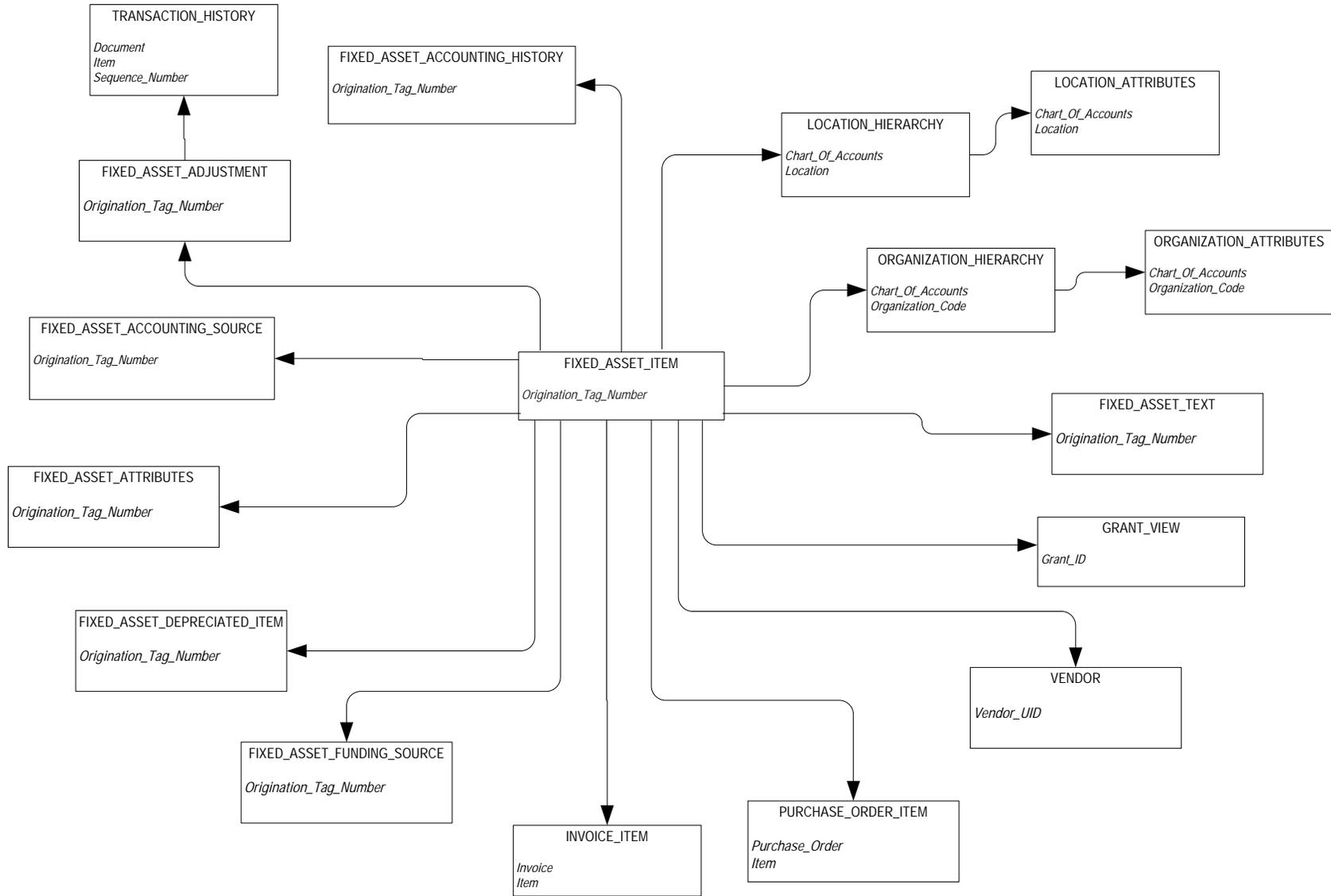
## Endowment Distribution



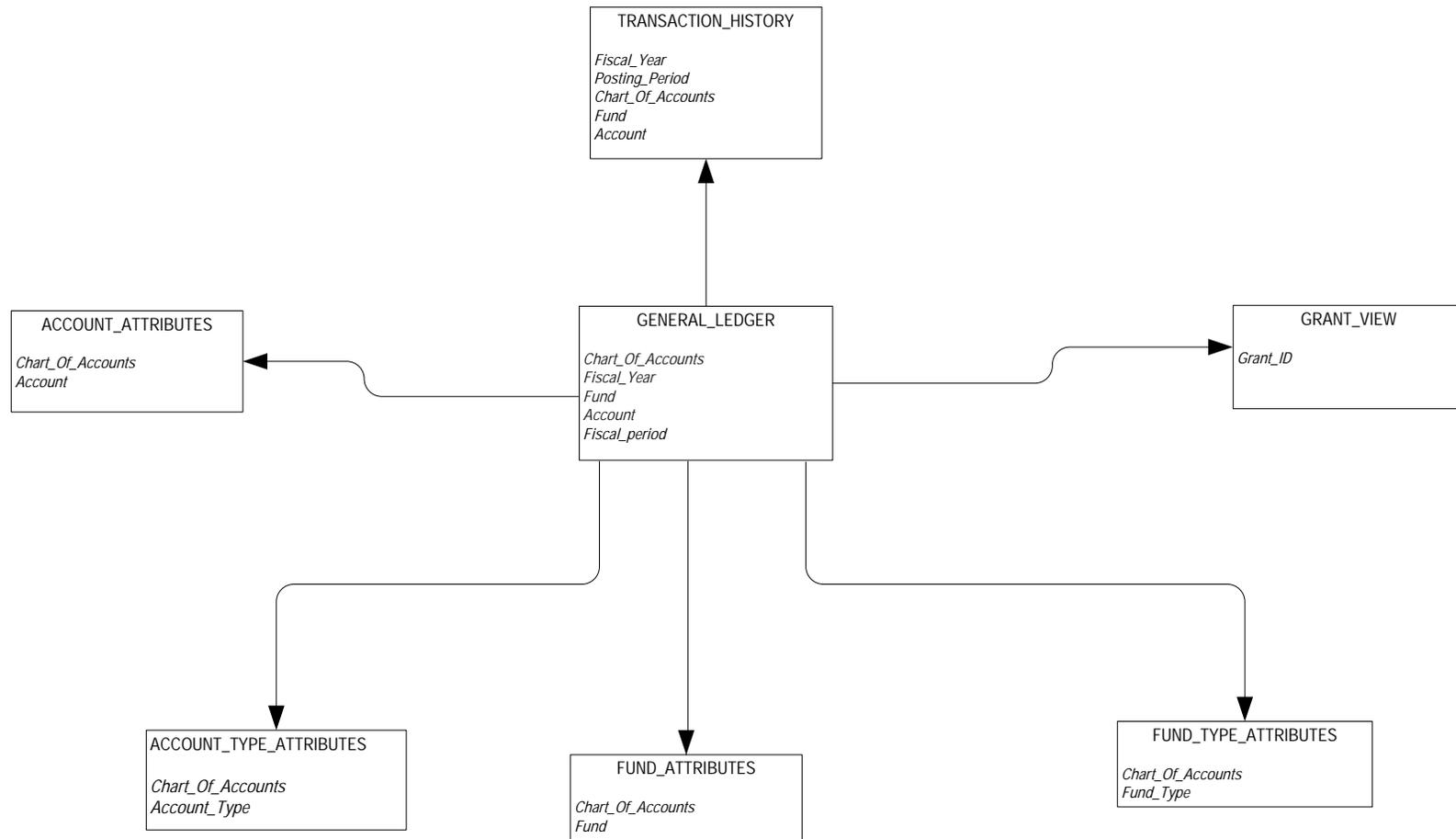
# Endowment Units



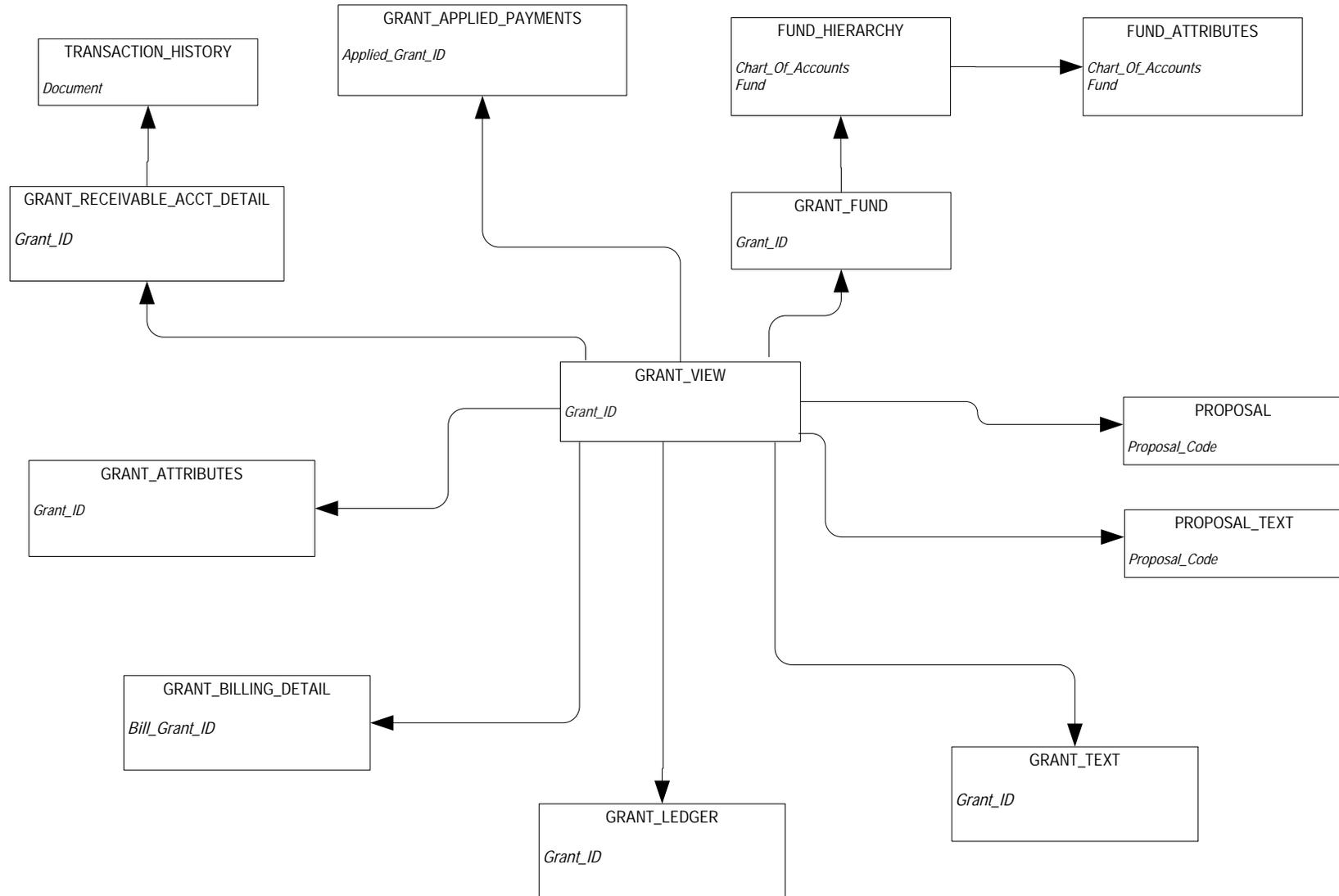
# Fixed Asset



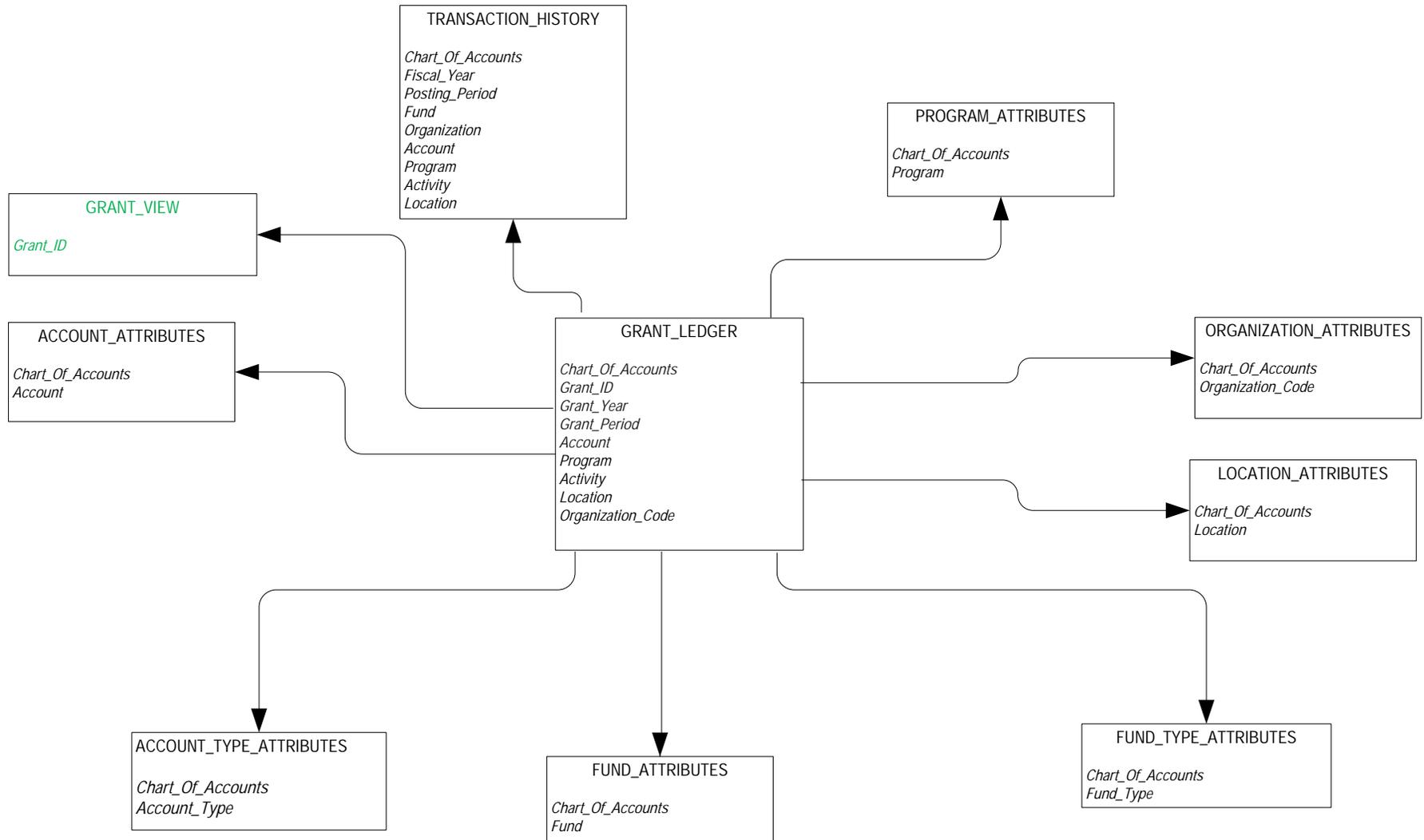
# General Ledger



## Grant and Project



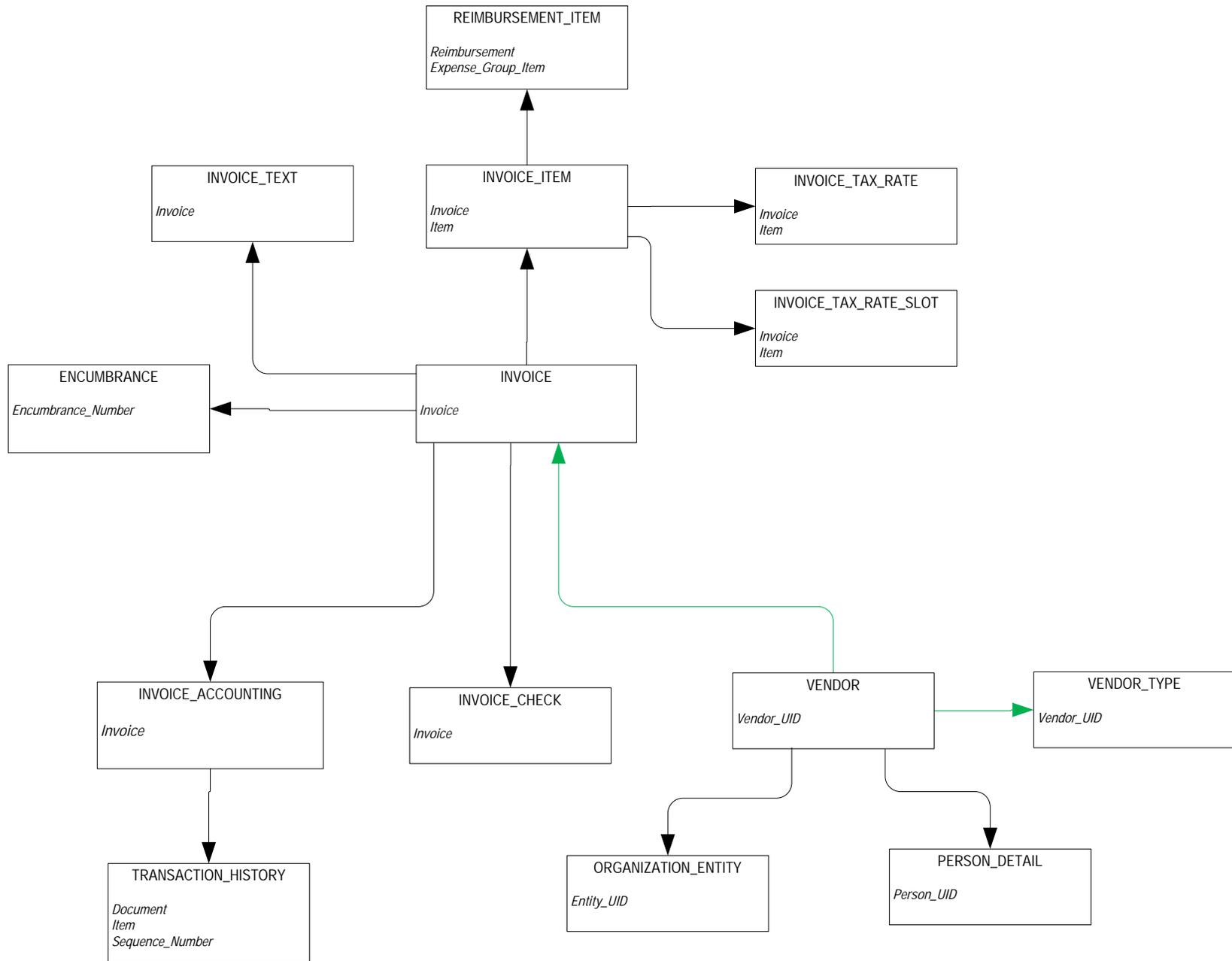
# Grant Ledger



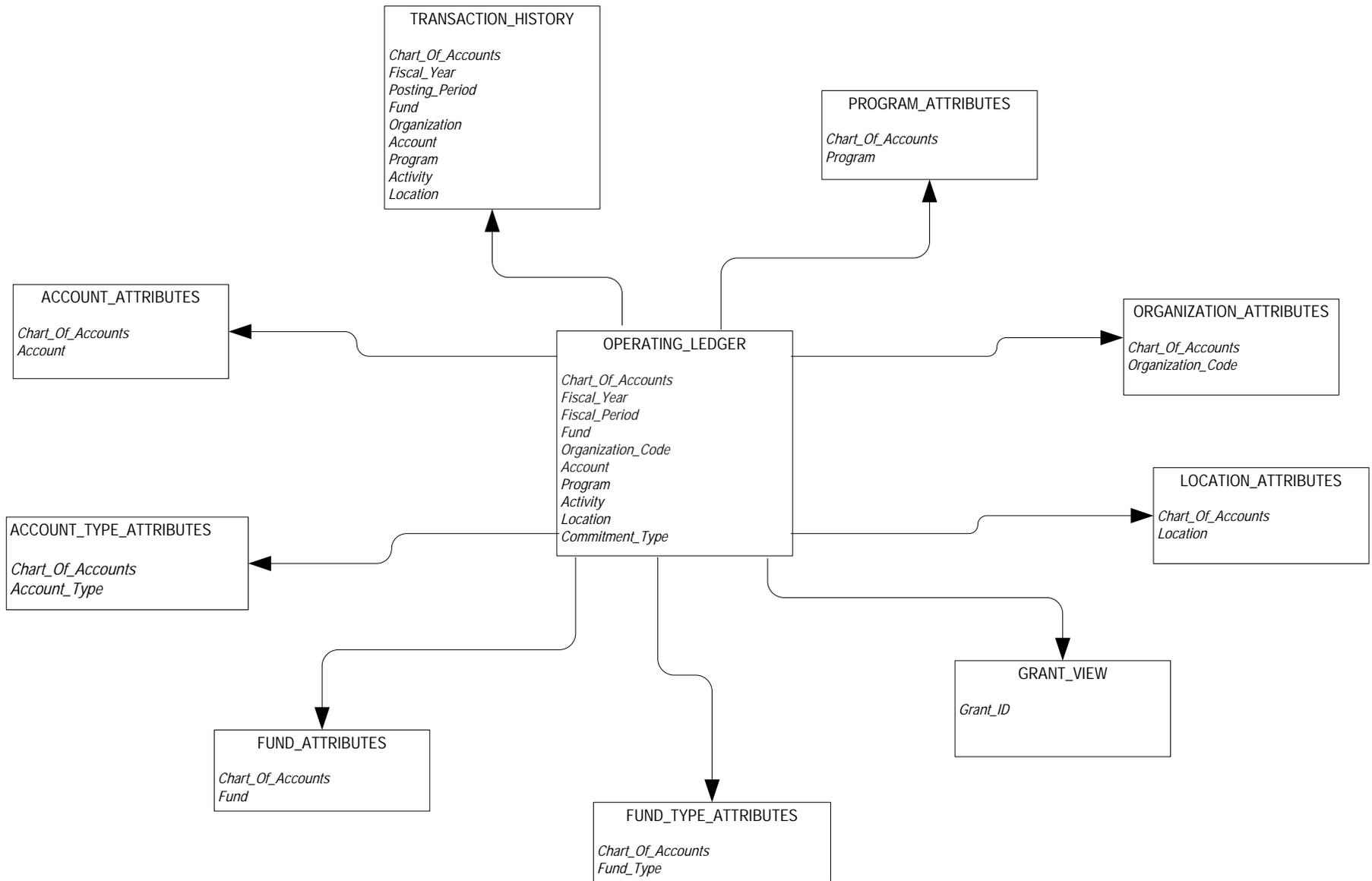
## Grant-Contract and Proposal

PROPOSAL
<i>Proposal Code</i>

# Invoice Payable



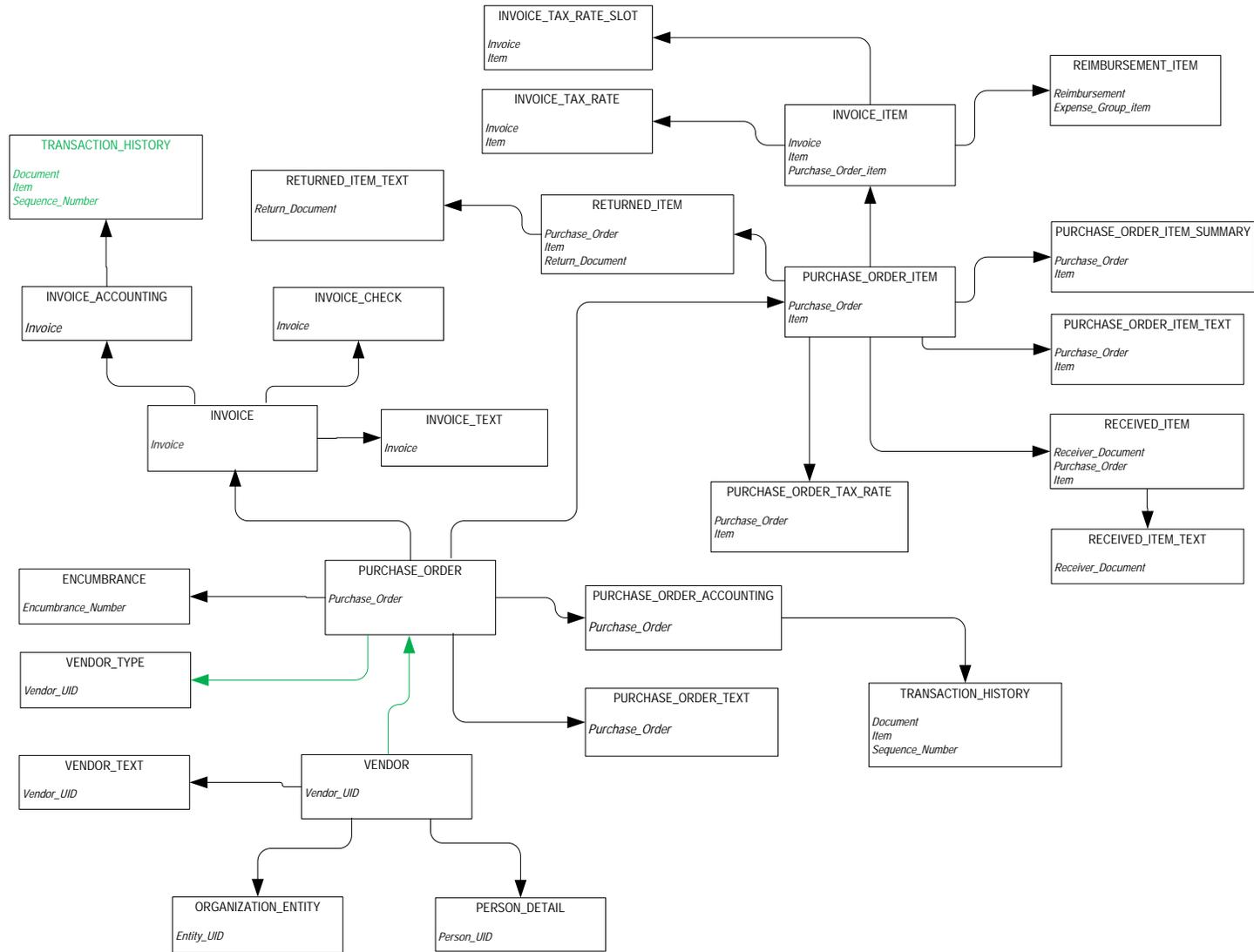
# Operating Ledger



## Operating Ledger Comparison

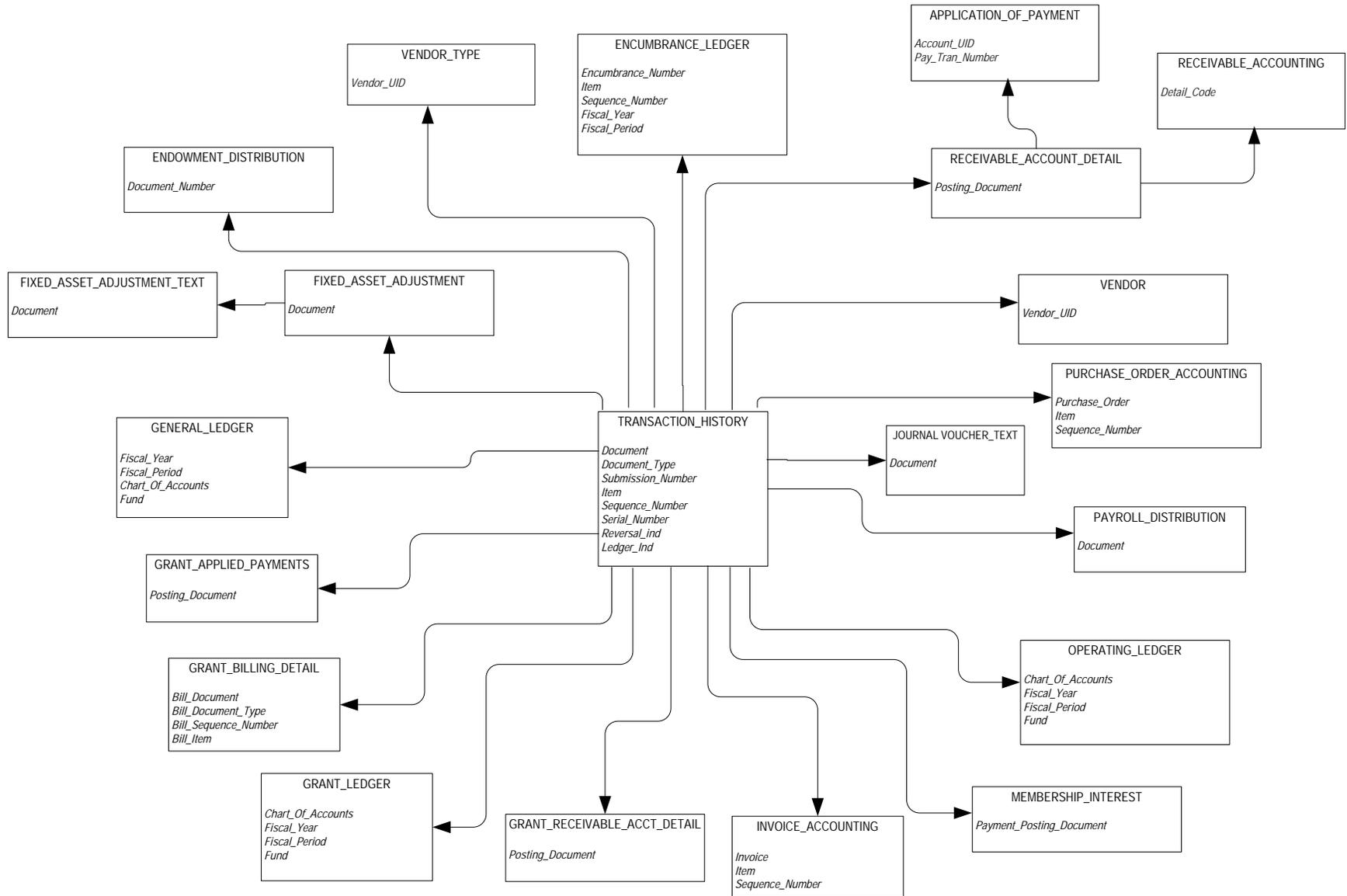
OPERATING_LEDGER
<i>Chart Of Accounts</i>
<i>Fiscal Year</i>
<i>Fiscal Period</i>
<i>Fund</i>
<i>Organization Code</i>
<i>Account</i>
<i>Program</i>
<i>Activity</i>
<i>Location</i>
<i>Commitment Type</i>

# Purchasing Payable



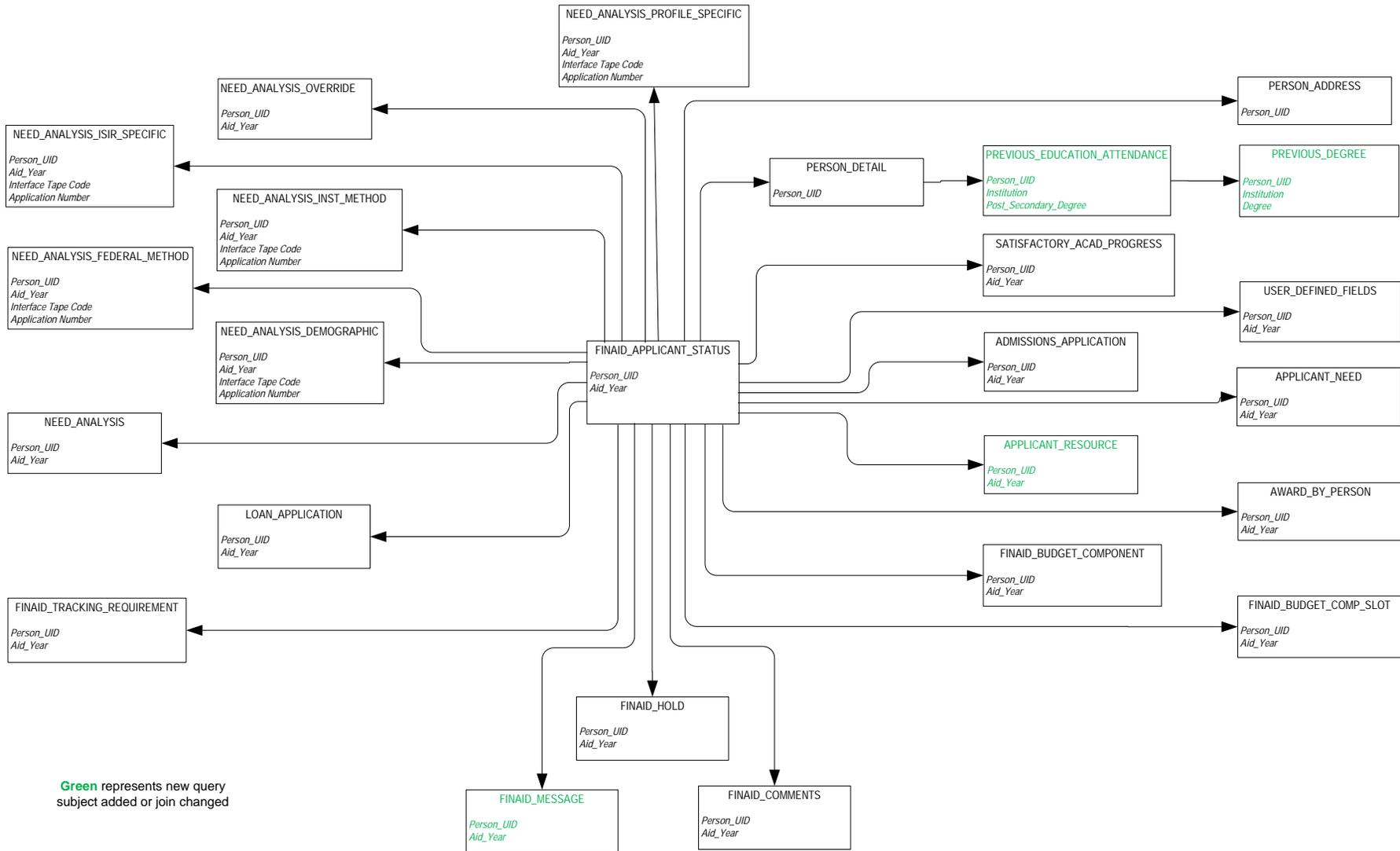
Green represents new query subject added or join changed.  
 NOTE: In order to ensure all vendor types display for the vendor and associated to each purchase order, the cardinality was left between Purchase\_Order and Vendor\_Type but changed to 1..1 0..N

# Transaction History



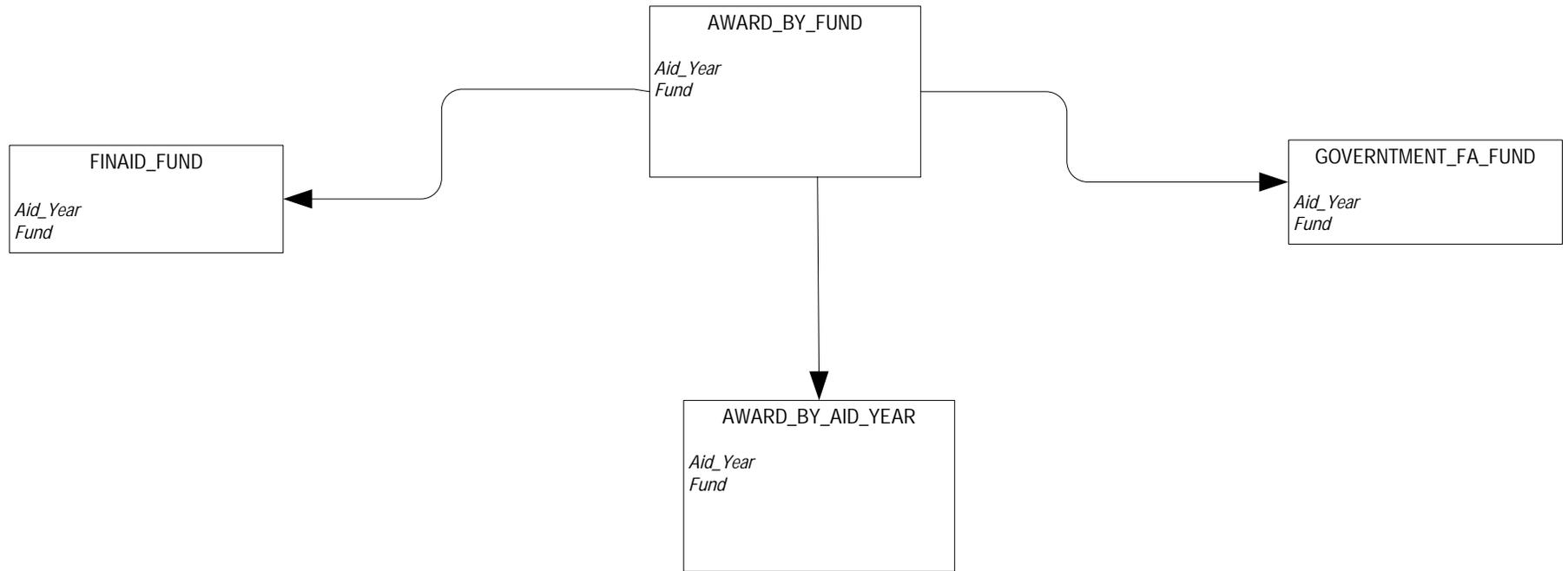
# Financial Aid

## Financial Aid Application

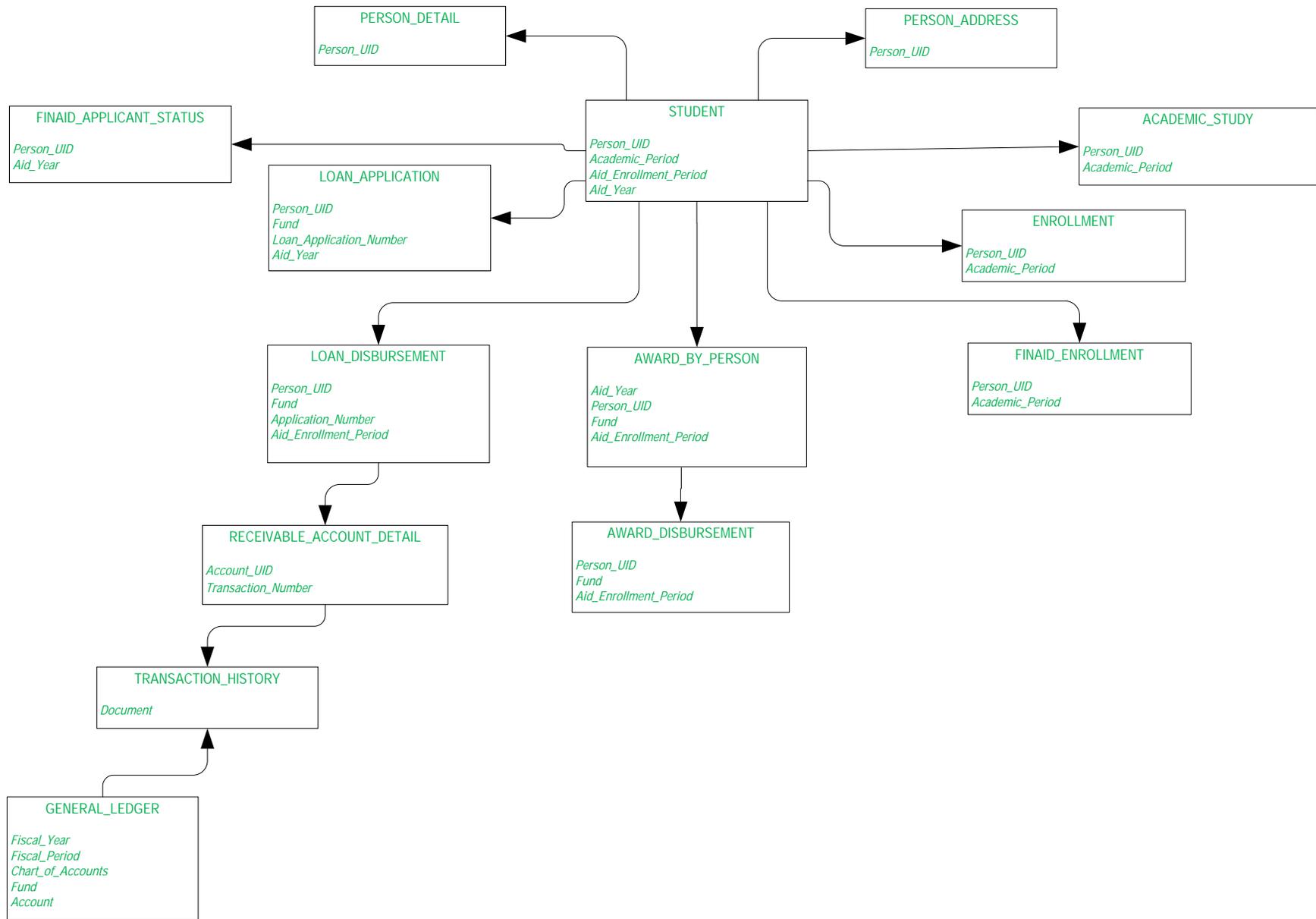




## Financial Aid Fund

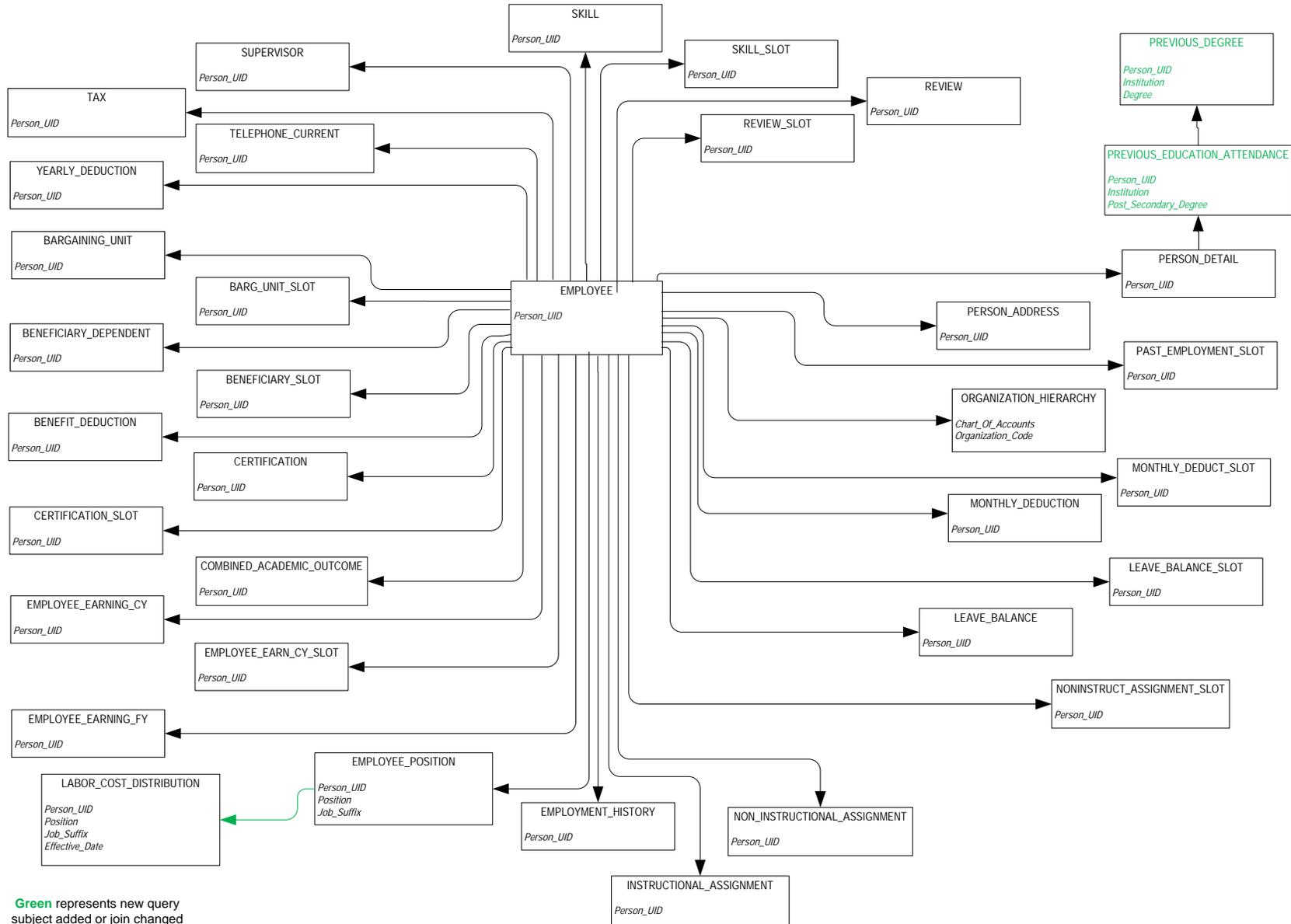


# Loan Disbursement

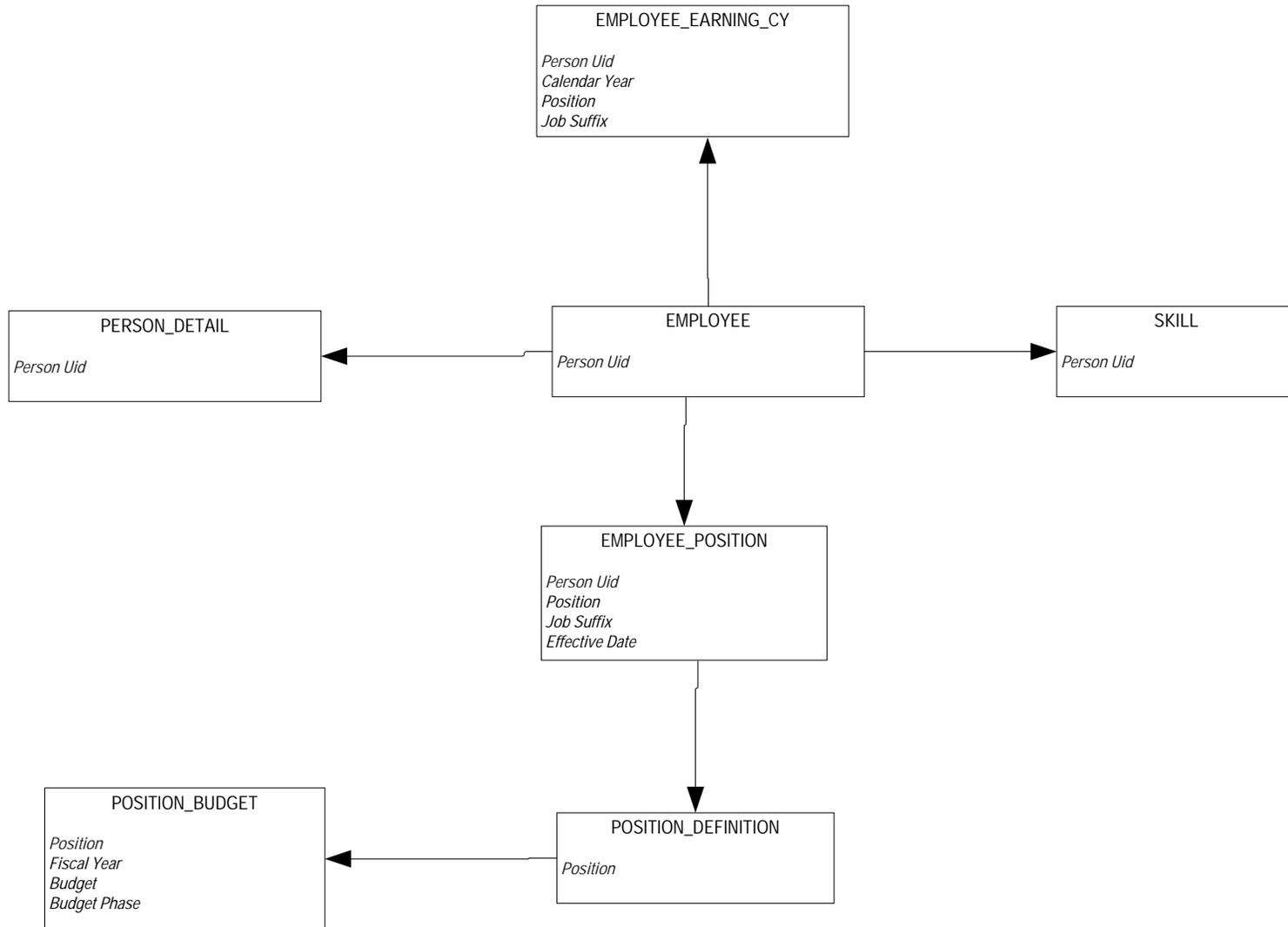


# Human Resources

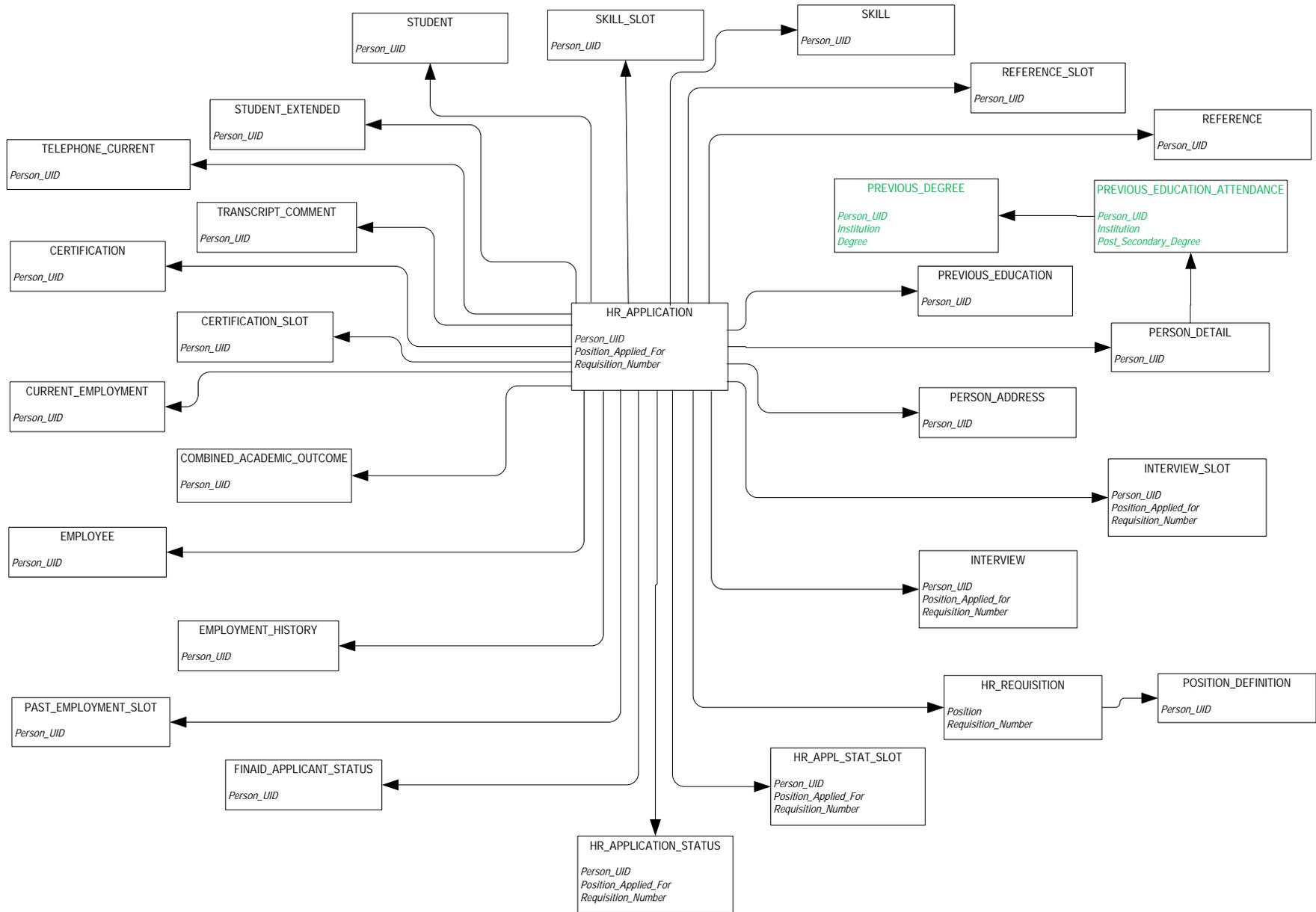
## Employee



## Employee and Position

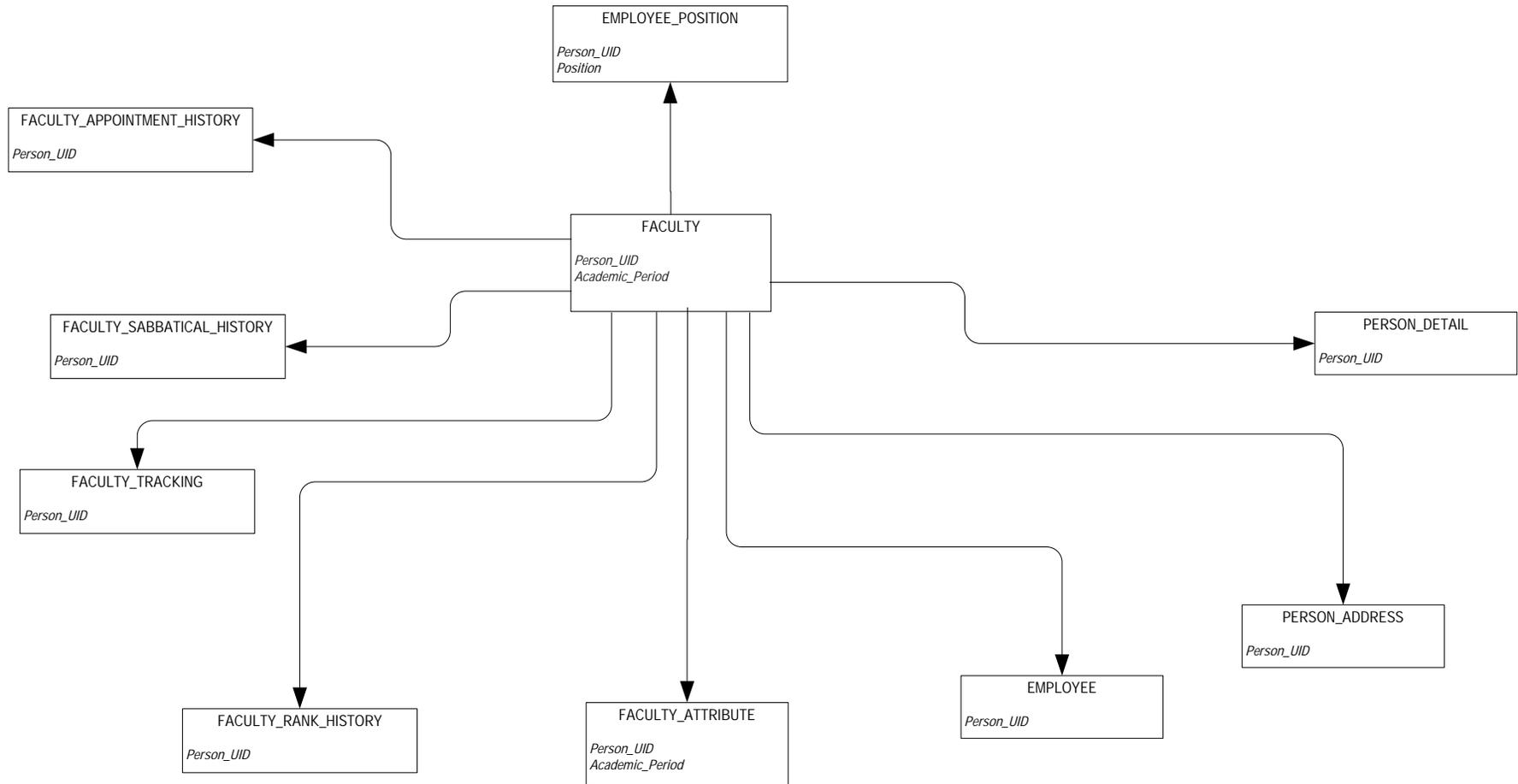


# Human Resource Application

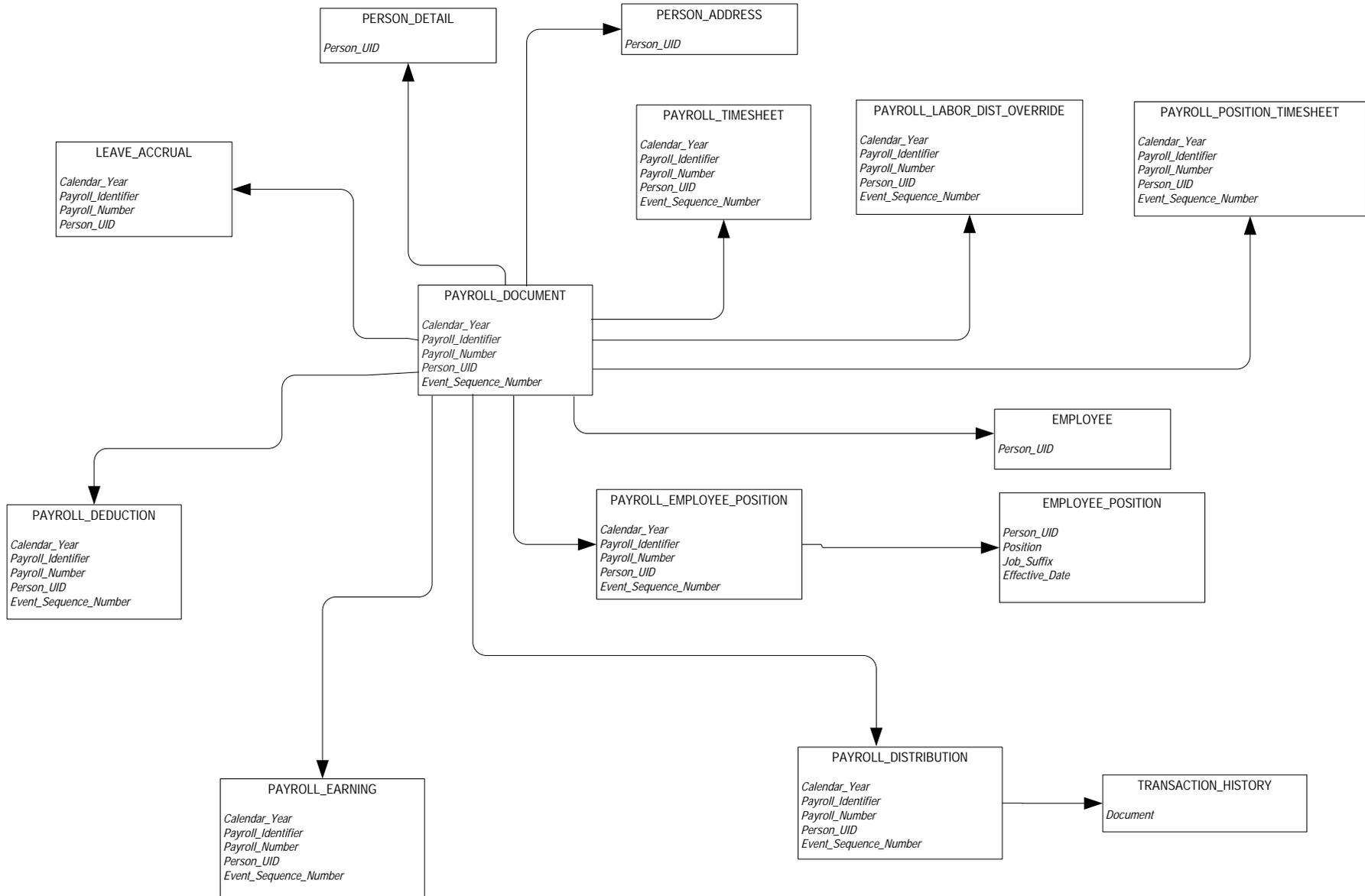


Green represents new query  
subject added or join changed

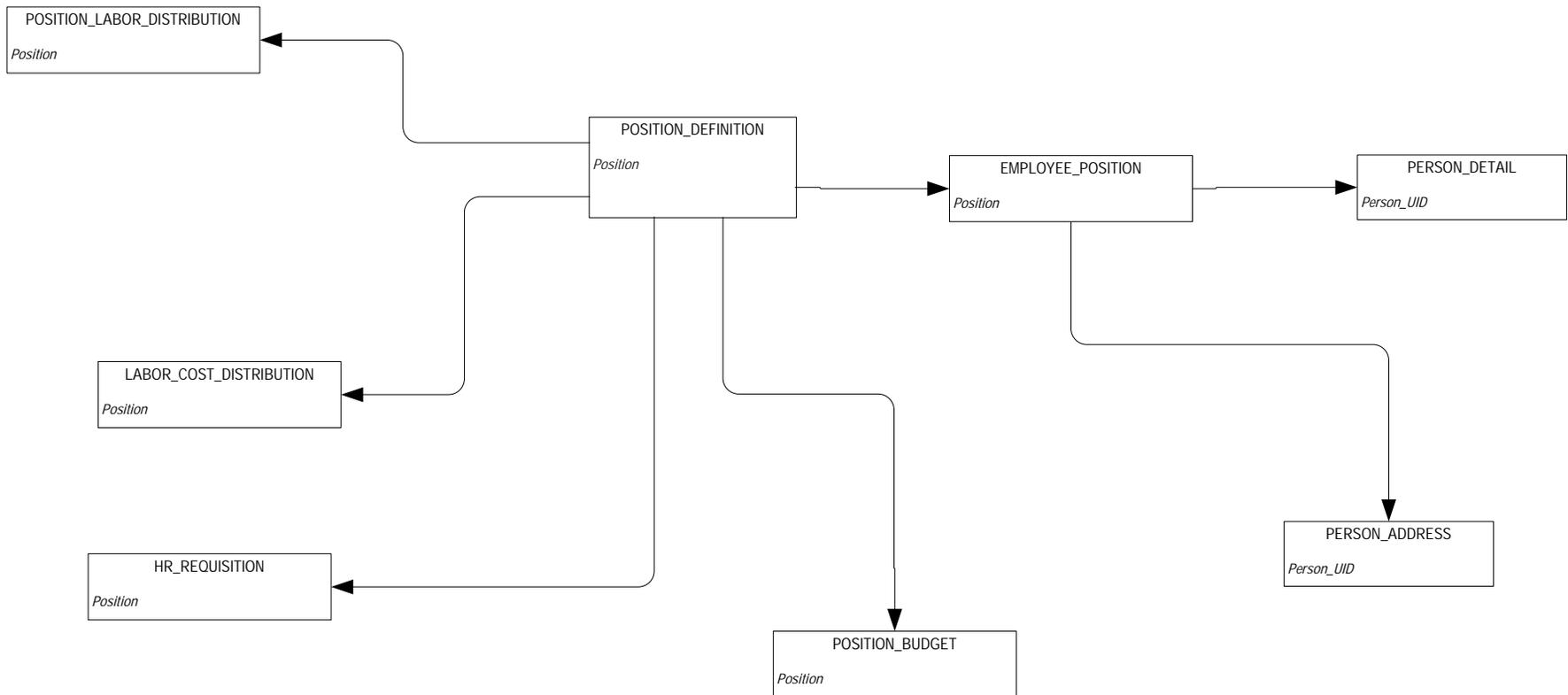
# Human Resource Faculty



# Payroll

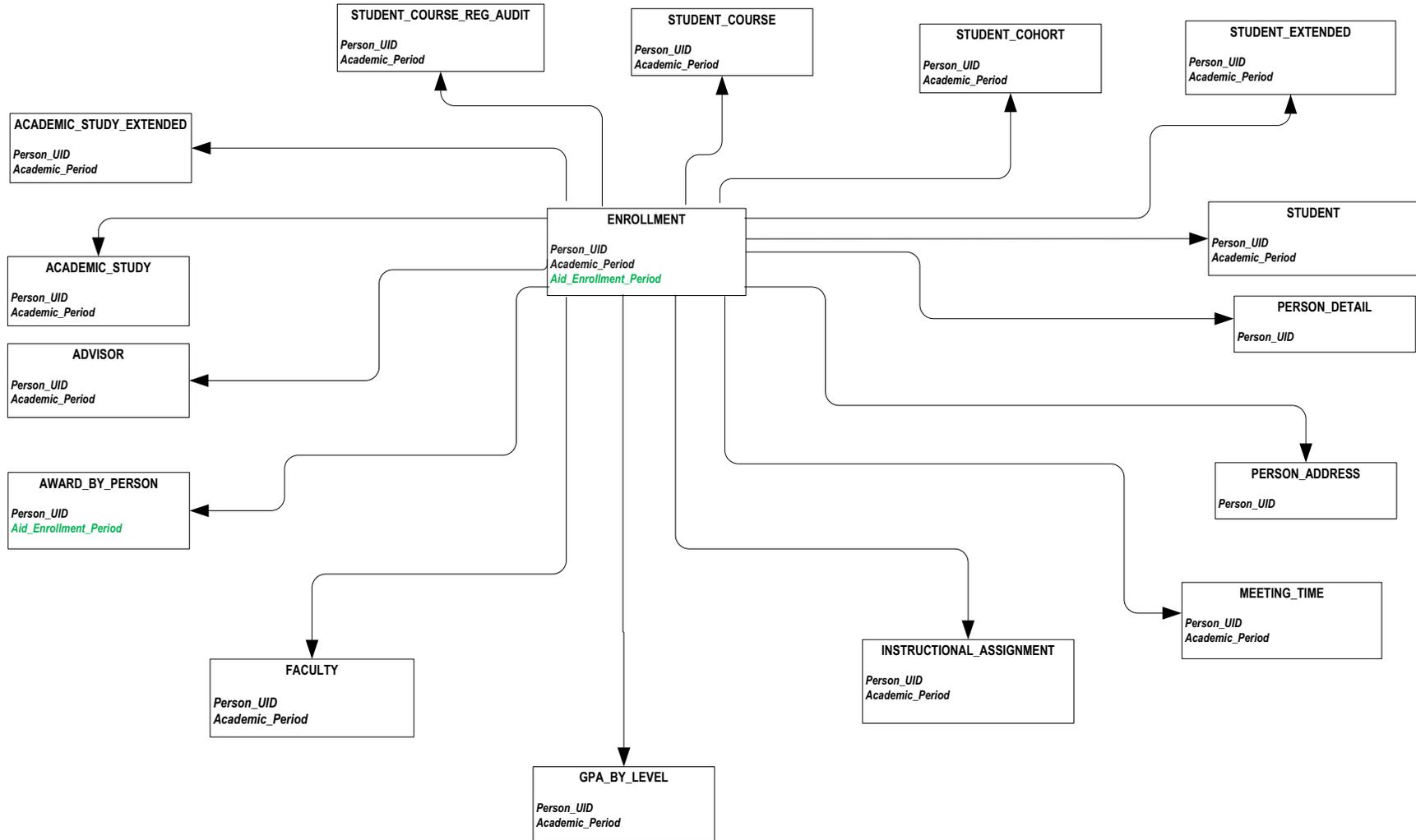


# Position

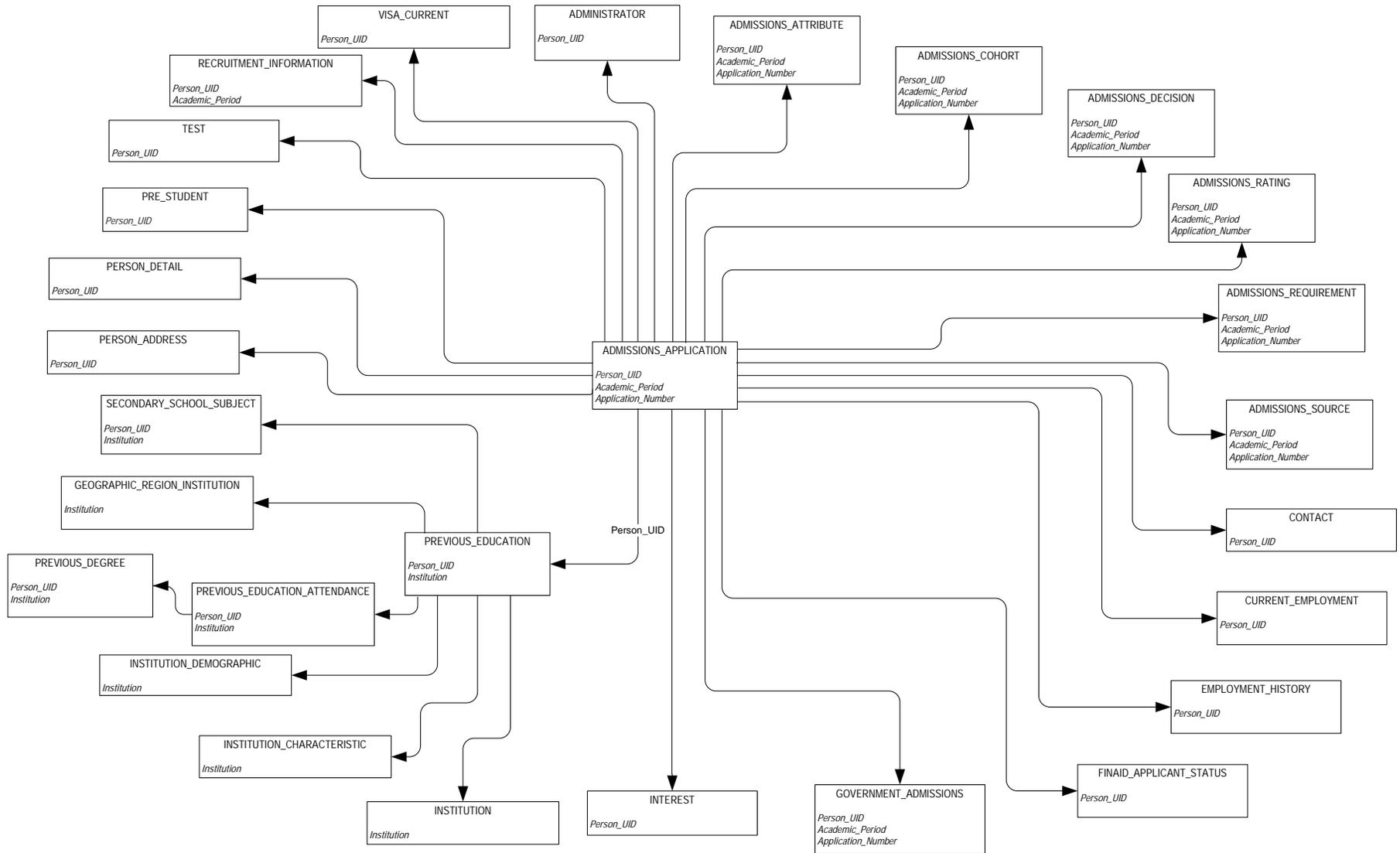


# Student

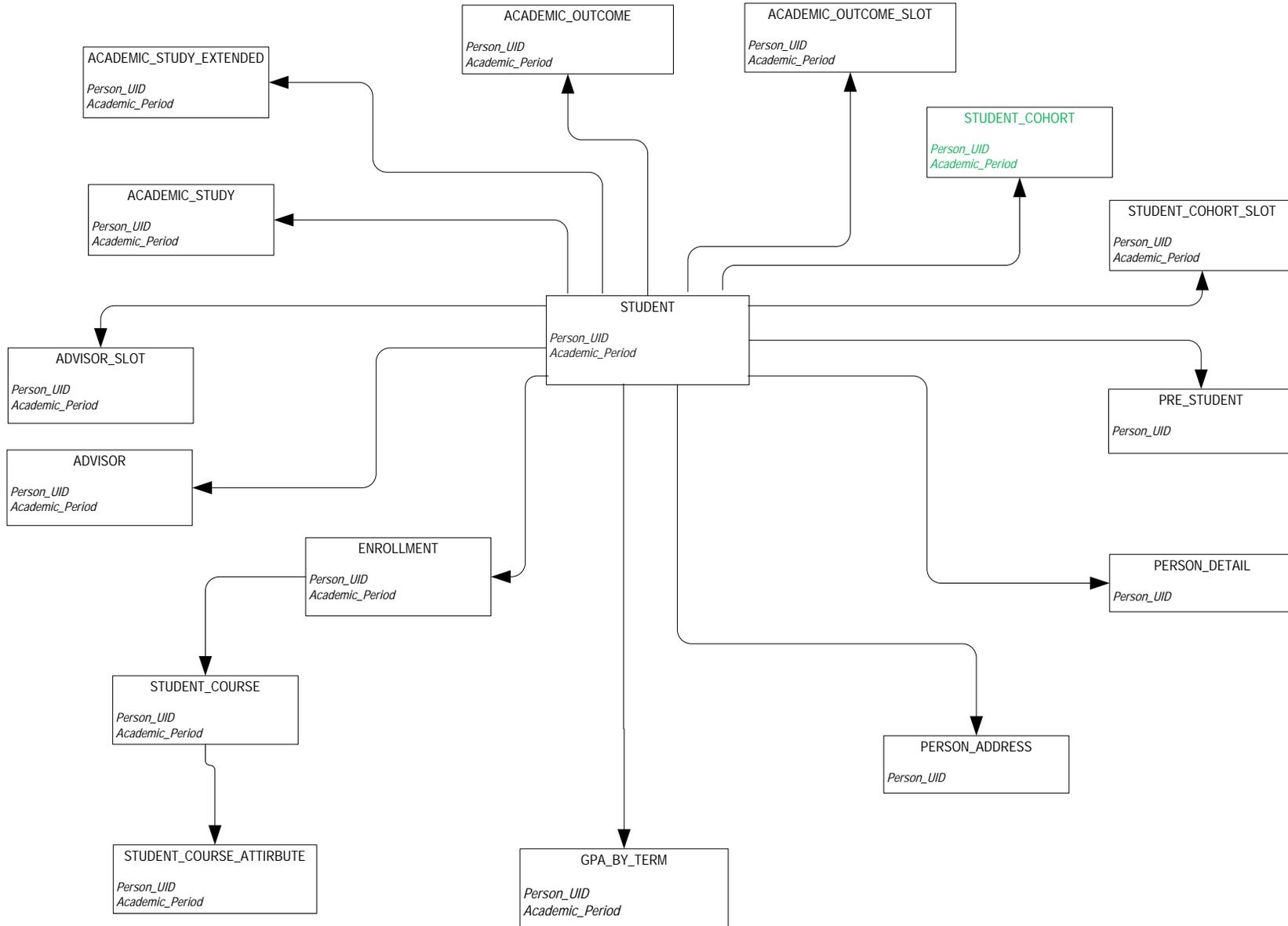
## Active Registration



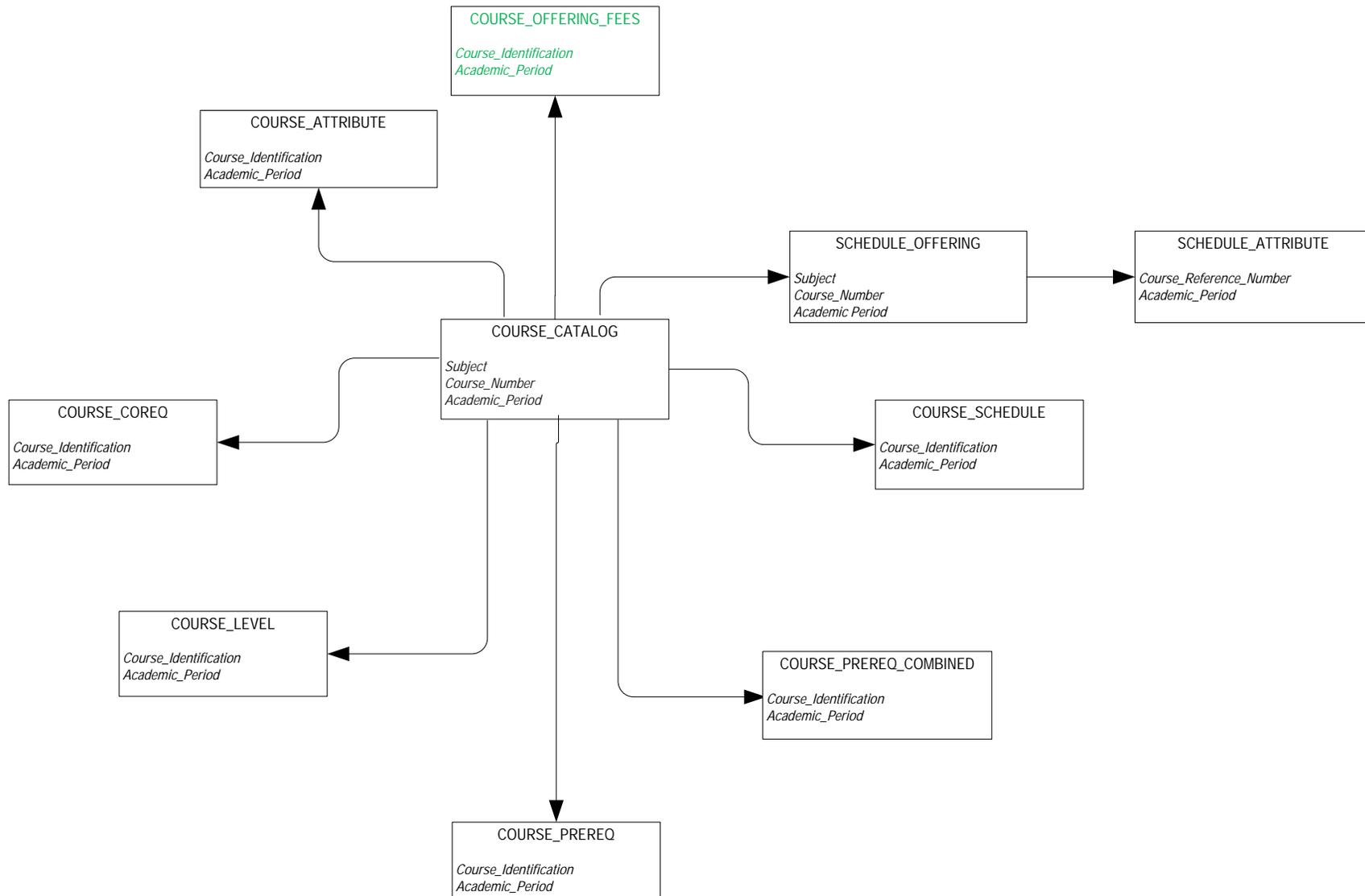
# Admissions Application



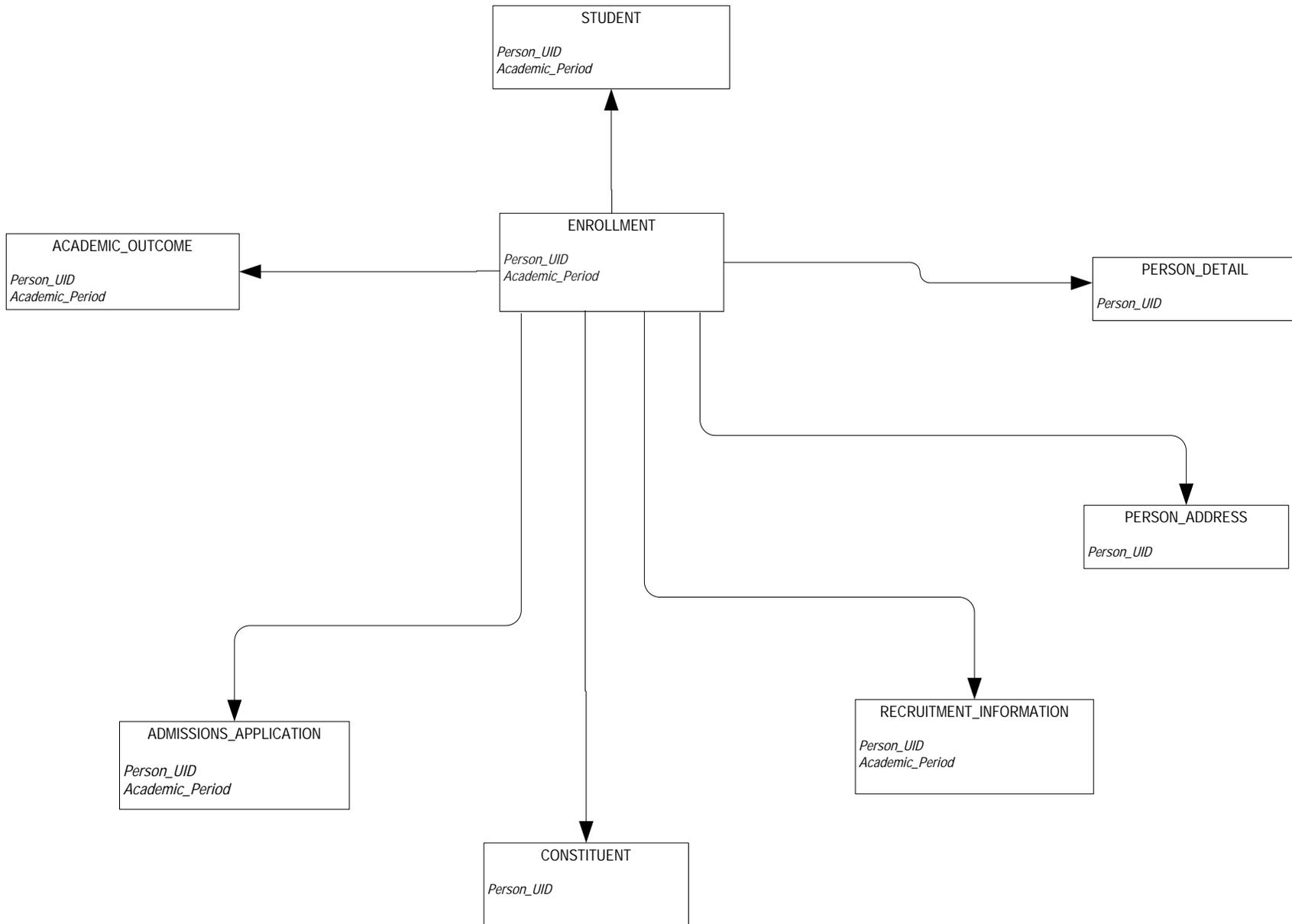
# Advisor Student List



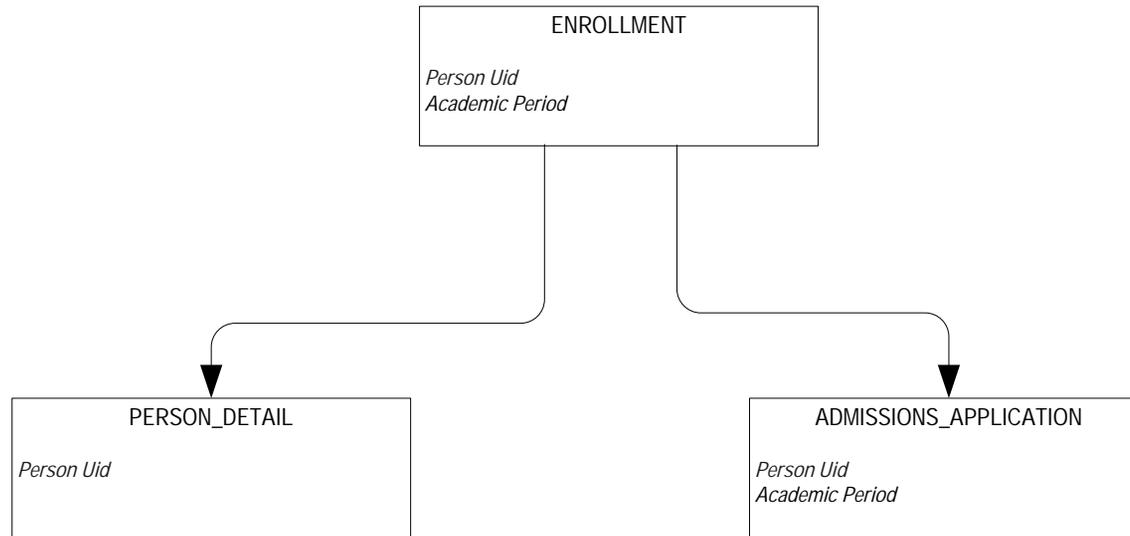
# Course Catalog



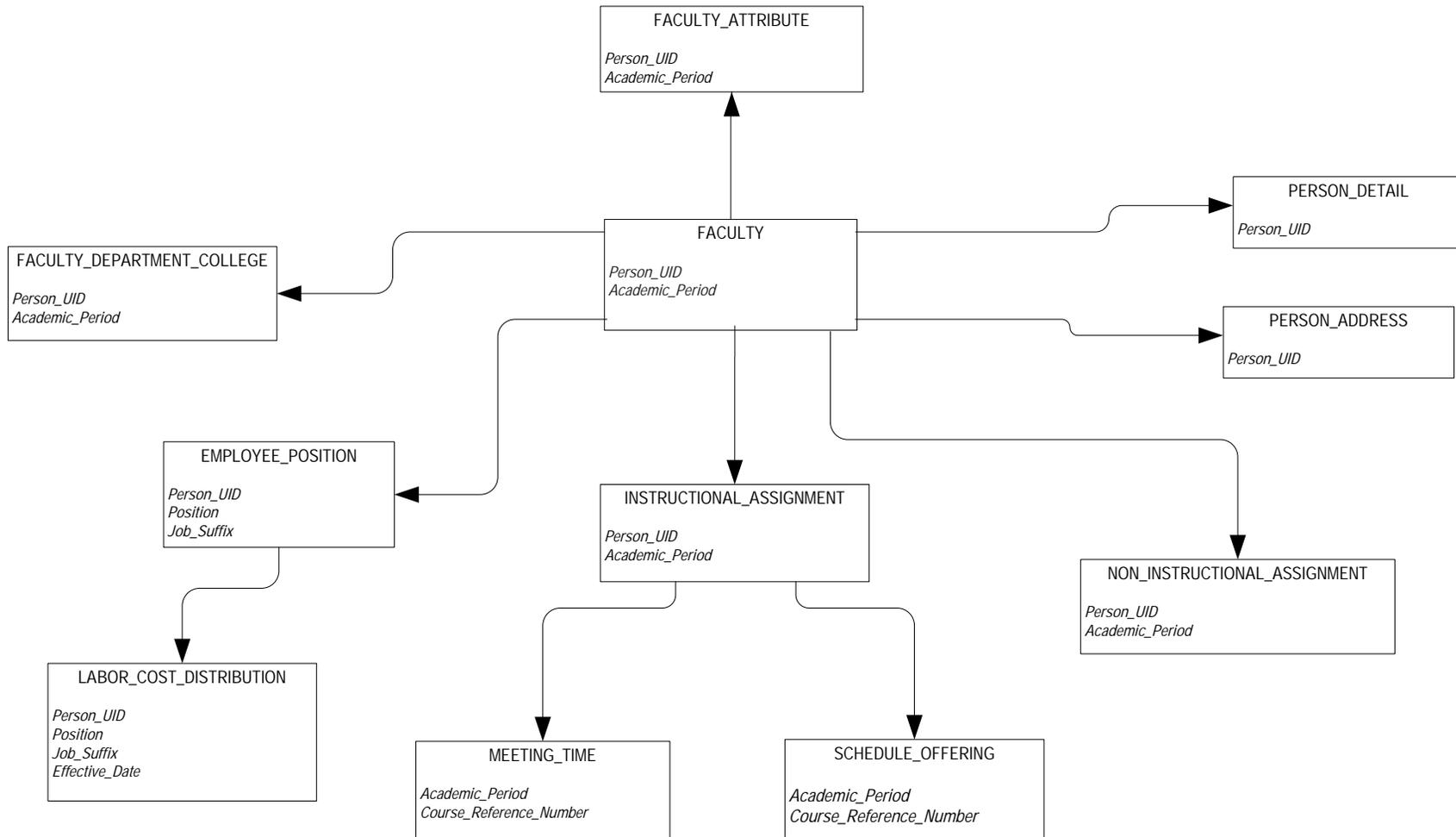
# Enrollment Management



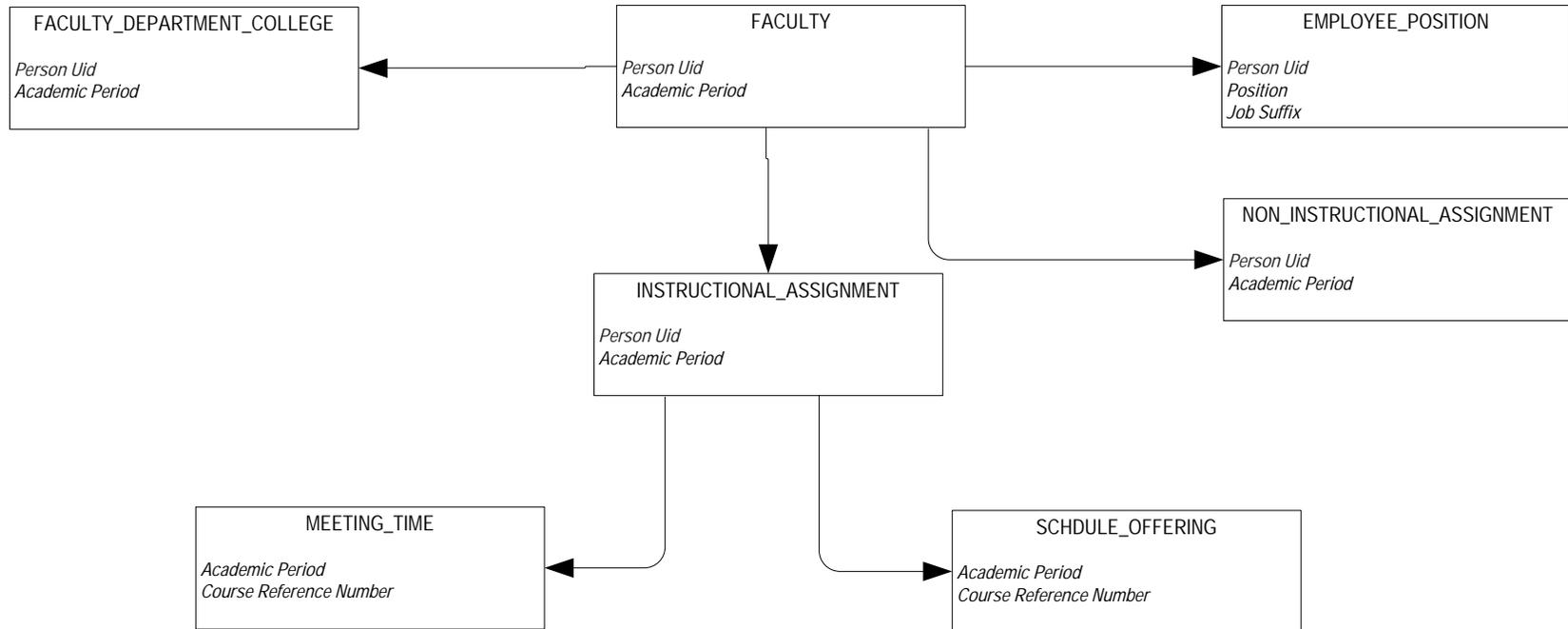
## Enrollment Management Subset



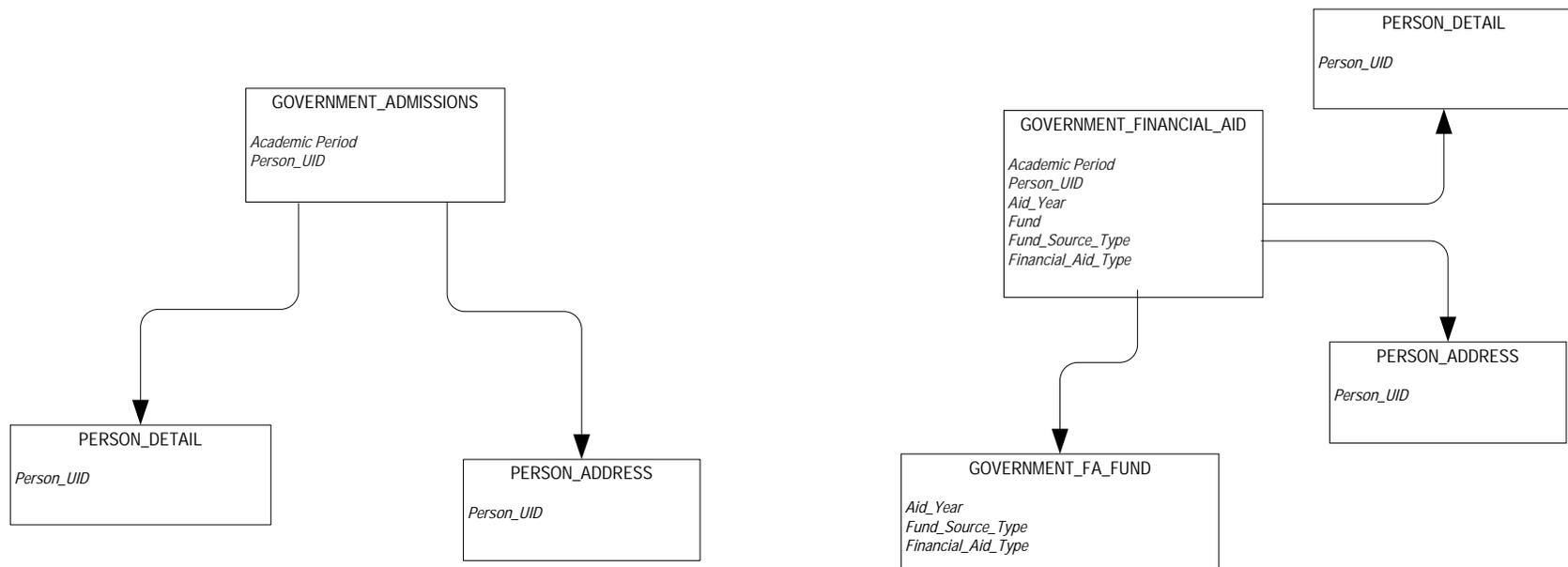
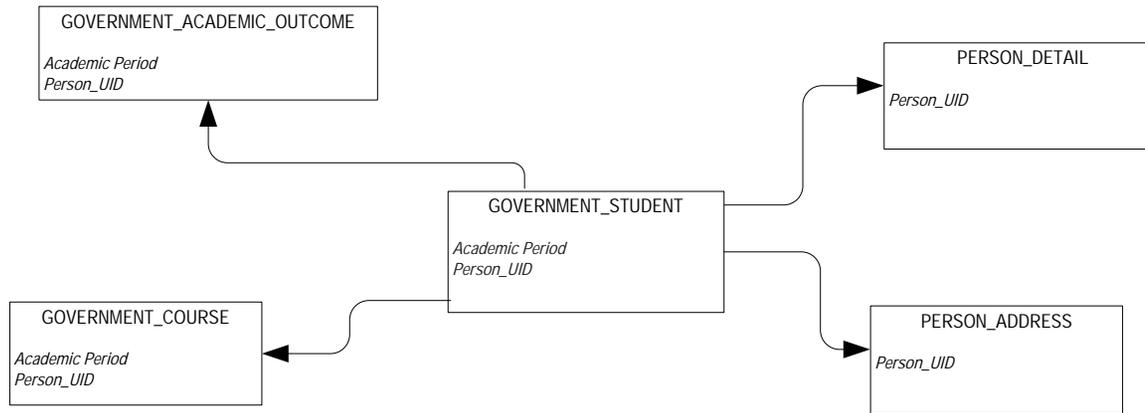
# Faculty Assignment



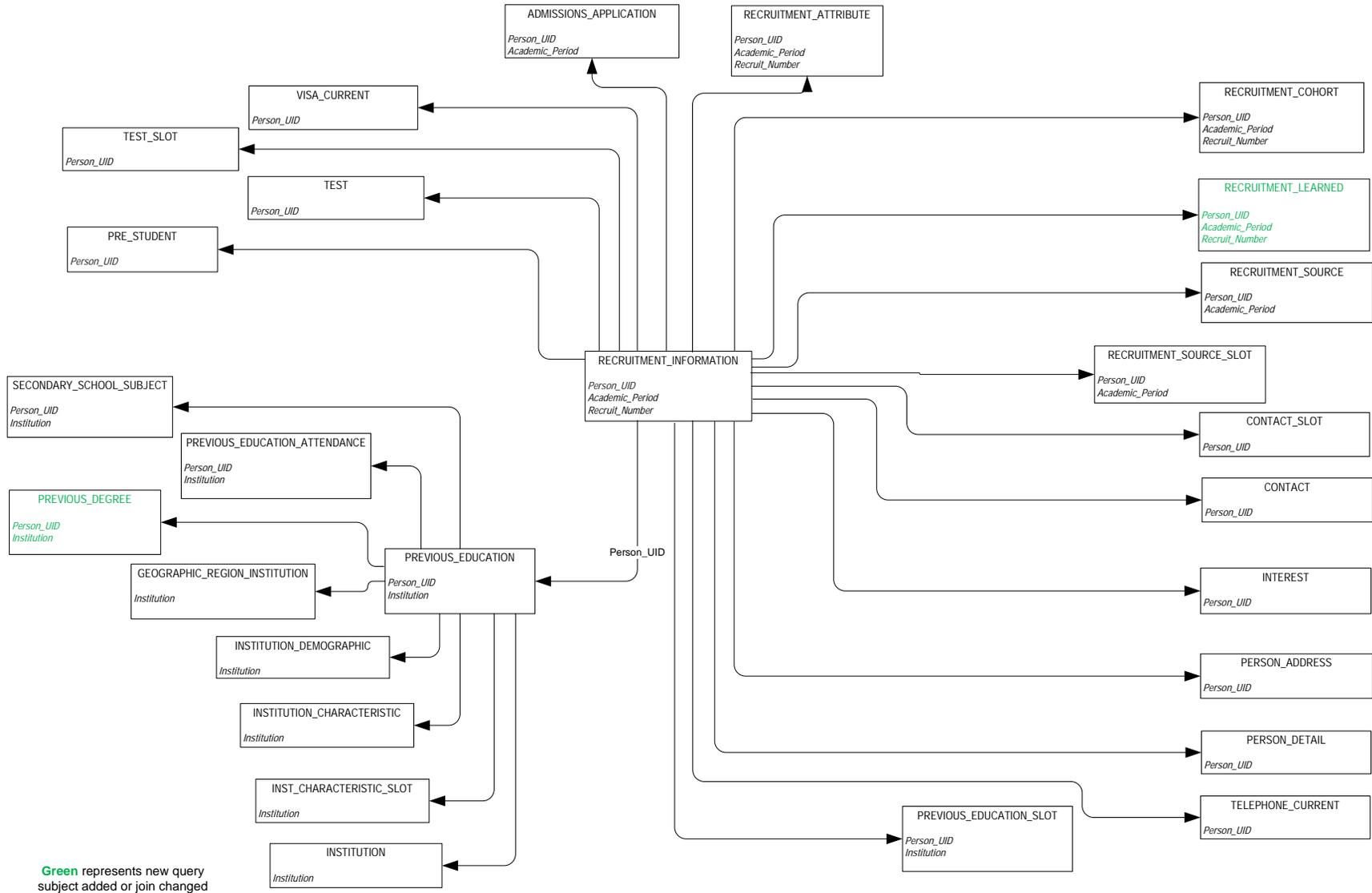
## Faculty Subset



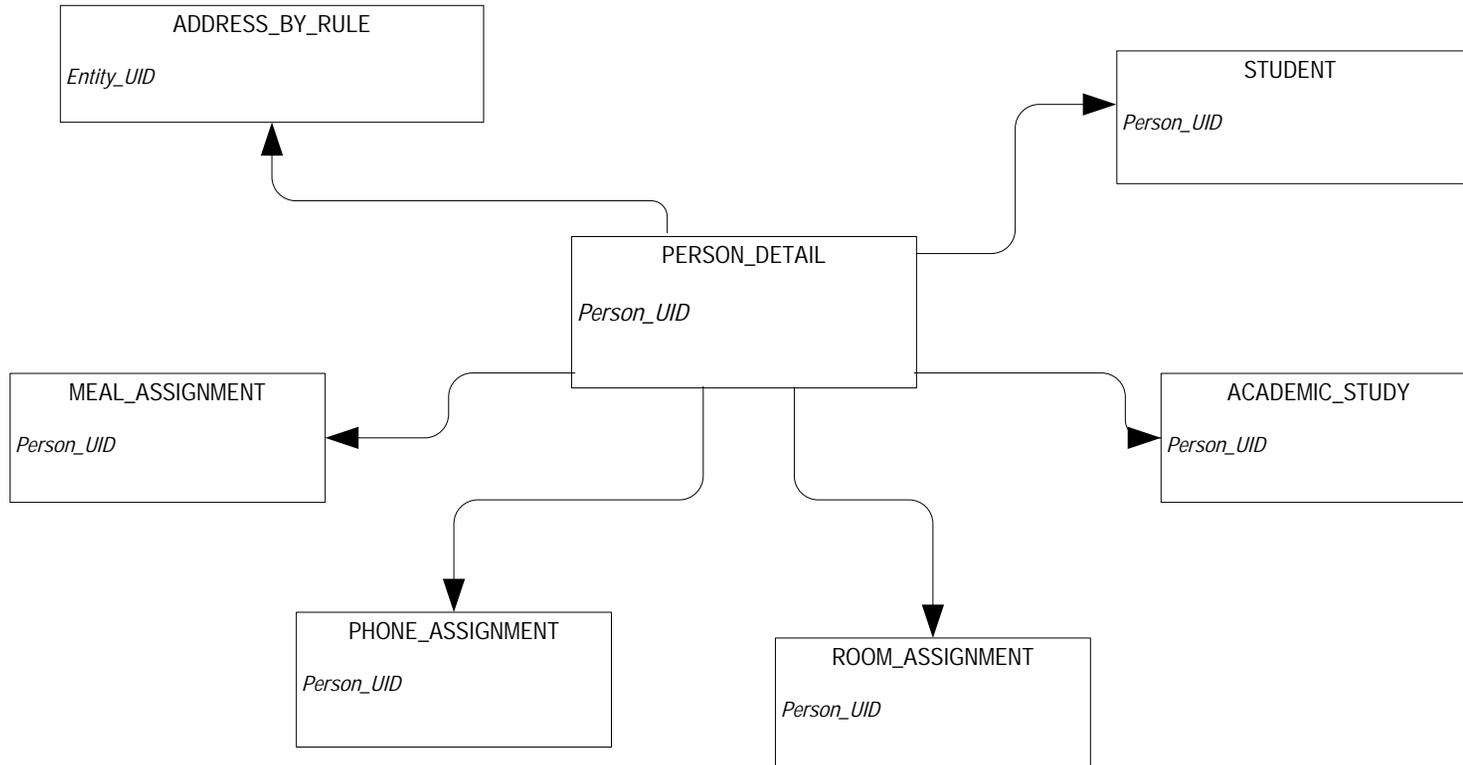
## Government Reporting



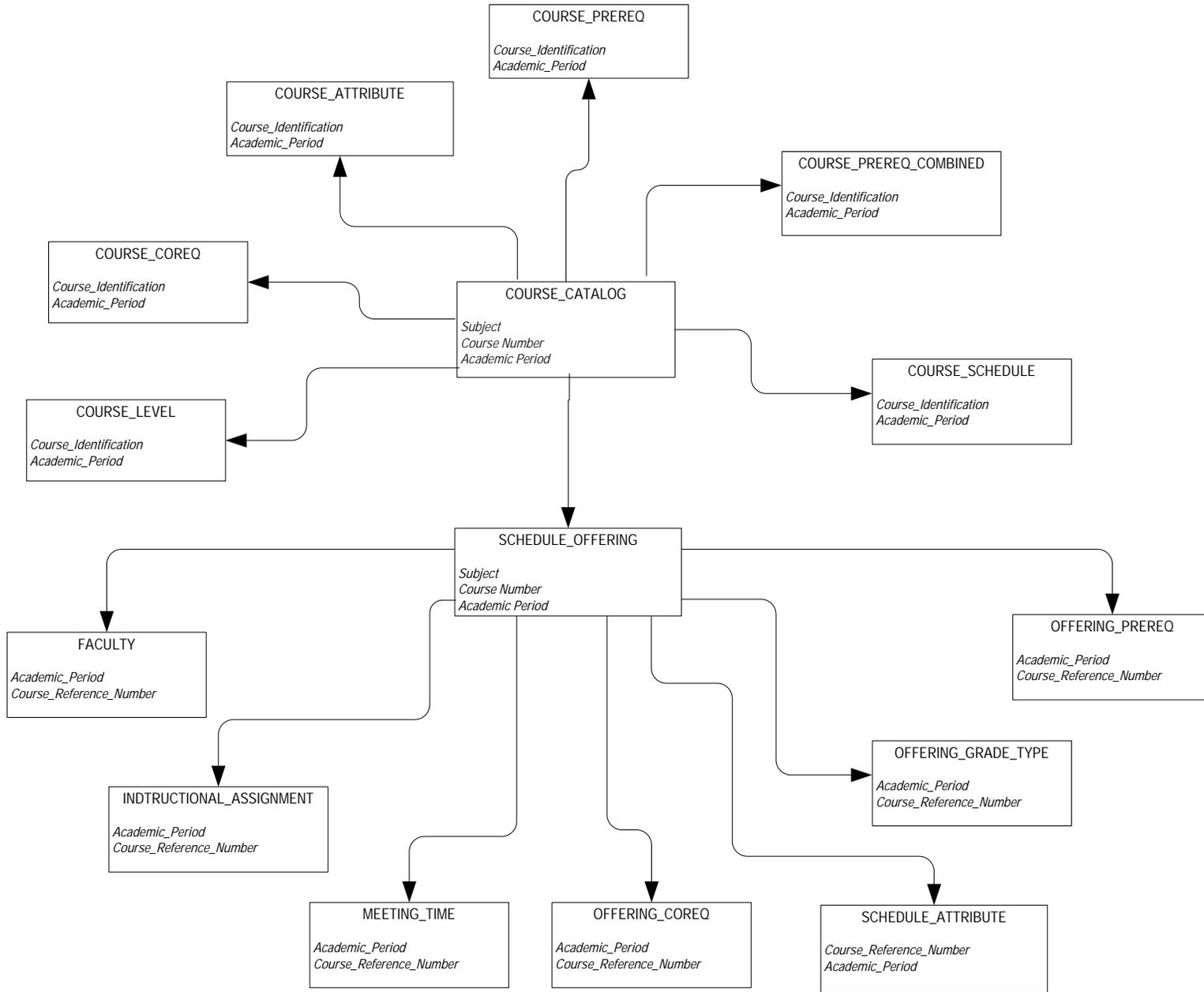
# Recruitment Information



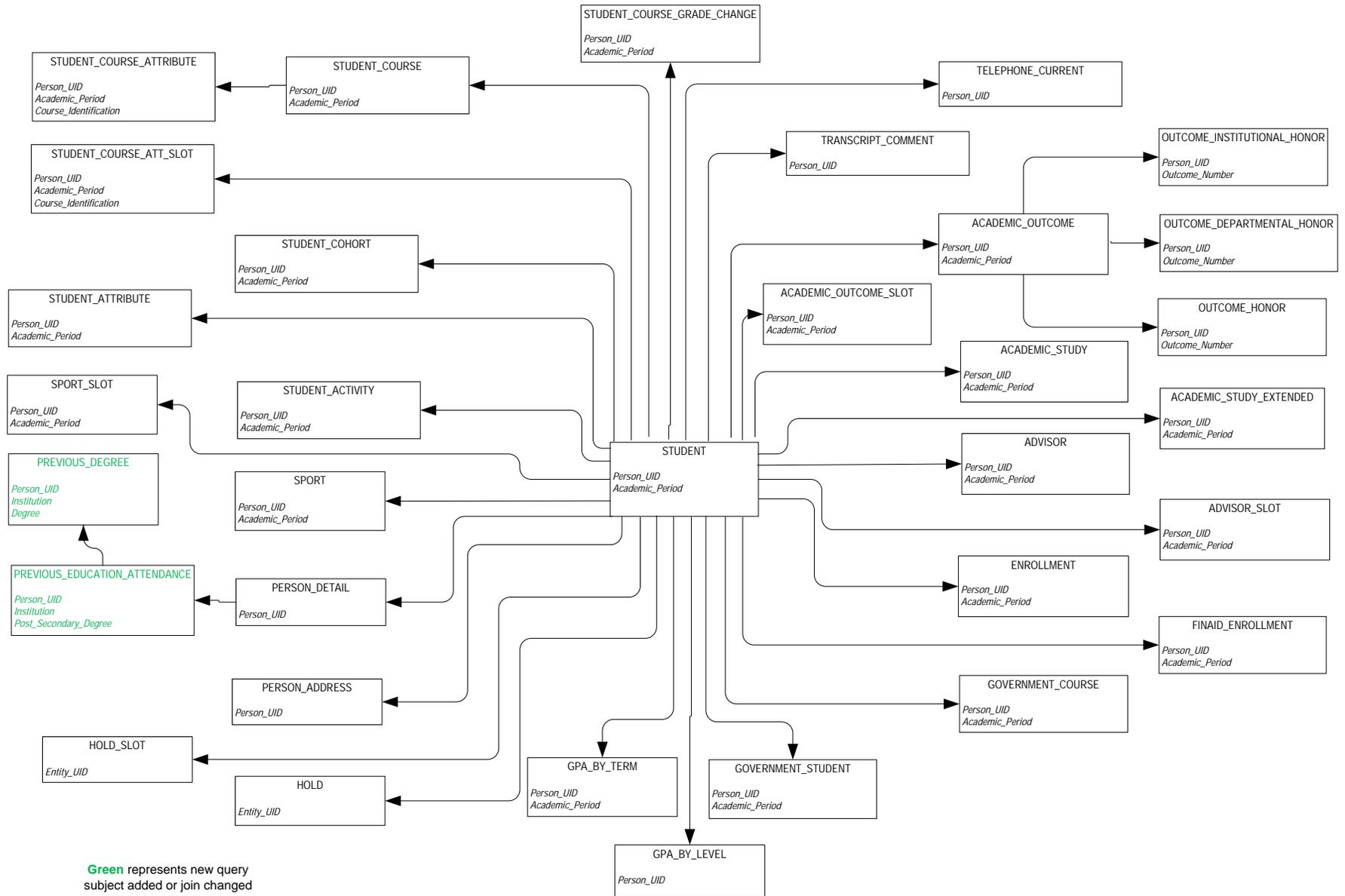
## Residential Life



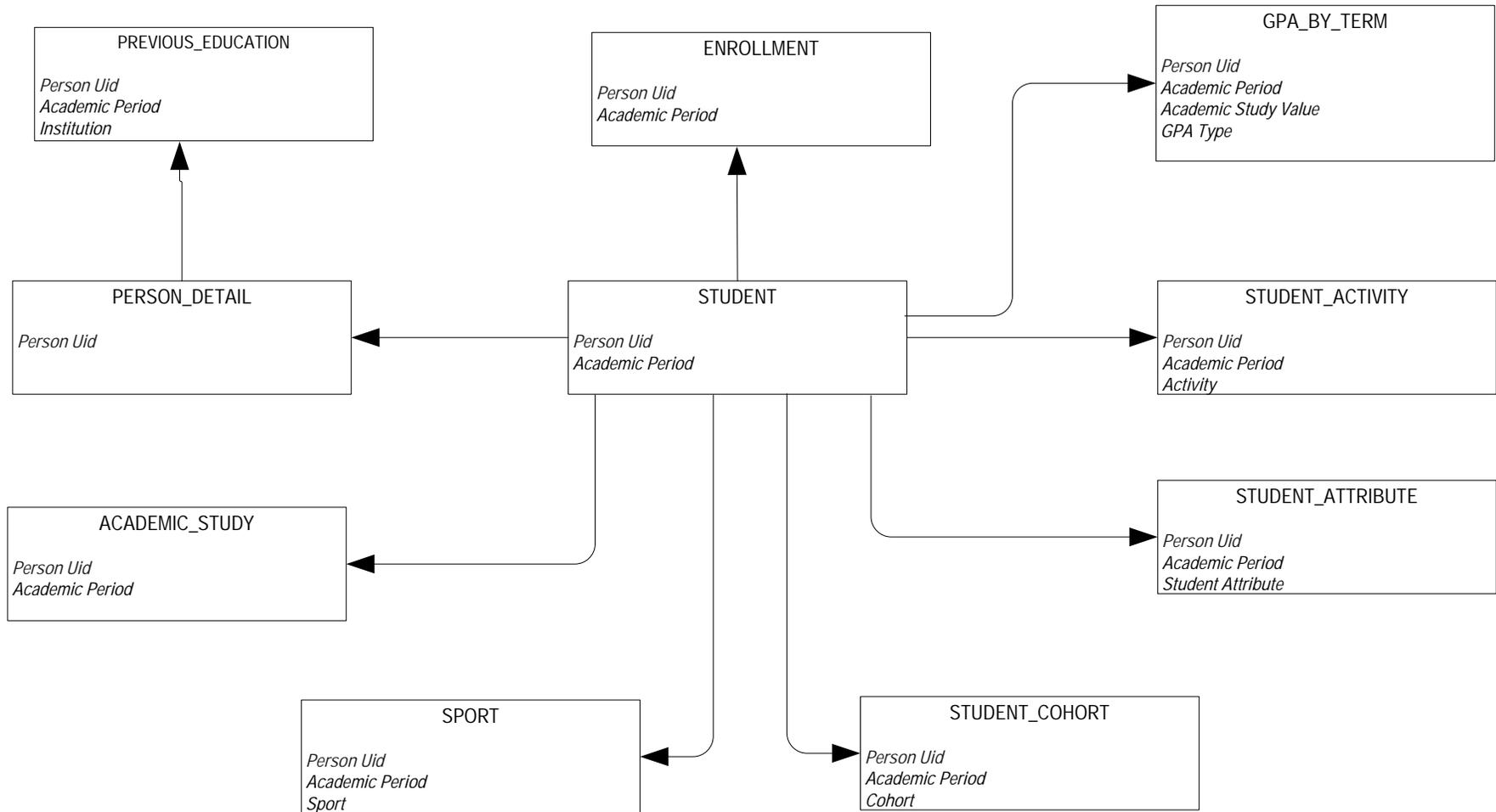
# Schedule Offering



# Student Detail

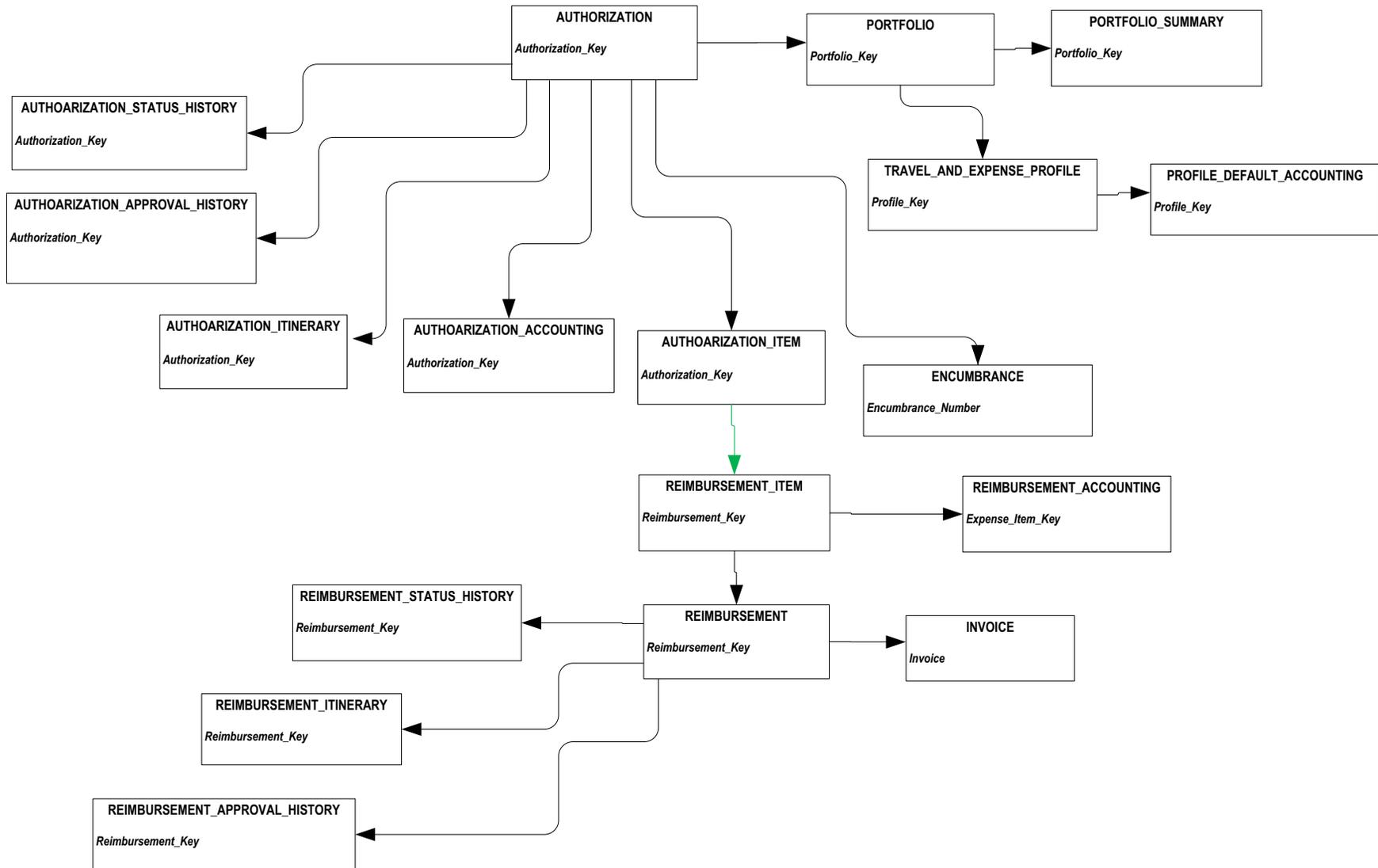


## Student Detail Subset

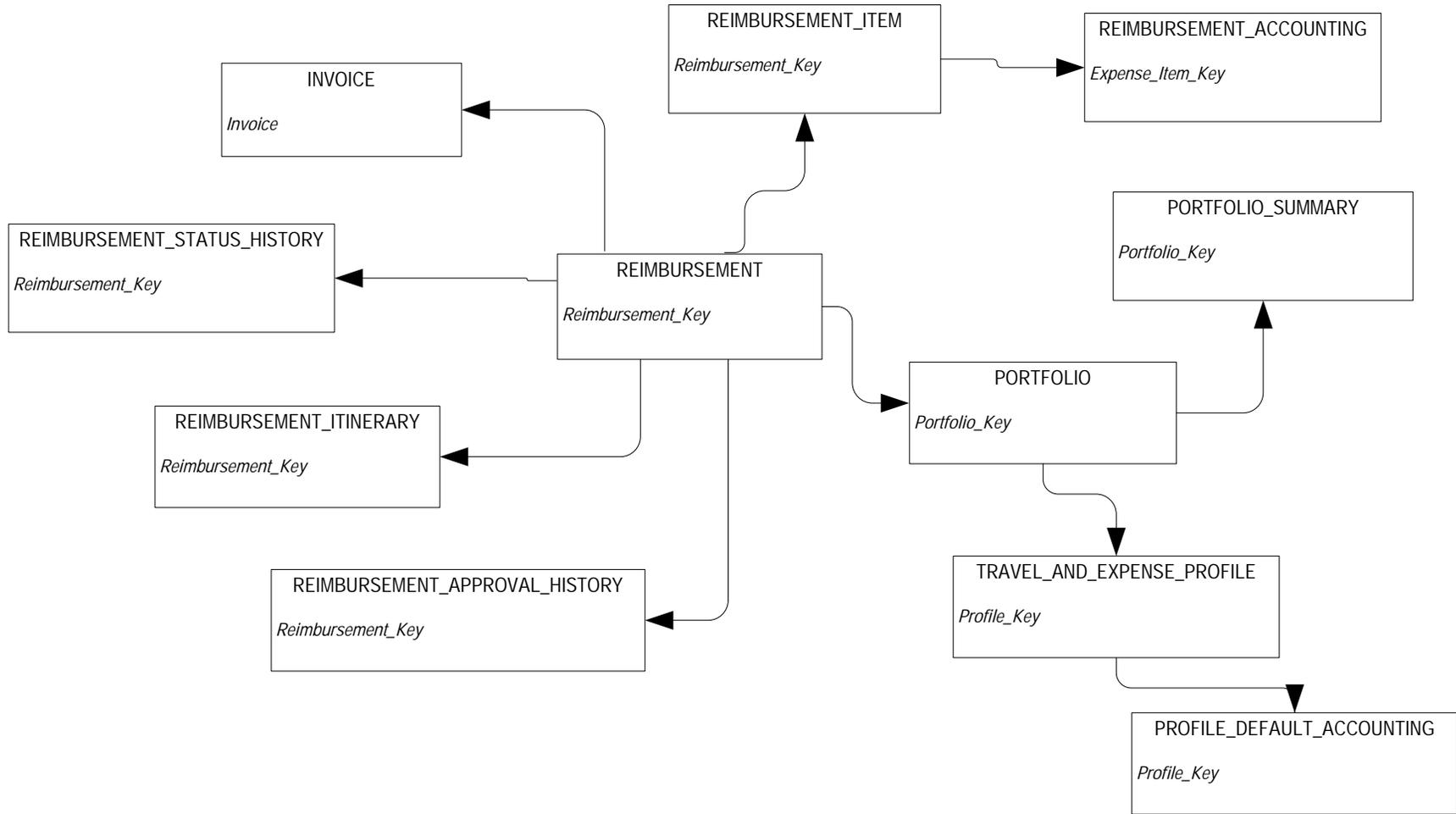


# Travel and Expense

## Authorization



## Reimbursement



# 4 Third Party Reporting Tools

---

A critical factor in determining the success of a reporting solution is the existence of a well defined and useful meta data layer. The meta data layer enables you to define relationships between objects in the database. It also enables additional filtering or formatting that can be useful to you when creating reports.

Cognos 8 Business Intelligence and Oracle Business Intelligence Discoverer meta data layers are delivered as part of the Banner Operational Data Store (Banner ODS). Relationships between the reporting views in the Banner ODS are included in these meta data layers for the supported reporting tools. The meta data layer provides the joins used by the database to connect the views or database tables so that you do not need to define those relationships when creating queries or reports using the reporting tools. You can use any reporting tool with the Banner Performance Reporting & Analytics products; however, you gain added value from using the Cognos and Discoverer meta data layers created and delivered with the products.

The Banner ODS defined reporting meta data contains reporting view and column definitions within the reporting views and manages columns that are number data type either to aggregate them or to treat them as identifiers. It can contain hierarchies for drilling into aggregate number columns at different levels, and lists of values (LOV's) for which you can create drop-down lists in prompts and filters for queries on the reporting views.

In Cognos Business Intelligence, the reporting meta data is defined using the Cognos Business Intelligence Framework Manager (FM) to create FM models.

In Oracle Business Intelligence Discoverer, the reporting meta data is defined using the Discoverer Administrator to create the End User Layer (EUL).

## Cognos Business Intelligence

---

The Cognos Business Intelligence (BI) meta data layer is delivered as part of the Banner ODS and the data warehouse. The Cognos BI meta data layer includes the following layers:

- Database view
- Business view
- Presentation view

## Framework Manager models

Databases are typically designed to store data captured through business processes. The stored data is not easily accessible for reporting and analysis to make enterprise decisions in business terms. Because of this, data requires metadata, the ‘data about data’, so that it can be effectively retrieved for analysis and reporting. The Cognos Framework Manager (FM) tool allows you to redefine the data in the database to answer business questions.

Cognos is designed to deliver centralized metadata via the FM model. The model provides a common definition of data in business terms that add value across the organization. The database is redefined so that you can publish metadata in a package and make it available through the Cognos Connection to the Cognos BI reporting tools Report Studio, Query Studio, and Analysis Studio to answer business questions.

The Framework Manager model presents the data using business terms and definitions. This enables you to use, build, and modify your own reports and enables consistent understanding and use of data and metrics across your institution. The logical relationships between data are defined within the model to enable complete data integration so that you spend less time gathering and organizing data.

For more information about data modeling, see the “Framework Manager User Guide” or “Metric Designer User Guide”.

## Metadata layers

Cognos Framework Manager provides the ability to layer metadata as a means to insulate end users from changes made to the underlying data sources and the defined data relationships within the database. When changes to an existing model are required, Framework Manager can identify the impact to existing reports. This enables your institution to manage model changes without having to rewrite reports.

The delivered FM models use two layers to manage the metadata content: the database view and the business view. A third layer, the presentation layer, is used to publish the data in logical groupings.

### Database view metadata layer

The database view metadata layer is the layer into which Framework Manager imports all database objects. There is no difference between the database views in this layer and the database itself. The database view contains all of the reporting views for a product including all List of Value Views coming from the ODSLOV schema.

## **Business view metadata layer**

The business view metadata layer organizes content around a specific business process or processes. The business view layer references objects from the database view and relationships among them are defined to support the associated business process.

The business view layer includes business concepts with relationships (joins) defined between related reporting views. These joins define the SQL generated behind the scenes by the various Cognos BI Reporting Tools.

The business view layer also includes a link back to the database view reporting views, which you can use to create custom SQL and queries outside of the business concepts. If you cannot create a query to answer a business question against a particular business concept, you can create reports and custom queries against the database views.

## **Presentation view metadata layer**

The presentation view metadata layer is the layer in which information is reorganized into useful logical groups of data that you can use together for reporting. The query subjects in the presentation layer include data elements and folders of data elements that present the data in an intuitive fashion so it is easy for you to locate desired data for any report.

The following standards were applied when creating the presentation layer:

- Related data or query items are grouped in the same query subject.
- Subsets of data that are typically used together are organized into folders.
- Commonly used filters are defined to enhance functionality. Examples of delivered filters include Student Level Undergraduate, Student Level Graduate, and Student Level Professional.
- Commonly used calculations have been added to make reporting easier.
- Additional range and aging concepts have been added that work in conjunction with parameter maps. Each has an accompanying 'order' concept to ensure they appear in proper order when you use them.

From the presentation layer you can publish a complete package of all the data in that presentation view and or a number of smaller packages of information that target specific types of analysis and users. These packages allow you to create and use dashboards, run reports, build ad hoc reports, and analyze trends without the need to sift through large amounts of unneeded information.

## Cognos and BPR Meta Data Integration

Banner ODS includes Cognos Framework Manager (FM) models and packages for the ODS Reporting Views organized in groupings called business concepts.

The layers of Cognos content relate to the underlying Banner ODS and Banner EDW data structures that include the Banner ODS reporting views and the Banner EDW fact and dimension tables. The bulk of the data dictionary that describes these data structures is defined as BPR Meta Data that is stored in the “IA-Admin Meta Data”, (IAMD) and is delivered with each BPR product.

The BPR Meta Data includes column level “business definitions” that describe the data and is stored for each target column along with its source system, table, and column. Another other key part of the BPR Meta Data is the mapping from one layer to the next. For a given Banner EDW column, which Banner ODS column it comes from, and in turn, which Banner column that ODS column is sourced from. (This mapping information is hereafter referred to as “lineage”.)

To provide a meaningful relationship between the BPR Meta Data information and the Cognos reporting tools, the BPR Meta Data is integrated with the Cognos tools. The Meta Data business definition and lineage information are delivered in the FM models and packages and displayed in the Cognos reporting tools (Query Studio, Report Studio, and Analysis Studio). Within each query item in the FM models, the Description field includes the business definition and lineage, while the ScreenTip field includes the EDW source column name.

### View BPR Meta Data in Cognos

To view the business definition and lineage for a query item, use the arrows at the bottom of the navigation window to open the information for the selected query item. See examples of BPR Meta Data displayed in Query Studio and Report Studio as illustrated in the following figures.

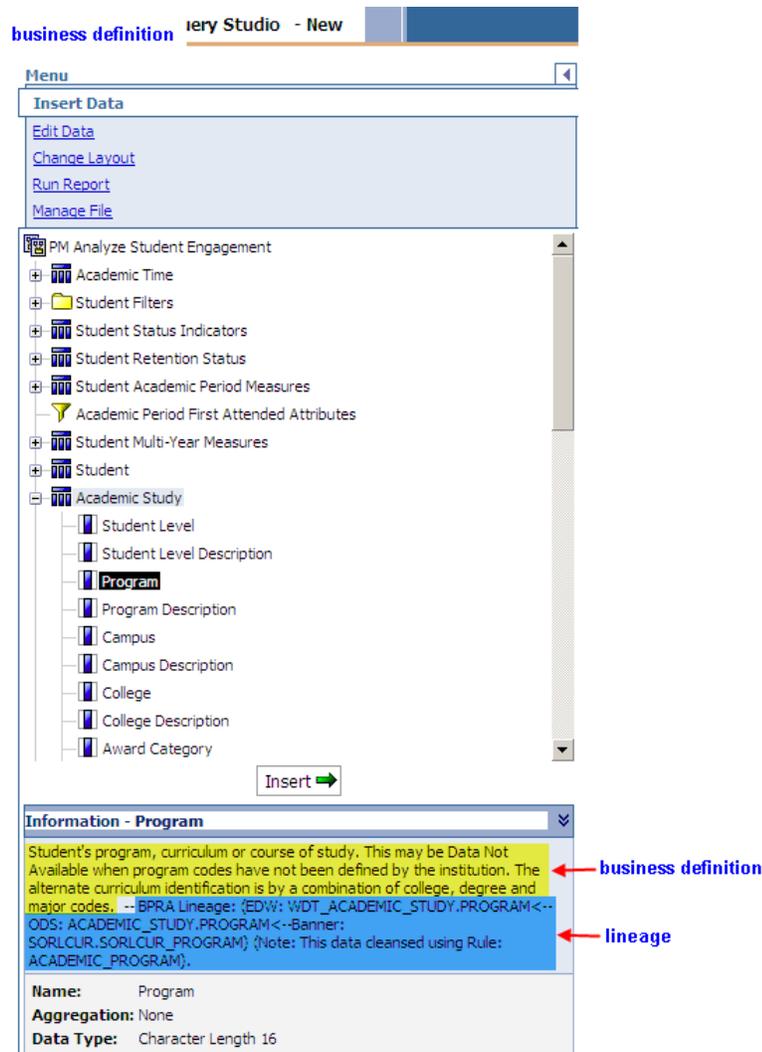


Figure 11: View BPR Meta Data in Query Studio

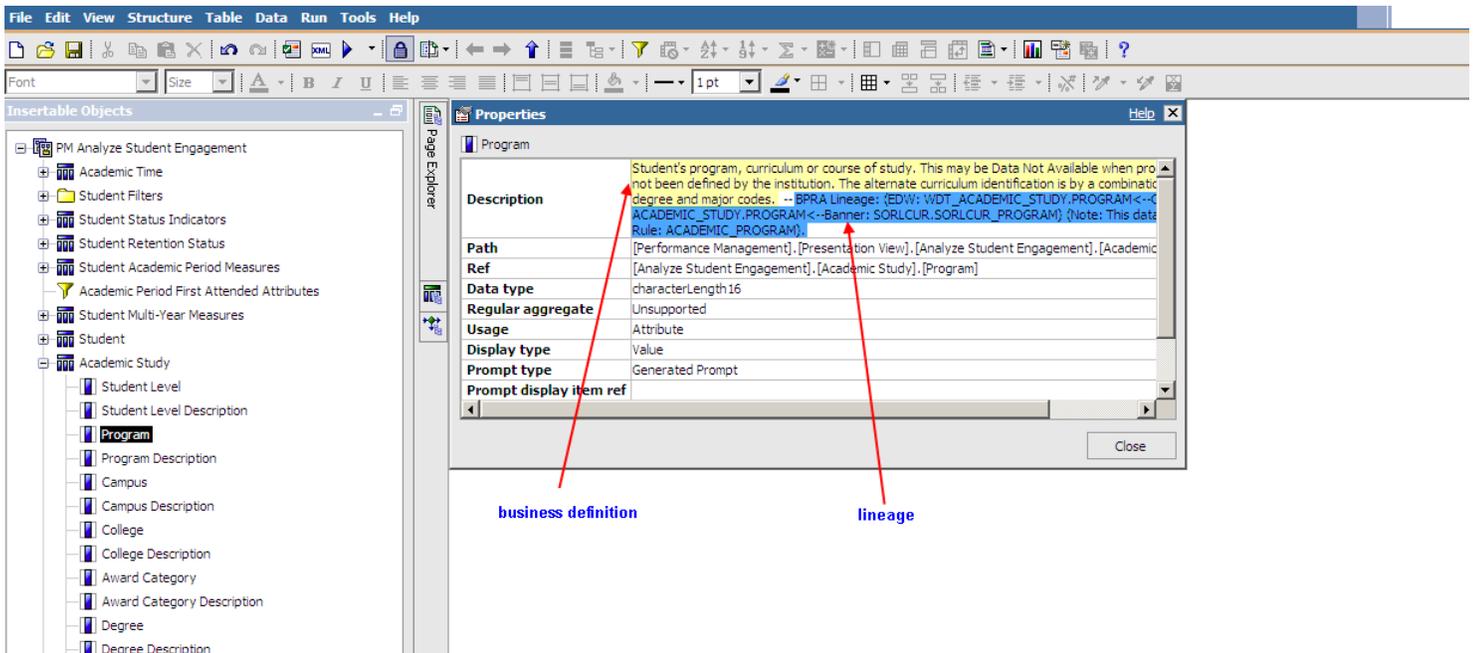


Figure 12: View BPR Meta Data in Report Studio

## View BPR Meta Data details

If you are using Cognos 8.4 or higher, you can also view more detailed information in both Query Studio and Report Studio. Right-click a query item and select the Lineage option to view the Database and Technical information displayed in Query Studio and Report Studio as illustrated in the following figures.

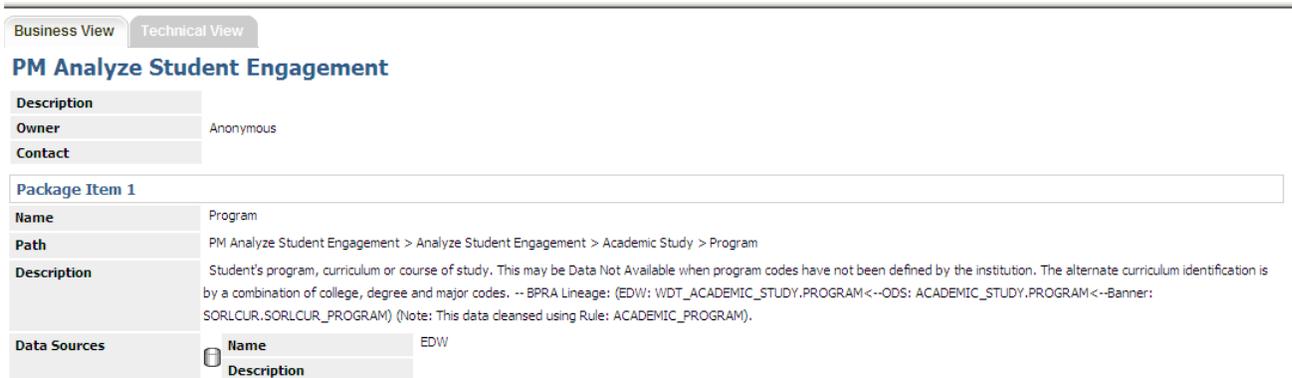


Figure 13: View Database and Technical BPR Meta Data in Query Studio

Business View Technical View

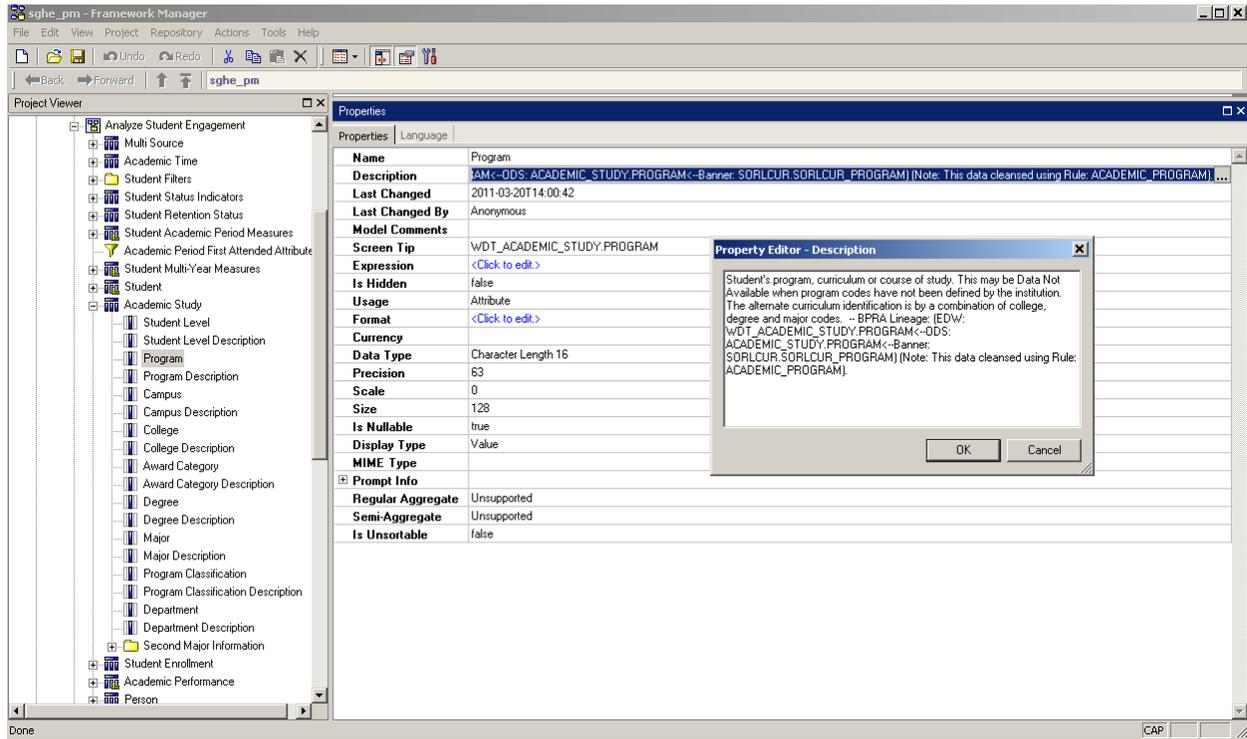
Package Data Sources

Properties for Program

Property	Value
ID	[Analyze Student Engagement].[Academic Study].[Program]
Name	Program
Type	Query Item
Description	Student's program, curriculum or course of study. This may be Data Not Available when program codes have not been defined by the institution. The alternate curriculum identification is by a combination of college, degree and major codes. -- BPR Lineage: (EDW: WDT_ACADEMIC_STUDY.PROGRAM<--ODS: ACADEMIC_STUDY.PROGRAM<--Banner: SORLCUR.SORLCUR_PROGRAM) (Note: This data cleansed using Rule: ACADEMIC_PROGRAM).
Screen Tip	WDT_ACADEMIC_STUDY.PROGRAM
Expression	[Analyze Student Engagement BV].[Academic Study].[Program]
Datatype	Character Length 16
Precision	63
Scale	0
Size	128
Regular Aggregate	Unsupported Value
Semi Aggregate	Unsupported Value

Figure 14: View Database and Technical BPR Meta Data in Report Studio

The BPRA Meta Data business definition and lineage are stored in the FM model query item Description as illustrated in the following picture.



**Figure 15: BPRA Meta Data Information in Framework Manager**

The Banner EDW source table and column are also stored in the FM model query item Screen Tip, which will be displayed when you move the cursor over a query item in either the Cognos Query Studio or Report Studio reporting tool.

## Packages

A package is a subset of data designed to support a specific set of reporting needs. Packages may contain content designed within Framework Manager. They are the means by which Query Studio, Report Studio, and Analysis Studio are able to access data using the Cognos BI reporting tools. They are essentially the data sources used for reporting and analysis.

Within the various Cognos studios you can report against only one package at a time. It is important to use the correct package for the intended business purpose. When creating a new report, you are prompted to select which package to use.

## Banner ODS packages

Each package defined as part of the Banner ODS includes attributes and measures related to one business area of reporting. Many of the packages also include pre-defined filters that you can use to narrow the information returned in a report.

### Note

Refer to the “Banner ODS Cognos Filters” document for a definition of each filter included in these packages. ■

The following Cognos packages are delivered with the Banner ODS.

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Account Index Audit		
Active Registration	<p>Enrollment, academic study, advisor, awards by person, faculty, GPA by level, instructional assignment, meeting time, person address and detail, current telephone and student related information: cohort, course, course attribute, course registration audit, and extended.</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Enrollment, Academic Study
Admission Application	<p>Admissions related information: applications, attributes, cohorts, decisions, ratings, requirements, sources; and contacts, employment, Financial Aid applicant status, geographic region, institution, interests, person detail recruitment, and tests.</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Operating Ledger
Advancement Prospect	<p>Prospects, advancement ratings, constituent contacts, designations, funding, organizations, person detail, prospect contacts, prospect moves, prospect proposals, and prospect staff contacts</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> and <a href="#">“Advancement Rating”</a> packages.</p>	Prospect Info, Constituent Contact, Constituent Staff Assignment, Person Address, Project Interest, Prospect Moves Management, Prospect Proposal Master, Prospect Staff Assignments, Prospect Staff Contact, Telephone Current

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Advancement Rating	<p>Advancement ratings, constituents, organization entities, person address, and person detail.</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	No filters
Advisor Student List	<p>Student, academic outcome, academic study, advisors, advisor person address, enrollment, GPA by term, person address and detail, pre-student, student cohort, student course and course attribute, current telephone, transcript comments</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Academic Study
Annual Giving	<p>Annual giving, campaign giving history, designation financial aid fund, designation giving history, constituent, donor category, organization entity and address, organizational constituent, person address and detail, and current telephone</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Person Address
Annual Giving Comparison	<p>A subset of the reporting views that support giving information for the most recent gift, highest gift amount, and most recent pledge information for each prospect; and total giving information by Campaign and by Designation</p>	Time, Donor Category
Authorization	<p>Authorizations, authorization-related information: accounting, approval history, item, itinerary, status history, encumbrance, portfolio, portfolio summary, profile default accounting, reimbursement information: accounting, approval history, item, itinerary, and status history, and travel and expense profile</p>	Authorization, Authorization Item, Authorization Itinerary, Portfolio
Budget Availability Comparison	<p>A subset of the columns from the Budget_Availability_Ledger reporting view to simplify reporting on available budget, only includes hierarchy information for organization</p>	Time, Budget

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Budget Availability Ledger	Budget availability ledger, account and account type attributes, fund and fund type attributes, organization attributes, and program attributes	Time, Budget Availability Ledger
Budget Detail	Budget detail, account and account type attributes, budget request text, fund and fund type attributes, location attributes, organization attributes, and program attributes	Time, Budget Detail
Campaign Giving History	Campaign giving history, annual giving, constituent, constituent entity, organization entity and address, organizational constituent, person address and detail, and current telephone  This package also includes the <a href="#">“Person Supplemental”</a> package.	No filters
Campaign Management	Campaign, gift, pledge, and campaign related information: comment, designation, expense, fundraiser, mail, and solicitation	No filters
Cashier Session Analysis	A subset of the columns from the Receivable_Account_Detail reporting view emphasizing information supporting analysis for a cashier session	Time
Constituent	Constituent, annual giving, campaign giving history, constituent related information: contact, entity, plan, spouse, and staff assignment, current employment, degree, designation giving history, donor category, employment history, exclusion, gift, gift society, gift transaction, interest, mail, membership, person detail, pledge, pledge transaction, previous degree, prospect information, relationship, solicitation, special activity, special purpose group  This package also includes the <a href="#">“Person Supplemental”</a> package.	Constituent, Annual Giving, Campaign Giving History, Constituent Spouse, Constituent Staff, Current Employment, Designation Giving History, Donor Category, Employment History, Exclusion, Gift, Gift Transaction, Person Address, Pledge, Pledge Transaction

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Constituent Entity	Constituent entity, advancement rating annual giving, campaign giving history, constituent confidential comment, cross-reference, designation giving history, donor category, exclusion, gift, gift society, mail, pledge, special activity, and special purpose group	Advancement Rating, Annual Giving, Campaign Giving History, Designation Giving History, Donor Category, Exclusion, Gift, Gift Society, Pledge, Pledge Transaction
Course Catalog	Course catalog, course related information: attribute, corequisite, prerequisite, level, schedule, attribute, and offering	Time
Designation	Designation, gift, pledge, and the following designation related information: accounting, adjustment, attribute, class year, Financial Aid fund, comment, ID	No filters
Designation Giving History	Designation giving history, annual giving, constituent, constituent entity, organization entity, organizational constituent, and person detail  This package also includes the <a href="#">“Person Supplemental”</a> package.	No filters
Employee	Employee, bargaining unit, beneficiary, benefit deduction, certification, combined academic outcome, employee earning current and fiscal year, employee position, employment history, instructional assignment, labor cost distribution, leave balance, monthly deduction, non-instructional assignment, organization hierarchy, past employment, person detail, previous degree, previous education, review, skill, supervisor, tax, and yearly deduction  This package also includes the <a href="#">“Person Supplemental”</a> package.	Employee, Combined Academic Outcome, Employee Earning Fiscal Year, Employee Position, Employment History, Instructional Assignment, Labor Cost Distribution, Non-instructional Assignment, Organization Hierarchy, Person Address
Employee and Position	A subset of columns from the Employee, Employee_Position and Employee_Earning reporting views to simplify reporting on employee position information	Time, Position, Employee

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Encumbrance	Encumbrance accounting, account attributes, account hierarchy, encumbrance, encumbrance ledger, encumbrance text, fund attributes and hierarchy, location attributes and hierarchy, organization attributes and hierarchy, program attributes and hierarchy, purchase order, transaction history, and vendor	Time, Encumbrance, Encumbrance Accounting, Encumbrance Ledger, Purchase Order
Endowment Distribution	Endowment distribution, account type attributes, endowment attributes, fund attributes and hierarchy, fund text, and transaction history	Endowment Distribution, Transaction History
Endowment Unit	Endowment unit, account attributes, account hierarchy, account type attributes, endowment attributes, endowment summarized units, fund attributes and hierarchy, fund text, organization attributes and hierarchy, program attributes and hierarchy, and summarized units endowment attributes	Endowment Unit
Enrollment Management	Enrollment, academic outcome, admissions application, constituent, person address and detail, recruitment information, student, student extended, and current telephone  This package also includes the <a href="#">“Person Supplemental”</a> package.	Time, Enrollment, Academic Outcome
Enrollment Management Subset	A subset of the reporting views and columns that encompass the existing Enrollment Management model, drives off of the Enrollment reporting view and includes columns from Admissions Application.	Time, Student
Event	Event, event invitee, organization entity, person detail, and organization entity address  This package also includes the <a href="#">“Person Supplemental”</a> package.	No filters

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Faculty Assignment	<p>Faculty, employee position, faculty attribute, faculty department college, instructional assignment, labor cost distribution, meeting time, non-instructional assignment, person address and detail, schedule offering, and current telephone</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Faculty
Faculty Subset	<p>A subset of the reporting views and columns that encompass the Faculty model, drives off of the Faculty reporting view and includes joins to faculty related data in the Faculty_Department_College, Employee_Position and Instructional_Assignment reporting view</p>	Time
Financial Aid Application	<p>Financial Aid applicant status, admissions application, applicant need, applicant resource, award by person, financial aid message, loan application, loan disbursement, need analysis, person detail, previous degree, previous education attendance, satisfactory academic progress, user defined fields, and the following financial aid related information: budget component, comments, holds, tracking requirements</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Financial Aid Applicant Status
Financial Aid Award and Disbursement	<p>Awards by person, awards by aid year, awards by fund, award disbursement, academic study, designation Financial Aid fund, enrollment, general ledger, GPA by term, person detail, receivable account detail, student, transaction history, and the following financial aid related information: applicant status, enrollment, fund, hold, and government</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Award by Person, Award Disbursement
Financial Aid Fund	<p>Awards by fund, awards by aid year, Financial Aid fund, and government Financial Aid fund</p>	Time, Financial Aid Fund

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Fixed Asset	The following fixed asset related information: item, accounting history, accounting source, adjustment, attributes, depreciated item, funding source, and text; grant view, invoice item, location attributes and hierarchy, organization attributes and hierarchy, purchase order item, transaction history, and vendor	Fixed Asset Item, Fixed Asset Accounting History, Fixed Asset Accounting Source, Fixed Asset Adjustment
General Ledger	General ledger, account attributes, account type attributes, fund attributes, fund type attributes, grant view, and transaction history	Time, General Ledger, Transaction History
Gift	Gift transaction, constituent, constituent entity, designation Financial Aid fund, endowment units, organization entity, organizational constituent, person detail, pledge, pledge transaction, solicitation, and the following gift related information: associated entity, auxiliary, matching, matching transaction, memo, and multiple.  This package also includes the <a href="#">“Organizational Constituent”</a> package.	Time, Constituent, Pledge, Pledge Transaction
Government Reporting	The following government related information: student, academic outcome, course, financial aid, financial aid fund, admissions; and person detail  This package also includes the <a href="#">“Person Supplemental”</a> package.	Government Student, Government Academic Outcome, Government Financial Aid, Government Admissions
Grant-Contract and Proposal	A subset of the Proposal reporting view, reorganizes common agency, organization, and date information into separate folders, contains the grant awarded as well as the awarded grant amount	Proposal Information
Grant and Project	Fund attributes and hierarchy, proposal, proposal text, transaction history, and the following grant related information: view, applied payments, attributes, billing detail, fund, ledger, receivable account detail, and text	Grant View, Grant Applied Payments, Grant Billing Detail, Grant Ledger, Grant Receivable Account Detail, Proposal, Transaction History

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Grant Ledger	Grant ledger, account attributes, account type attributes, fund attributes, fund type attributes, location attributes, organization attributes, program attributes, and transaction history	Time, Grant Ledger, Transaction History
Human Resource Application	Human Resource application, certification, combined academic outcome, current employment, employee, employment history, financial aid applicant status, human resource application status, human resource requisition, interview, past employment, person detail, position definition, previous degree, previous education, reference, skill, student, and transcript comment  This package also includes the <a href="#">“Person Supplemental”</a> package.	Human Resource Application, Combined Academic Outcome, Employee, Human Resource Requisition, Person Address, Position Definition, Telephone Current
Human Resource Faculty	Faculty, employee, employee position, person detail, and the following faculty related information: appointment history, attribute, sabbatical history, rank history, tracking  This package also includes the <a href="#">“Person Supplemental”</a> package.	Time, Faculty, Employee, Employee Position, Person Address, Telephone Current
Institution	Institution, degree, person detail, previous degree, previous education, previous education attendance, secondary school subject, and the following institution related information: geographic region, characteristic, degree, demographic, and subject	Institution
Invoice Payable	Invoice item, encumbrance, organization entity, person detail, transaction history vendor, and the following invoice related information: accounting, check, tax rate, and text	Invoice, Encumbrance, Invoice Accounting, Invoice Check, Vendor
Loan Disbursement	Student, enrollment, academic study, award by person, loan disbursement, receivable account detail, transaction history, and general ledger	

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Operating Ledger	Operating ledger, account attributes, account type attributes, fund attributes, fund type attributes, grant view, location attributes, organization attributes, program attributes, and transaction history	Time, Operating Ledger, Grant View
Operating Ledger Comparison	A subset of the columns from the Operating_Ledger reporting view to simplify departmental reporting, Account and Organization information is reorganized into separate folders, only includes hierarchy information for organization, only displays inception to date amounts (summary)	Time, Operating Ledger
Organization Entity	Organization entity, address current, contract, current employment, employment history, geographic region, internet address, current internet address, organization entity address, organizational constituent, receivable account, and vendor	Address Current, Contract, Organization Entity Address, Organizational Constituent
Organizational Constituent	Organizational constituent, annual giving, campaign giving history, constituent contact, constituent entity, cross reference, designation giving history, donor category, exclusion, funding interest, gift, gift society, gift transaction, mail, membership, organization contact, organization entity including address, organization funding area, person detail, pledge, pledge transaction, prospect, relationship, solicitation, special activity, special purpose group, and telephone	No filters
Payroll	Payroll document, employee, employee position, leave accrual, person address, person detail, transaction history, and the following payroll related information: deductions, distribution, earnings, employee position, labor distribution override, timesheet, and position timesheet  This package also includes the <a href="#">“Person Supplemental”</a> package.	Time, Payroll Document, Person Address, Telephone Current, Transaction History

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Person Demographic	Person detail, address, geographic region, internet address, current internet address, person sensitive, preferred address, relationship, salutation, and telephone	Person Detail, Address By Rule
Person Role	Person detail, administrator, admissions application, advisor, constituent, employee, faculty, Financial Aid applicant status, Human Resource application, organization chart, pre-student, prospect, receivable account, recruitment, student, and vendor	Person Detail
Person Supplemental	Person detail, activity, certification, combined academic outcome, communication, contact, current employment, employment history, hold, interest, internet address, current internet address, mail, medical information, international, veteran, previous degree and education, skill, test, and Visa,	No filters
Pledge	<p>Pledge transaction, constituent, constituent entity, gift, gift transaction, organization entity, organizational constituent, address, person detail, pledge, solicitation, telephone, and the following pledge related information: associated entity, auxiliary, conditional, installment, and matching</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Pledge Transaction, Constituent, Gift Transaction, Person Address, Pledge
Position	Position definition, employee position, human resource requisition, labor cost distribution, organization hierarchy, position budget, position labor distribution, person address, and current telephone	Position Definition, Employee Position, Labor Cost Distribution, Position Budget, Position Labor Distribution, Person Address, Telephone Current

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Purchasing Payable	Purchase order item, encumbrance, organization entity, person detail, received item and related text, returned item and related text, transaction history, vendor and related text, vendor type, invoice related information: accounting, check, item, tax rate, text, and purchase order related information: accounting, item summary, item text, tax rate, and text	Purchase Order, Invoice Accounting, Invoice Check, Purchase Order Item, Purchase Order Accounting,
Receivable Customer	Receivable account, contract third party, deposit, enrollment, exemption, hold, installment plan, miscellaneous transaction, organization entity and address, person address and detail, receivable account detail, and current telephone	Receivable Account, Contract Third Party, Enrollment, Exemption, Hold
Receivable Revenue	Receivable account detail, application of payment and detail accounting, award disbursement, contract, contract third party, deposit, exemption, general ledger, hold, installment plan, miscellaneous transaction, organization entity and address, person address and detail, receivable account, receivable accounting, receivable summary, current telephone, and transaction history  This package also includes the <a href="#">“Person Supplemental”</a> package.	Time, Receivable Account Detail, Contract Third Party, Exemption, General Ledger, Receivable Account, Transaction History
Recruitment Information	Recruitment information, admissions application, contact, geographic region institution, institution characteristic and demographic, interest, person address and detail, pre-student, previous degree and education, secondary school subject, current telephone, test, current Visa, and recruitment related information: attribute, cohort, learned, and source  This package also includes the <a href="#">“Person Supplemental”</a> package.	Time, Recruitment Information

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Relationship	<p>Relationship, first and related party information for constituent, constituent entity, organization address, organizational constituent, organization, person address, person, and current telephone</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	No Filters
Residential Life	<p>Person detail, address, meal assignment, phone assignment, room assignment, academic study, and student</p>	No filters
Schedule Offering	<p>Schedule offering, faculty, instructional assignment, meeting time, offering corequisite, offering grade type, offering prerequisite, schedule attribute, and course related information: attribute, catalog, corequisite, level, offering fees, prerequisite, prerequisite combined, and schedule</p>	Time
Solicitation Effort	<p>Solicitor assignment, solicitor assignment result, constituent entity, organizational constituent, constituent, solicitor organization fundraiser and goal, solicitor organization, and solicitor organization comment and goal</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	<p>Solicitor Organization Fundraiser, Solicitor Org Fundraiser Goal, Solicitor Organization, Solicitor Organization Goal</p>
Student Detail	<p>Student, academic outcome, academic study and extended, advisor, enrollment, financial aid enrollment, government course and student, GPA by level, GPA by term, hold, outcome departmental honor, outcome honor, outcome institutional honor, person address and detail, previous degree, previous education attendance, sport, current telephone, transcript comment, and student related information: activity, attribute, cohort, course, course attribute, and course grade change</p> <p>This package also includes the <a href="#">“Person Supplemental”</a> package.</p>	Time, Academic Study

**Table 1: Banner ODS Packages**

Package	Information Includes	Filter Folders
Student Detail Subset	A subset of the reporting views and columns that encompass the existing Student Detail model, drives off of the Student reporting view and includes information pertaining to a student's activities, GPA, attributes, cohorts, sports, enrollment and academic study	Time, Student
Transaction History	Transaction history, application of payment, encumbrance ledger, endowment distribution, fixed asset adjustment, general ledger, grant applied payments, grant billing detail, grand ledger, grant receivable account detail, invoice accounting, journal voucher text, membership interest, operating ledger, payroll distribution, purchase order accounting, receivable account detail, receivable accounting, vendor, vendor type	Time, Transaction History

## List of Values

A list of values is a set of valid values for a column in a reporting view. List of value (LOV) views are contained within the ODSLOV schema within the Banner ODS. The LOV views get their information from the Banner ODS composite MGT\_VALIDATION table.

The meta data layers are shipped containing lists of values to be used for drop-down lists or filters in queries and reports. The views contained within the ODSLOV schema provide the data which populates these lists of values. The values exist in Cognos 8 Business Intelligence with the exact same names as the LOV views, but without the underscores.

There is a Business view in Cognos 8 Business Intelligence that contains a model query subject for each of the ODSLOV views. The business view and area are called "List of Values".

See the "[List of Value Views](#)" section in [Chapter 5, "Data Models"](#) for the complete list of ODSLOV list of value views.

## Prompts

Prompts are Lists of Values query subjects in Cognos 8 Business Intelligence that can be used to create lists for prompts in the report writing component of Cognos 8 Business Intelligence Report Studio. There is an additional list of values query subject in the Cognos 8 Business Intelligence model called All Values LOV. This query subject contains

all the lists of values in the MGT\_VALIDATION table. If there is a list of values in the MGT\_VALIDATION table for which there is no ODSLOV view, you can use this query subject to create a list for a prompt in Report Studio.

## Filters

Filtering capabilities simplify and enhance reporting. When using the reporting tool metadata to write a report, you can apply a filter on any columns of the report so that specific report will retrieve a subset of the data in the database.

There are multiple ways to add filters to the metadata layer. One way is to add a query item to the metadata that will filter a subset of data that is used on a regular basis. This type of filter is referred to as a stand-alone or pre-defined filter. A stand-alone filter can be included in multiple data model packages. For example, the time filter “Year of Giving” is included in several packages. The filter definition is the same across all packages that include it. When you place a stand-alone filter on a report, the report will select only the data defined with that filter.

Another way to define a filter is to apply it to an entire set of data, like a query subject in the Cognos FM Model. When there is a need to define a subset of data by one of the attributes, a role based or alias query subject is defined. This type of filter would have the specific restriction embedded in the filter query item.

The Banner ODS includes both types of filters. Your institution can define additional filters of either type within the Framework Manager tool to meet your specific reporting requirements.

### Banner ODS Pre-defined Cognos Filters

The Banner ODS data model includes pre-defined stand-alone filters in almost all of its packages. Refer to the `ods_cognos_filters.csv` file to see a list of package filters and the filter definitions. This .csv file is delivered as part of your product documentation. The .csv format allows you to open the file in any spreadsheet application where you can then review the filter definitions. You can also sort the data by package or filter.

## Cognos Security Integration

### Cognos Authorization and fine-grained access

In a security context, authorization refers to permissions or defining “who can see what.” Cognos provides a complete infrastructure to define rules regarding “object” permissions (the ability to see folders or reports) as well as “data” permissions (which rows or columns of data individual users or groups are permitted to see). Cognos picks up its list of users and groups from the authentication providers defined at a given site, and maintains its own list of data permissions internally.

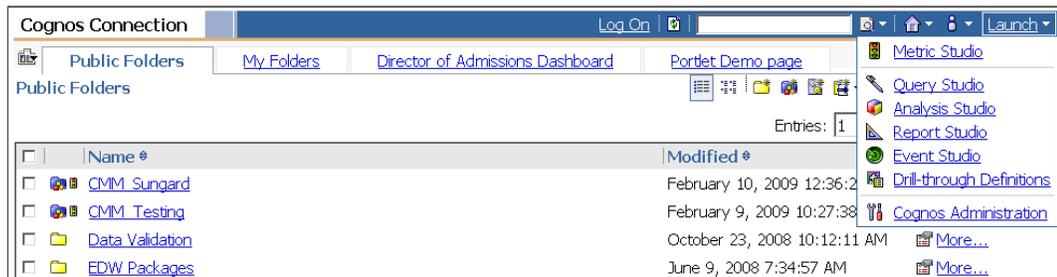
Data permissions can also be defined within the Banner Performance Reporting and Analytics (BPRA) database using the Fine-Grained Access (FGA) facility which allows for centralized maintenance of those rules for any non-Cognos based access as well. A typical Cognos configuration uses a single database connection (using a single Oracle username and password) for the BPRA database which does not allow for use of the BPRA FGA feature. However, it is also possible to configure Cognos to use multiple database connections, which then use the BPRA Fine-Grained Access rules.

## Cognos and BPRA authorization

Authorization enables you to create logins so that each user can access the same data source while still allowing them to use the fine-grained access rules already defined for them in the Administrative User Interface. Authorization could be used to set up more general Oracle users whose associated fine-grained access rules might apply to a type of report writer instead of a single person. Multiple Cognos users or roles could then be set up to secure the actual Cognos content (reports, dashboards, etc.), and then matched with data source signons which would provide the means to secure the actual data contained in the database.

For existing users, you would remove or disable the extra users so that as each user performs a query, their fine-grained access rules would be used. This should be done because their signon would be using their actual Oracle username to access the database.

1. Open Cognos Connection.
2. Click **Launch**.



3. Click **Cognos Administration**.
4. Click the **Configuration** tab.

The named data source connections display. The connections provide detailed connectivity information as to where to retrieve data.

5. Click one of the data sources to view the possible servers on which source data may reside.

In the screen samples, we have chosen the warehouse data source. By default, the defined server connection has the same name as the data source connection. (See the navigation bread crumbs at the top of the screen.)



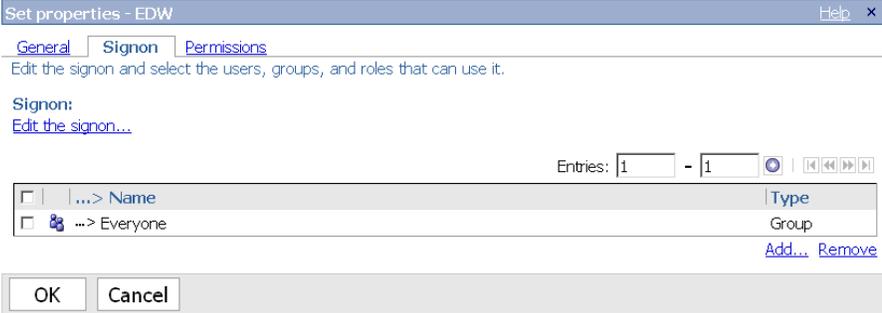
6. Navigate to the next layer of detail to define what users connect to this data source.

Again, as with the server connection name, the user connection name is inherited by the data source connection unless otherwise specified.

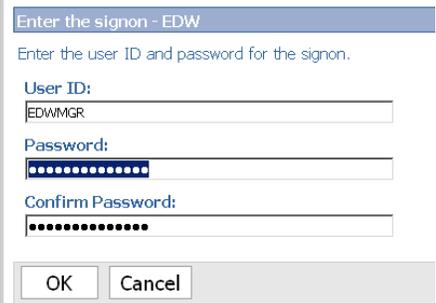
7. Click the **Set Properties** icon in the Actions column.



8. Click the **Signon** tab.



9. Click the **Edit the signon. . .** link to view or change the Oracle username and password for this connection.



In this case you'll see the warehouse data source connection defined with a username of EDWMGR, which would have access to all data.

Let's say, for example, that your institution has two Cognos users: John Doe and Bob Smith. You would like to make use of the Oracle fine-grained access (FGA) rules that are already defined for these two users in your Administrative User Interface. Accomplishing this is a simple matter of defining two different logins to the warehouse data source that is proprietary to each user.

**10.** To create a new connection for the warehouse data source, return to the user connection screen within the Cognos Administration.

**11.** Click the **New Signon** icon.



**12.** Create a signon for John Doe and call the signon "JDOE".

**13.** Click **Next**.

You are prompted for the Oracle username and password that will be used for this signon.

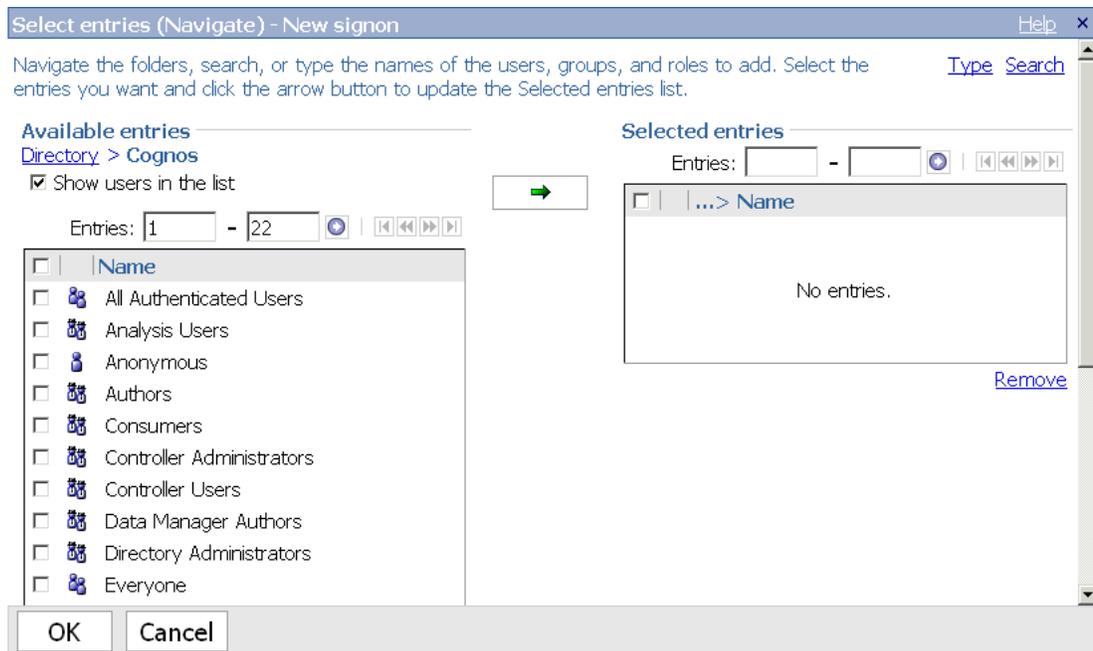
A screenshot of the 'Specify a signon - New Signon wizard' dialog box. It contains three input fields: 'User ID' with 'JDOE' entered, 'Password' with seven dots, and 'Confirm Password' with seven dots. At the bottom are buttons for 'Cancel', '< Back', 'Next >', and 'Finish'.

**14.** Enter the information, then click **Next**.

You are prompted for which Cognos users can access this signon.

**15.** Add JDOE to the list of users able to use this signon.

**16.** Click **OK**.



**17. Click Finish.**

You'll see that now there is a second signon for the warehouse data source.

**18. Repeat the above steps for Bob Smith.**

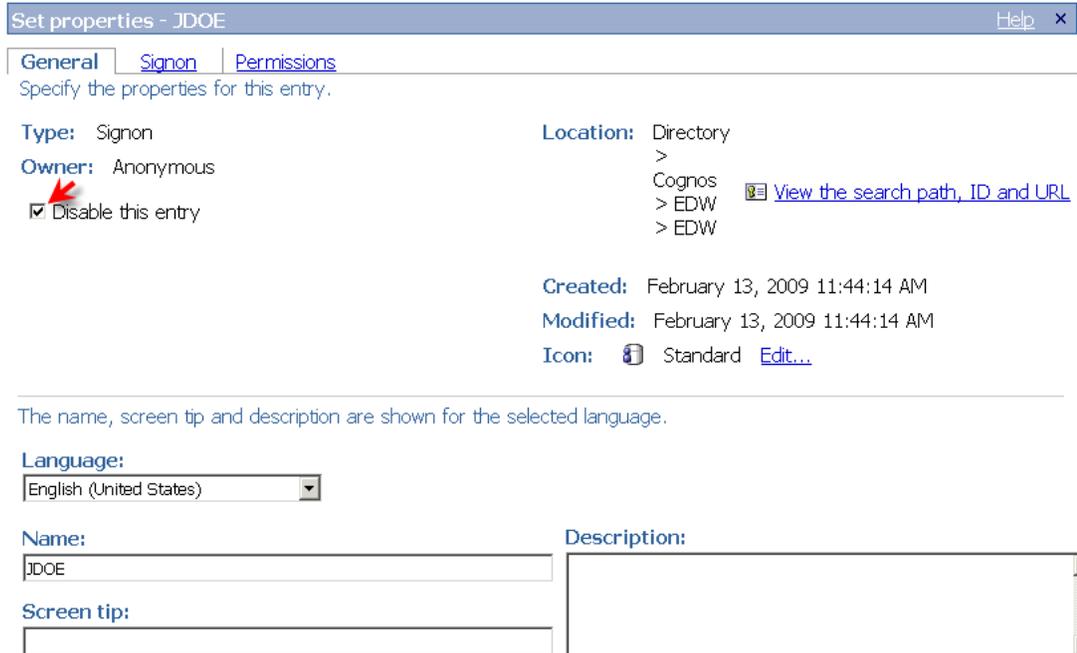
You will view three distinct signons for the warehouse.

<input type="checkbox"/>	Name	Modified	Actions
<input type="checkbox"/>	BSMITH	February 13, 2009 11:46:07 AM	<a href="#">More...</a>
<input type="checkbox"/>	EDW	August 8, 2008 12:02:46 PM	<a href="#">More...</a>
<input type="checkbox"/>	JDOE	February 13, 2009 11:44:14 AM	<a href="#">More...</a>

Last refresh time: February 13, 2009 11:46:07 AM

At this point, if you logged in as John Doe, and ran a query within Cognos, you would be prompted for what signon to use. (John or Bob) This would not be an ideal situation, because you would be prompted for which connection to use each time you accessed Cognos, and the warehouse signon is not FGA secured. You, therefore, would want to remove John or Bob's access to the warehouse signon, delete the signon, or disable it.

How to view or change what users have access to a signon was detailed previously. Deleting a signon is a straight forward activity. You select a signon and delete it. Disabling a signon is most likely the preferred method so that the overall warehouse signon is retained, but simply not active. This is a simple matter of checking the **Disable this entry** check box within the general properties of the signon.



Once this signon is disabled, the signons John and Bob will be the only two active signons. Therefore, if John Doe now signs into Cognos and performs a query, he will no longer be prompted to choose a signon (because he does not have permission to use the Bob signon) and his FGA rules would be enforced on his query because his signon is using his actual Oracle username of JDOE to access the database. Similarly, if Bob Smith signs into Cognos and performs a query, his FGA rules will be enforced because his signon is using his Oracle username of BSMITH.

To put this into more practical application, one might set up more general Oracle users within the data warehouse whose associated FGA rules might apply more broadly to a type of report writer as opposed to a single person. Multiple Cognos users or roles could then be set up to secure the actual Cognos content (reports, dashboards, etc.), and matched with data source signons which would provide the means to secure the actual data contained in the database.

For additional detailed information on Cognos security, see the *Cognos Administration and Security Guide*.

## Luminis authentication (single sign-on)

Authentication is the process of logging into a secured application. This section describes integrating Authentication considerations when using Cognos BI with BPRA solutions using the Luminis portal.

Usually Luminis and Cognos are configured to require users to enter a username and password to access their content. And usually, these credentials are stored and maintained separately. This requires users to log in once for Luminis and then again for Cognos every

time you use Cognos within the Luminis Portal. However, this dual log-in problem can be avoided by configuring Luminis to perform Single Signon (SSO) into Cognos. Luminis provides various techniques to accomplish SSO with external applications, but the simplest is their Generic Connector Framework (GCF). (This is documented extensively in the Luminis SDK / Generic Connector Framework Implementation Guide), but basically what happens after setting up a GCF is this:

- The user sees a Cognos BI link in a Luminis page and clicks it.
- The first time a user clicks a Cognos link within Luminis they are prompted for their Cognos username/password.
- Luminis passes that through to Cognos. If it authenticates, Luminis redirects that link to the appropriate Cognos page.
- Luminis also stores that Cognos username/password, so that for future attempts, the user doesn't have to enter anything. Luminis automatically passes through the username/password and authenticates the user for them.

An important consideration regarding Cognos security is that, unlike other applications, Cognos does not have its own security infrastructure. That is, it does not have its own "user store" (where it stores usernames/passwords). Instead, it interfaces with standard security providers (such as LDAP, NTLM, Windows Active Directory, etc.) so that users can re-use existing security setups without having to duplicate them. This is fully documented in the Cognos Setup/Install documentation, as well as various other Cognos extensibility documents. So this provides an opportunity to re-use an existing user store, so that clients only have to enter/remember a single username/password.

Combining reusing an existing user store for Cognos authentication with the Luminis GCF construct simplifies SSO because users can re-use existing usernames/passwords and (after an initial Luminis session) not have to re-enter credentials to access Cognos from Luminis. The only exception to this is when their password changes. They will have to re-enter it in Luminis once.

Luminis also supports different user stores as well. By default, Luminis uses its default LDAP implementation (the SunOne Directory server) as the location where it stores security credentials, but it can also be configured to use other external systems (such as Windows Domain, or other LDAP implementations). This flexibility regarding authentication storage between Luminis and Cognos provides the client the ability to centralize their authentication processes, which can further help with the SSO process.

Determining where to store security credentials is a client-specific choice, but for SSO illustration purposes, this documentation describes how to implement that using the default Luminis LDAP implementation. Some of the concepts are applicable to other configurations as well and are noted.

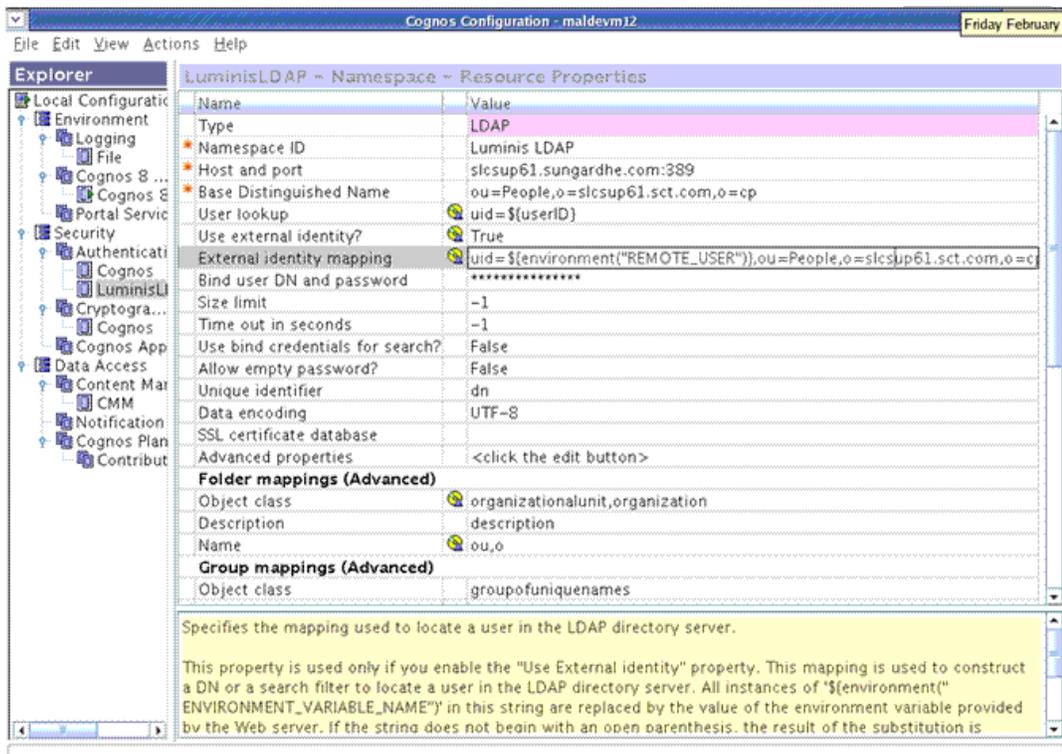
## Setting up Luminis single sign-on to Cognos using Luminis LDAP authentication

These steps were written for Cognos Business Intelligence 8.3 and Luminis 4.0.2. Later releases may follow the same steps. Refer to the release-specific versions of each product's associated documentation for more details.

All sample configuration files referenced can be found in the `luminis_sso` folder, under the `ods\reports\cognos_8` folder in the ODS source tree.

### In Cognos:

1. Configure an LDAP authentication namespace in Cognos to point to the Luminis LDAP instance. The properties page for the new namespace should look similar to the screen capture:



The majority of the default settings for an LDAP namespace can be retained with the following exceptions (as noted in the screen above, either with a red asterisk or a yellow circle icon):

## *NameValue*

Property Name	Property Value
Namespace ID	A unique name for the namespace - can be whatever you choose
Host and port	Needs to point to the Luminis machine and LDAP listener port
Base Distinguished Name	Ou=People, o=<machine>,o=cp
User lookup	uid=\$(userID)
Use external identity?	True
External identity mapping	uid=\${environment("REMOTE_USER")},ou=People, o=<machine>,o=cp
Bind user DN and password	user="cn=Directory Manager" password=<Luminis LDAP pw>

### *Under folder mappings (advanced):*

Property Names	Property Value
Object Class	organizationalunit, organization
Name	ou,o

2. Once configured, disable the Anonymous Login property in the default Cognos namespace. (Your Cognos content now requires login.)
3. Place a copy of the Luminis `pickup.html` file in the document root location of the Cognos web/application server, where it can be accessed from the Luminis machine.

### **In Luminis:**

1. Place a copy of the `cognos.xml`, `cognos.properties` and `cognos.config` files from the distribution in the GCF connector configuration folder, specifically:

#### *Luminis IV*

`$CP_ROOT/webapps/cpipconnector/WEB-INF/config`

#### *Luminis III*

`$CP_ROOT/products/sso(or gcf)/config`

2. Edit the `cognos.properties` file and update the values of the following fields to represent your Cognos installation:

Field	Description
<code>cognos.externalSystemURL</code>	Point to the main URL for your Cognos environment.
<code>cognos.pickup.remoteurl</code>	Point to the copy of the file you placed in the Cognos environment in step <a href="#">3</a> of the previous section.

3. Edit the `cognos.properties` file only if Cognos and Luminis are not authenticating to the same LDAP to allow the credentials to be entered the first time a person selects the link:

```
cognos.cpipconnector.getconfig.createonlogin = 0
cognos.cpipconnector.getconfig.usePDScredentials = false
```

4. Edit the `cognos.config` file and make sure the property:

```
es.cognos.configURL
```

points to your Luminis installation.

5. Edit the `cpipconnector.properties` file and append `cognos.properties` to the end of the `property.files` line toward the top of the file.
6. Perform the following configuration. Import the configuration parameters within `cognos.config` into the Luminis configuration:

```
configman -i cognos.config
```

7. Alter the `es.systems` parameter to include the cognos connector:

```
configman -g es.systems This gets the current list of connectors
```

```
configman -s es.systems "<current list> cognos"
```

8. Restart Luminis to reload the cache with the new configuration values.
9. Build a channel using a portal admin account and the following URL:

```
http://<Site Luminis Server>/cp/ip/login?sys=cognos&url=${urlPass}
```

*or*

refer to the next section [“Cognos channels in Luminis” on page 4-32](#).

10. Once these changes have been made, restart Luminis, or at least the cipconnector service.

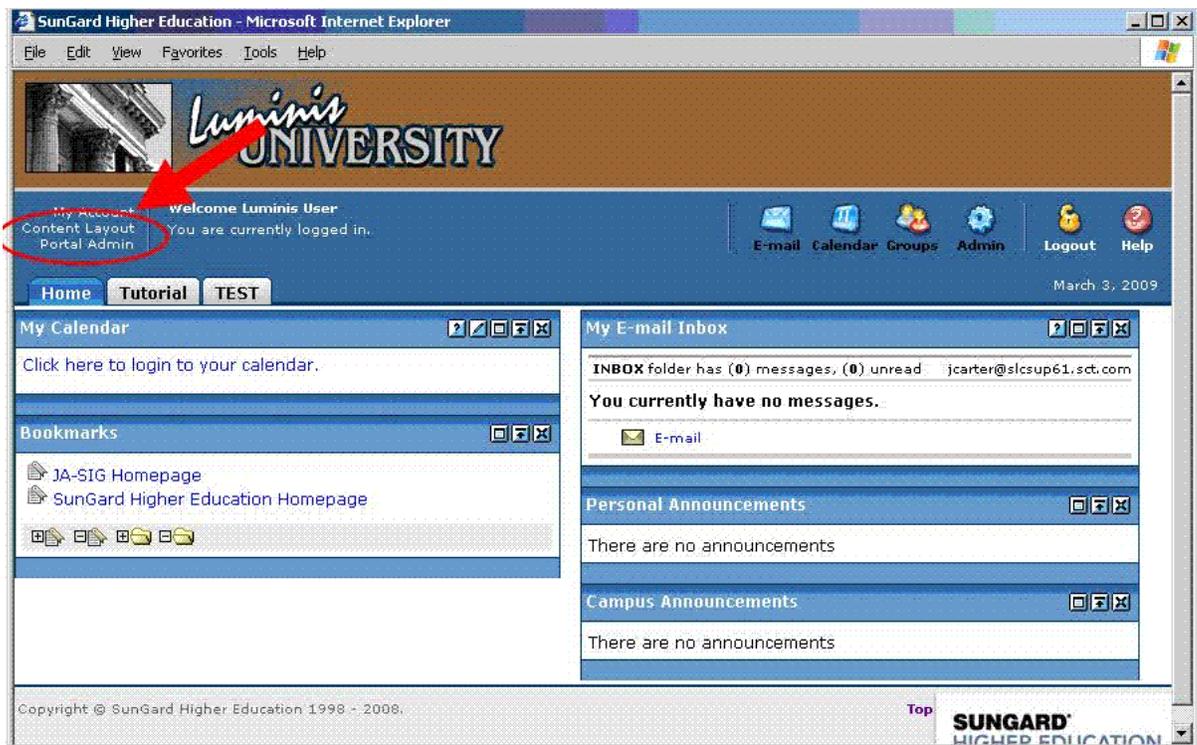
This explains how to configure Luminis and Cognos to share a username/password using Luminis's LDAP implementation. However, both Luminis and Cognos can be configured to use other authentication sources, even potentially different ones. When they are configured to use the same source, then the password information can be maintained in a single place. If they point to different sources, Luminis can store the username/password information and then it can be configured to prompt the user to re-enter the password whenever it changes.

For more information on configuring a Luminis GCF implementation, refer to the Luminis SDK / Generic Connector Framework Implementation Guide.

You can also find more information about configuring Cognos security in the *Cognos Configuration and Installation Guide*.

### Cognos channels in Luminis

Once SSO has been established, you can create links to Cognos within Luminis. Typically, this is accomplished using channels within the Luminis tabs. This process is documented in the "Luminis SDK/Channel Developer Guide". The end result is to be able to display Cognos content within Luminis, such as in the example screen below:



The simplest way to set up the links is using CPIP Inline Frames, which can then be defined for an entire tab, or as a column (portion) of a specific tab. These tabs can then be associated with a Luminis fragment definition, which can then be rolled out to specific Luminis users, or audiences (based on Luminis role). The key is to define which Cognos content should be displayed within a channel. That is done by capturing the actual URL used to access the Cognos content, and defining that in the `cognos.xml` file as a variable, which can then be referenced in the Luminis channel definition URL.

For example, consider the following URL definition that is delivered in the `cognos.xml` file delivered (in the `luminis_sso` folder in the data store source tree):

```
<SET a:symbol="urlPass" a:value=
"${properties.externalSystemURL}/${properties.cognosSystemID}/
cgi-bin/cognos.cgi?b_action=xts.run&m=portal/
cc.xts&gohome=&ui=" />
```

This defines a CPIP variable called `urlPass` which points to the base Cognos URL for Cognos Connection viewer.

#### Note

Notice the use of the `es.externalSystemURL` and `es.cognosSystemID` variables, which are defined in the `properties` file for the Cognos CPIP definition. This convention allows you to parameterize commonly used portions of URL definitions.

Also notice the conversion of all ampersand characters to the URL-encoding equivalent. This is required for proper parsing of the URL in the XML syntax. ■

This variable “`urlPass`” can now be used when referencing Cognos via a Luminis channel definition, as per:

```
http://<Site Luminis Server>/cp/ip/
login?sys=cognos&url=${urlPass}
```

which points to the Cognos Connection viewer (based on the definition of `urlPass`). By defining CPIP variables in the XML file to point to the desired Cognos reports/pages you wish to expose in Luminis, you can then create Luminis channels using those variables.

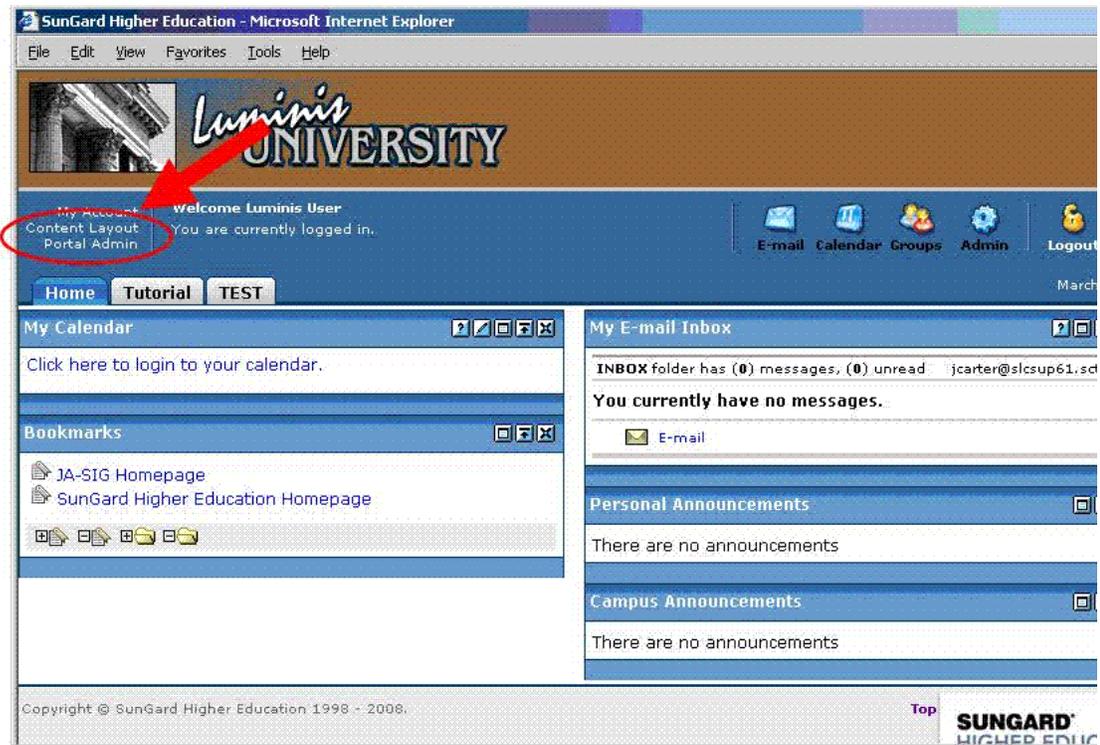
A series of example Cognos URLs are delivered as variables in the `cognos.xml` file (`urlPass`, `cogURL1`, `cogURL2`, `cogDash1` - `cogDash4`). These demonstrate the ability to define various Cognos content (reports) that can be viewed specifically using a Luminis channel, and these can be modified/updated/deleted as needed. Note that these URL values need to be URL-encoded when they are stored in the XML file for proper parsing by Luminis.

## Setting up Cognos channels in Luminis

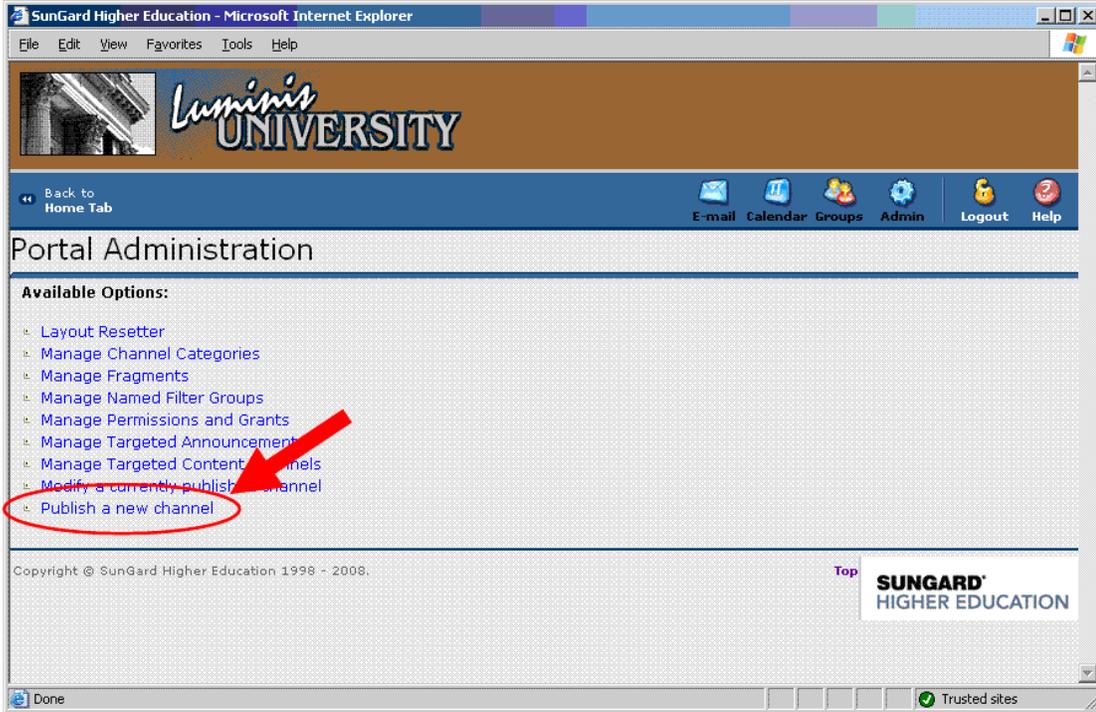
Now we will use all the pieces of what we have defined so far to create a basic Luminis channel to display the standard Cognos Connection viewer application. To start, assume a

new user is defined in Luminis (who has Luminis Administrative privileges, in order to administer the portal and content layout).

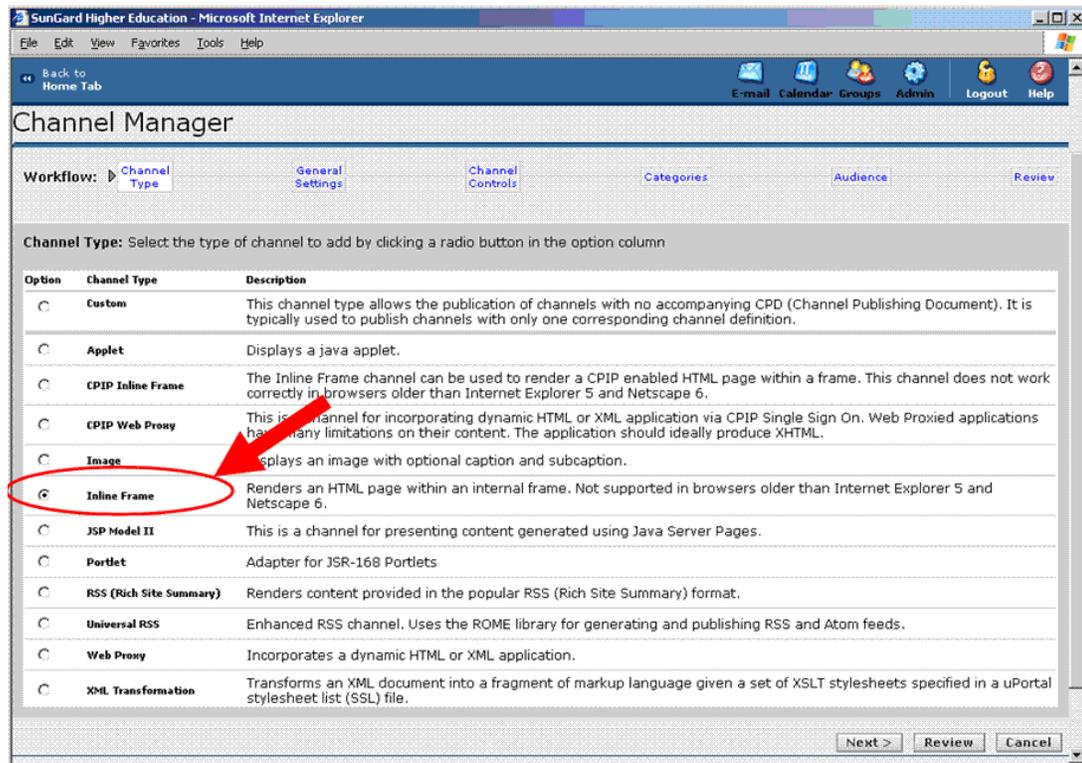
1. Click the **Portal Admin** link to define the channel.



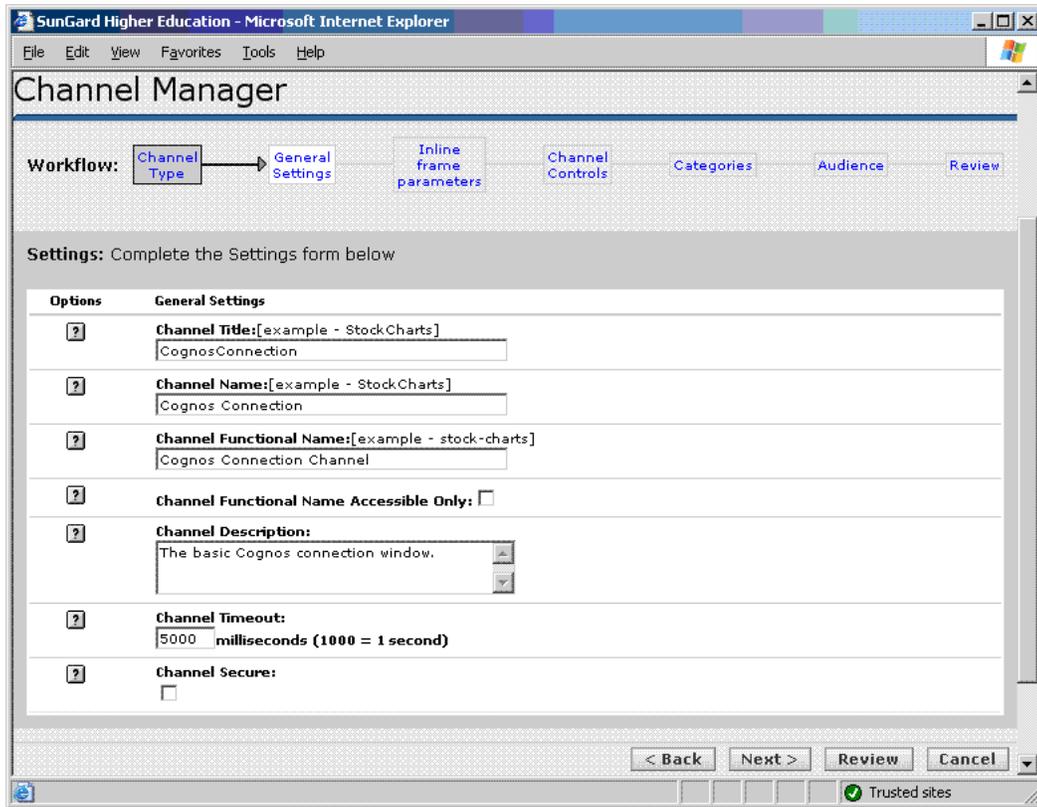
2. Select **Publish a new channel**.



3. Select **Inline Frame** as the Channel type.



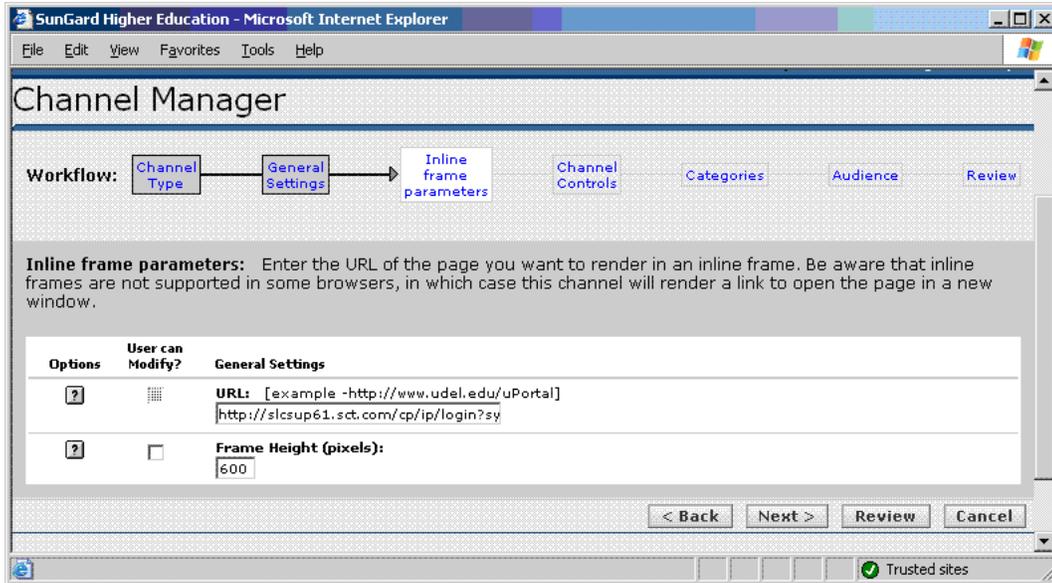
4. Click **Next**
5. Enter the title, names, and description information for the channel.



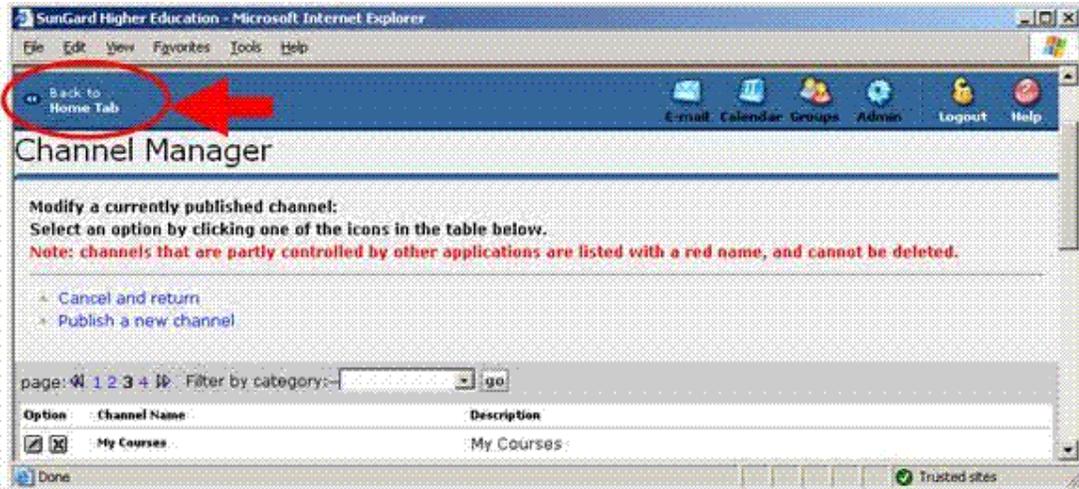
6. Click **Next**.
7. Enter the URL for this channel, which is the CPIP definition described earlier:
 

```
http://<Site Luminis Server>/cp/ip/login?sys=cognos&url=${urlPass}
```

Include the CPIP variable “urlPass” which points to the desired Cognos content.

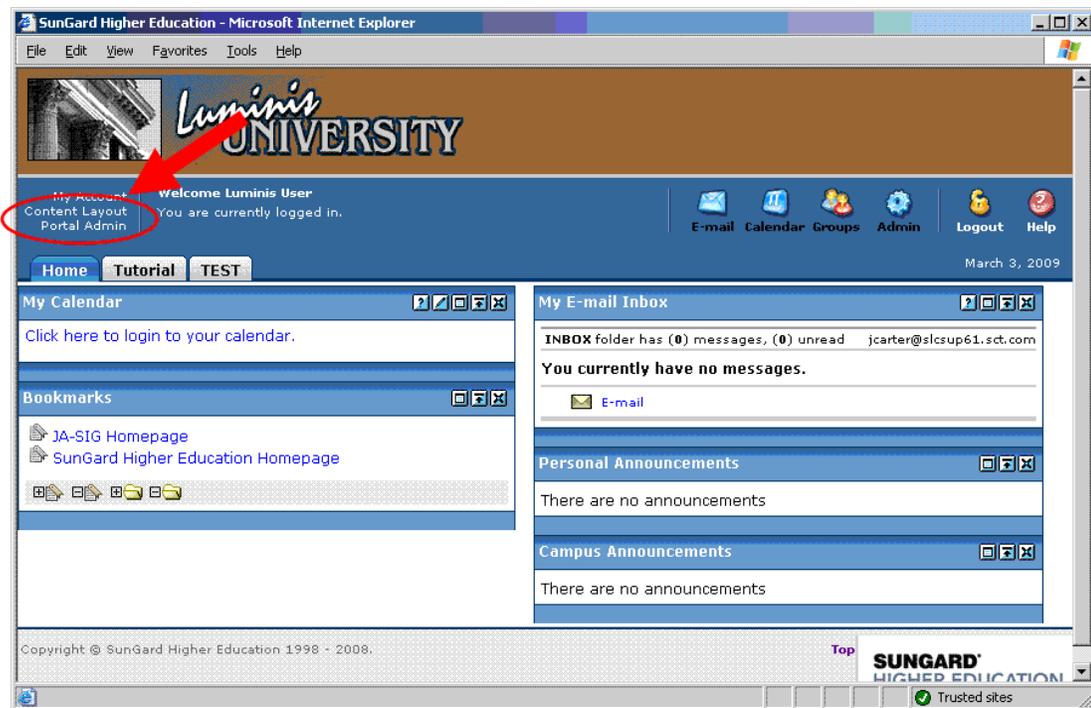


8. Click **Next**.
9. Click **Next** to accept the default values for Channel Controls.
10. Select a category (or categories) for the channel to be associated with.
11. Click **Next**.  
(The category is used to locate channels when searching for them later)
12. Click **Next** to accept the default values for Audience.
13. Click **Finished** to publish this channel.
14. Click the **Back to Home Tab** link in the upper left-hand side of the screen to return to the main Luminis page.

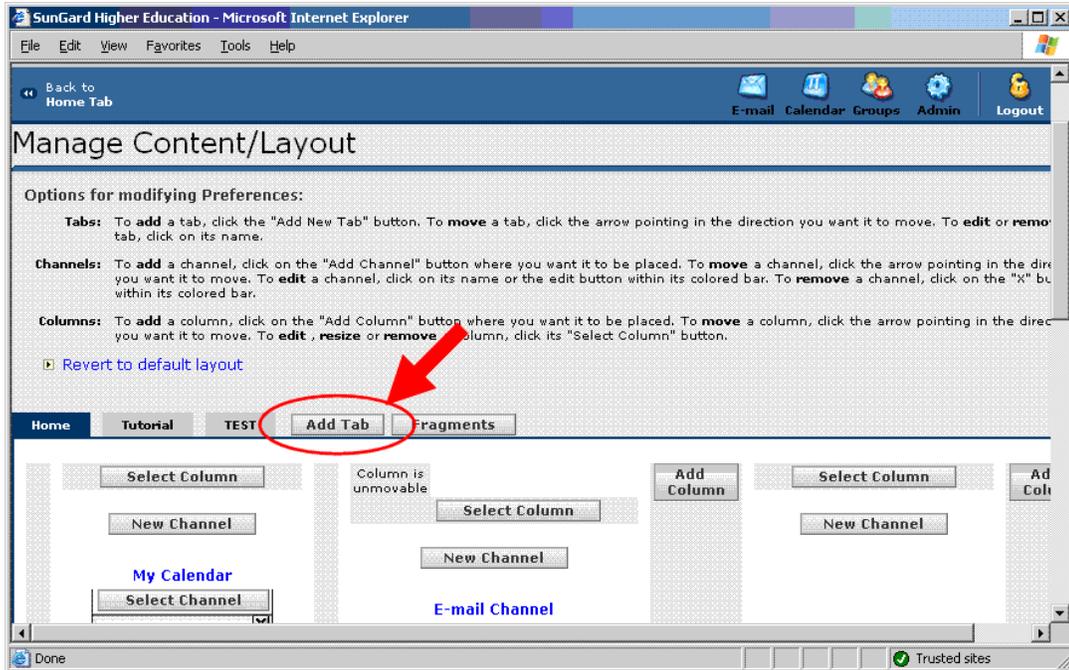


The next step is to associate this channel with a tab on the portal.

**15. Click the Content Layout link**

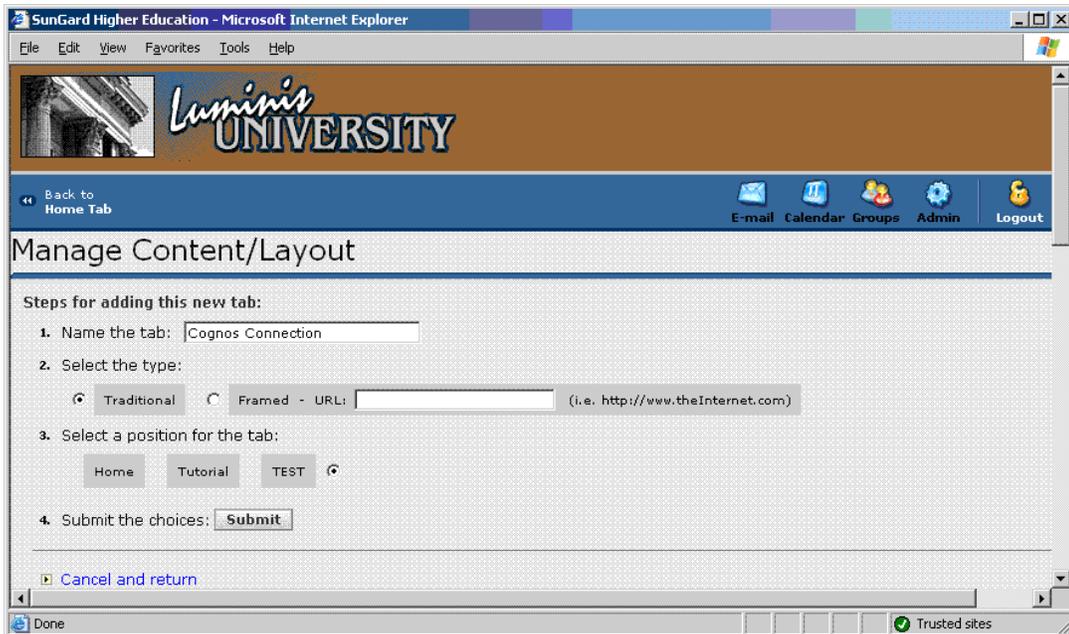


**16. Click the Add Tab button to create a new tab.**



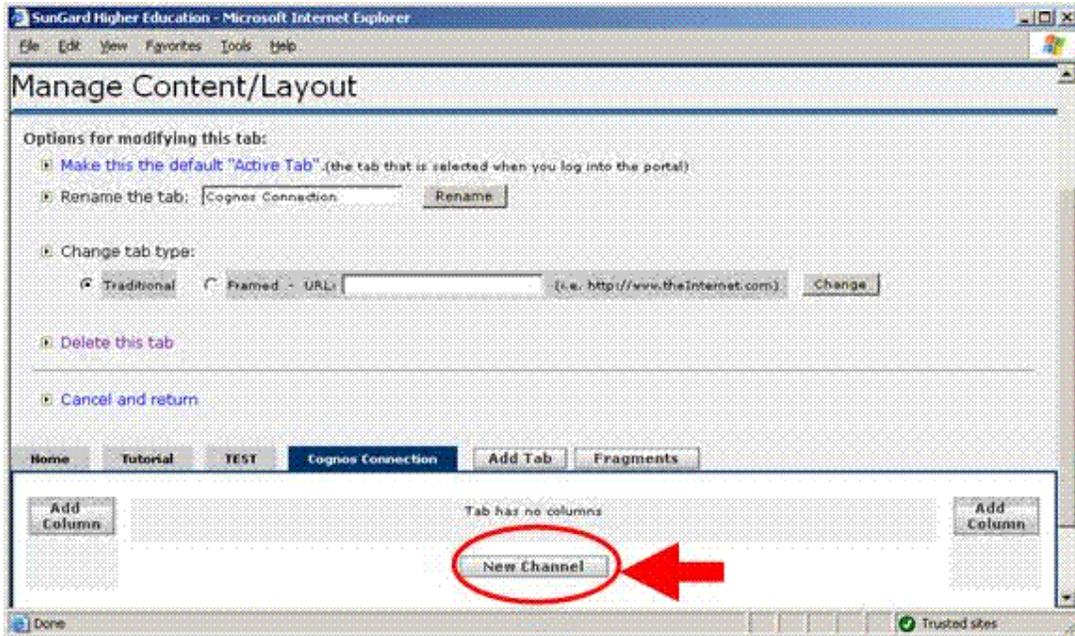
17. Enter the name for this Tab as **Cognos Connection**.

18. Click **Submit**.

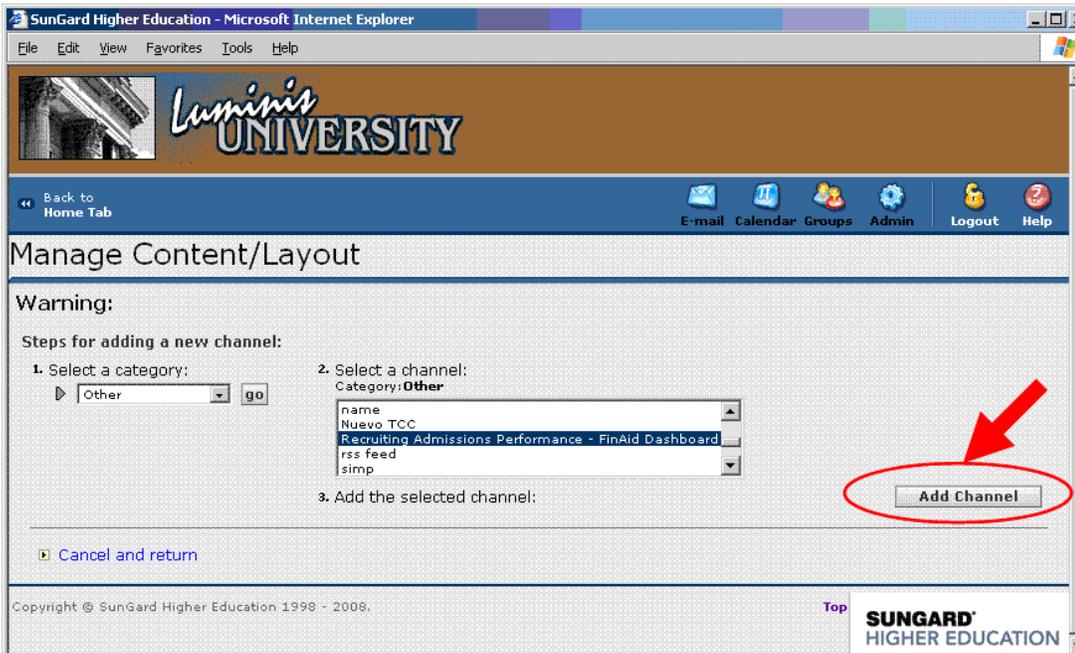


19. Select the new Cognos Connection tab.

20. Click the **New Channel** button:



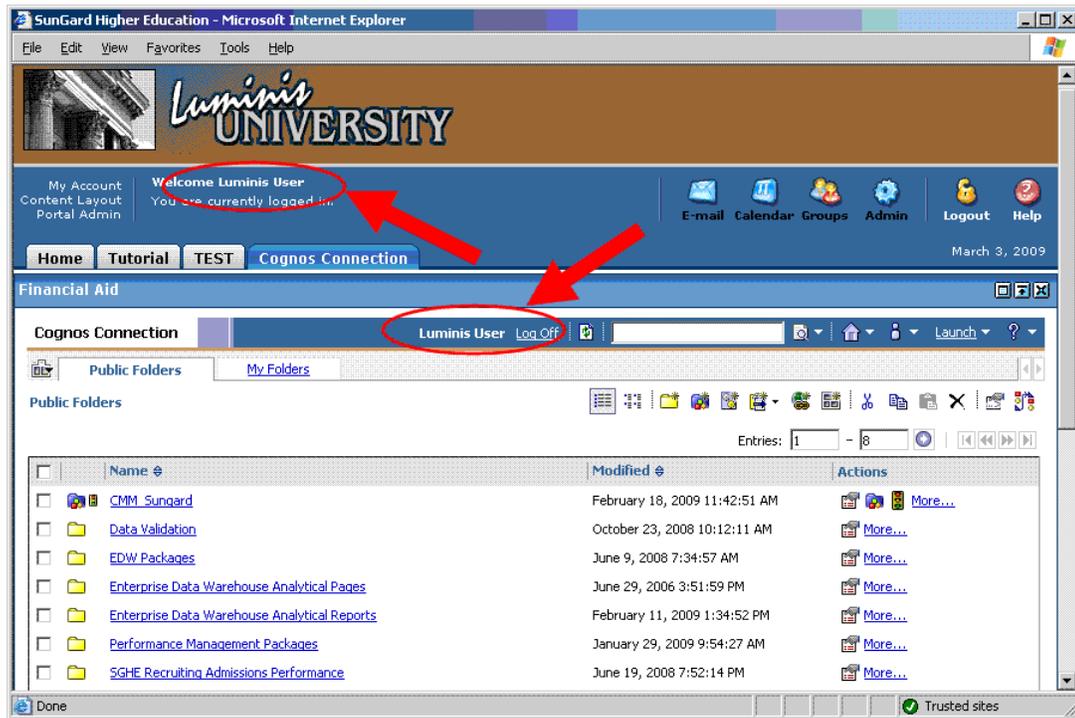
21. Select the channel by first entering the category (or **Select All**)
22. Click **Go**.
23. Select the channel from the listbox.
24. Click **Add Channel**:



25. Click the **Back to Home Tab** link to return to the main Luminis page,

26. Click the new **Cognos Connection** tab to see the new channel.

You will see the authenticated user in both Luminis and Cognos, with the name coming from the common user store (Luminis LDAP):



Following the same basic process, any Cognos pages can be deployed within Luminis, such as in the example screen below:



This screen is using the Cognos URL for the “Director of Financial Aid Dashboard”, which is defined in the cognos.xml file as:

```
<SET a:symbol="cogURL2"
a:value="\${properties.externalSystemURL}/
\${properties.cognosSystemID}/cgi-bin/
cognos.cgi?b_action=dashboard&pathinfo=/cm&frag-
header=true&path=storeID(%22i04AD276242AF47B680223538F724B0
6C%22)&ui=h1h3h4" />
```

so that the channel definition of this is then:

```
http://<Site Luminis Server>/cp/ip/
login?sys=cognos&url=\${cogURL2}
```

In the Cognos URL definition, note the use of the `path=storeID` parameter to refer to the Cognos object (the dashboard report) to display. This ID number is unique within a given Cognos installation so it can be advisable to use the actual search path for the object instead of the object ID when referencing it in the URL. The search path for a page/object is found in the Properties dialog, which is available in the Cognos Connection navigator interface.

## Further Cognos UI customization

While the Cognos applications (such as Cognos Connection viewer or the Studio applications) can be embedded within a Luminis page using the channel concepts discussed, some of the Cognos UI features may be unnecessary and distract from the overall usability of the page. To address this issue, Cognos provides various URL-based parameters which can control some aspects of the UI for these applications. This section describes those parameters and describes a few examples of setting these up.

Consider again the Cognos URL used to launch the Cognos Connection viewer previously, that was defined in the cognos.xml file:

```
<SET a:symbol="urlPass" a:value=
"${properties.externalSystemURL}/${properties.cognosSystemID}/
cgi-bin/cognos.cgi?b_action=xts.run&m=portal/
cc.xts&gohome=&ui=" />
```

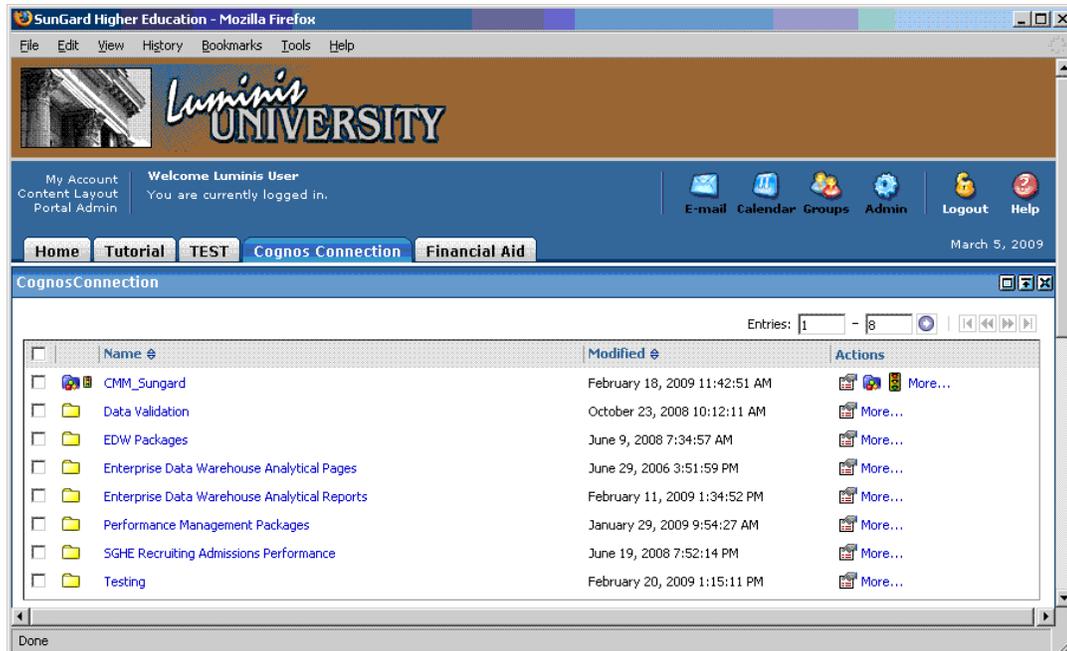
### Note

Consider the use of the trailing “ui=” parameter in the URL above. Cognos supports using URL parameters to customize the appearance and functionality of the web pages displayed by the Cognos Connection/Viewer interface. The “ui” parameter can take different values to display (or hide) various parts of the page. For example, ui=h1h2h3h4 will display all 4 header bars on a Cognos Connection page, whereas “ui=h1” would only display the first header bar. Similarly, the “frag-header” dparameter (=true/false) can be used to customize the appearance of Cognos dashboard reports displayed in Cognos connection. Following the technique described here, these values would get added to the URLs defined in the cognos.xml file, so they could then be referenced in Channel definitions.

For more information on using these parameters, see information about customizing the functionality of Cognos in the *Cognos Administration and Security Guide*. ■

Putting this into action, you can modify the Cognos Connection URL used earlier (defined in the cognos.xml as urlPass) as:

```
<SET a:symbol="urlPass" a:value=
"${properties.externalSystemURL}/${properties.cognosSystemID}/
cgi-bin/cognos.cgi?b_action=xts.run&m=portal/
cc.xts&gohome=&ui=h1h3h4" />
```



#### Note

Luminis can be configured to cache certain internal configuration data (such as channel definitions) so you may need to restart Luminis in order for channel definition changes to take effect.

For additional details on defining Luminis channels and UI elements, see the Luminis SDK/Channel Development Guide.

## Transaction history tracking process

In Framework Manager, you can view and play back actions performed on the project. An action log is an XML file that contains a set of transactions. Each transaction has a sequence number and one or more actions. The action log file is stored in the project folder.

For example, you make changes to a project in a test environment. When it is time to move the project to production, you can use log files to play back every action, or series of actions, that you performed in the test environment to create an identical project in the production environment. Similarly, as an alternative to branching and merging projects one might want to track a series of customizations applied to a project to enable the identical customizations to be applied to an upgraded version of that model.

There are two action log files. The `log.xml` file contains all the transactions that have been run and saved in the project. This file is created the first time you save the project and exists until you delete the project. The temporary file contains transactions that have been

run during the current session, but not saved. The temporary file is deleted when you close the project.

 **Note**

Previously you had the option to use the Cognos Branch Merge functionality to retain your customizations. It is recommended using the Cognos Transaction History Tracking functionality instead to maintain institution-specific changes and upgrade your Framework Manager model. ■

## View and save transaction history

You can view the transaction history in an action log file and then save it as a script.

1. From the **Project** menu, click **View Transaction History**.

 **Tip**

To make the dialog box larger, double-click the caption. Double-click again to restore the dialog box to its original size. ■

2. Click the transaction numbers that you want.

 **Tip**

To view the details of a transaction, click the plus sign (+) next to a transaction number. ■

3. Click **Save as Script**.
4. Type a name for the file.
5. Click **Save**. Do not save the file in the **logs** folder.
6. Click **Close**.

## Play back transactions from a log file

You can choose to play back a specific transaction or a combination of transactions in a project or segment action log file.

When you play back transactions from a log file, the script player applies the commands in the log file to the contents of the existing model. Errors appear if objects created by the log file already exist in the model.

After the script in a log file has run successfully, a backup of the original project is created in the parent directory of the project. If you want to undo the transactions performed in the script, you can use the backup to restore the project to its original state.

You must disable or clear any commands that will conflict with the contents of the model. You can then run the script again.

1. From the **Project** menu, click **Run Script**.
2. Select the script you want, and click **Open**.
3. If you want to view the details of a transaction, click the transaction.
4. Set the starting or stop point that you want.
  - To set the starting point for running the script, select the script and then click **Set the starting point**. You can do this at any time to skip an instruction or run instructions that have already been executed .
  - To set a stop point for the script, select the script and then click **Set the stop point** .  
You can stop the script to make a manual fix and then start it again.
  - To remove the stop point, click **Remove the stop point** .
5. Using the toolbar buttons, choose the run action that you want.

Buttons	Description
	Runs the script. After an error is encountered, clicking this button attempts to re-execute the failed instruction.
	Skips to the next transaction and runs the script to the end
	Runs the selected transaction only
	Skips to the next transaction and stops, but does not run any transactions

6. The project window is updated as the script is run.
7. Fix any errors encountered by the script either by retargeting objects or modifying the temporary project as required.
8. When the script has completed, click **Accept** to accept the changes or click **Revert** to undo the changes.

 **Note**

After you click **Accept** or **Revert**, you cannot use **Undo** and **Redo** for the current session. ■

## Brand Cognos Connection page

You can brand the Cognos Connection page to meet your institution's needs by customizing the banner, display text fonts, gradient, display color, and other format aspects.

To brand the Cognos Connection, perform the following steps.

1. Go to the `[install root]\cognos\c8\webcontent\skins` folder in your Cognos installation server.
2. Modify the following style sheets and XML files to reflect the desired display settings:
  - fonts.css
  - default.css
  - banner.css
  - system.xml

For detailed instructions on how to modify these files, refer to the *Customizing the IBM Cognos 8 UI* document provided with Cognos.

3. You can also replace the images in the following folders and update the style sheets and XML files (see step [2](#)) accordingly:
  - `[install root]\cognos\c8\webcontent\skins\sungardhe\branding`
  - `[install root]\cognos\c8\webcontent\skins\sungardhe\shared\images`

## Customize the welcome splash screen

You can customize the Welcome splash screen to reflect the desired look and feel.

To modify the default splash screen, perform the following steps.

1. Go to the `[install root]\cognos\c8\webcontent\skins` folder in your Cognos installation server.
2. Replace the following image files with your branded files:
  - cognos\_product\_label.gif
  - portal\_splash.gif

For additional information on customizing the splash screen, refer to the *Administration and Security Guide* document delivered with your Cognos installation.

# Oracle Business Intelligence Discoverer (Banner ODS)

---

## Delivered EULs

The structure of the business areas delivered in the Oracle Discoverer EUL is designed to mimic the business concepts in the ODS metadata. The EUL contains 51 organized Business Concepts which help users identify which reporting views to use when trying to write a report in a specific business area.

The older style EUL delivered in the Banner ODS 2.0 up to the Banner ODS 3.0 was also delivered for the Banner ODS 3.1. This EUL consists of just 3 Business Area. The main Business Area being the Banner ODS – Reporting Views that comprised all of the Banner ODS Reporting Views and their logical joins to other reporting views. This EUL has been synchronized with the subsequent Banner ODS releases and delivered so that existing reports written against this style EUL could continue to run without breaking in Banner ODS 3.1. Currently clients have the option of installing 1, both or none of these EULs.

Below are the scenarios that may help your institution decide which EUL to import moving forward:

### Existing institutions that wrote reports against the EUL in Banner ODS 3.0 or earlier

- Import and continue to use the old style EUL and not import the new style EUL. SunGard Higher Education will continue to support the old style EUL.
- Continue to use the old style EUL and also create a separate schema to import the new EUL.

In this scenario institutions can run their old reports against the older style EUL and create their new ones against the new EUL if they like the newer style EUL. In order for this scenario to happen 2 EUL schemas would need to be up and running. You cannot import both .eex files into the same EUL.

- Only import and use the new EUL.

This option would require institutions to potentially have to modify existing reports that were written against the old style EUL.

Clients new to the Banner ODS can use either EUL, but SunGard recommends that you use the new style EUL to benefit from the additional functionality.

## Lists of Values

A list of values (LOV) is a set of valid values for a column in a Banner ODS reporting view. List of value views are contained within the ODSLOV schema within the Banner ODS. The LOV views obtain their information from the Banner ODS composite table called MGT\_VALIDATION. The meta data layers are shipped containing lists of values to be used for drop-down lists or filters in queries and reports. The views contained within the ODSLOV schema provide the data which populates these lists of values. See the ODSLOV List of Values section in [Chapter 5, “Data Models”](#) for the complete list of ODSLOV list of value views. The values also exist in Oracle Business Intelligence Discoverer and Cognos 8 Business Intelligence with the exact same names as the LOV views but without the underscores.

In Oracle Business Intelligence Discoverer there is a business area that contains folders for each of the ODSLOV views. The business view and area are called “List of Values”.

### Lists of Values – Item Classes

List of value folders in Oracle Business Intelligence Discoverer are used to create item classes in Oracle Business Intelligence Discoverer Administrator. Item classes are groups of items that share some similar properties. An item class enables you to define item properties once, and then assign the item class to other items that share similar properties.

#### Example

The Academic Period LOV folder includes an item called Academic Period that describes each academic period. A similar item also called Academic Period is contained in the Academic Outcome folder.

To enable both items to share common properties (for example, a list of values), SunGard Higher Education created an item class from the list of value folder to define the properties, and applied it to both items. So, the list of values only had to be defined once from the ODSLOV view.

#### Note

You may notice that when using this approach there may be academic periods in the Academic Period LOV folder that are not in the Academic Outcome folder. However, accessing the list from the ODSLOV view is faster than accessing one created from the reporting view. If you need a list of values that exactly matches the values in the reporting view column, you can create an item class from a reporting view column similarly to how it was created from the ODSLOV views.) ■

Oracle Business Intelligence Discoverer end users use lists of values to display values or enter values in parameters and conditions.

A table of Oracle Business Intelligence Discoverer item classes that have been assigned to reporting view columns displays below. There are many more item classes that can be create from the list of value views. The following table lists the ones that are currently provided:

Item Class	Item Class	Item Class
Lov Academic Period.Value	Lov Division.Value	Lov Native Language.Value
Lov Academic Period.Value Description	Lov Division.Value Description	Lov Native Language.Value Description
Lov Academic Standing.Value	Lov Earnings.Value	Lov Organization Level 1.Value
Lov Academic Standing.Value Description	Lov Earnings.Value Description	Lov Organization Level 2.Value
Lov Academic Title.Value	Lov Educational Goal.Value	Lov Organization Level 3.Value
Lov Academic Year.Value	Lov Educational Goal.Value Description	Lov Organization Level 4.Value
Lov Academic Year.Value Description	Lov Eeo Skill.Value	Lov Organization Level 5.Value
Lov Account Class.Value	Lov Eeo Skill.Value Description	Lov Organization Level 6.Value
Lov Account Class.Value Description	Lov Employee Class.Value	Lov Organization Level 7.Value
Lov Account Level 1.Value	Lov Employee Group.Value	Lov Organization Pool.Value
Lov Account Level 2.Value	Lov Employee Group.Value Description	Lov Organization Pool.Value Description
Lov Account Level 3.Value	Lov Employee Status.Value	Lov Packaging Group.Value
Lov Account Level 4.Value	Lov Employee Status.Value Description	Lov Packaging Group.Value Description
Lov Account Pool.Value	Lov Employer Category.Value	Lov Position Change Reason.Value
Lov Account Pool.Value Description	Lov Employer Category.Value Description	Lov Position Change Reason.Value Description
Lov Account Type Level 1.Value	Lov Employer Industrial Type.Value	Lov Position Class.Value
Lov Account Type Level 2.Value	Lov Employer.Value	Lov Position Class.Value Description
Lov Activity Category.Value	Lov Employer.Value Description	Lov Position Deferred Pay.Value
Lov Activity Category.Value Description	Lov Employment Status.Value	Lov Position Deferred Pay.Value Description
Lov Activity Type.Value	Lov Employment Status.Value Description	Lov Position Location.Value
Lov Activity Type.Value Description	Lov Enrollment Status.Value	Lov Position Location.Value Description
Lov Activity.Value	Lov Enrollment Status.Value Description	Lov Position Status.Value

Item Class	Item Class	Item Class
Lov Activity.Value Description	Lov Fiscal Period.Value	Lov Position Status.Value Description
Lov Address Type.Value	Lov Fiscal Year.Value	Lov Position.Value
Lov Address Type.Value Description	Lov Foreign Currency.Value	Lov Pref Class.Value
Lov Admissions Attribute.Value	Lov Foreign Currency.Value Description	Lov Pref Class.Value Description
Lov Admissions Attribute.Value Description	Lov Fund Level 1.Value	Lov Prim Disability.Value
Lov Admissions Population.Value	Lov Fund Level 2.Value	Lov Prim Disability.Value Description
Lov Admissions Population.Value Description	Lov Fund Level 3.Value	Lov Program Classification.Value
Lov Advisor Name Lfmi.Value	Lov Fund Level 4.Value	Lov Program Classification.Value Description
Lov Advisor Type.Value	Lov Fund Level 5.Value	Lov Program Level 1.Value
Lov Advisor Type.Value Description	Lov Fund Pool.Value	Lov Program Level 2.Value
Lov Aid Year.Value	Lov Fund Pool.Value Description	Lov Program Level 3.Value
Lov Aid Year.Value Description	Lov Fund Source.Value	Lov Program Level 4.Value
Lov Application Status.Value	Lov Fund Source.Value Description	Lov Program.Value
Lov Application Status.Value Description	Lov Fund Type Level 1.Value	Lov Program.Value Description
Lov Assignment Grade.Value	Lov Fund Type Level 2.Value	Lov Progress Evaluation.Value
Lov Assignment Pay Id.Value	Lov Fund Type.Value	Lov Progress Evaluation.Value Description
Lov Assignment Pay Id.Value Description	Lov Fund Type.Value Description	Lov Project.Value
Lov Assignment Salary Group.Value	Lov Gender.Value	Lov Project.Value Description
Lov Assignment Salary Group.Value Description	Lov Gender.Value Description	Lov Prospect Status.Value
Lov Award Category.Value	Lov Grade Type.Value	Lov Rating Type.Value
Lov Award Category.Value Description	Lov Grade Type.Value Description	Lov Rating Type.Value Description
Lov Benefit Category.Value	Lov Grant.Value Description	Lov Rating.Value
Lov Benefit Category.Value Description	Lov Hold.Value	Lov Rating.Value Description
Lov Block Schedule.Value	Lov Hold.Value Description	Lov Recruiter.Value
Lov Block Schedule.Value Description	Lov Income Level.Value	Lov Recruiter.Value Description

Item Class	Item Class	Item Class
Lov Budget Group.Value	Lov Income Level.Value Description	Lov Registration Reason.Value
Lov Budget Group.Value Description	Lov Installment Plan.Value	Lov Registration Reason.Value Description
Lov Budget Phase.Value	Lov Installment Plan.Value Description	Lov Registration Status.Value
Lov Budget Phase.Value Description	Lov Instruction Method.Value	Lov Registration Status.Value Description
Lov Budget.Value	Lov Instruction Method.Value Description	Lov Residency.Value
Lov Building.Value	Lov Instructional Method.Value	Lov Residency.Value Description
Lov Building.Value Description	Lov Instructional Method.Value Description	Lov Review Type.Value
Lov Campaign Type.Value	Lov Instructor Name.Value	Lov Review Type.Value Description
Lov Campaign Type.Value Description	Lov Internal Account Type.Value	Lov Schedule Type.Value
Lov Campaign.Value	Lov Internal Account Type.Value Description	Lov Schedule Type.Value Description
Lov Campus.Value	Lov Internal Fund Type.Value	Lov Secondary School.Value
Lov Campus.Value Description	Lov Internal Fund Type.Value Description	Lov Site.Value
Lov Certification.Value	Lov Job Leave Category.Value	Lov Site.Value Description
Lov Certification.Value Description	Lov Job Leave Category.Value Description	Lov State Province.Value
Lov Chart Of Accounts.Value	Lov Job Suffix.Value	Lov State Province.Value Description
Lov Chart Of Accounts.Value Description	Lov Leadership Role.Value	Lov Student Population.Value
Lov Cohort.Value	Lov Leadership Role.Value Description	Lov Student Population.Value Description
Lov Cohort.Value Description	Lov Leave Of Absence Reason.Value	Lov Student Status.Value
Lov Collection Agency Name.Value	Lov Leave Of Absence Reason.Value Description	Lov Student Status.Value Description
Lov College.Value	Lov Legacy.Value	Lov Sub Academic Period.Value
Lov College.Value Description	Lov Legacy.Value Description	Lov Sub Academic Period.Value Description
Lov Commodity.Value	Lov Location Level 1.Value	Lov Subject.Value
Lov Commodity.Value Description	Lov Location Level 2.Value	Lov Subject.Value Description
Lov Contract Number.Value	Lov Location Level 3.Value	Lov Termination Reason.Value
Lov Contract Type.Value	Lov Location Level 4.Value	Lov Termination Reason.Value Description

Item Class	Item Class	Item Class
Lov Contract Type.Value Description	Lov Mail.Value	Lov Test Rule.Value
Lov County.Value	Lov Mail.Value Description	Lov Test.Value
Lov County.Value Description	Lov Major.Value	Lov Test.Value Description
Lov Course Attribute.Value	Lov Major.Value Description	Lov Tracking Group.Value
Lov Course Attribute.Value Description	Lov Marital Status.Value	Lov Tracking Group.Value Description
Lov Course Identification.Value	Lov Marital Status.Value Description	Lov Vendor Type.Value
Lov Course Reference Number.Value	Lov Meal Plan.Value	Lov Vendor Type.Value Description
Lov Current Time Status.Value	Lov Meal Plan.Value Description	Lov Veteran Category.Value
Lov Current Time Status.Value Description	Lov Meeting Type.Value	Lov Veteran Category.Value Description
Lov Department.Value	Lov Meeting Type.Value Description	Lov Worker Compensation Class.Value
Lov Department.Value Description	Lov Nation.Value	Lov Worker Compensation Class.Value Description
Lov Designation.Value	Lov Nation.Value Description	

## Conditions

Information from reporting views in the Banner ODS can be filtered using objects in the reporting tool meta data layer. In Oracle Business Intelligence Discoverer, they are called conditions.

The table at the bottom of this section identifies which conditions are set up in the Oracle Business Intelligence Discoverer EUL.

Oracle Business Intelligence Discoverer Object		Definition
Condition Name		A name for the condition on the reporting view.
Formula		An expression that is used to generate a where clause when querying the Reporting View.
Folder Name		An Oracle Business Intelligence Discoverer folder that represents a Banner ODS Reporting View.
Business Area		A logical grouping of Banner ODS Reporting Views.
Optional vs. Mandatory		If a condition is optional, it is visible in Oracle Business Intelligence Discoverer Plus and can be added to a workbook. If it is mandatory, it is invisible, and always applied to a folder.

Condition Name	Formula	Folder Name	Business Area	Optional vs. Mandatory
Endowment Distribution Document Type	Document Type = 20	Transaction History	Endowment Distribution	Optional
			Transaction History	Optional
Fixed Asset Adjustment Document Type	Document Type = 60	Transaction History	Fixed Asset	Optional
			Transaction History	Optional
Invoice Document Type	Document Type = 3	Transaction History	Invoice Payable	Optional
			Transaction History	Optional
Purchasing Document Type	Document Type = 2	Transaction History	Purchasing Payable	Optional
			Transaction History	Optional
Encumbrance Ledger Indicator	Ledger Indicator = 'E'	Transaction History	Encumbrance	Optional
			Transaction History	Optional
General Ledger Indicator	Ledger Indicator = 'G'	Transaction History	General Ledger	Optional
			Receivable Revenue	Optional
			Transaction History	Optional
Operating Ledger Indicator	Ledger Indicator = 'O'	Transaction History	Grant Ledger	Optional
			Operating Ledger	Optional
			Transaction History	Optional
Status = 'A'	Status = 'A'	Employee	Benefit Deduction	Mandatory
Status Disposition = 'U'	Status Disposition = 'U'	Hr Application	Human Resource Application	Mandatory
			Person Role	Mandatory

## Date Hierarchies

Hierarchies are logical relationships between items that enable you to drill up and down to view more or less detail. To analyze information effectively, Oracle Business Intelligence Discoverer end users should:

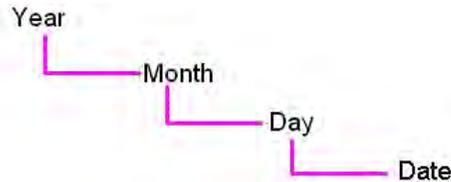
- Drill down to see more detail. The Year to Month to Day to Date, for example.

- Drill up to see how the detail contributes to information at a higher level. The Date to Day to Month to Year, for example.

 **Note**

Oracle Business Intelligence Discoverer automatically creates default date hierarchies against date items when you import a reporting view into the End User Layer (EUL) using Oracle Business Intelligence Discoverer Administrator. However, these default date hierarchies can cause performance issues. Oracle Business Intelligence Discoverer adds additional date items to a folder with default date hierarchies, using a function to populate the values returned for these items. This keeps queries that include these date items from using any indexes on the folder. Therefore the SunGard Higher Education date hierarchy replaces the Oracle Business Intelligence Discoverer default date hierarchies. ■

The EUL uses the CALENDAR\_DATE\_HIERARCHY reporting view to specify time periods needed for the hierarchy. Below is an example of the calendar date hierarchy used in the EUL.



A number of date folders are provided based on the CALENDAR\_DATE\_HIERARCHY reporting view. Each folder has a hierarchy defined on it. The table below lists the date folders in the Oracle Business Intelligence Discoverer EUL.

Date Folder	Date Folder
Award Status Date	Origination Tag Number Date
Birth Date	Package Completion Date
Collection Date	Pledge Date
Current Time Status Date	Pool Termination Date
Date Added Date	Position Begin Date
Deceased Date	Position Vacancy Date
Document Date	Posting Date
Enrollment Status Date	Profile Date
Function Start Date	Project Start Date
Highest Gift Amount Date	Purchase Order Date
Immigration Status Date	Rating Date
Income Spend End Date	Related Birth Date
Invoice Date	Related Deceased Date
Latest Decision Date	Start Date

Date Folder	Date Folder
Military Separation Date	Target Ask Date
Most Recent Gift Date	Tenure Date
Most Recent Pledge Date	Transaction Date
Operating Date	Visa Start Date

The EUL joins a date in a folder created for a typical reporting view to the date item in the date folder. The date is then accessible along with date items for Year, Month, and Day from the date folder. Drill up or down on these items to view more detail or a more generalized view of the information. The date folders are sometimes used in more than one business area.

The table below displays the date folders in the Oracle Business Intelligence Discoverer EUL by business area:

Business Area	Date Folder
Active Registration	Current Time Status Date
	Enrollment Status Date
Admissions Application	Latest Decision Date
Advancement Prospect	Target Ask Date
Advancement Rating	Rating Date
Annual Giving	Highest Gift Amount Date
Constituent	Most Recent Gift Date
	Most Recent Pledge Date
Constituent Entity	Birth Date
Employee	Profile Date
Endowment Distribution	Income Spend End Date
Endowment Unit	Pool Termination Date
Enrollment Management	Current Time Status Date
	Enrollment Status Date
Event	Function Start Date
Faculty Assignment	Tenure Date
Financial Aid Application	Package Completion Date
Financial Aid Award and Disbursement	Award Status Date
Fixed Asset	Origination Tag Number Date
Gift	Deceased Date
	Posting Date
Government Reporting	Visa Start Date
Grant and Project	Project Start Date
Human Resource Application	Position Vacancy Date
Human Resource Faculty	Tenure Date
Invoice Payable	Invoice Date

<b>Business Area</b>	<b>Date Folder</b>
Organizational Constituent	Most Recent Gift Date
	Most Recent Pledge Date
Payroll	Document Date
Person Demographic	Birth Date
	Deceased Date
	Immigration Status Date
	Military Separation Date
Person Role	Birth Date
	Deceased Date
	Immigration Status Date
	Military Separation Date
Person Supplemental	Birth Date
	Deceased Date
	Immigration Status Date
	Military Separation Date
Pledge	Deceased Date
	Pledge Date
	Posting Date
Position	Position Begin Date
Purchasing Payable	Invoice Date
	Purchase Order Date
Receivable Customer	Collection Date
Receivable Revenue	Operating Date
Recruitment Information	Date Added Date
Relationship	Related Birth Date
	Related Deceased Date
Residential Life	Birth Date
	Deceased Date
	Immigration Status Date
	Military Separation Date
Schedule Offering	Start Date
	Start Date
Transaction History	Transaction Date



# 5 Data Models

---

A typical data model indicates what information is in a database, how the information can be used, and how the items in the database relate to each other.

The Banner Operational Data Store (Banner ODS) is comprised of over 300 reporting views containing data across eight subject areas applicable to higher education; Accounts Receivable, Advancement, Common, Finance, Financial Aid, Human Resources, Student, and Travel and Expense.

Because of the size and scope of the Banner ODS data model, reporting views are grouped into logical “business concepts” to better illustrate the various business uses or reporting opportunities within the Banner ODS. These data models depict the reporting views contained in each business concept and how the reporting views, and the data within these reporting views, is related to each other.

The data models (Entity Relationship Diagrams or ERDs) in this chapter incorporate most of the reporting views available in the Banner ODS, and illustrate business concepts within and across all Banner ODS subject areas. However, this is not an inclusive representation as additional business concepts could be conceived and supported by the Banner ODS. There may also be alternative associations between the reporting views within any given data model depending on the type of report you are running.

## Entity Relationship Diagrams (ERD)

---

The most widely used method for representing a data model is the Entity Relationship Diagram (ERD). This chapter uses ERDs to represent the logical relationships between the reporting views within a given Banner ODS business concept. Each ERD represents a business concept. The entities within each ERD correspond to the reporting views associated with that business concept. They don't include all the columns in the reporting views. They only display the primary key columns.

The following legend explains the relationships used in the business concept ERDs.

## ERD Relationship Legend

The legend contains three categories:

- Identifying Relationships
- Optional Non-Identifying Relationships
- Special Relationships

### Identifying Relationships

Most relationships in the business concept ERDs are identifying relationships. Identifying relationships are represented by a solid line. An identifying relationship is a relationship between two entities in which an instance of a child entity is identified through its association with a parent entity, which means the child entity is dependent on the parent entity for its identity and cannot exist without it. The primary key attributes migrate from a parent entity to a child entity, so the primary key of the child has attributes from the parent entity primary key in it. These are called foreign keys, and they are marked with the characters (FK) beside them.

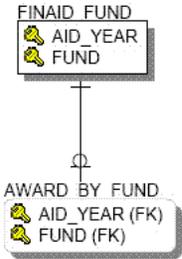
Identifying Relationship	Description
<i>One to Exactly One</i>	Each Person Detail has exactly one Person.

```
graph TD; PD[PERSON_DETAIL] ---|1 to 1| P[PERSON]; style PD stroke-width:2px; style P stroke-width:2px;
```

Identifying Relationship	Description
--------------------------	-------------

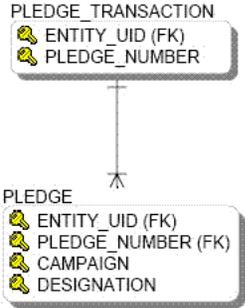
*One to Zero or One*

Each Finaid Fund has zero or one Award by Fund.



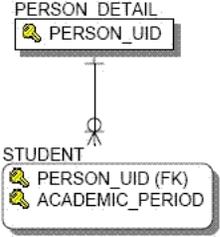
*One to One or More*

Each Pledge Transaction has one or more Pledges.



*One to Zero, One or More*

Each Person has zero, one, or more Students. This makes sense because a Student really represents a student for each academic period.

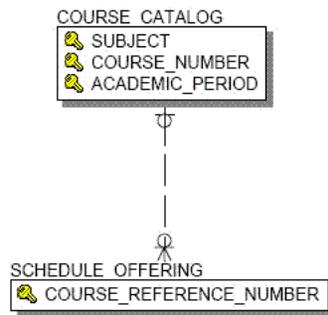


## Optional Non-Identifying Relationship

Non-identifying relationships are represented by a dashed line. A non-identifying relationship is a relationship between two entities in which an instance of the child entity is not identified through its association with a parent entity. This means the child entity is not dependent on the parent entity for its identity and can exist without it. In an optional non-identifying relationship, the attributes that are migrated into the non-key area of the child entity are not required in the child entity. Therefore, nulls are allowed in the foreign key.

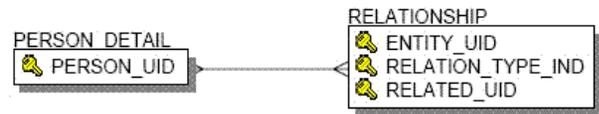
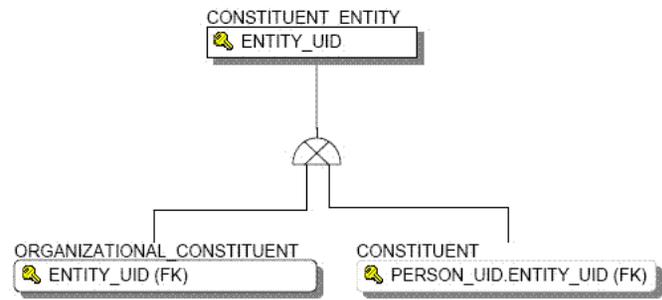
*Zero or One to Zero, One, or More*

Each Course Catalog entry has zero, one or more Schedule Offerings. There may be a Schedule Offering without a Course Catalog entry.



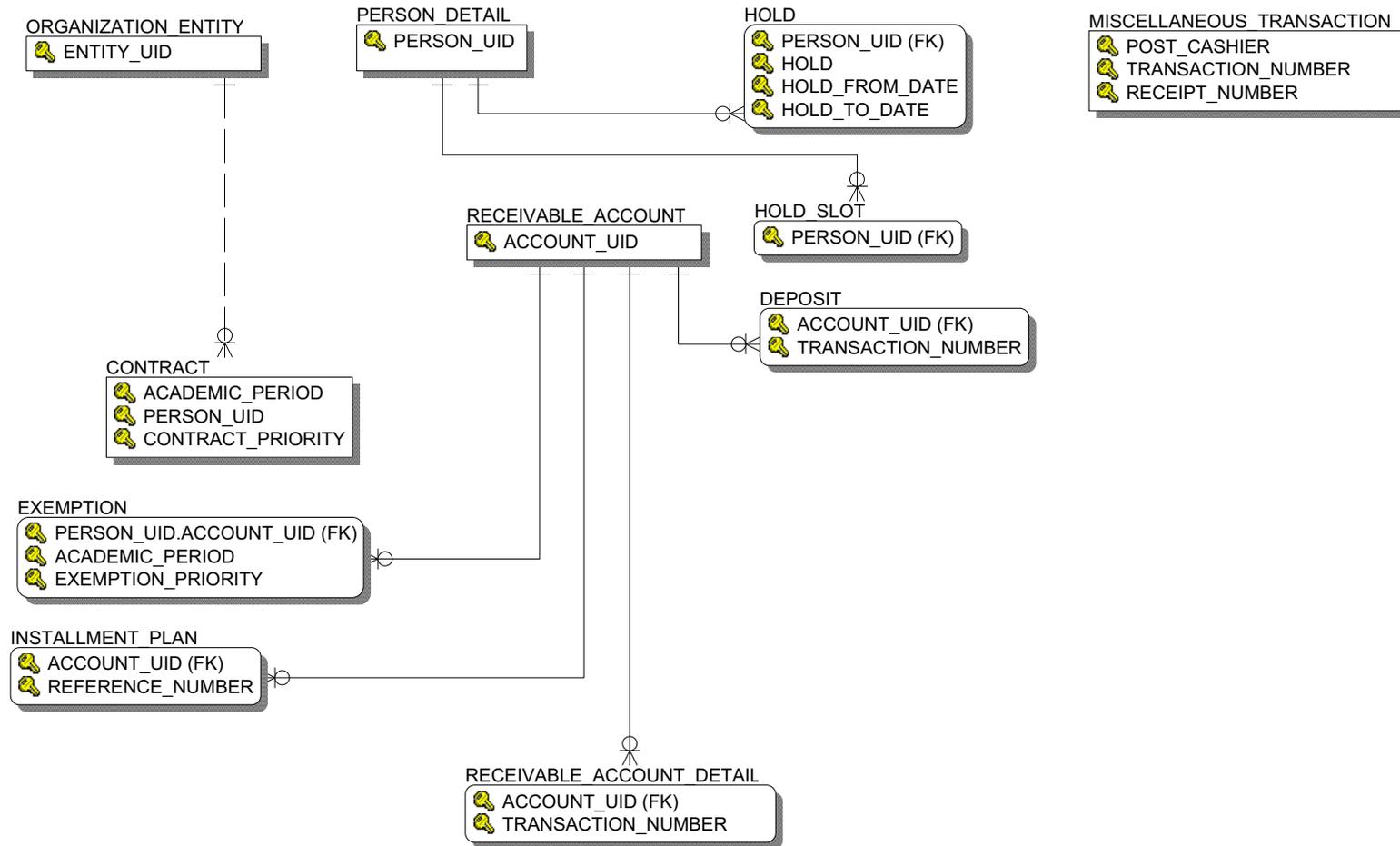
## Special Relationships

Special relationships are logical relationships that don't use foreign keys.

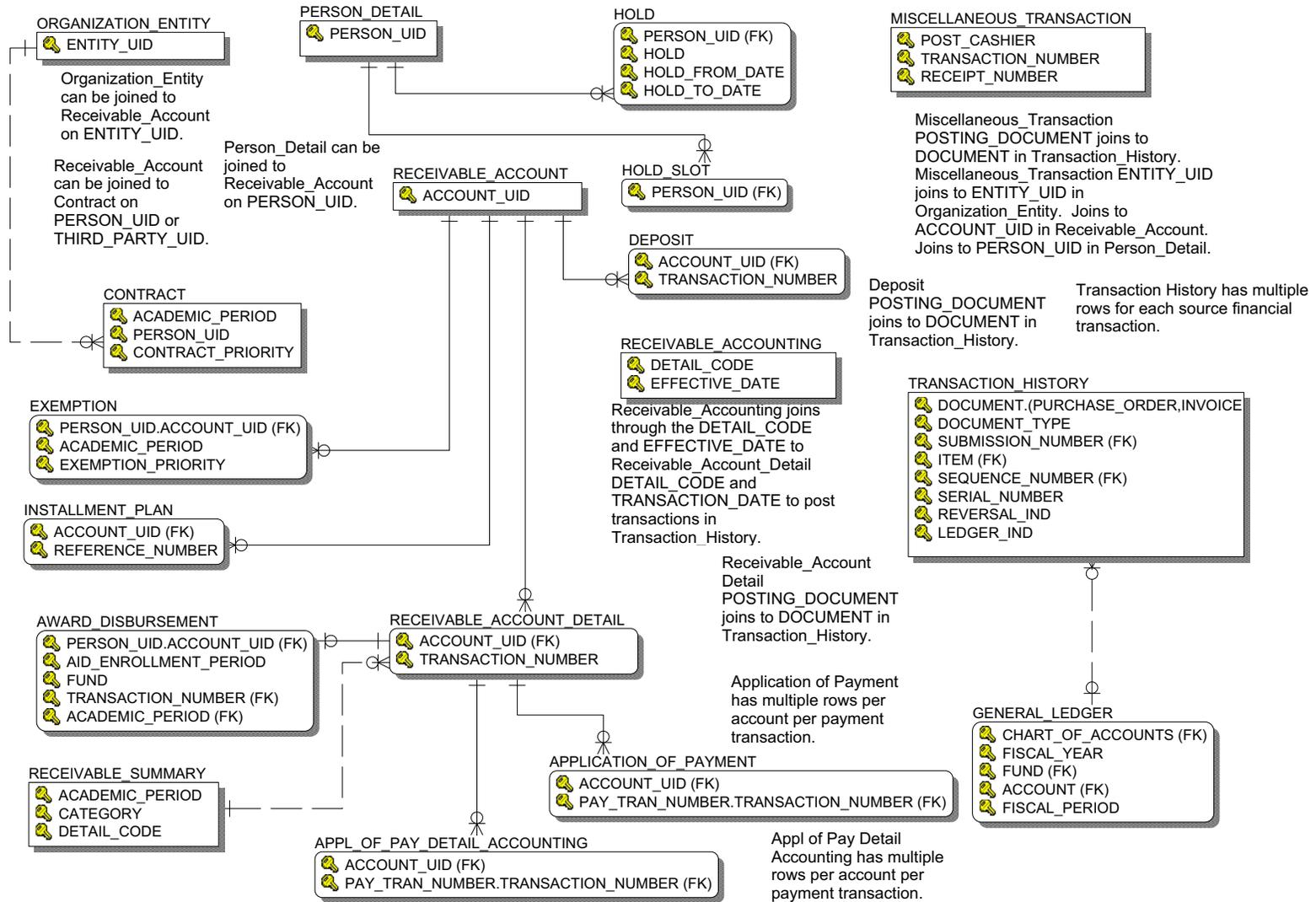
Special Relationship	Description
<p data-bbox="346 332 525 365"><i>Many to Many</i></p>  <pre>graph LR; PD[PERSON_DETAIL] --&gt; R[RELATIONSHIP]; style PD stroke:#f00,stroke-width:2px; style R stroke:#f00,stroke-width:2px;</pre>	<p data-bbox="1060 332 1701 527">A relationship between two entities where instances in one entity have zero, one, or more related instances in the other entity. In the example ERD relationship, each Person can have many Relationships, and each Relationship can be related to many (actually two) Persons.</p>
<p data-bbox="346 568 609 600"><i>Subtype Relationships</i></p>  <pre>graph TD; CE[CONSTITUENT_ENTITY] --- J(( )); OC[ORGANIZATIONAL_CONSTITUENT] --- J; C[CONSTITUENT] --- J; style CE stroke:#f00,stroke-width:2px; style J stroke:#f00,stroke-width:2px; style OC stroke:#f00,stroke-width:2px; style C stroke:#f00,stroke-width:2px;</pre>	<p data-bbox="1060 568 1701 836">In an ERD, you can show that organizational constituents and constituents are part of a larger category, Constituent Entity, by creating a subtype relationship. A subtype relationship connects an entity that defines the category and two or more additional entities that define each of the elements of the category. The parent entity of the category is considered the supertype and each child entity is considered a subtype.</p>

# Accounts Receivable

## Receivable Customer

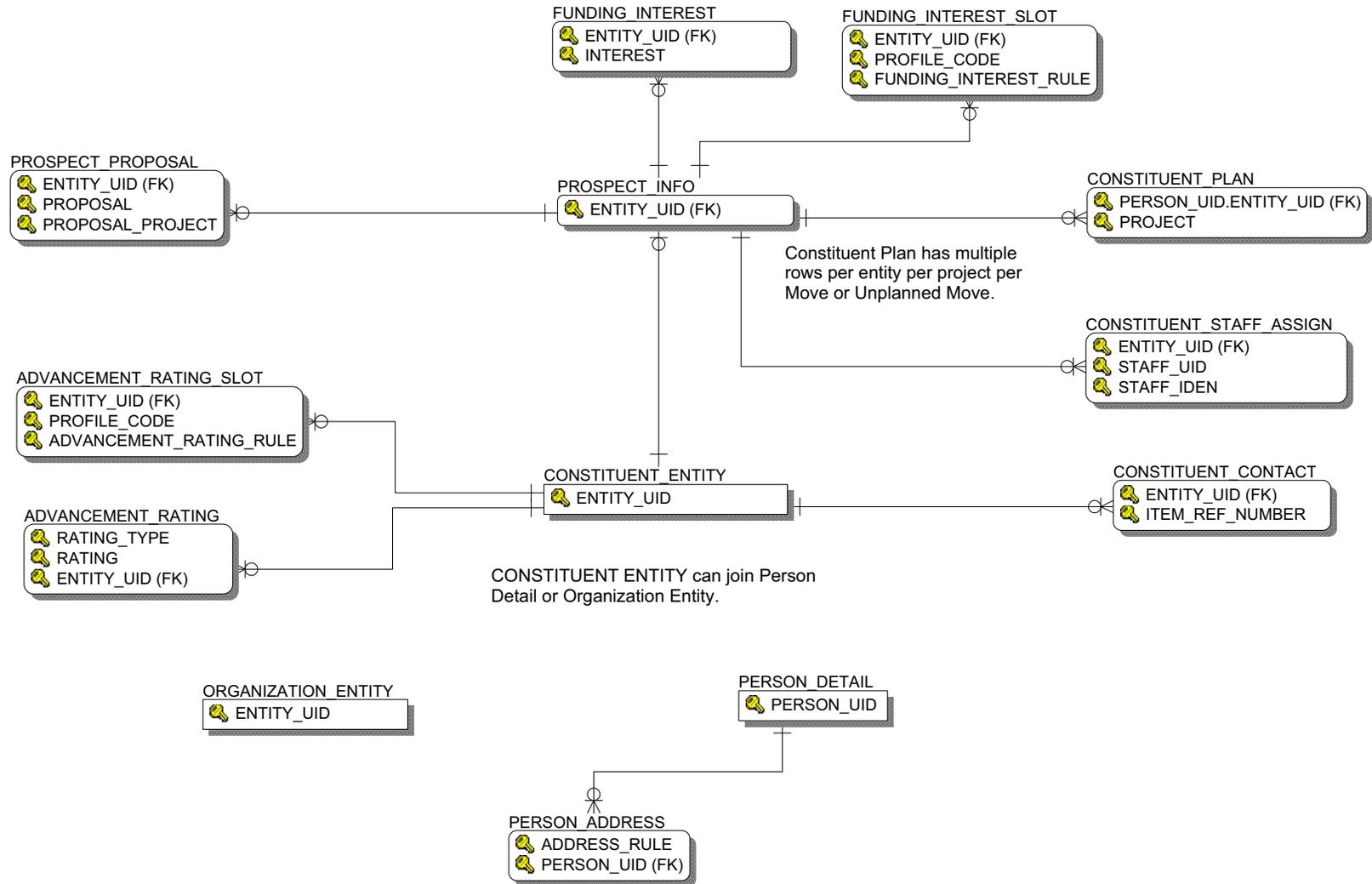


# Receivable Revenue



# Advancement

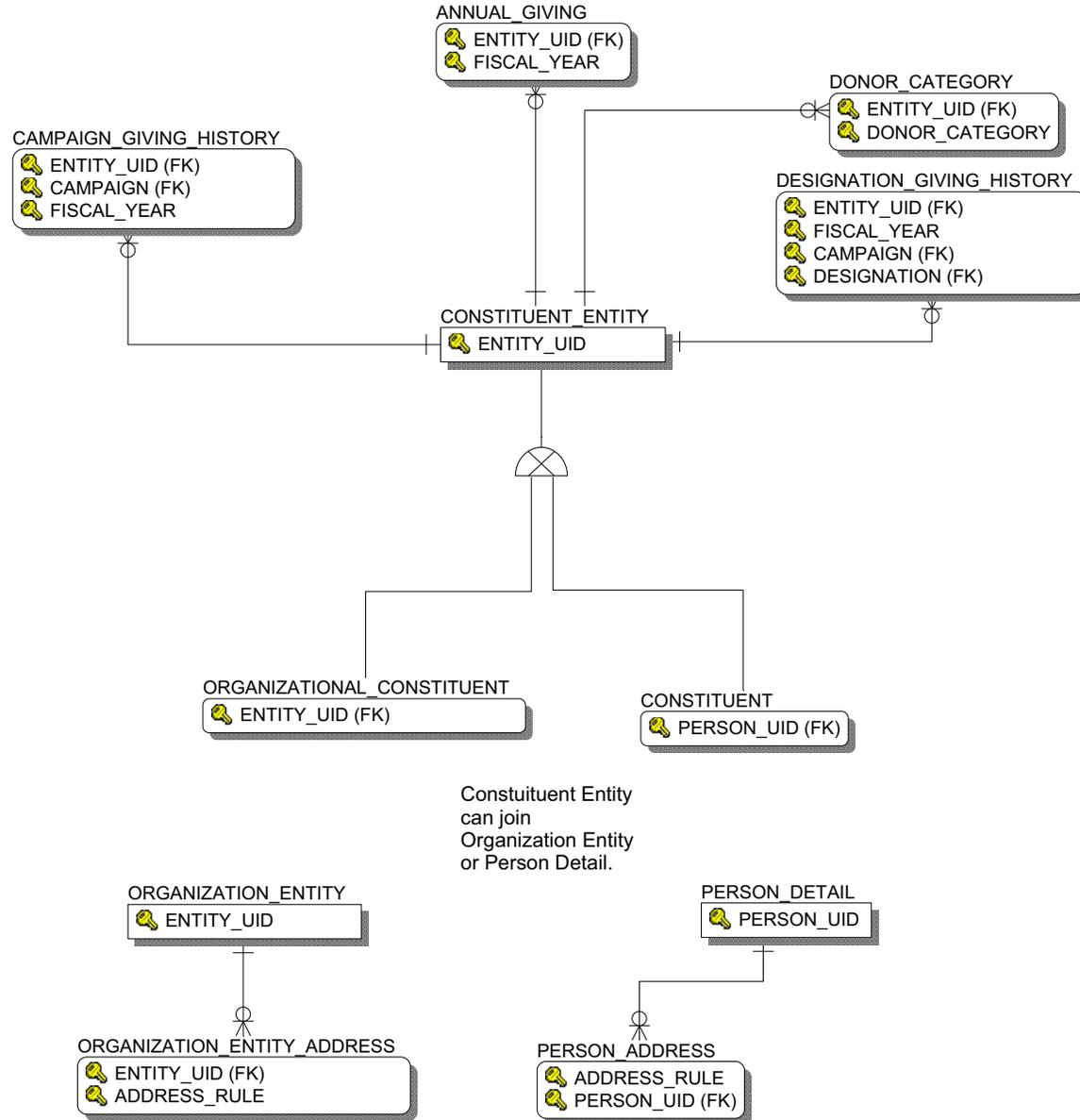
## Advancement Prospect



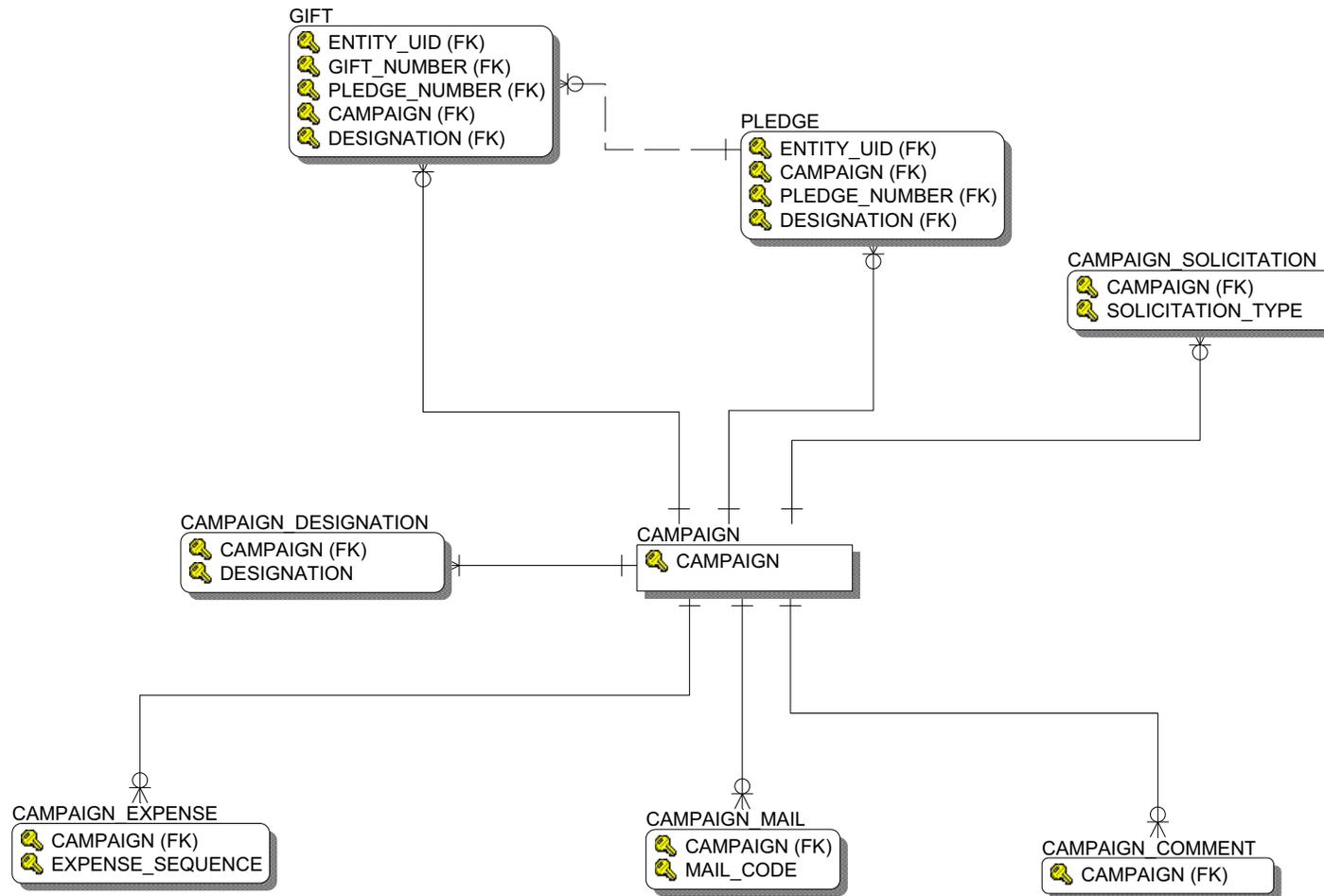
# Advancement Rating



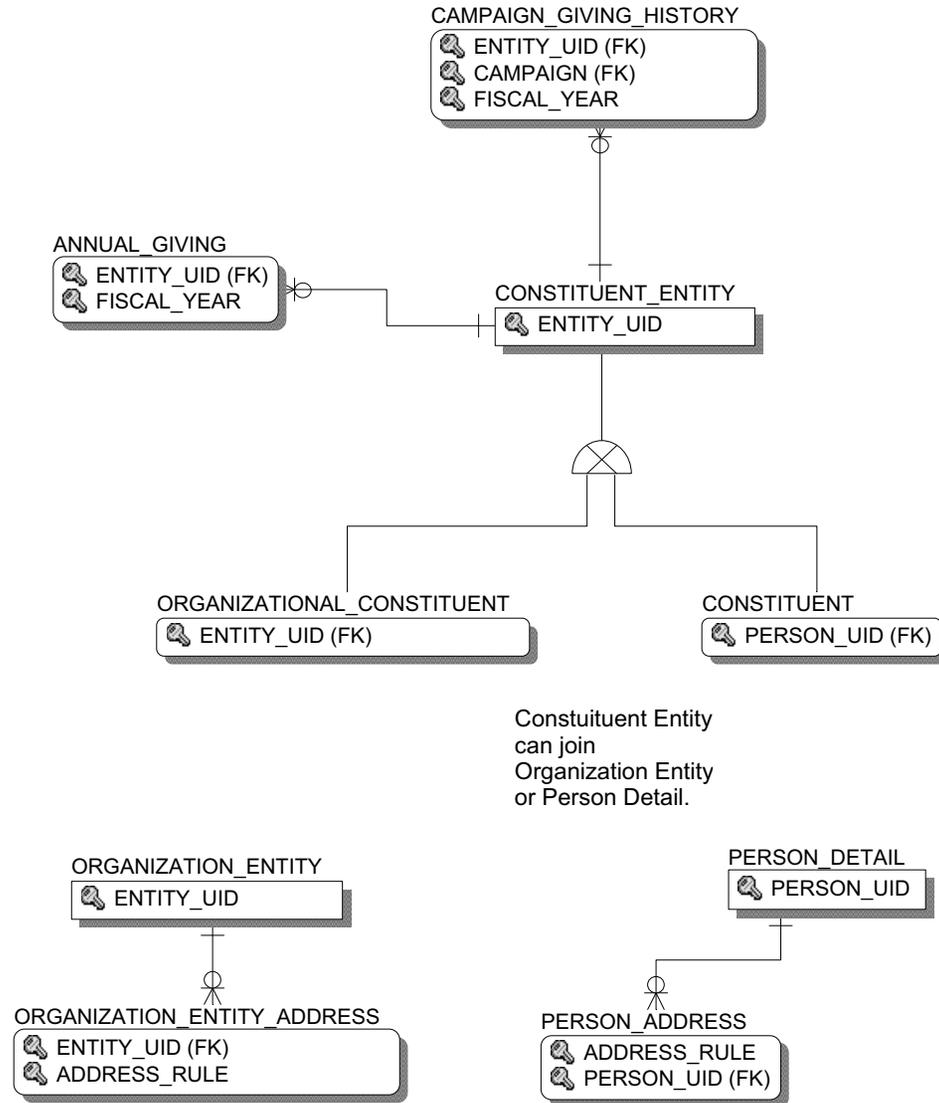
# Annual Giving



# Campaign Management

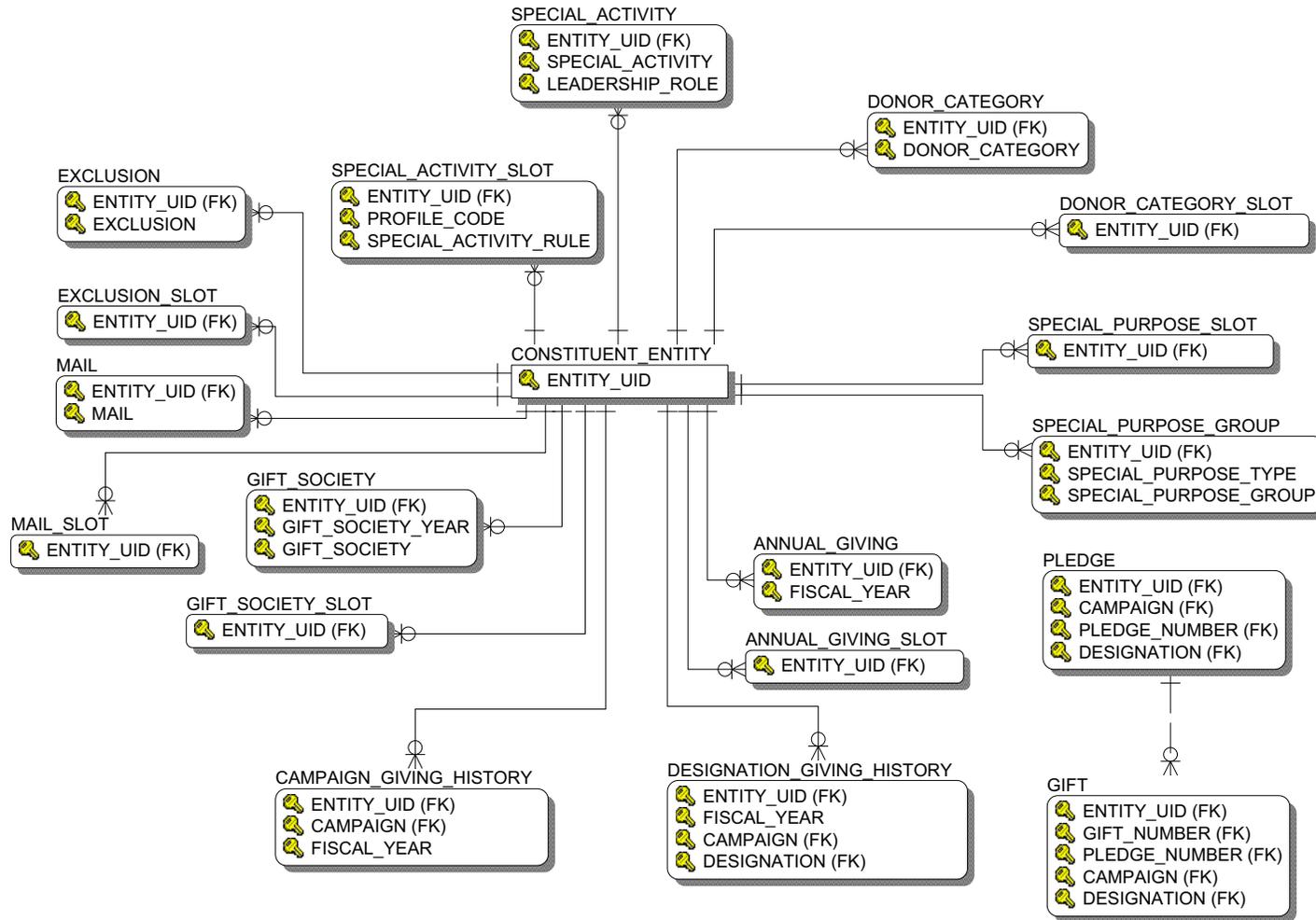


# Campaign Giving History

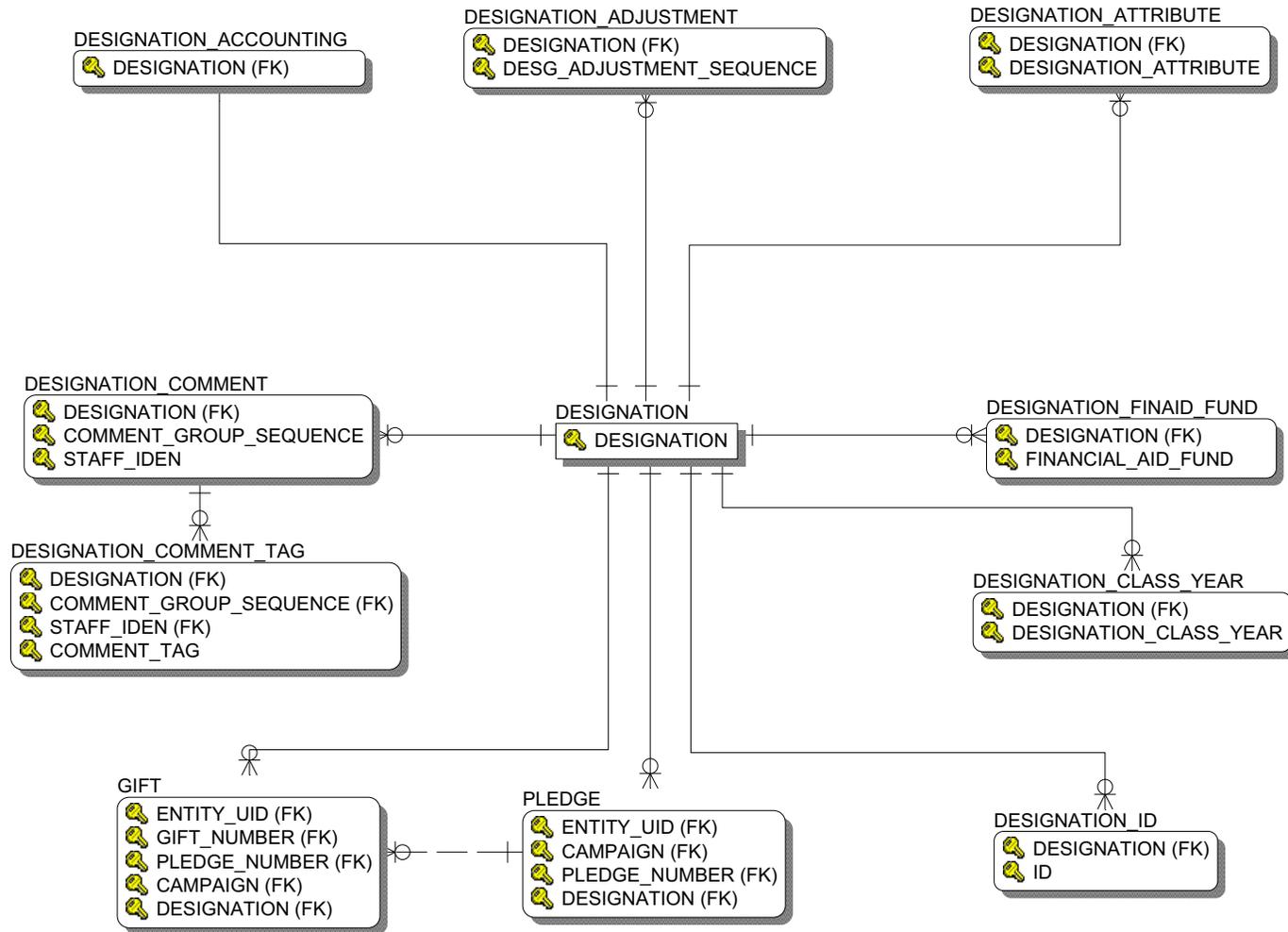




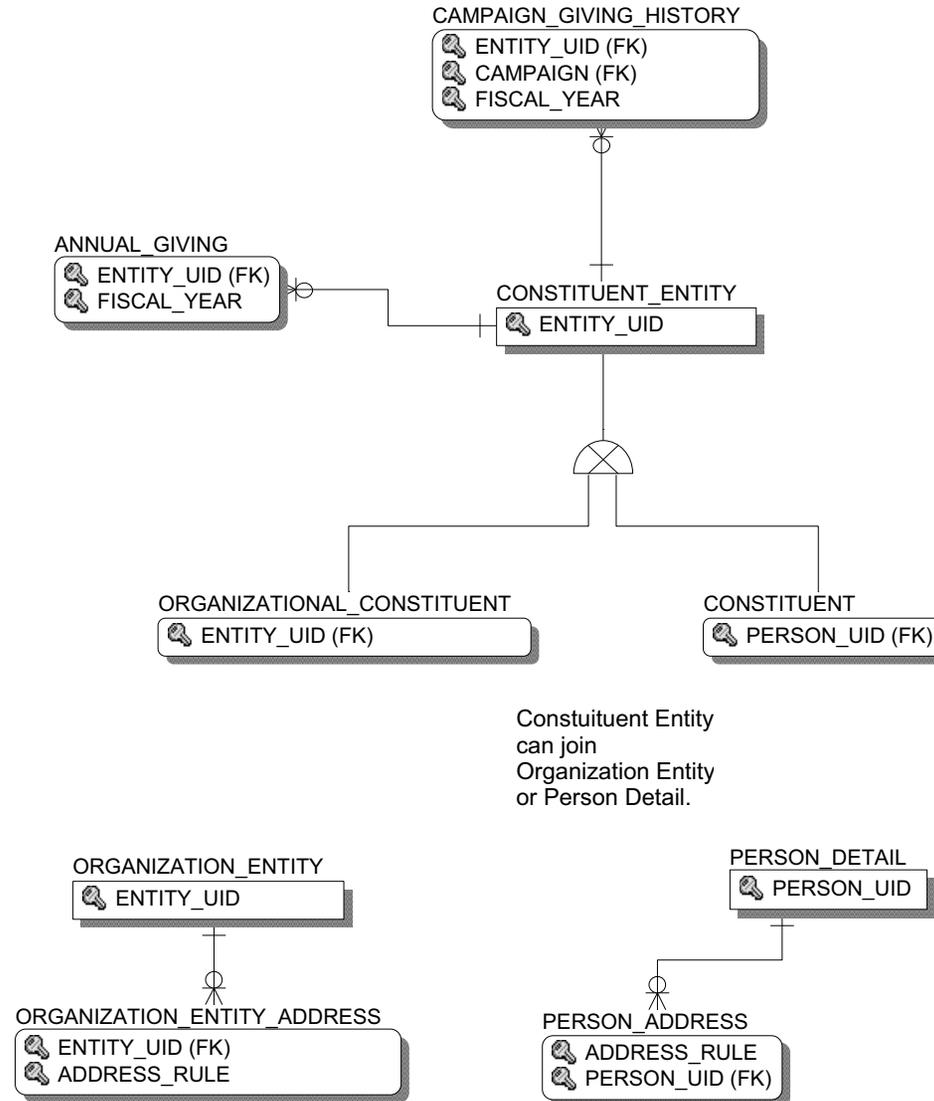
# Constituent Entity



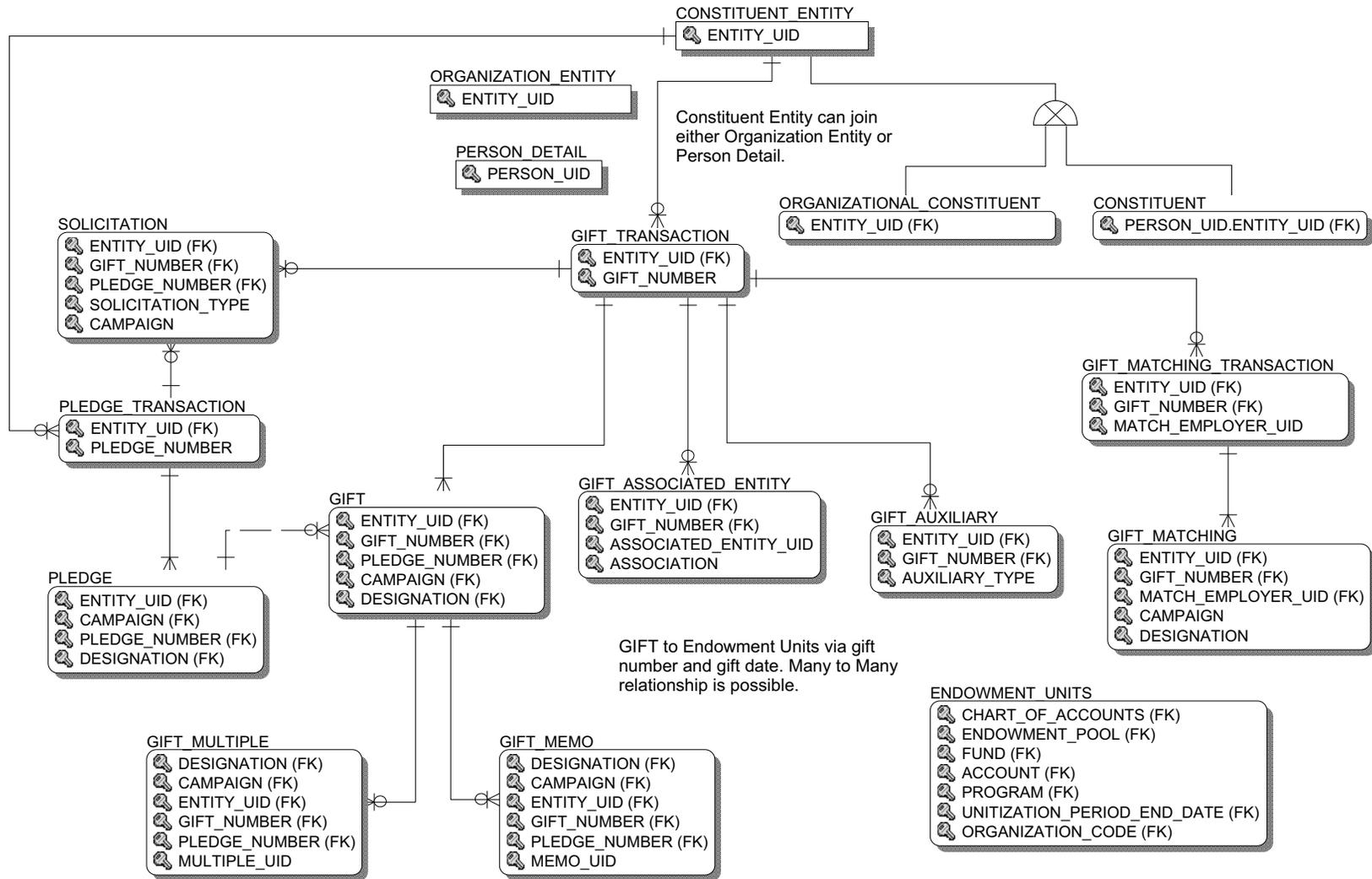
# Designation



## Designation Giving History

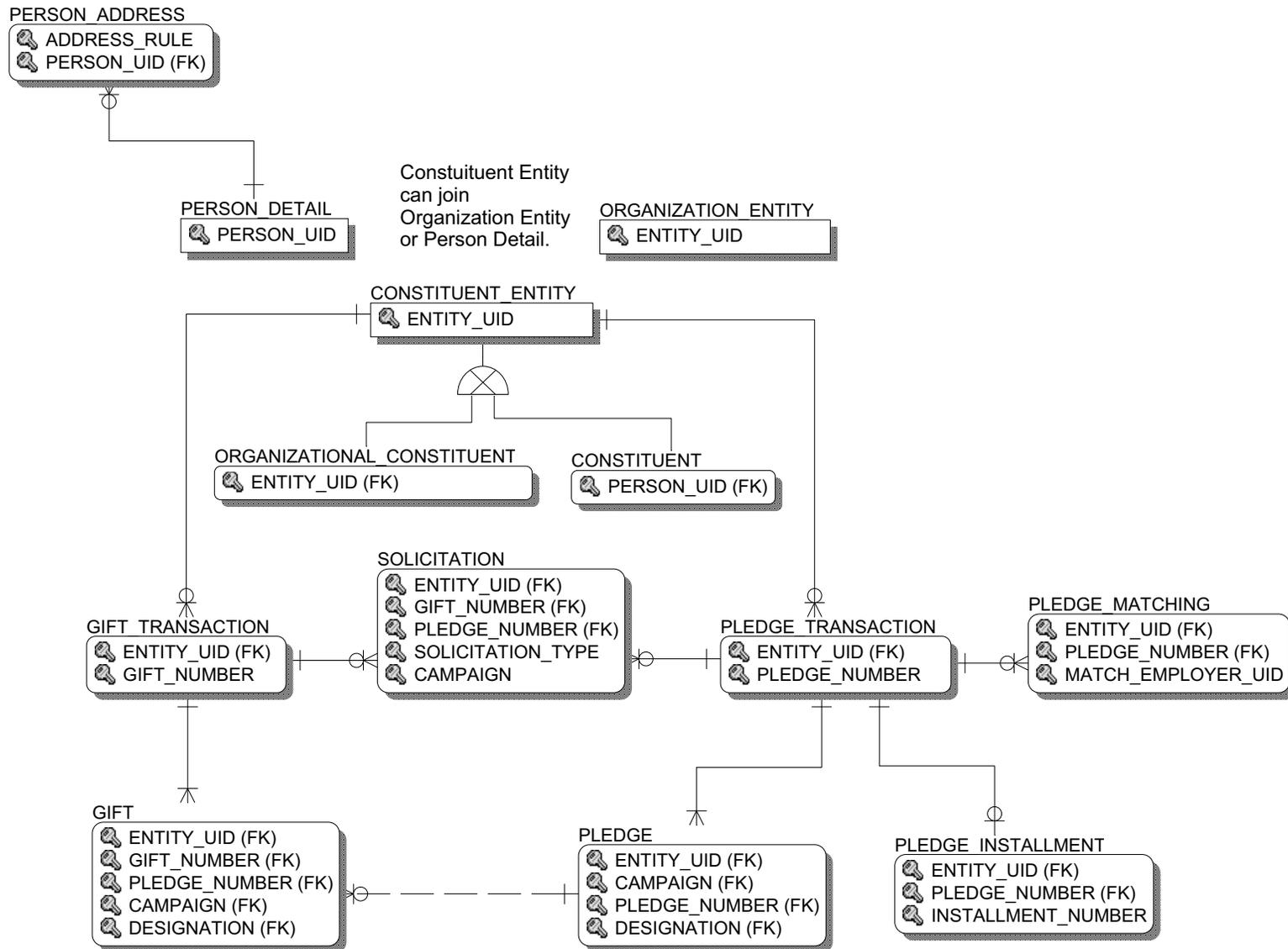


# Gift

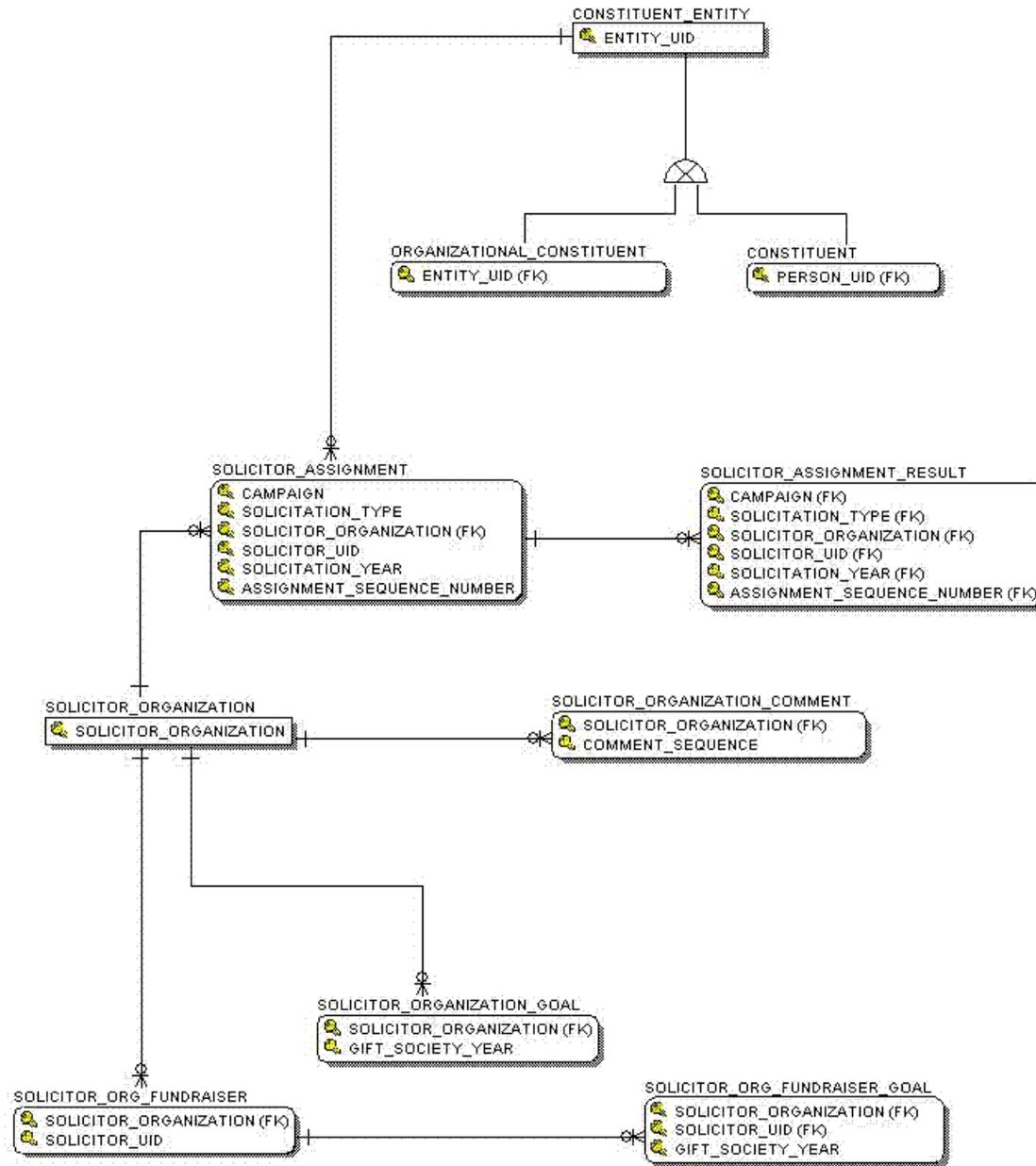




# Pledge

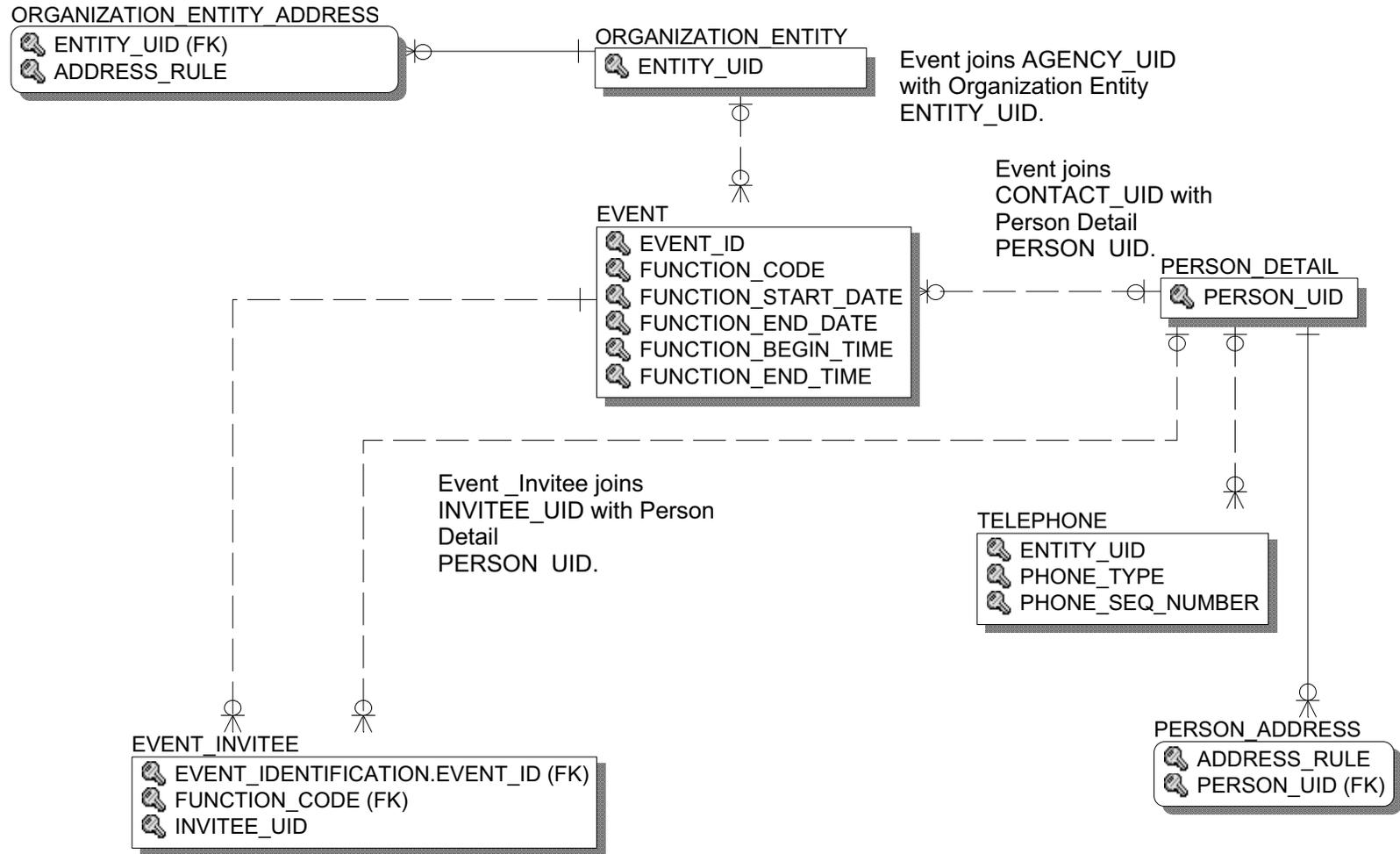


## Solicitation Effort

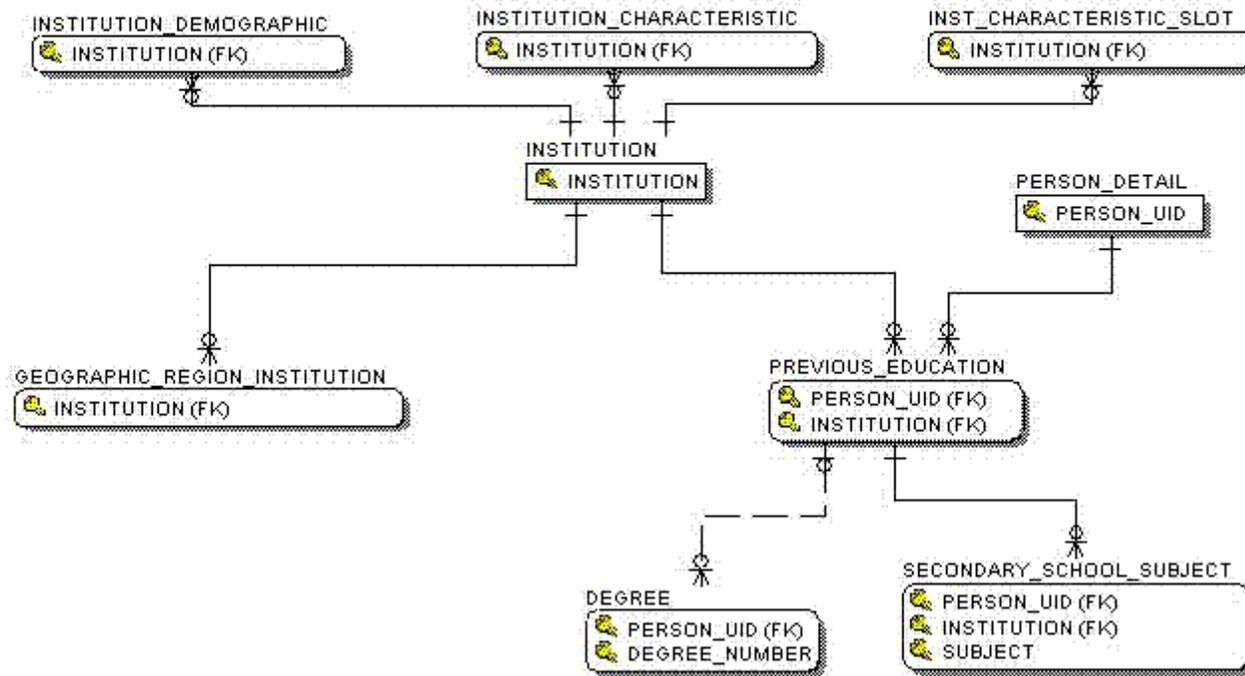


# Common

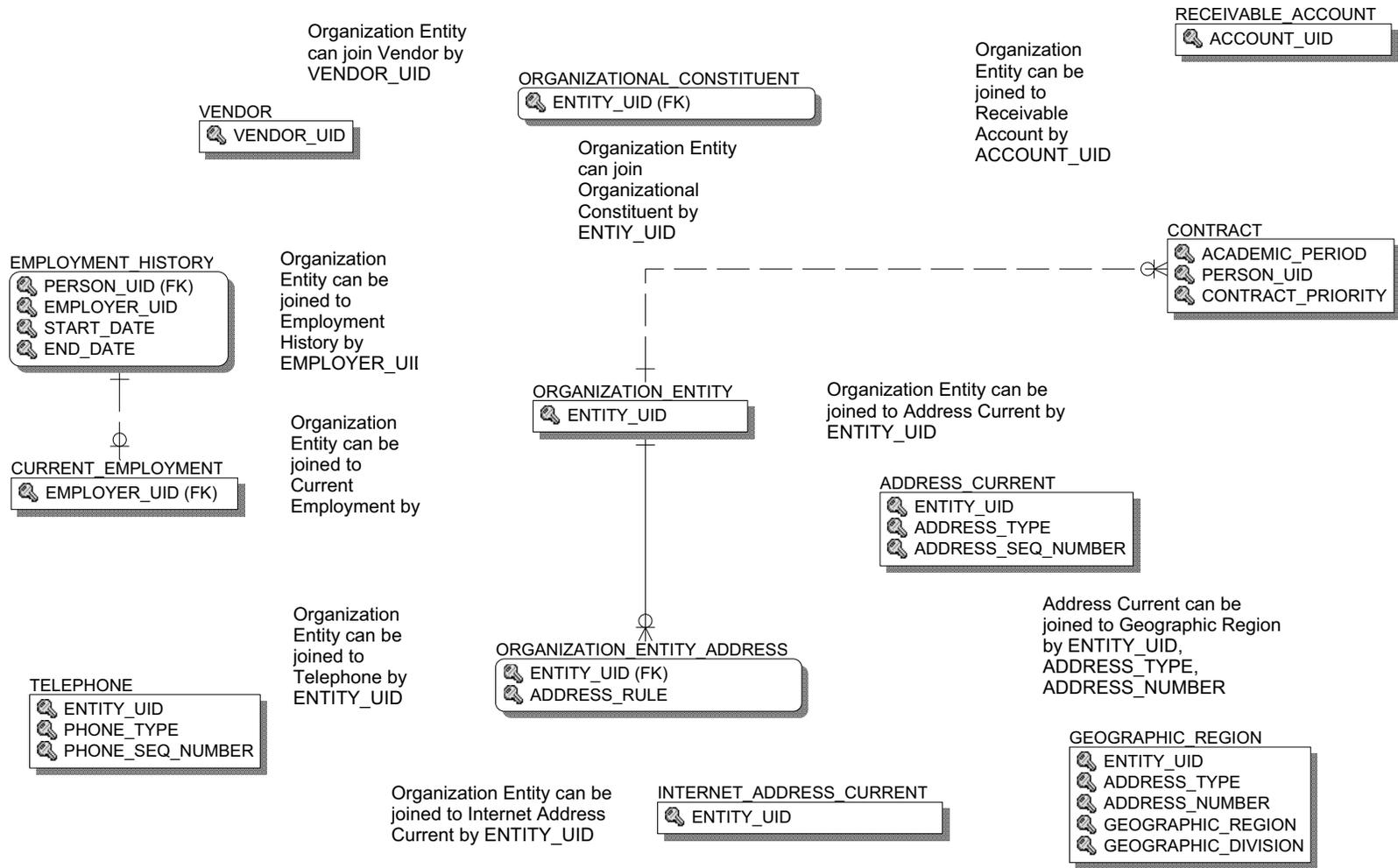
## Event



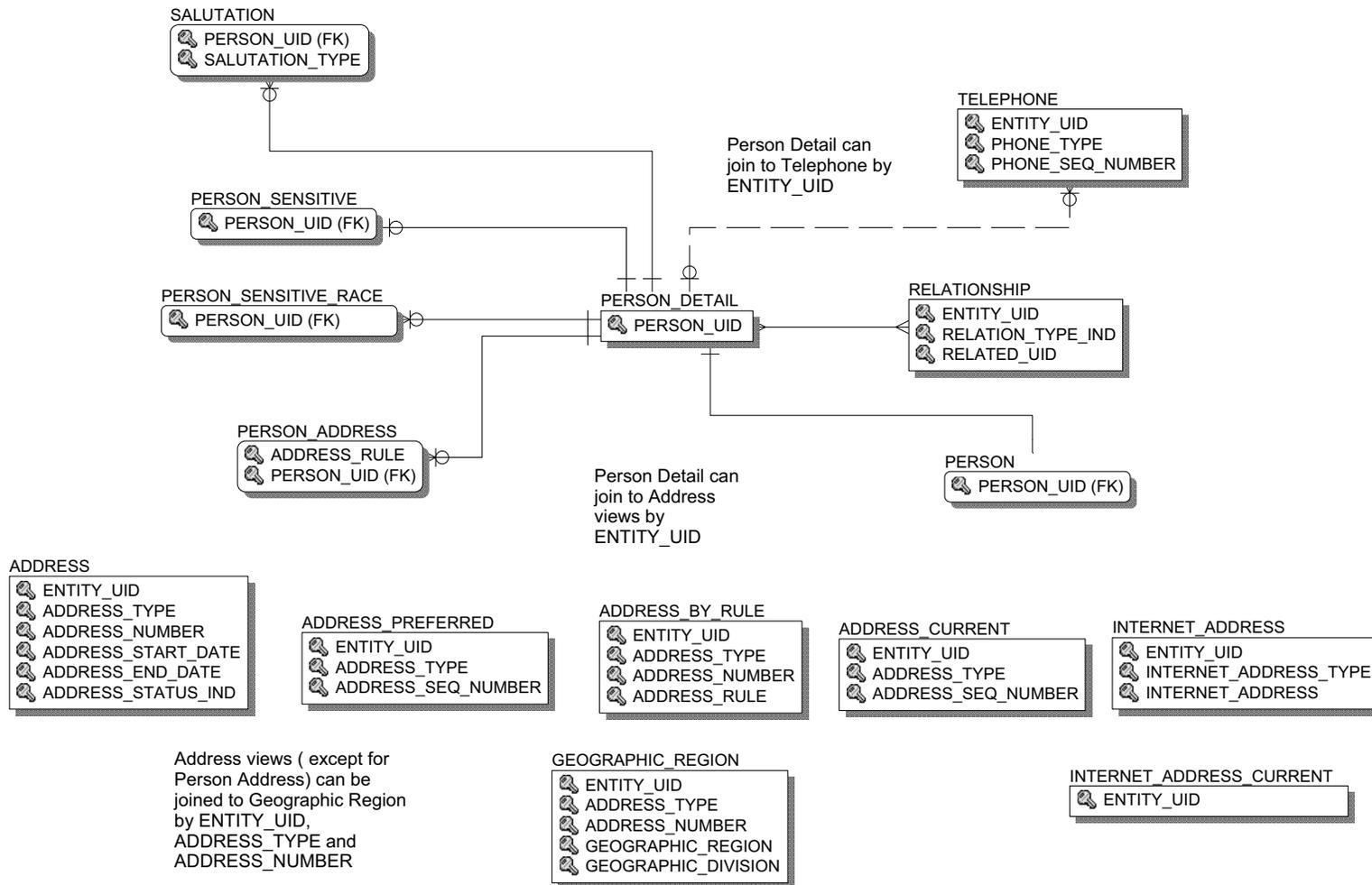
# Institution



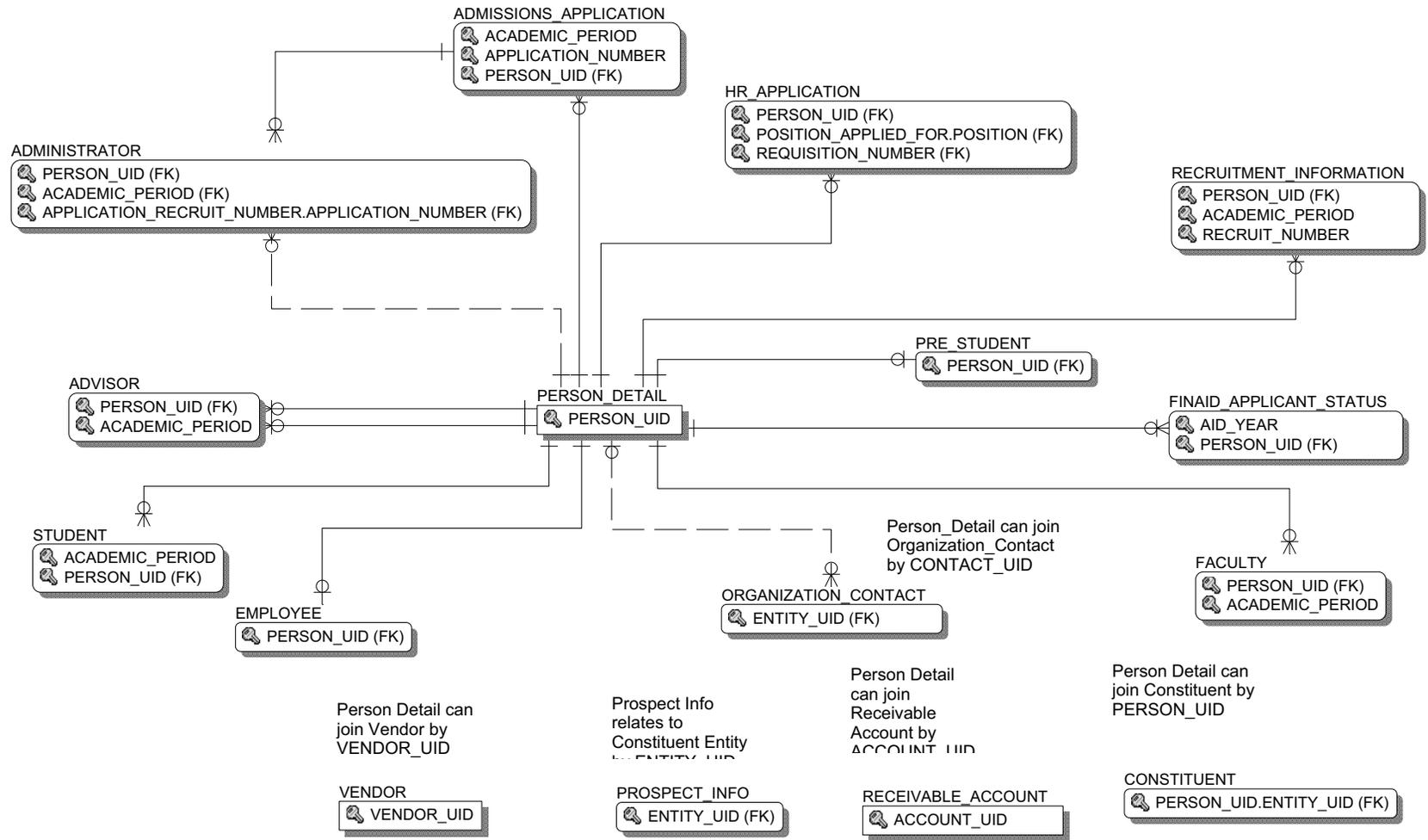
# Organization Entity



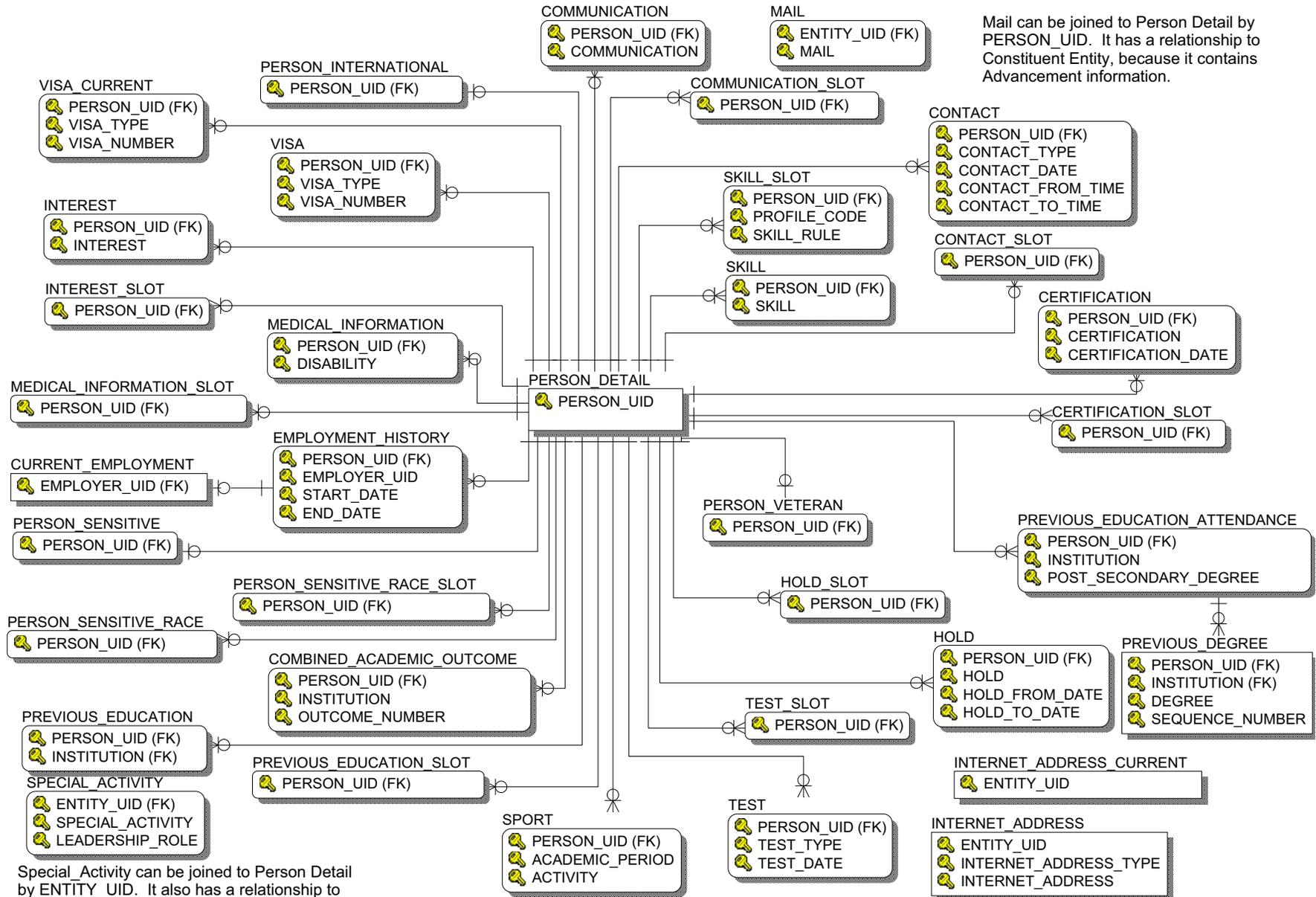
# Person Demographic



# Person Role



# Person Supplemental



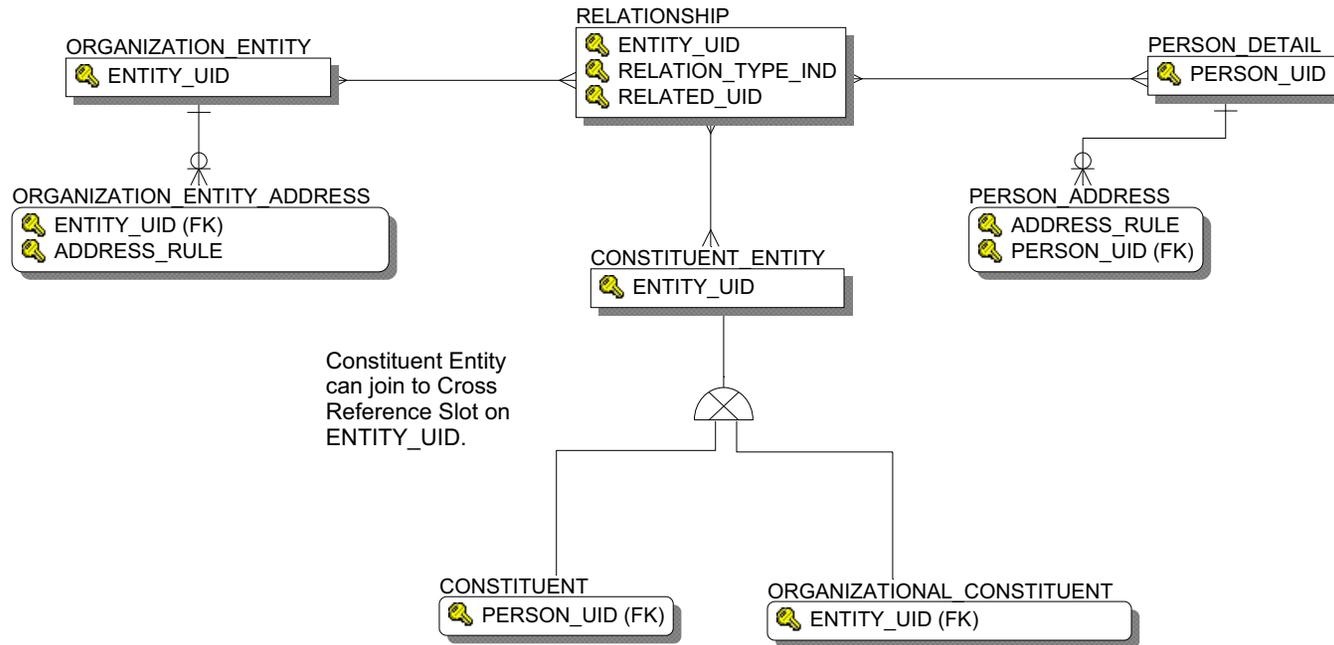
Mail can be joined to Person Detail by PERSON\_UID. It has a relationship to Constituent Entity, because it contains Advancement information.

Special\_Activity can be joined to Person Detail by ENTITY\_UID. It also has a relationship to Organizational\_Constituent by ENTITY\_UID.

Internet Address and Internet Address Current can be joined to Person Detail by ENTITY\_UID.

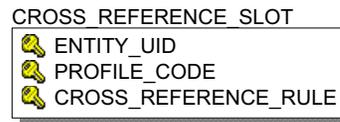
# Relationship

## Relationships using a Relationship Type



## Relationships using a Slotted View

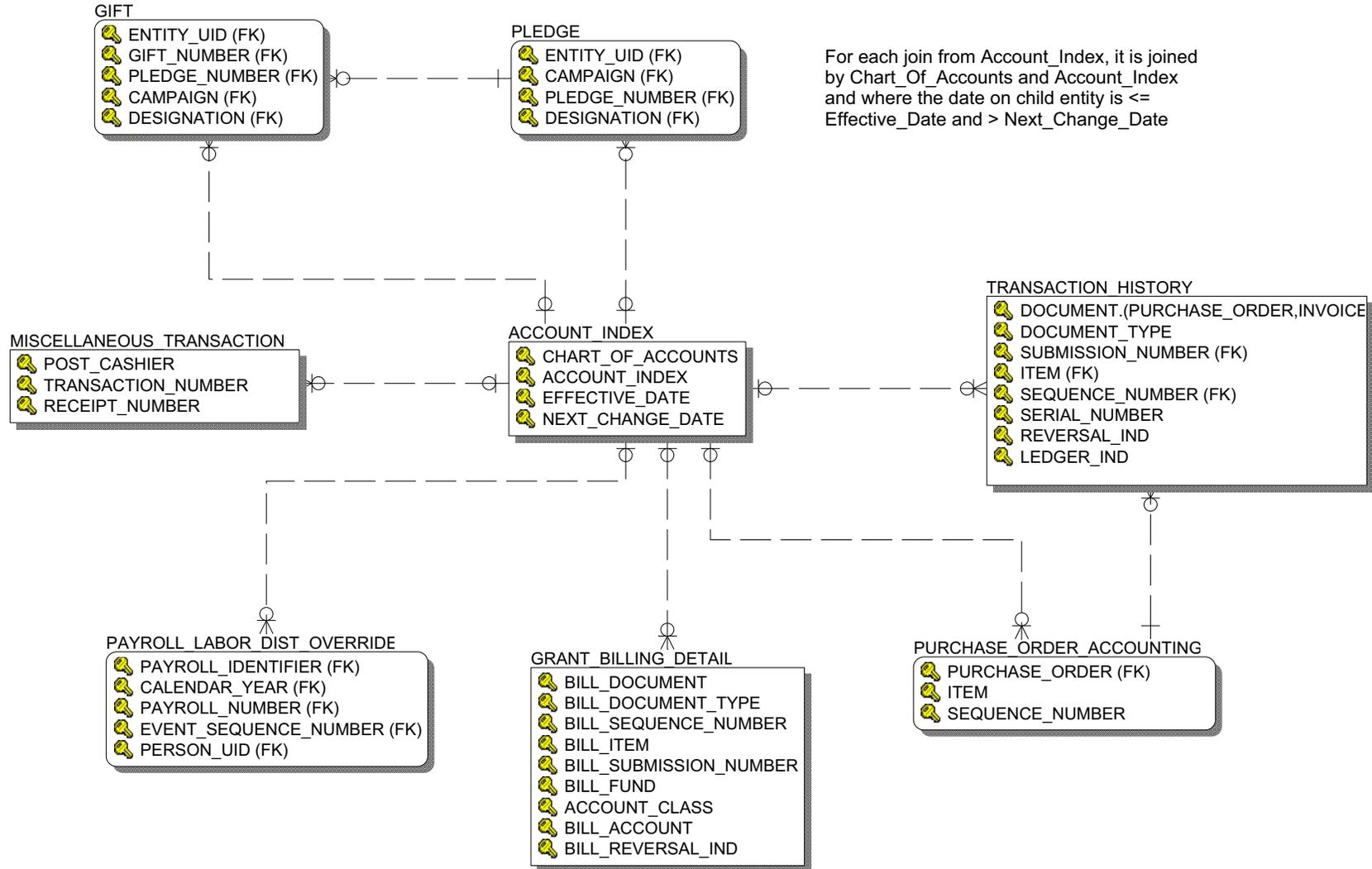
Organization Entity can join to Cross Reference Slot on ENTITY\_UID.



Person Detail can join to Cross Reference Slot on PERSON\_UID.

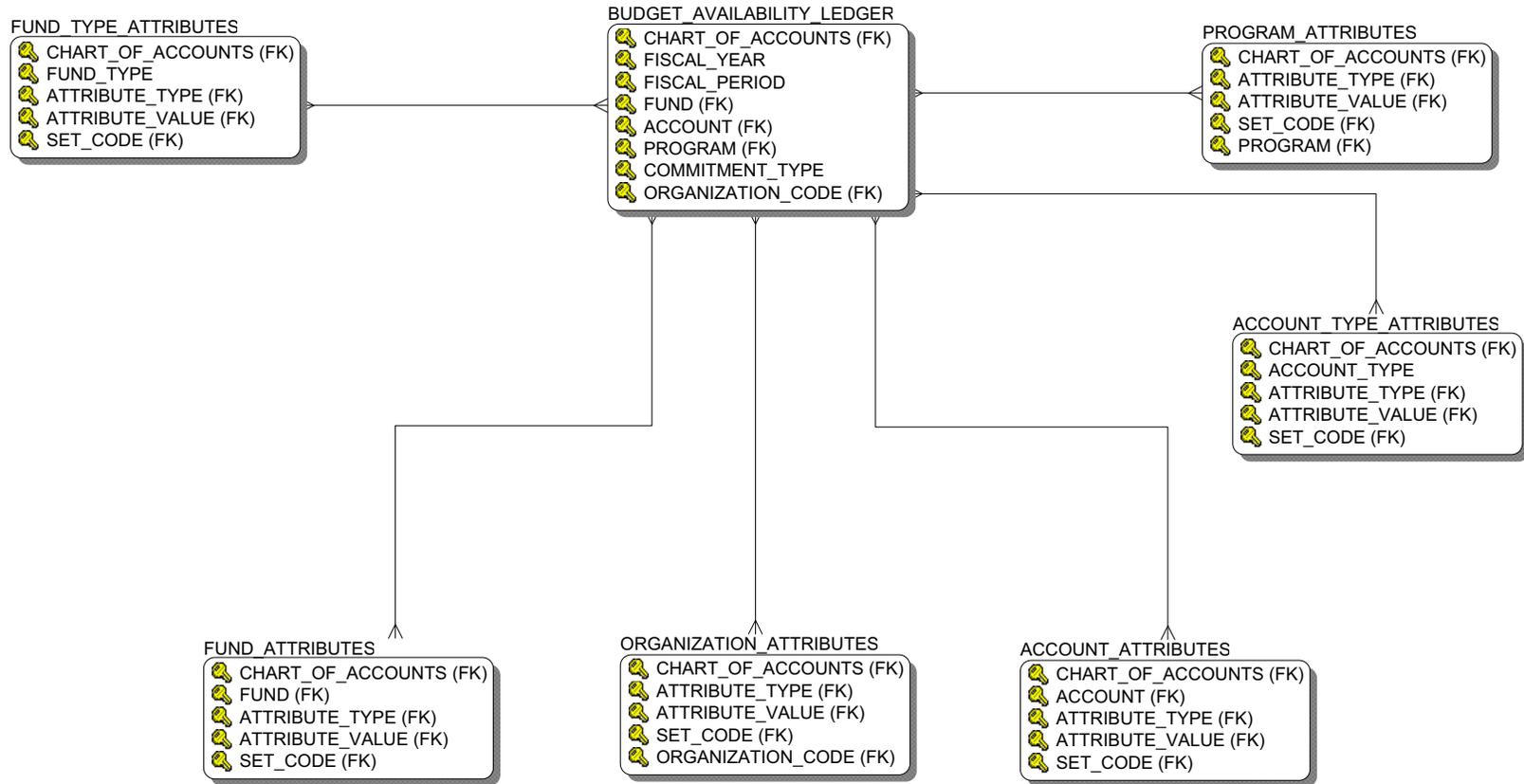
# Finance

## Account Index Audit

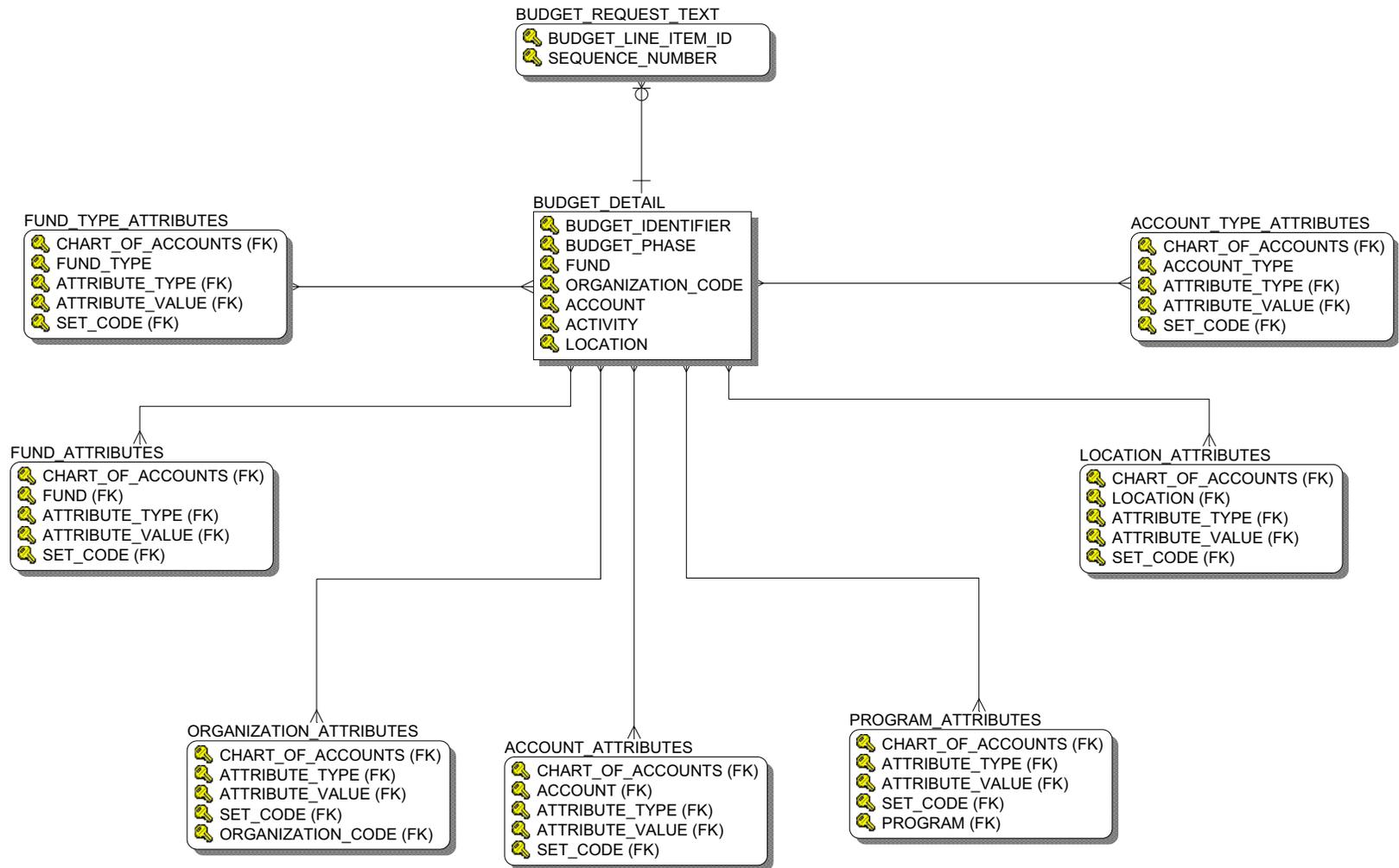


For each join from Account\_Index, it is joined by Chart\_Of\_Accounts and Account\_Index and where the date on child entity is <= Effective\_Date and > Next\_Change\_Date

# Budget Availability Ledger

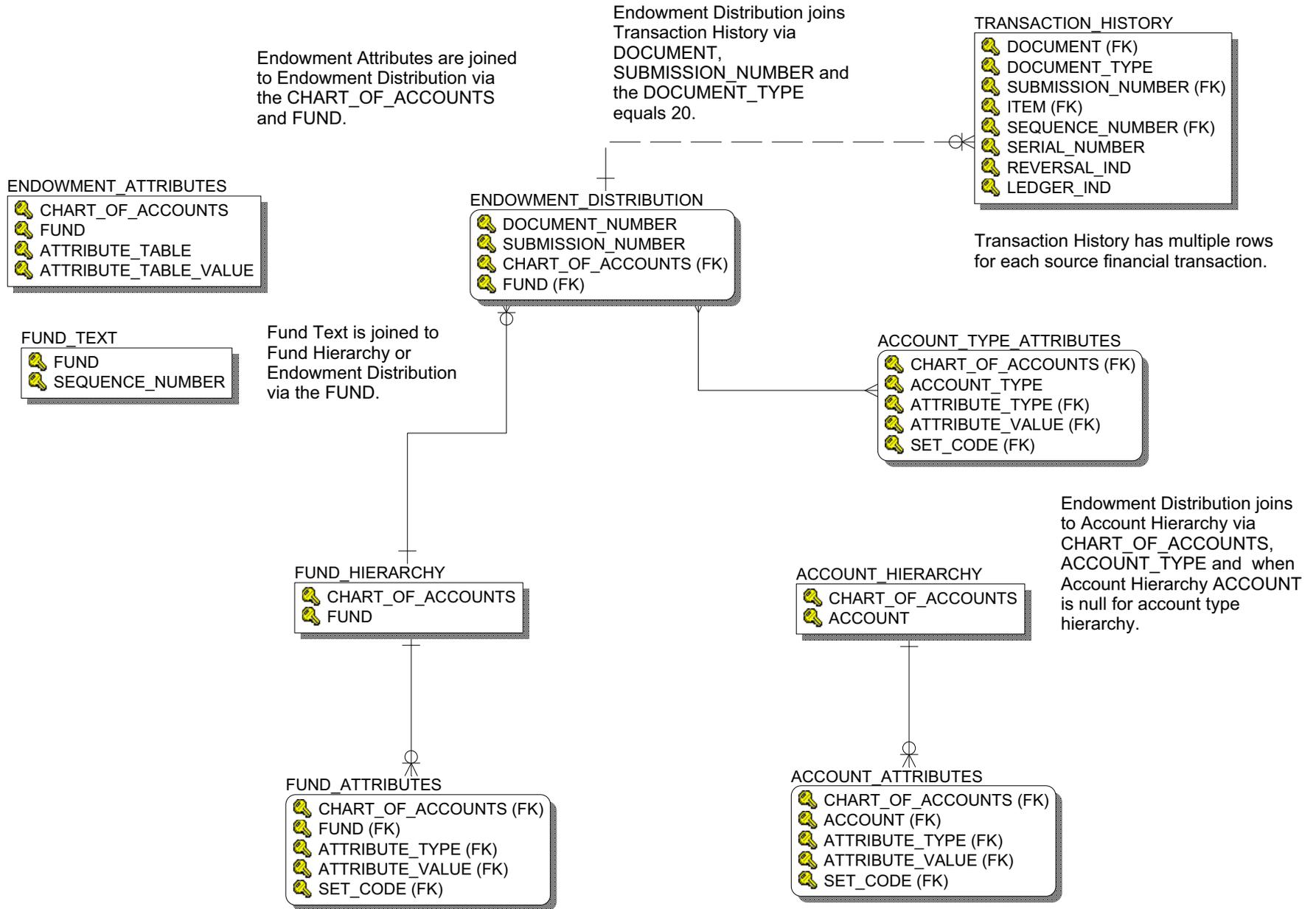


# Budget Detail

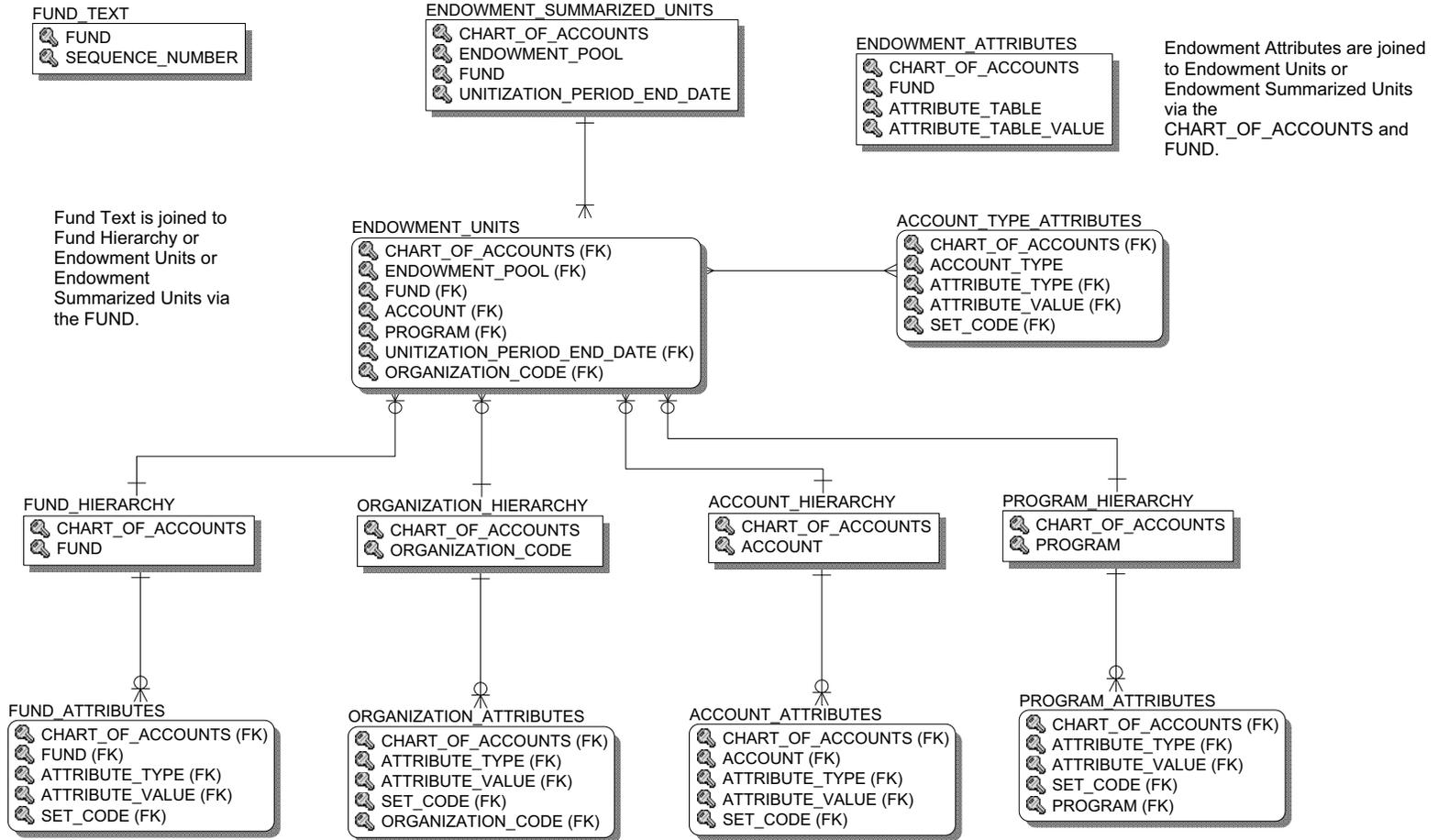




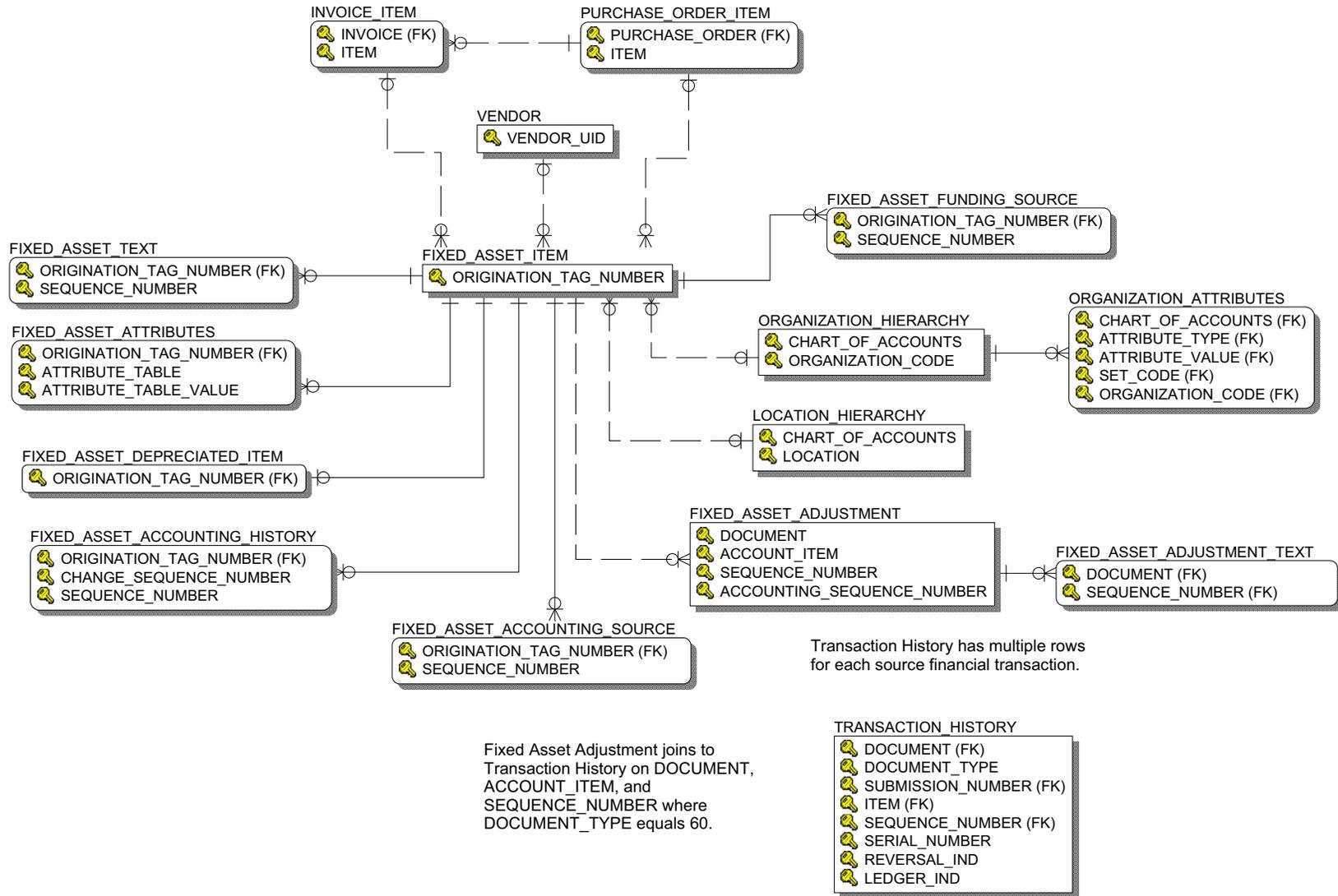
# Endowment Distribution



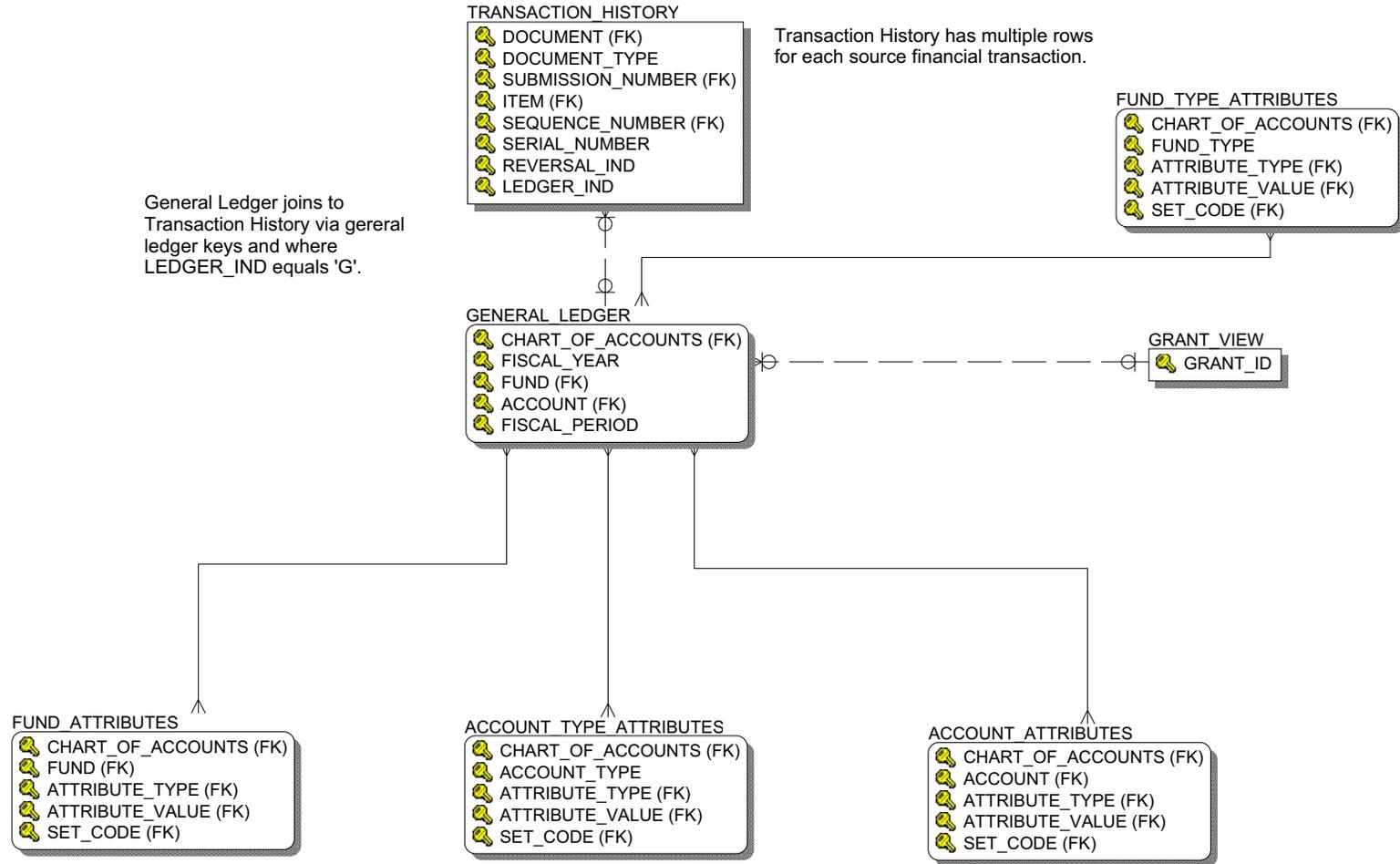
# Endowment Units



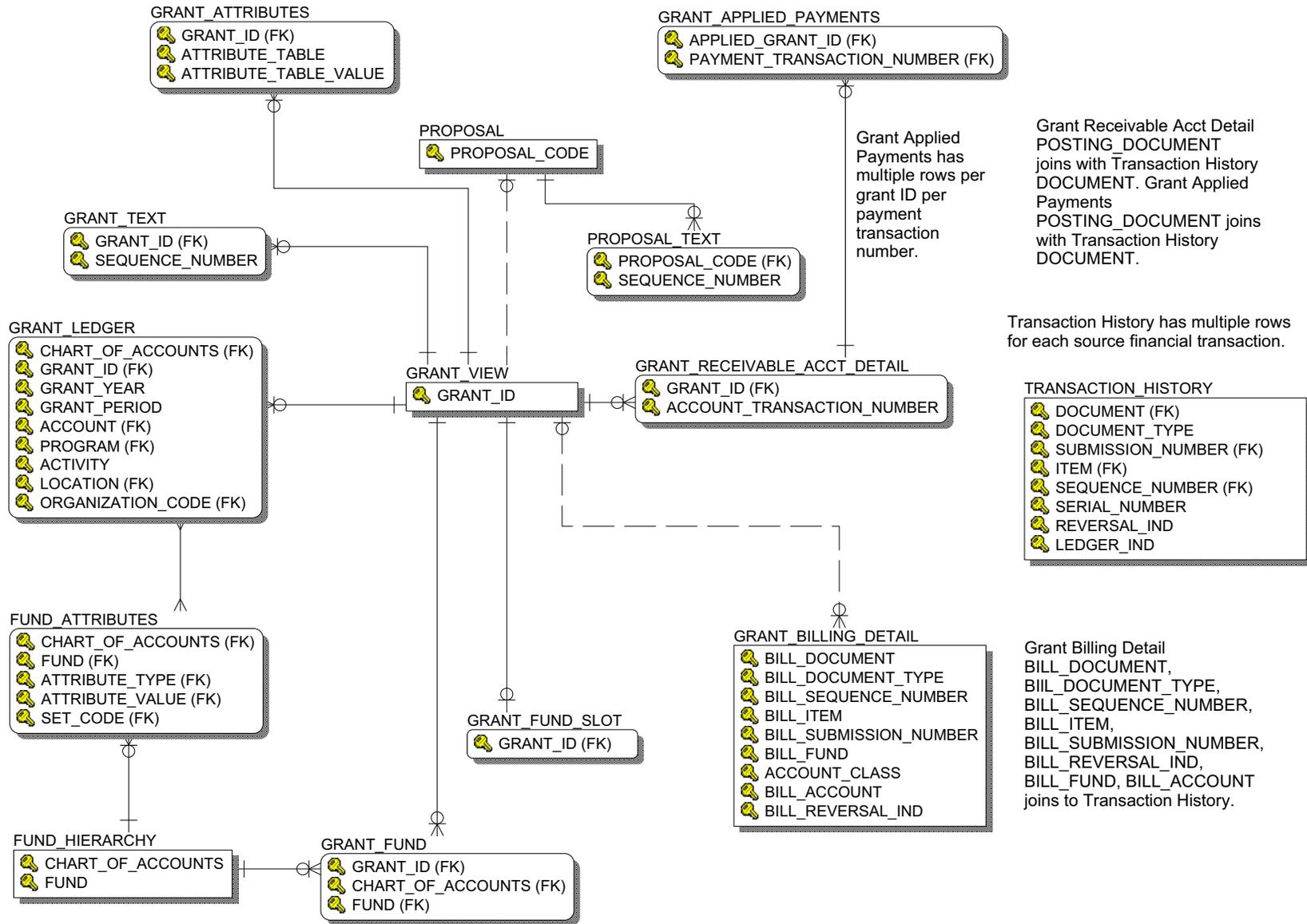
# Fixed Asset



# General Ledger



# Grant and Project



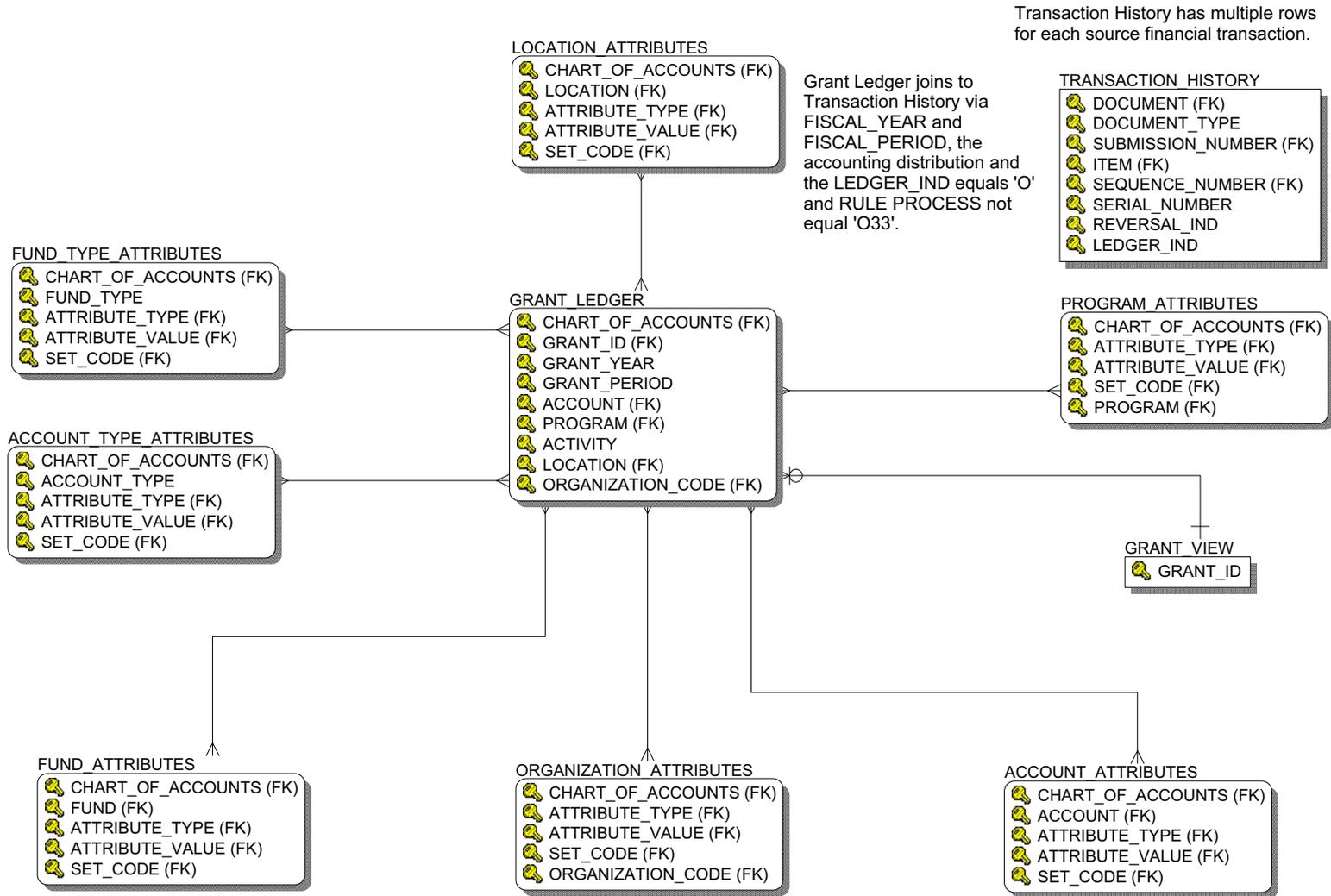
Grant Receivable Acct Detail POSTING\_DOCUMENT joins with Transaction History DOCUMENT. Grant Applied Payments POSTING\_DOCUMENT joins with Transaction History DOCUMENT.

Transaction History has multiple rows for each source financial transaction.

GRANT\_RECEIVABLE\_ACCT\_DETAIL joins with TRANSACTION\_HISTORY on GRANT\_ID and ACCOUNT\_TRANSACTION\_NUMBER.

GRANT\_BILLING\_DETAIL joins with TRANSACTION\_HISTORY on BILL\_DOCUMENT, BILL\_DOCUMENT\_TYPE, BILL\_SEQUENCE\_NUMBER, BILL\_ITEM, BILL\_SUBMISSION\_NUMBER, BILL\_FUND, BILL\_REVERSAL\_IND, BILL\_ACCOUNT, and BILL\_REVERSAL\_IND.

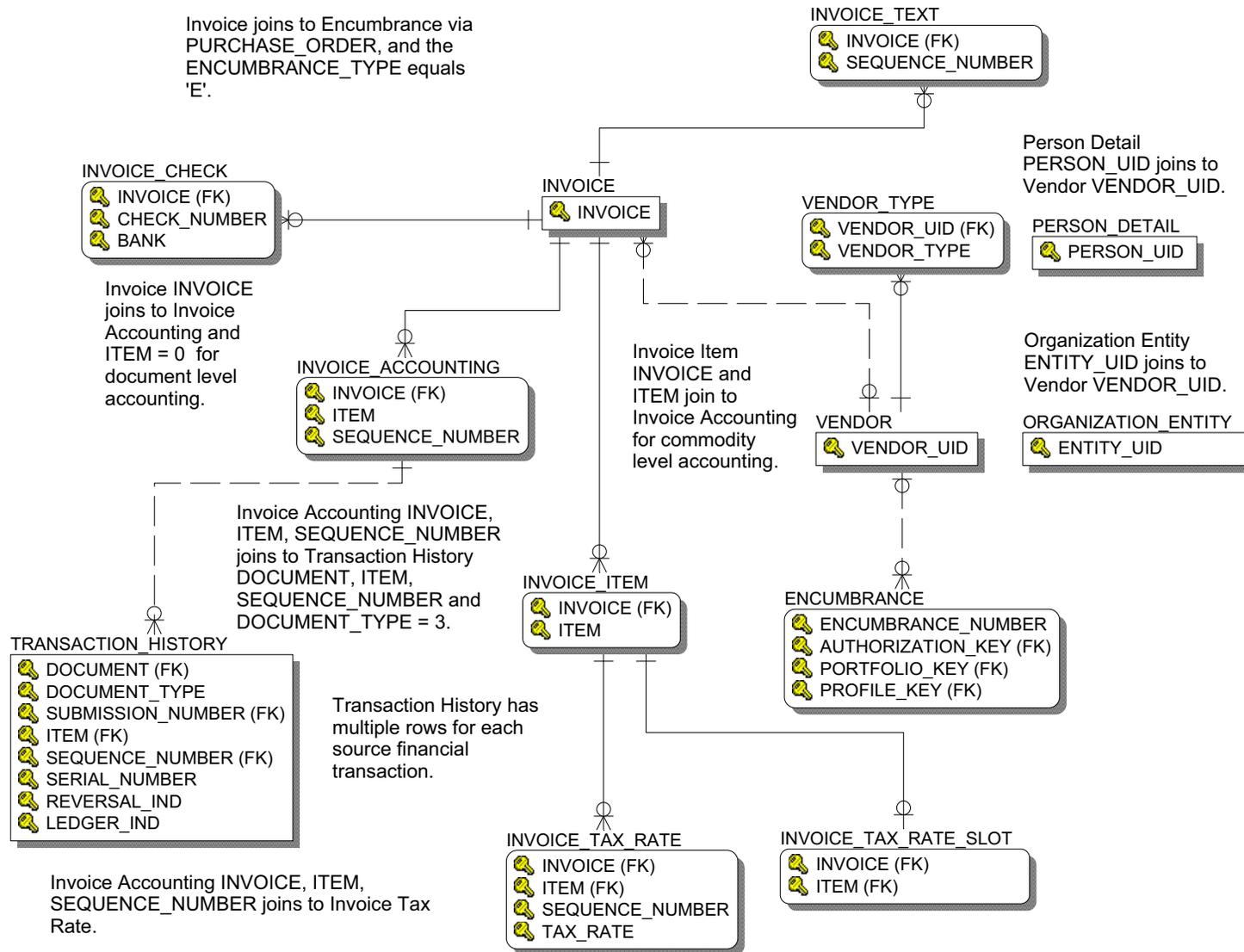
# Grant Ledger



Transaction History has multiple rows for each source financial transaction.

Grant Ledger joins to Transaction History via FISCAL\_YEAR and FISCAL\_PERIOD, the accounting distribution and the LEDGER\_IND equals 'O' and RULE\_PROCESS not equal 'O33'.

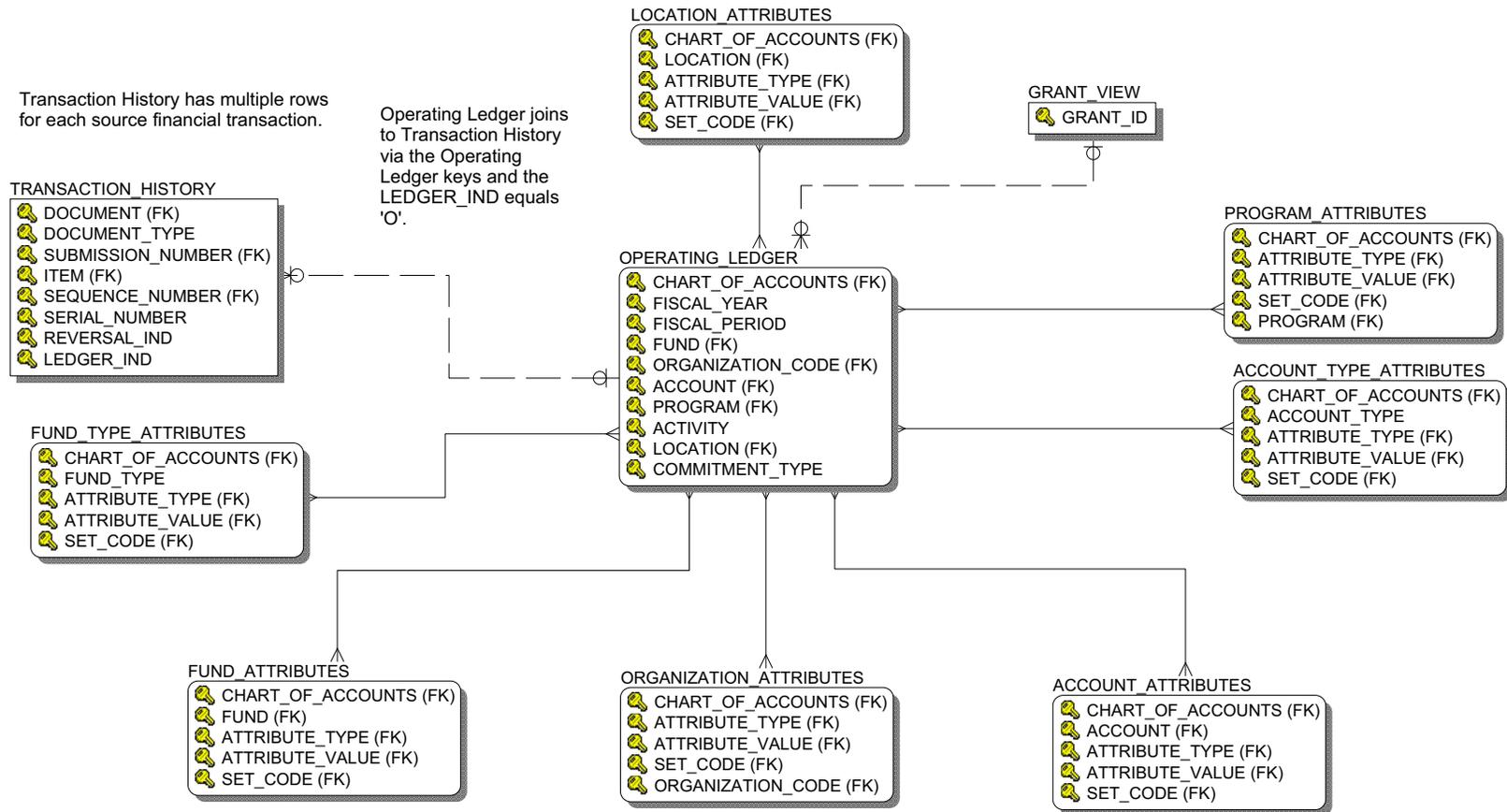
# Invoice Payable



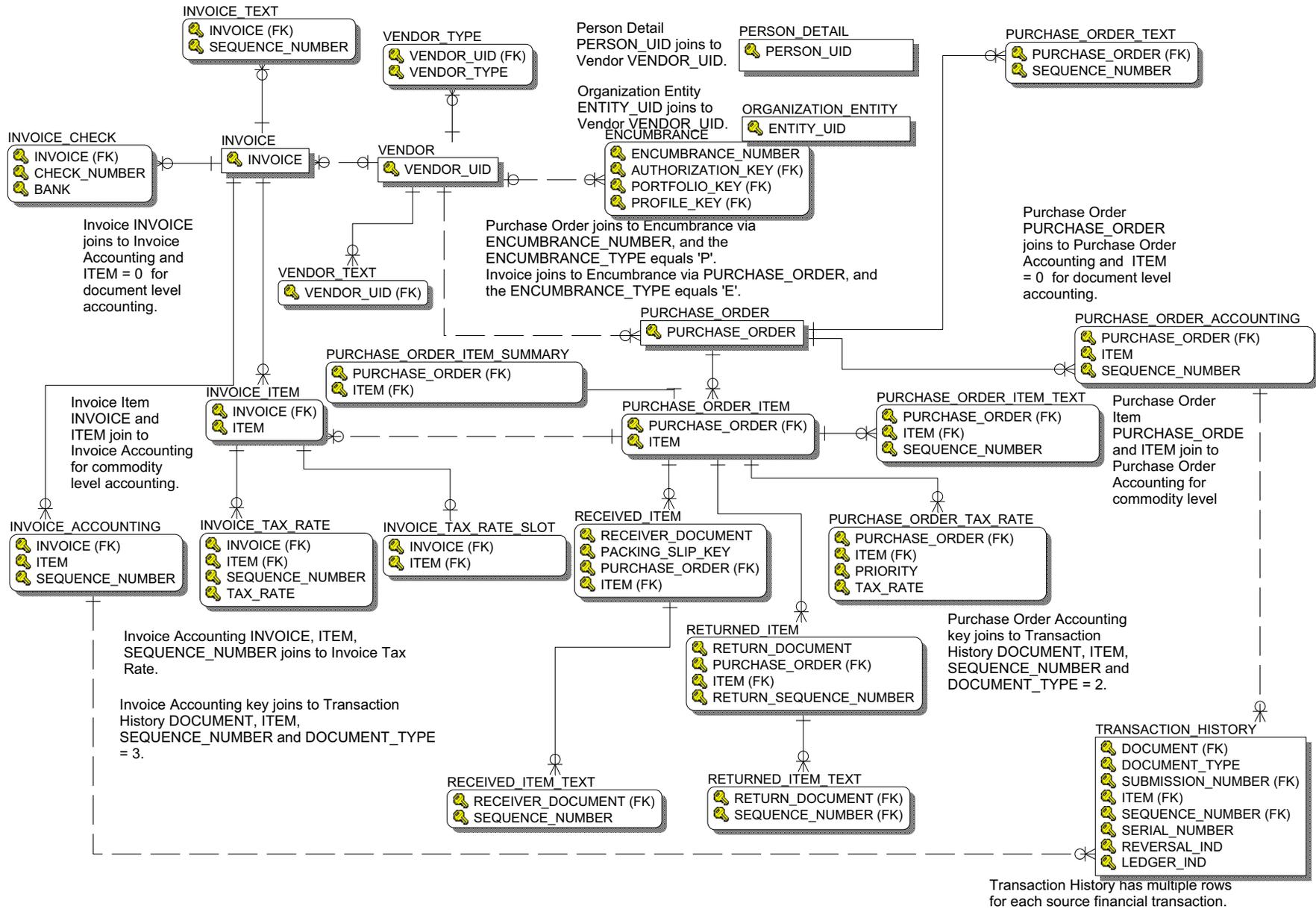
# Operating Ledger

Transaction History has multiple rows for each source financial transaction.

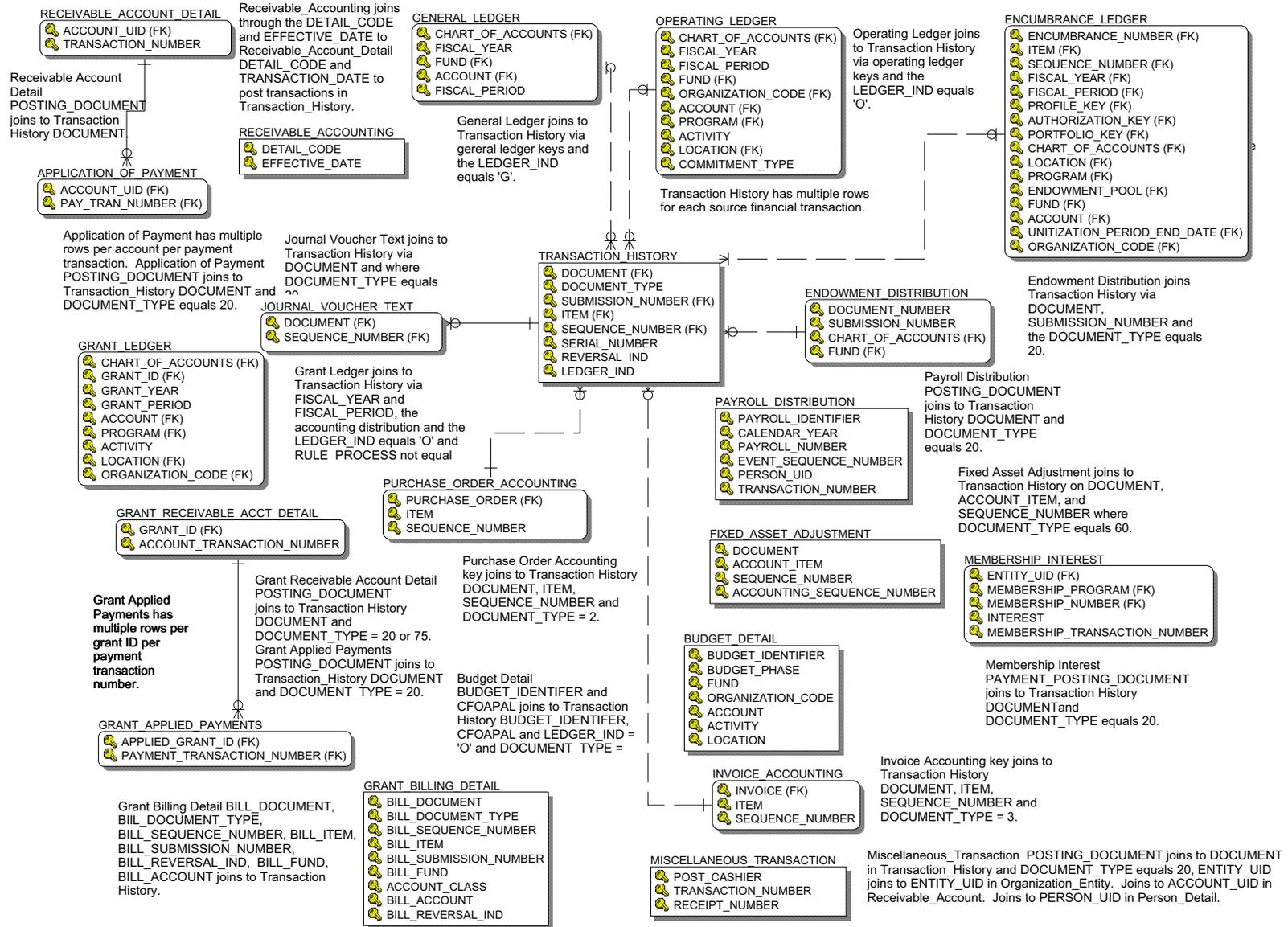
Operating Ledger joins to Transaction History via the Operating Ledger keys and the LEDGER\_IND equals 'O'.



# Purchasing Payable

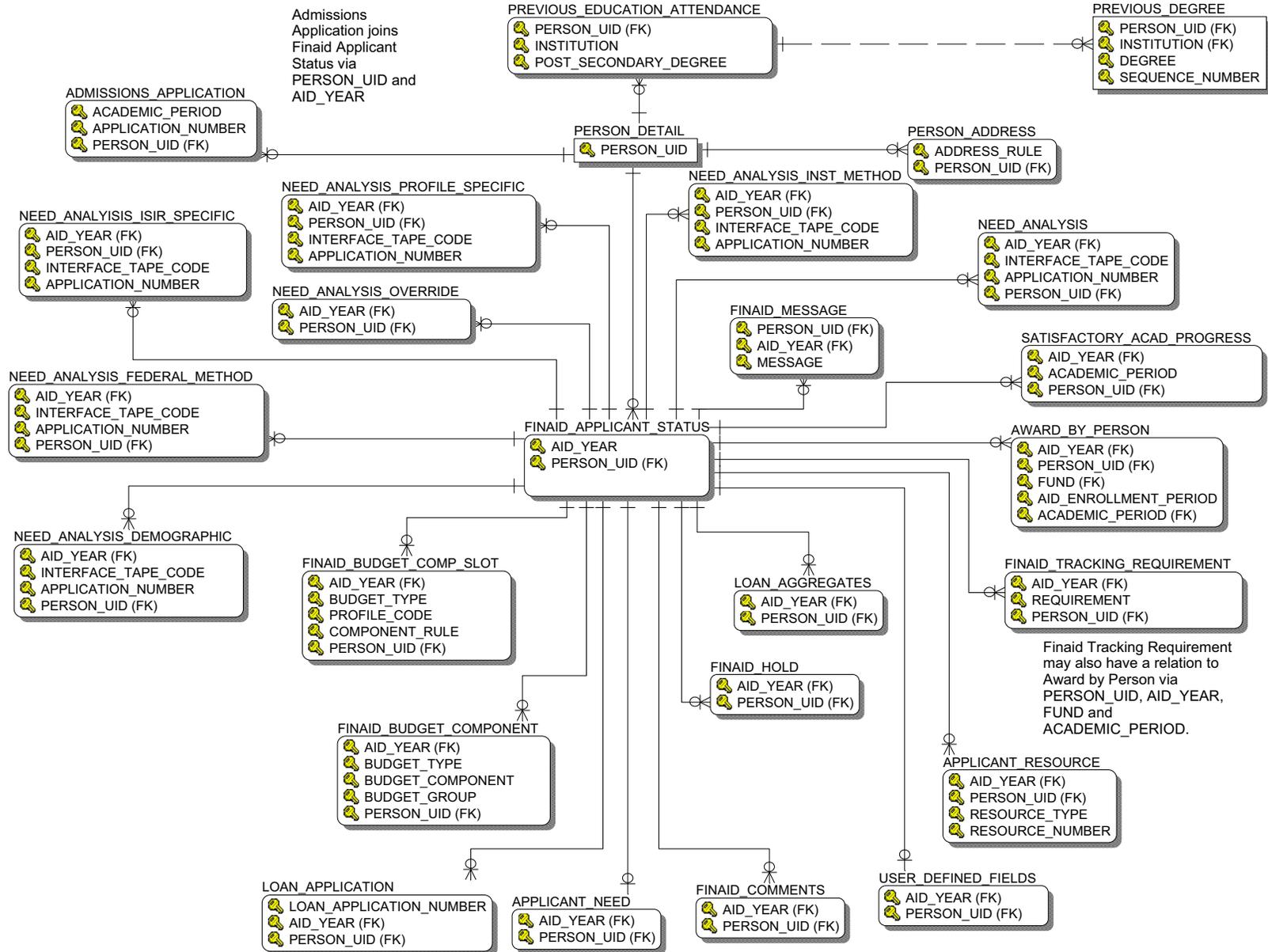


# Transaction History

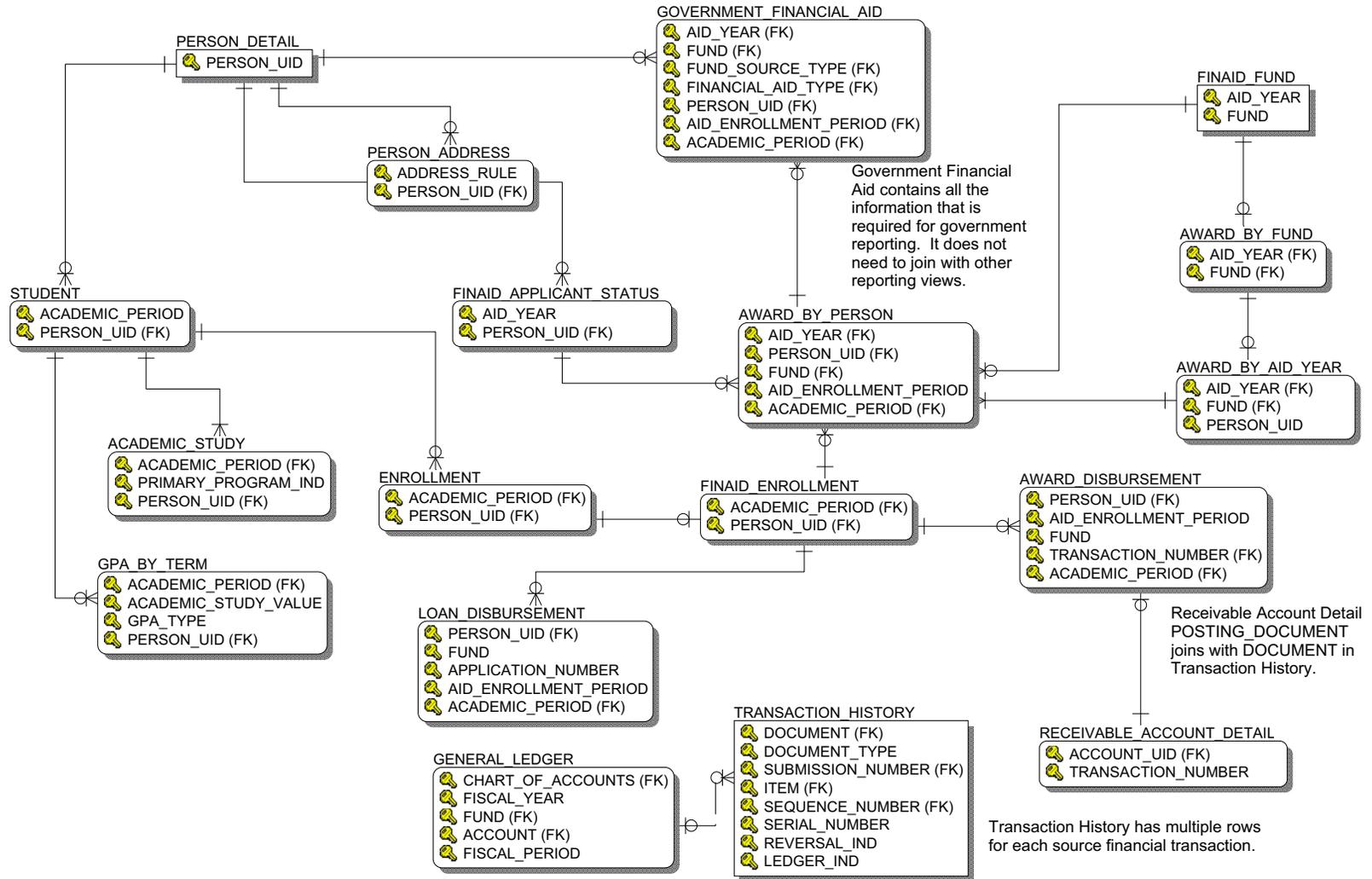


# Financial Aid

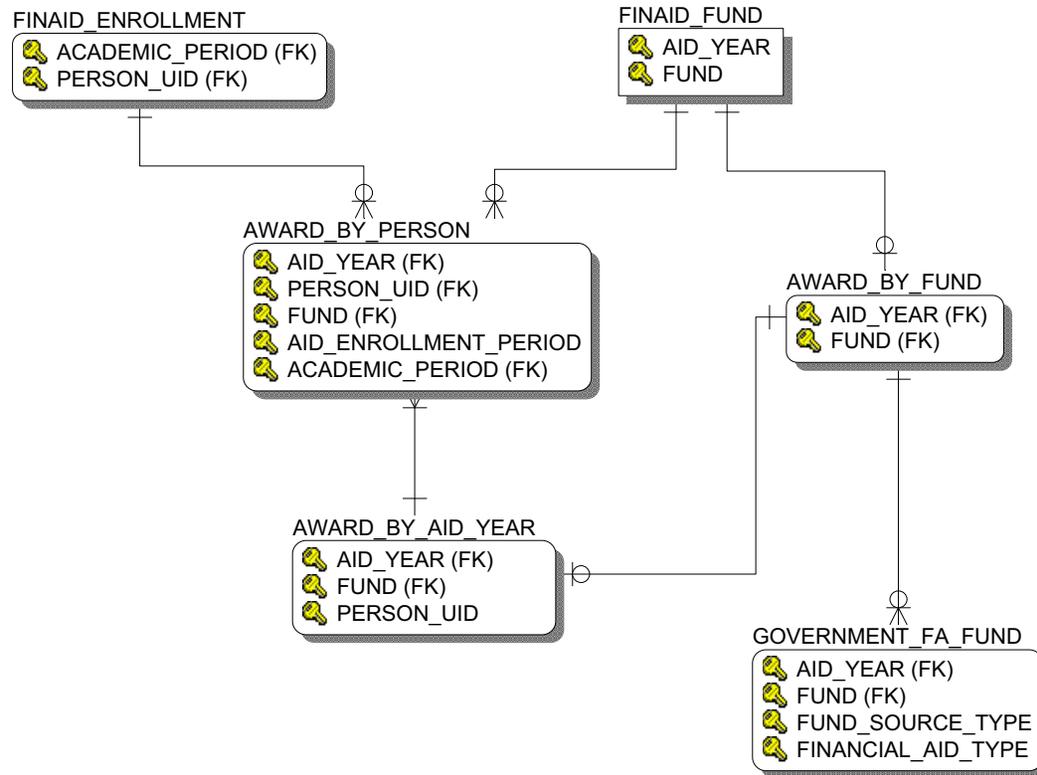
## Financial Aid Application



# Financial Aid Award and Disbursement

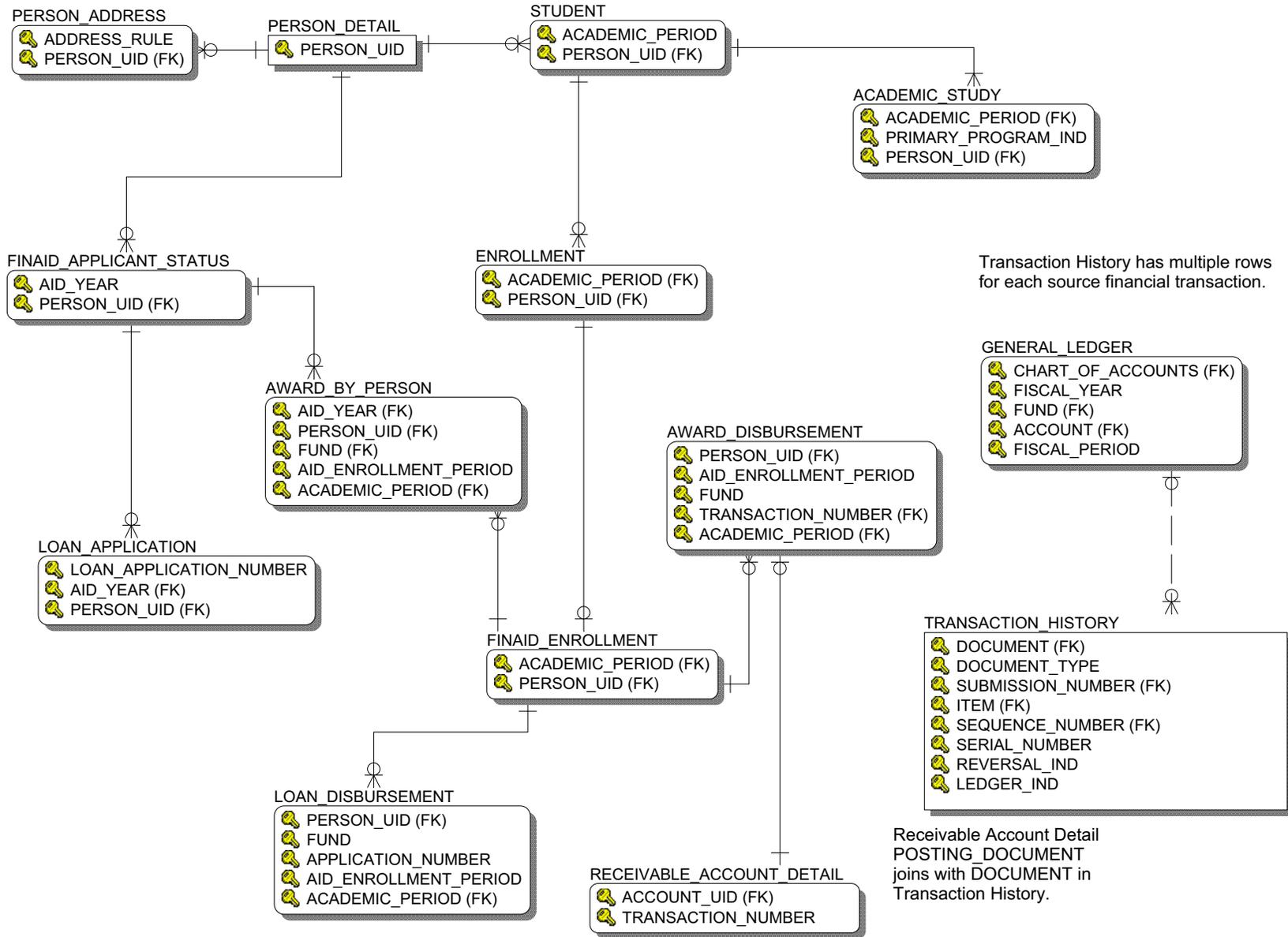


## Financial Aid Fund



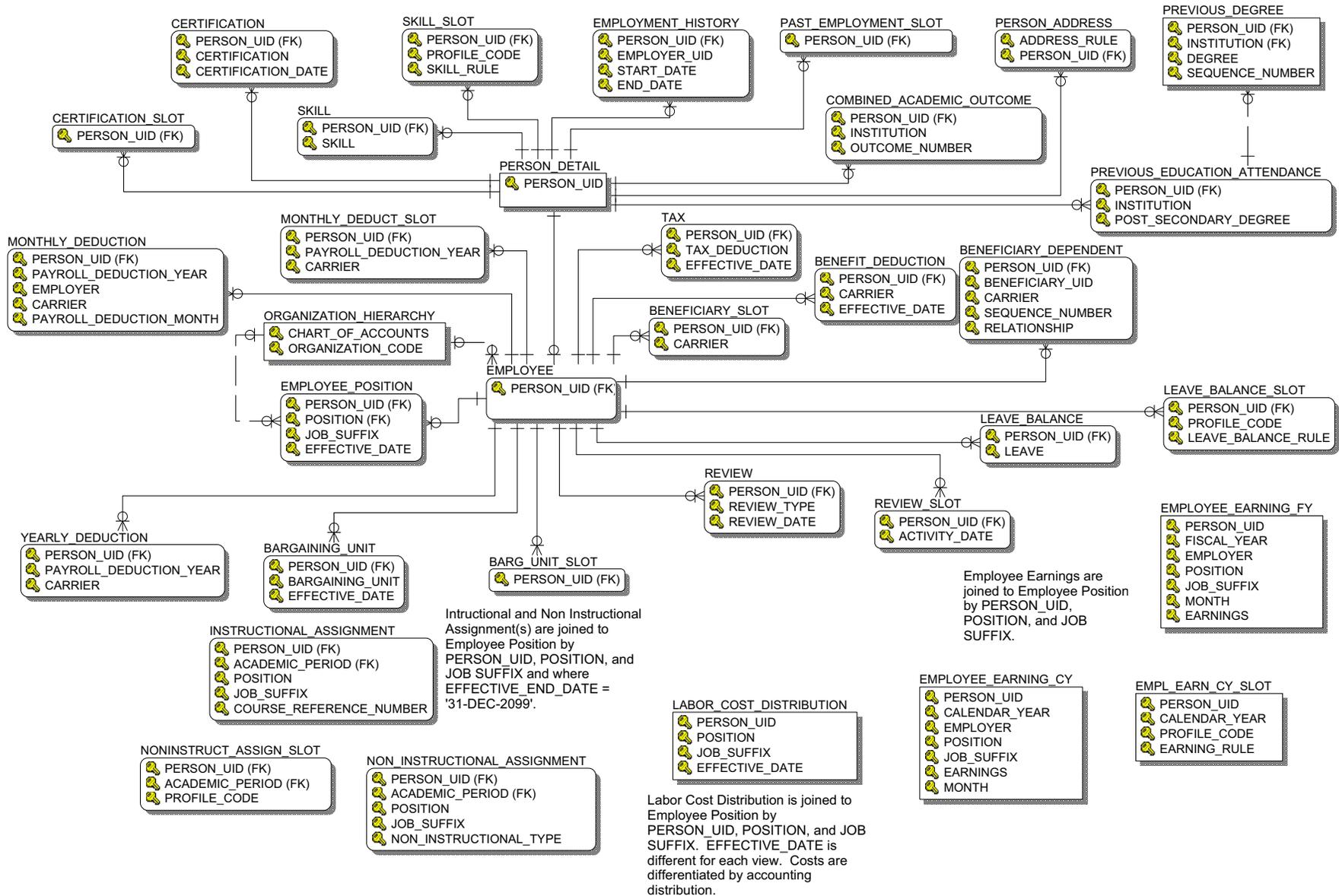
Government Financial Aid Fund contains all the information that is required for government reporting. It does not need to join with other reporting views.

# Loan Disbursement

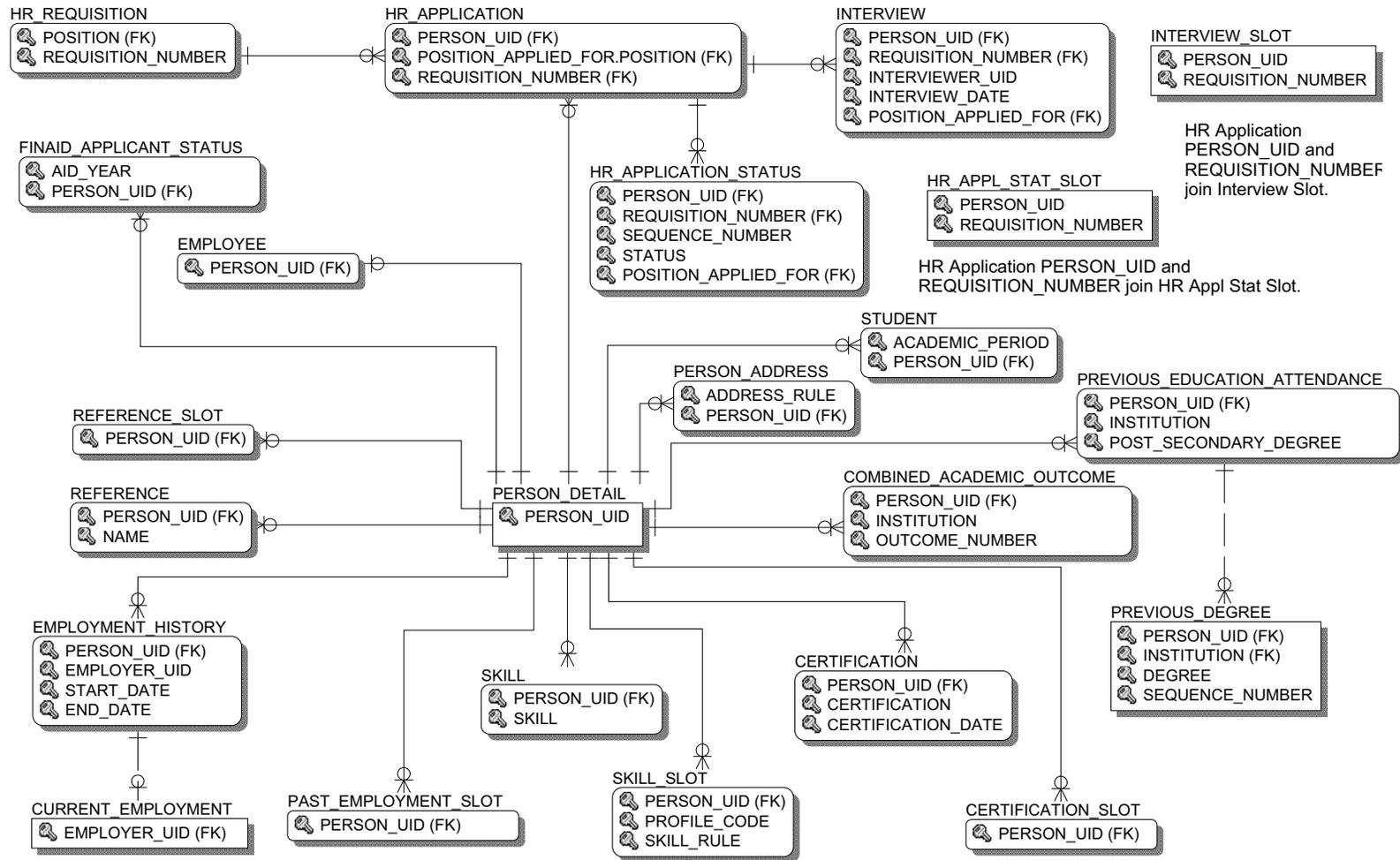


# Human Resources

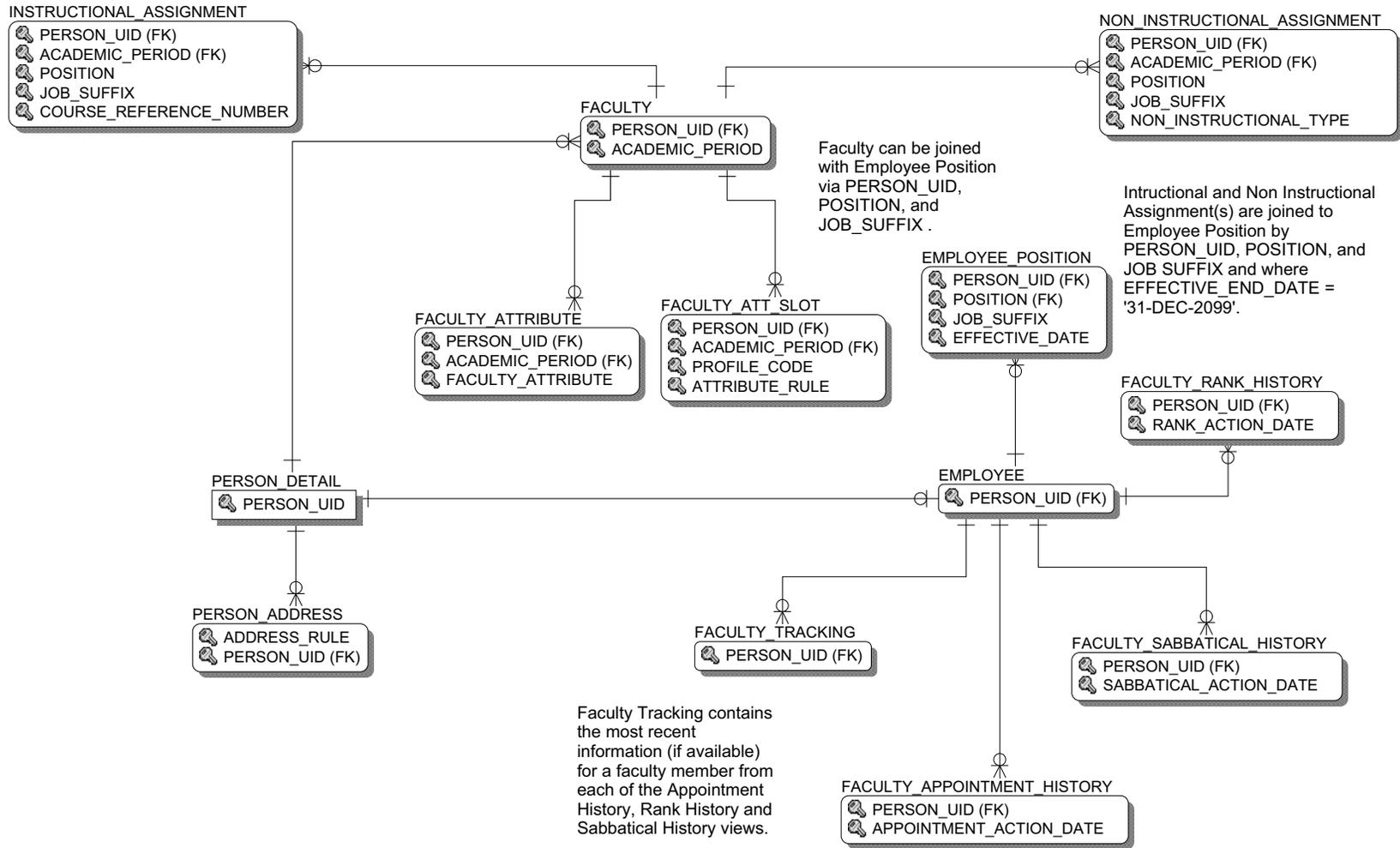
## Employee



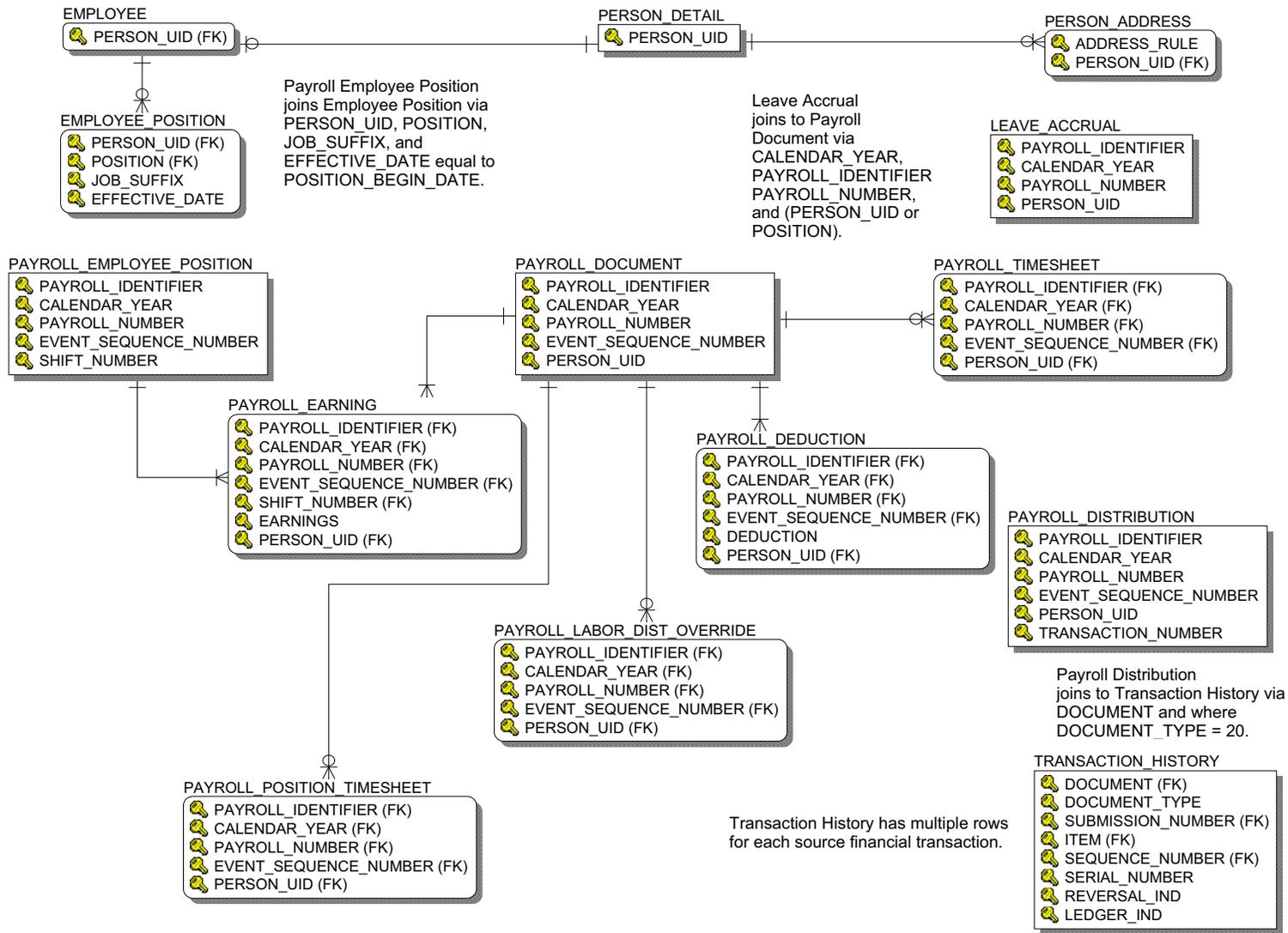
# Human Resource Application



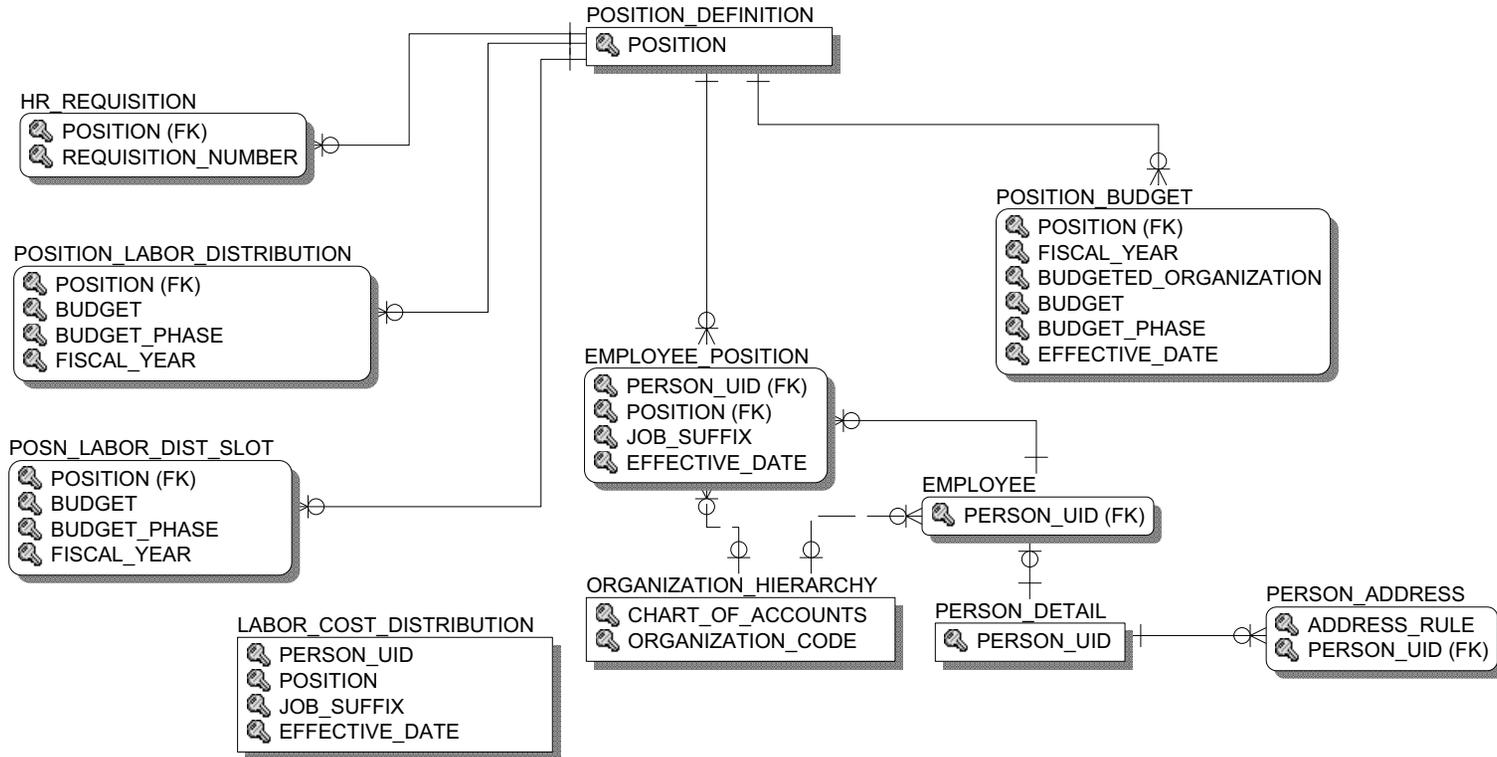
# Human Resource Faculty



# Payroll



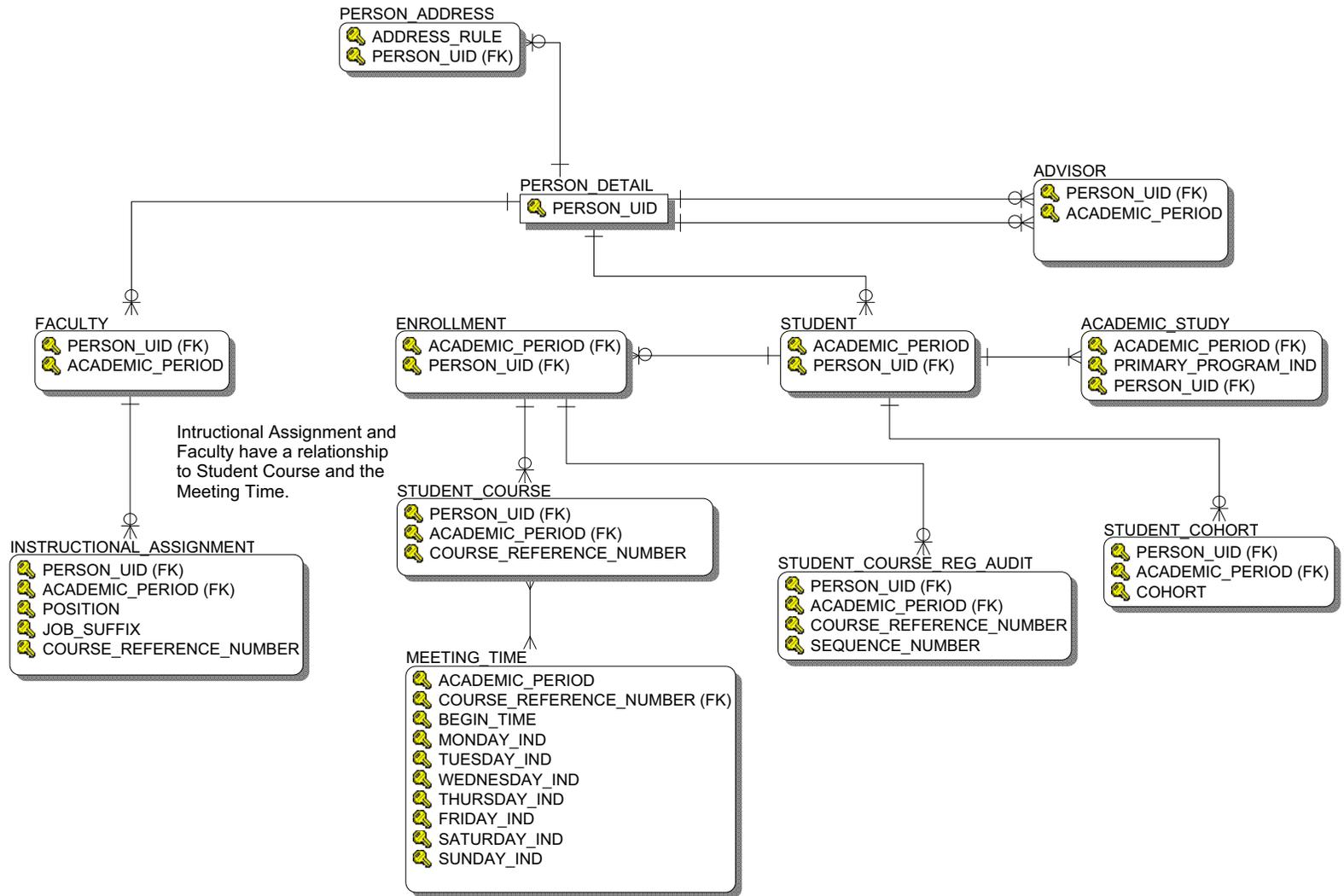
## Position



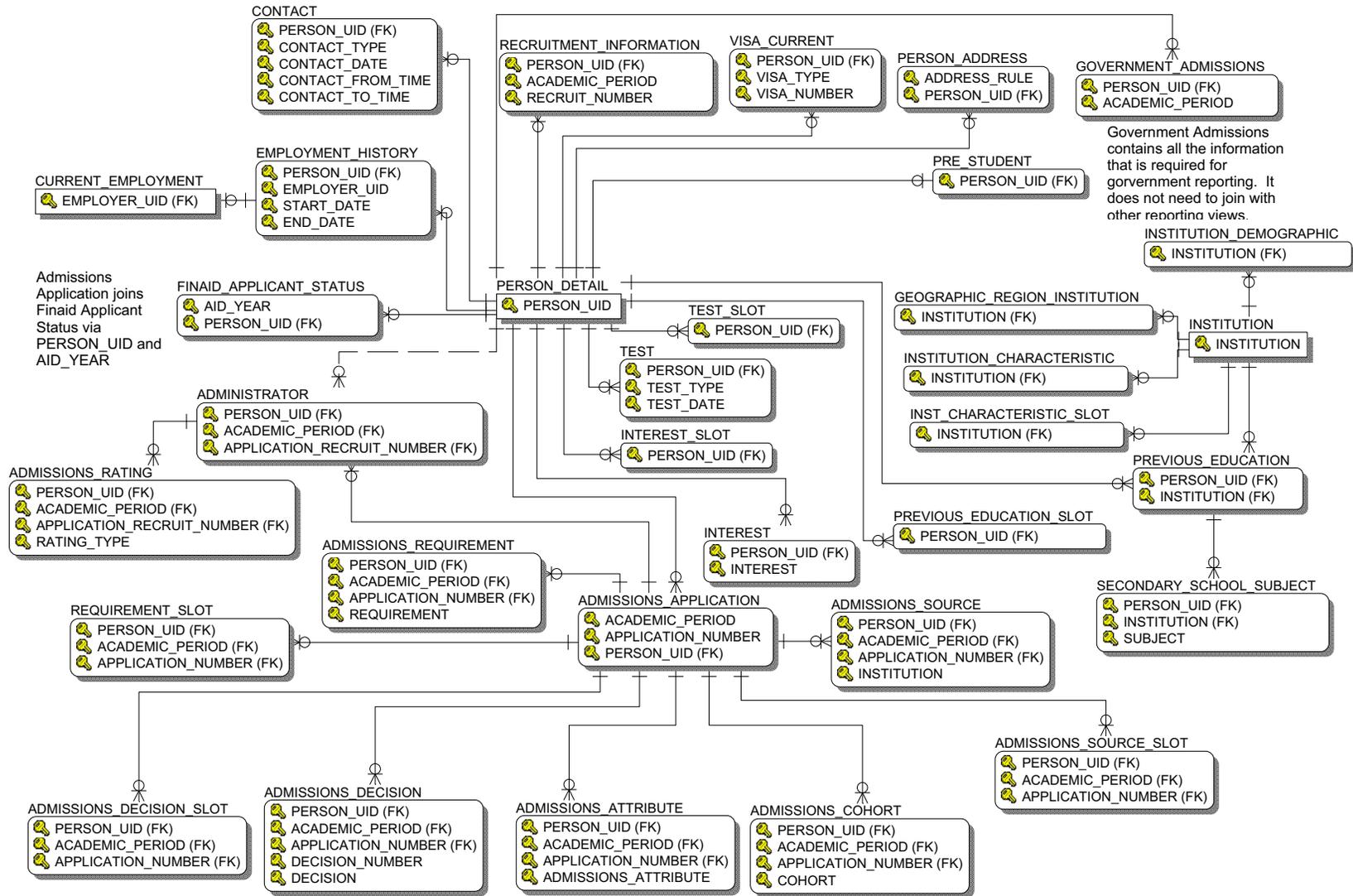
Labor Cost Distribution is joined to Employee Position by PERSON\_UID, POSITION, and JOB SUFFIX. EFFECTIVE\_DATE is different for each view. Costs are differentiated by accounting distribution.

# Student

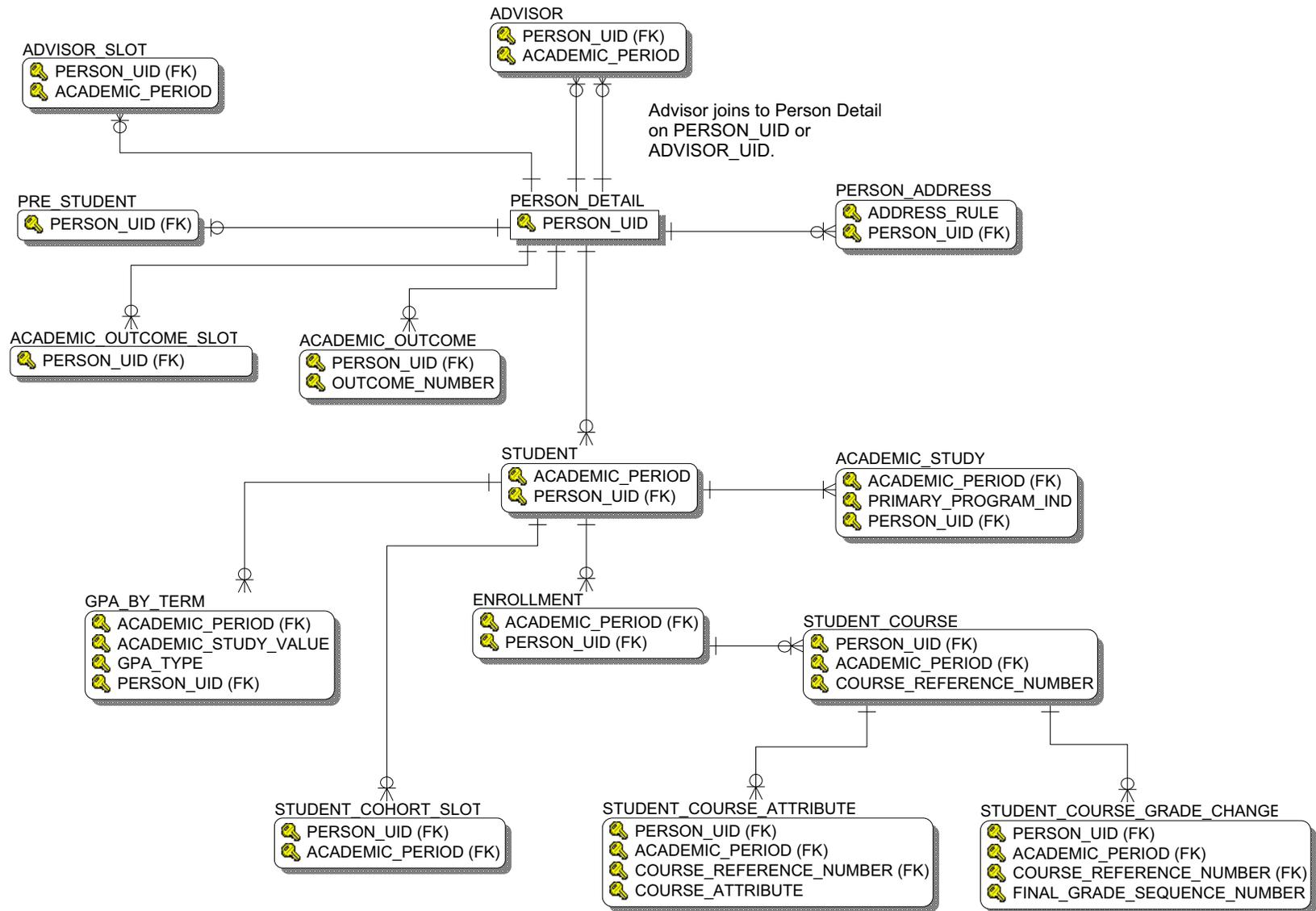
## Active Registration



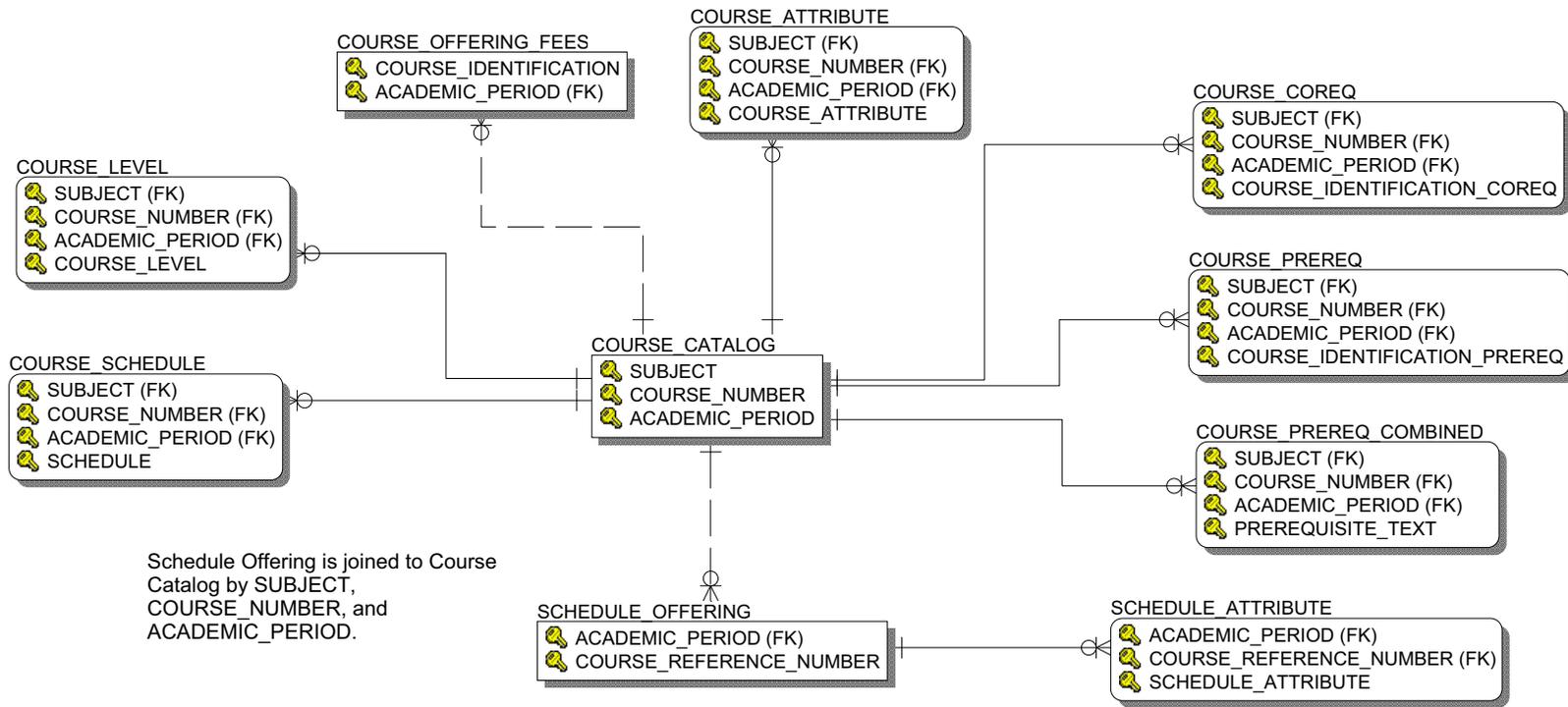
# Admissions Application



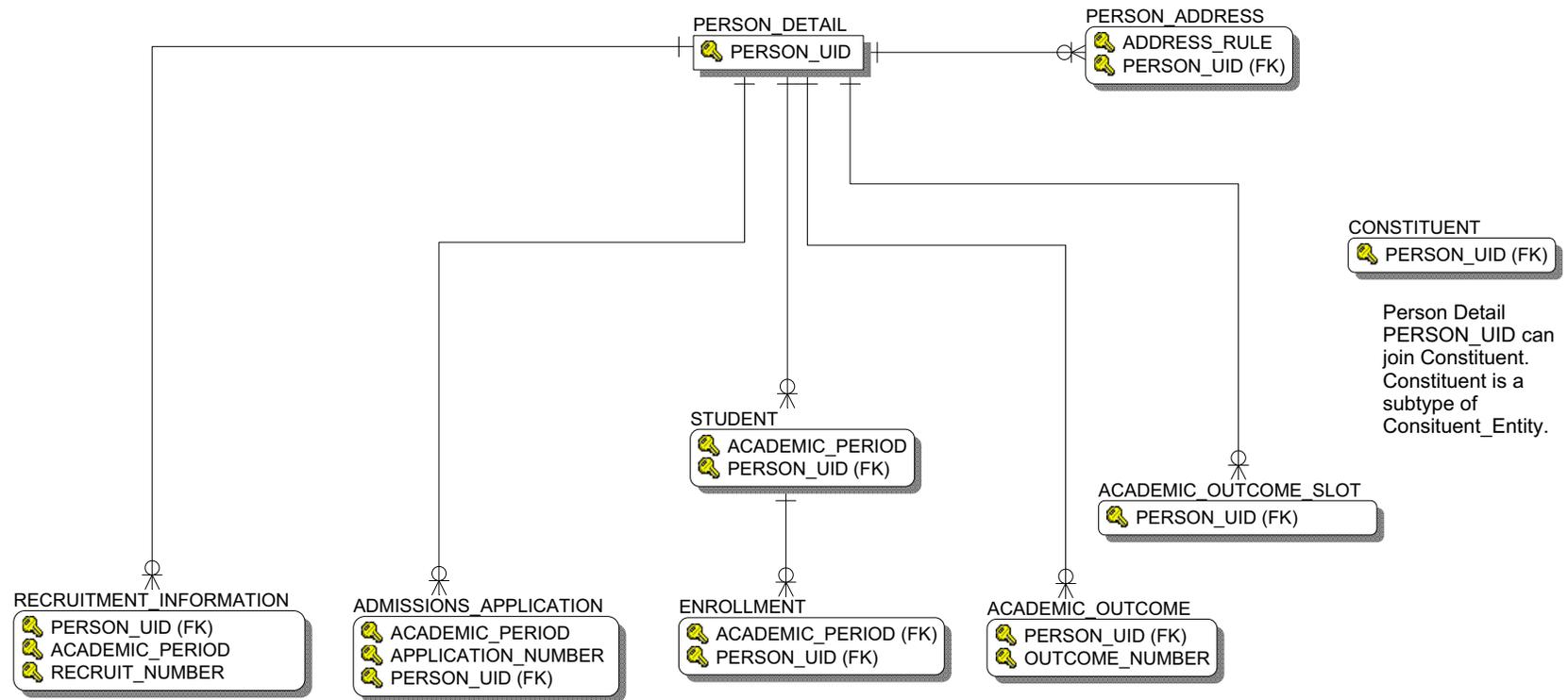
# Advisor Student List



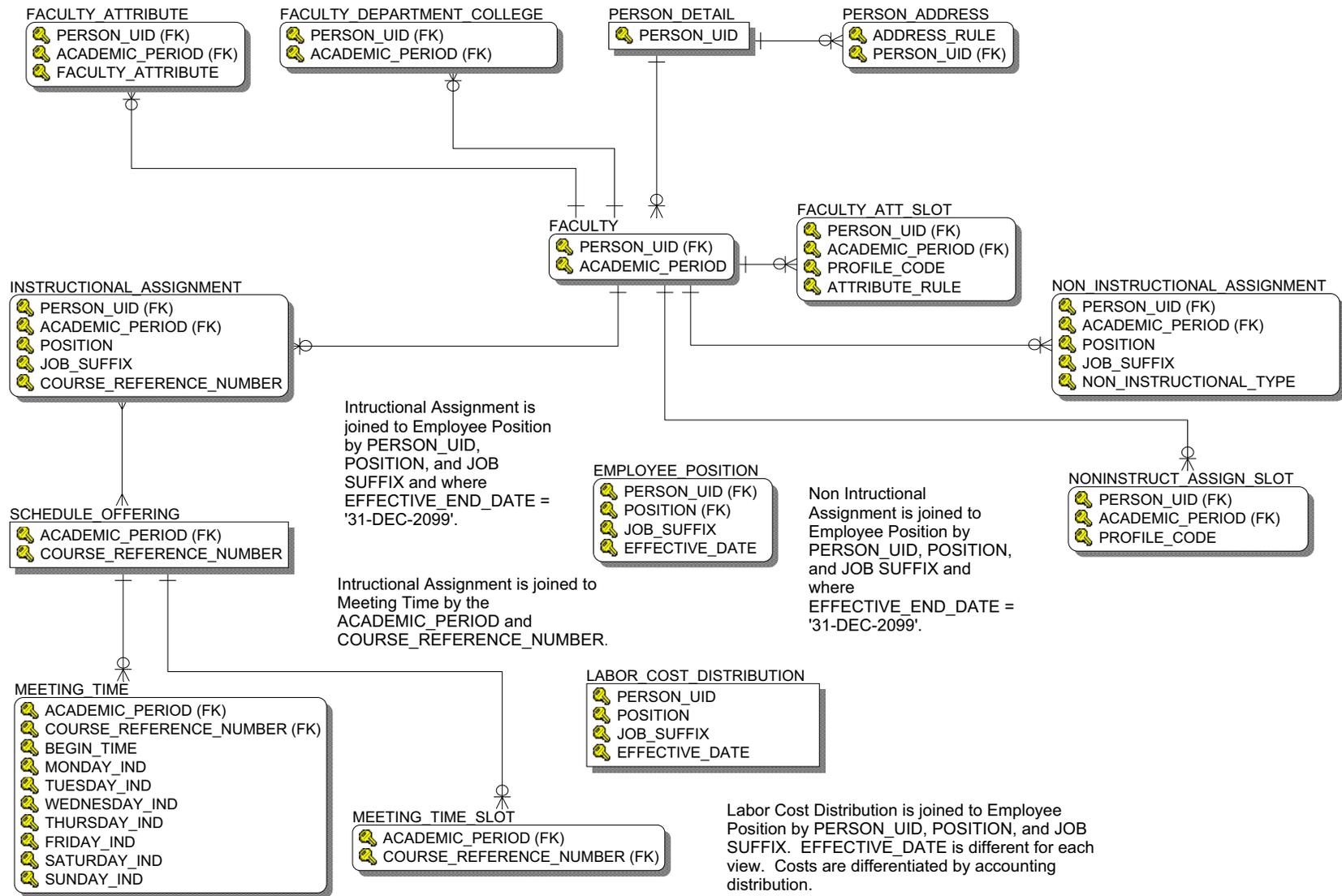
# Course Catalog



# Enrollment Management

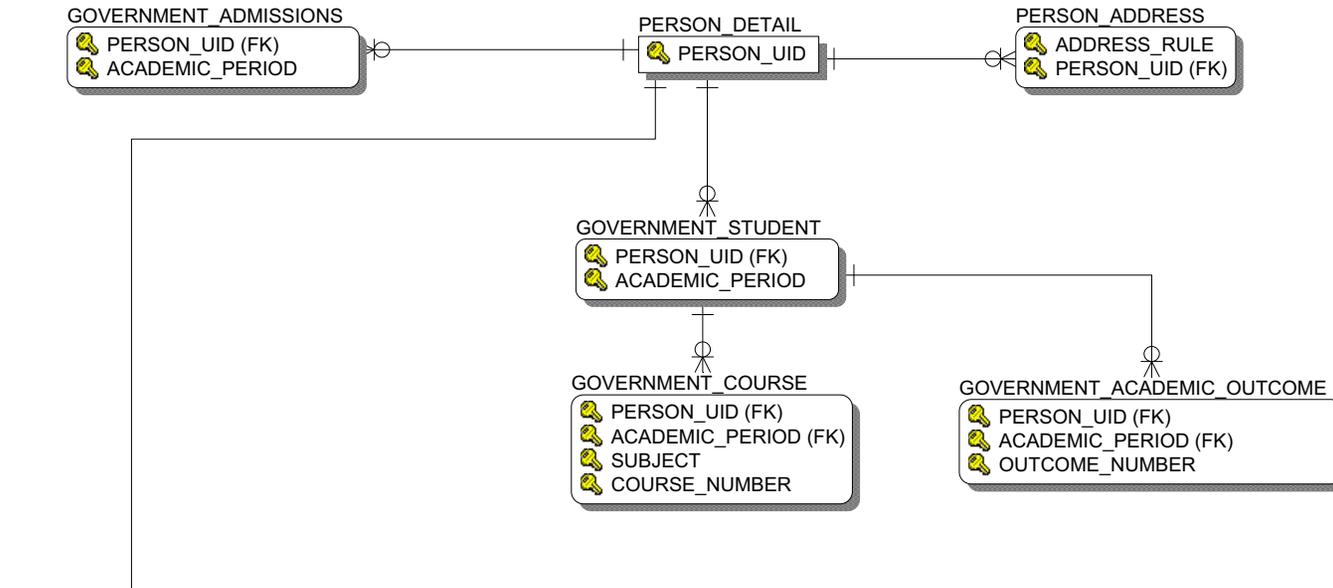


# Faculty Assignment

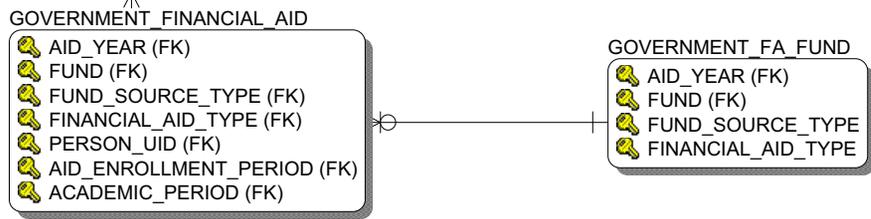


# Government Reporting

## Student Related Government Reporting

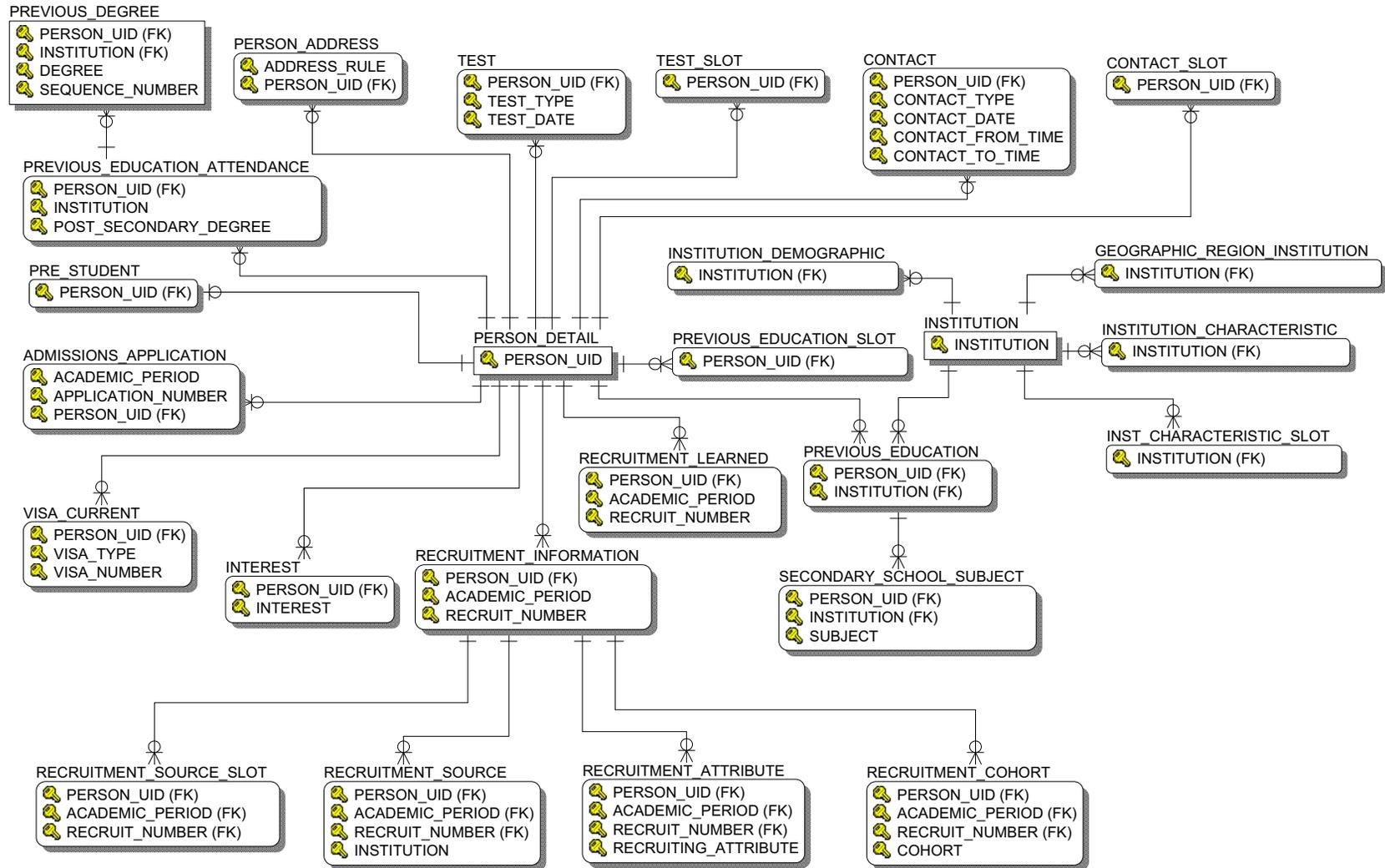


## Financial Aid Related Government Reporting

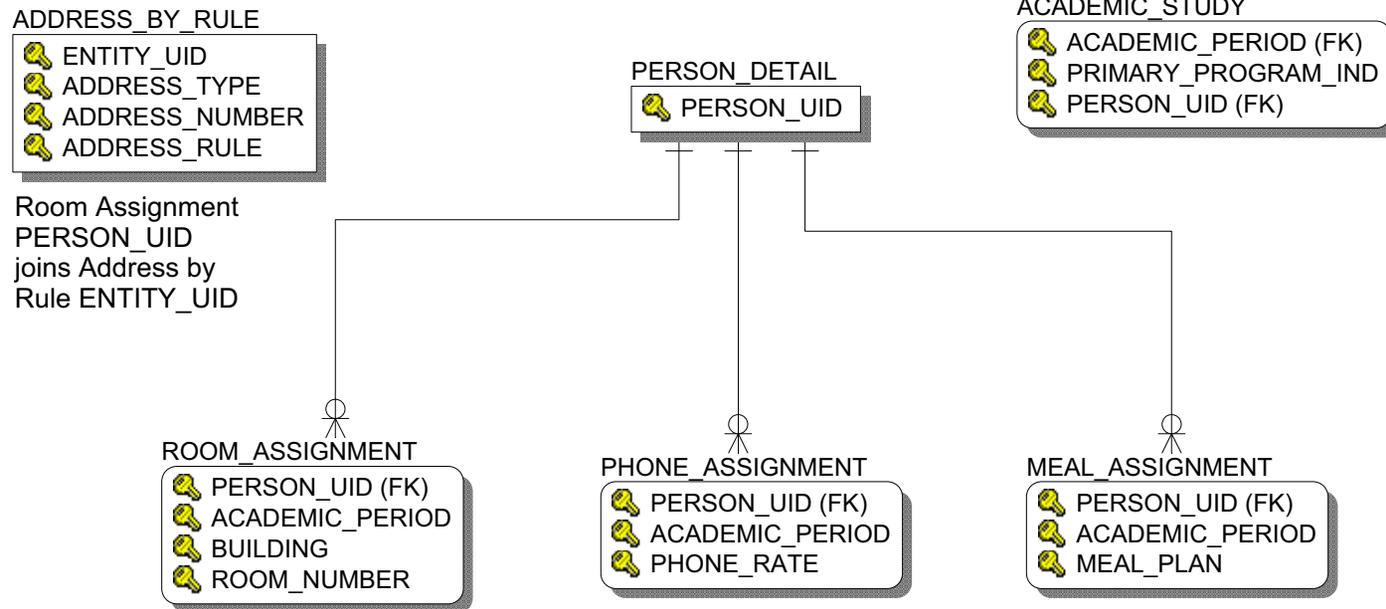


ACADEMIC\_PERIOD is NULL in Government Financial Aid Fund.

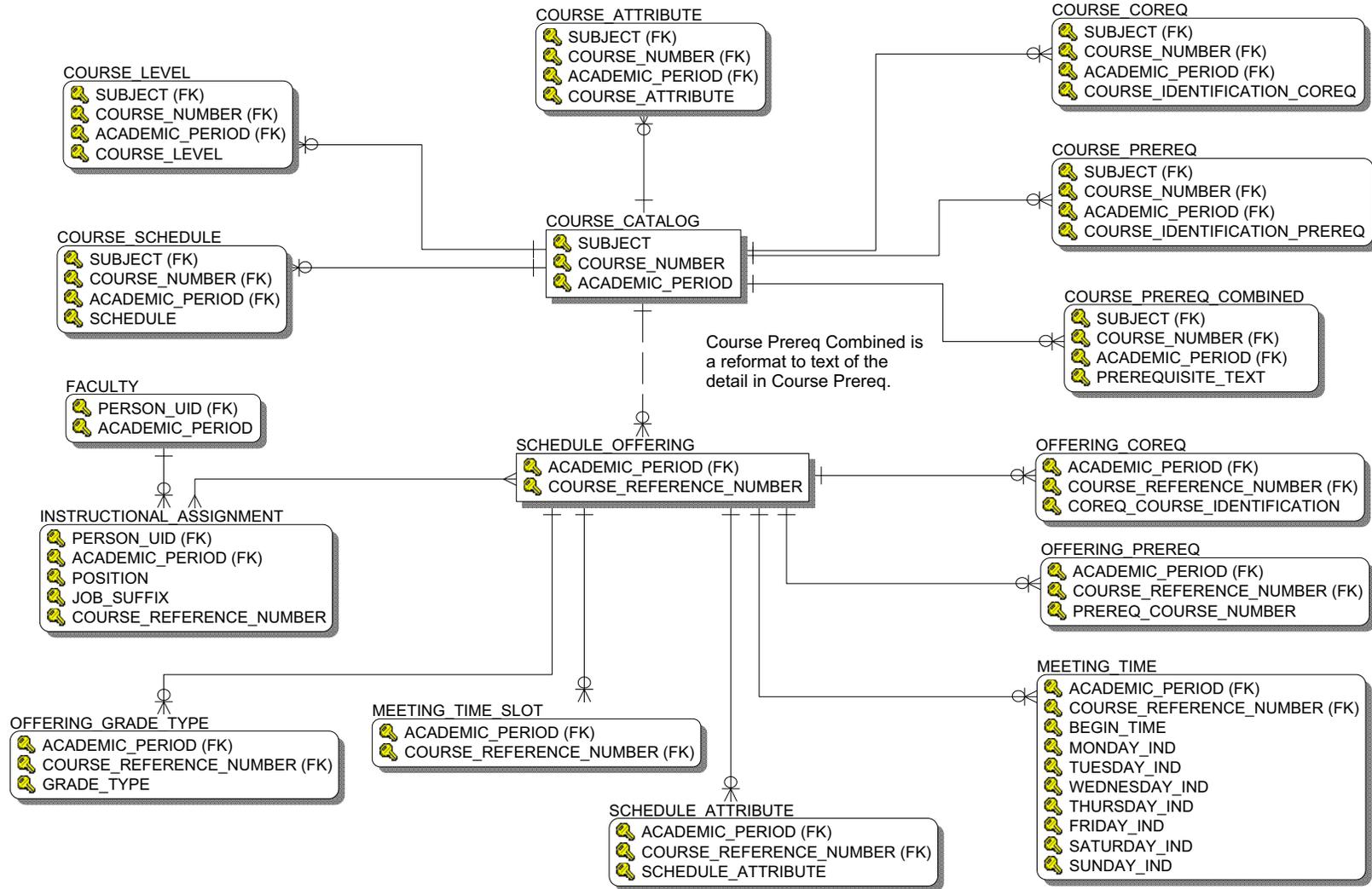
# Recruitment Information



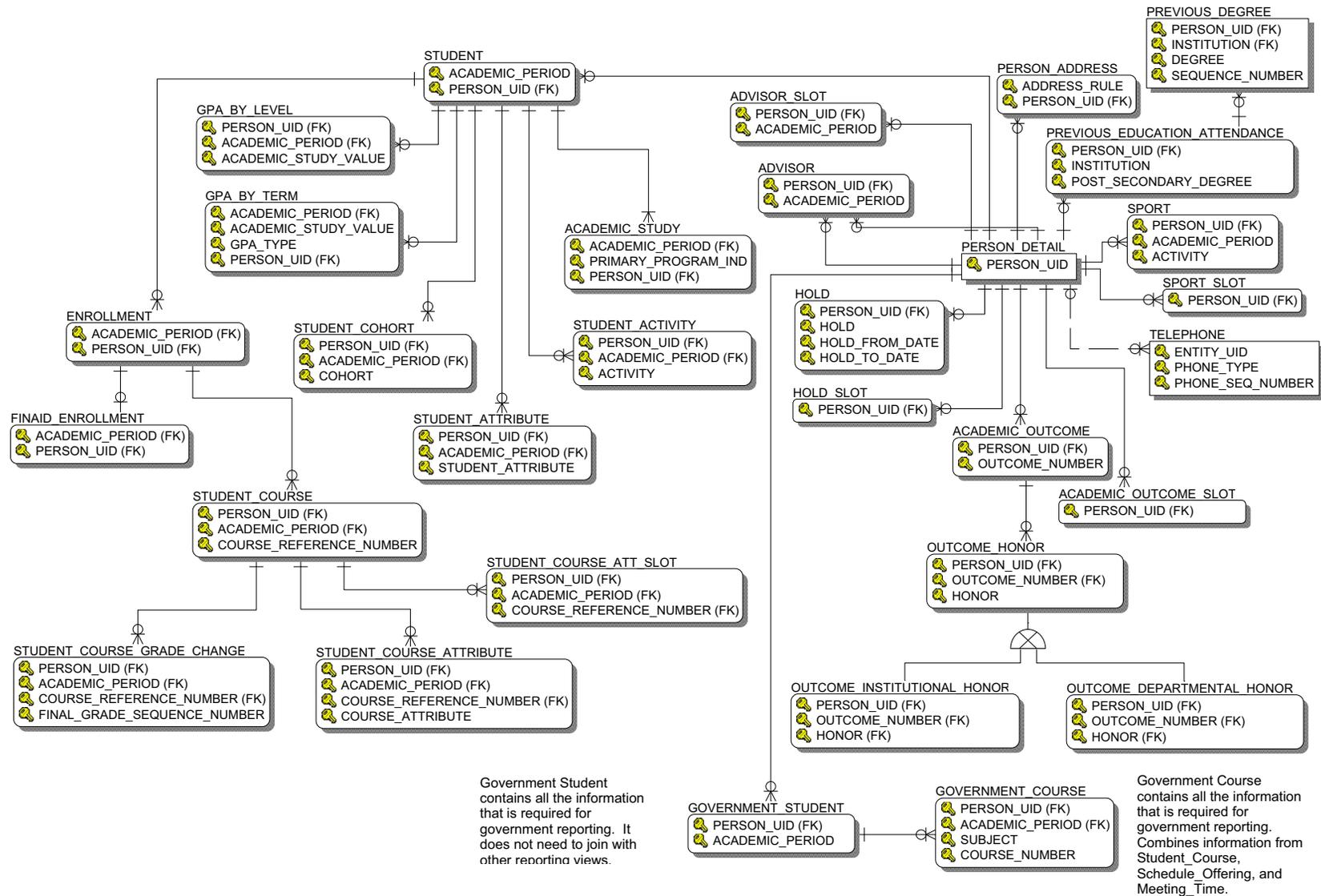
## Residential Life



# Schedule Offering

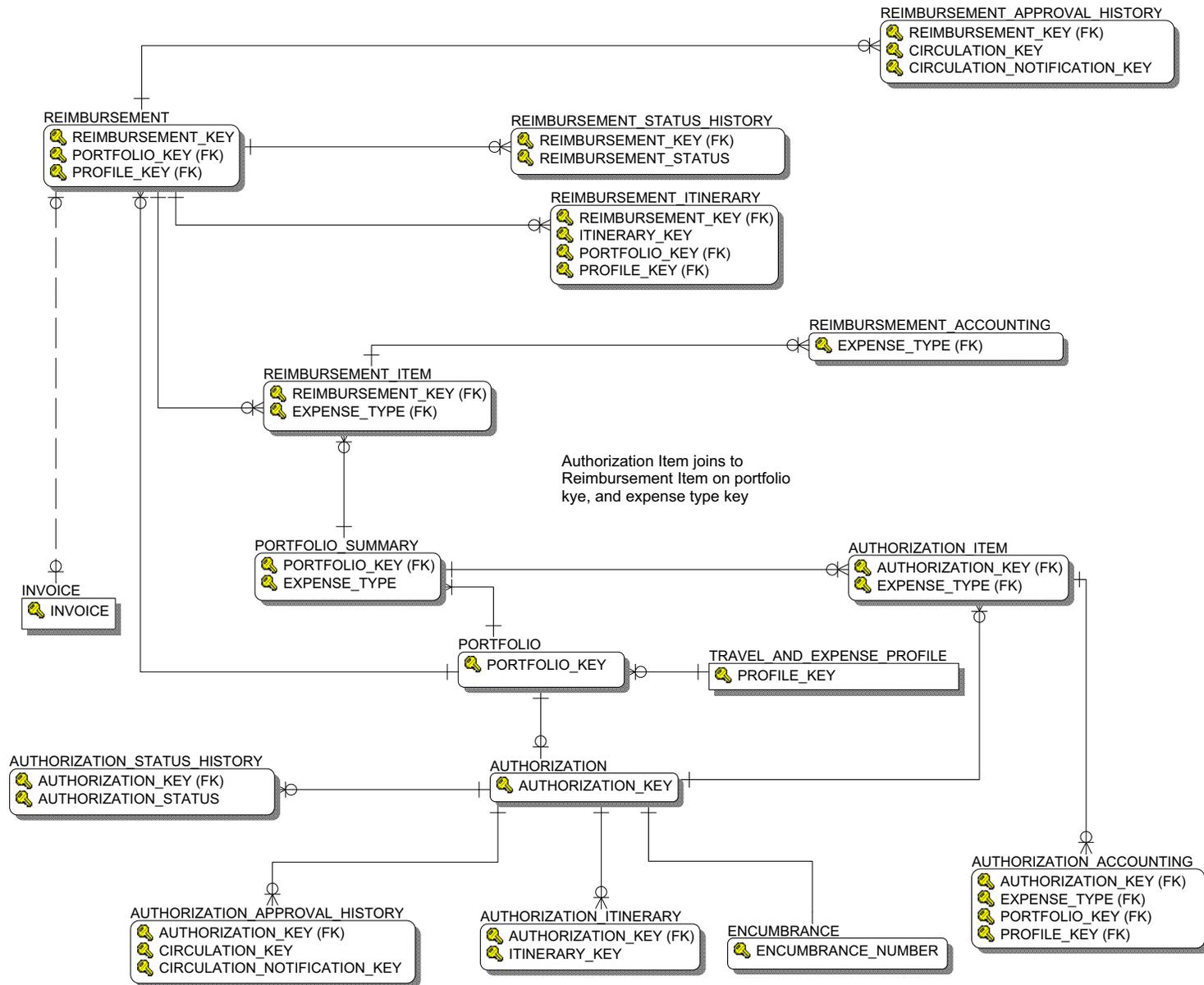


# Student Detail

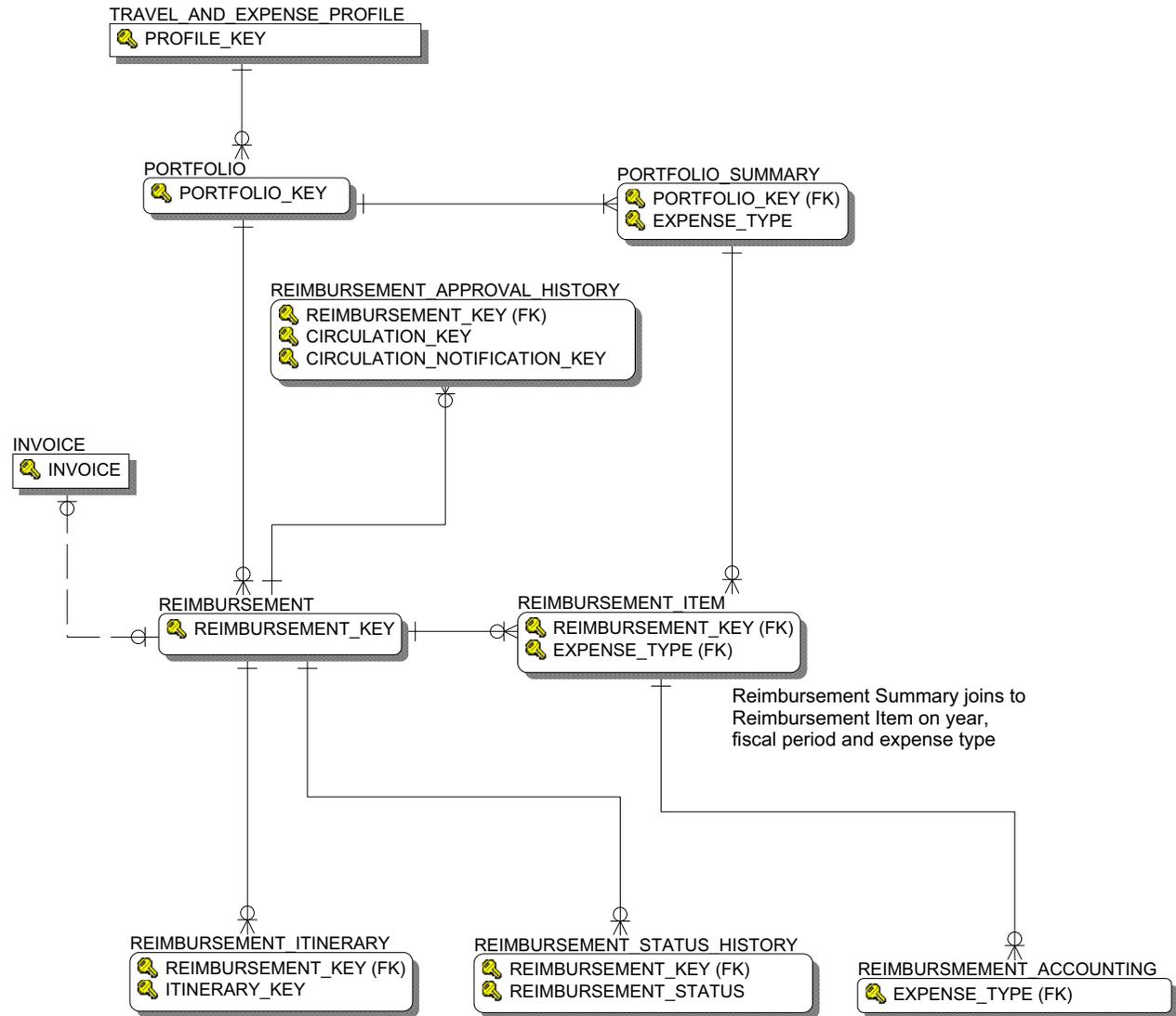


# Travel and Expense

## Authorization



# Reimbursement



## Travel and Expense Reporting Views

The following reporting views enable you to report on the Travel and Expense product.

Reporting View	Description
AUTHORIZATION	Reports on expense report authorization requests from travel and expense. This enables reporting by authorization status and contains information about the request for an authorization, its current status, expense owner information, and summary authorization amounts for reimbursable and non-reimbursable expenses. If the authorization has associated reimbursements, then this also contains summary amounts for the reimbursements as reimbursable and non-reimbursable.
AUTHORIZATION_ACCOUNTING	Reports on an expense authorization request's accounting distribution. This enables reporting of expense amounts by department and contains general expense accounting data including, fiscal year/period, accounting distribution, and approved amount. Additional reporting can be by financial manager or the various hierarchy levels of the fund, organization code, account, program, and location.
AUTHORIZATION_APPROVAL_HISTORY	Reports on the approval cycle and notifications for authorization requests from travel and expense. This contains approval history information for the authorization for each change in notification, including the person who is approving the document, what action was taken and when, and when email notification was sent.
AUTHORIZATION_ITEM	Reports on expense authorization request details from travel and expense. This enables reporting of requested expenses at a detail level for each authorization containing what the expense description is, expected expense date, the expense type, unit rate for distance calculations or per diem calculations, payment method, if the expense is reimbursable or not, the calculated amount for the expense, and if the expense is to be paid for by personal credit card or institution card.

Reporting View	Description
AUTHORIZATION_ITINERARY	Reports on a travel itinerary associated with an authorization. This contains information about a person's or group's plans for traveling with the start and end dates of travel as well as each starting and ending locations. The itinerary can be created at the time of the authorization request. This enables reporting to determine where and when a person may be at any one point in time while traveling.
AUTHORIZATION_STATUS_HISTORY	Reports on the various statuses of an expense authorization request as it migrates through various stages of completion. This enables reporting to determine how long the life cycle of an authorization may take as well as the history; if it was returned for corrections, denied, or approved. This contains each status change, the date it changed, the time in hours since the prior change, and the time in hours from inception to current status.
PORTFOLIO	Reports on the portfolio information from travel and expense. This enables reporting on travel and non-travel related expenses, travel start and end date, expense owner, business purpose, and various summarized amounts for the authorization and reimbursements within the portfolio. The portfolio is used to associate reimbursements with an authorization.
PORTFOLIO_SUMMARY	Reports on summarized amounts by expense type within a portfolio for the authorization and the reimbursement. This enables management to verify the reimbursements against the authorization.
PROFILE_DEFAULT_ACCOUNTING	Reports on default accounting information for a travel and expense owner profile. This contains one or more accounting distributions and can be used to audit what accounting was actually used for reimbursements versus what account distributions were set up as defaults.
REIMBURSEMENT	Reports on expense report reimbursement requests from travel and expense. This enables reporting by reimbursement status and contains information about the request for a reimbursement, its current status, expense owner and payee information, and summary reimbursement amounts for reimbursable and non-reimbursable expenses. If the reimbursement is associated with an authorization, then this also contains summary amounts for the authorization as reimbursable and non-reimbursable.

Reporting View	Description
REIMBURSEMENT_ACCOUNTING	Reports on an expense reimbursement request's accounting distribution. This enables reporting of expense amounts by department and contains general expense reimbursement accounting data including, fiscal year/period, accounting distribution, and approved amount. Additional reporting can be by bank, and financial manager or the various hierarchy levels of the fund, organization code, account, program, and location.
REIMBURSEMENT_APPROVAL_HISTORY	Reports on the approval cycle and notifications for reimbursement requests from travel and expense. This contains approval history information for the reimbursement for each change in notification, including the person who is approving the document, what action was taken and when, and when email notification was sent.
REIMBURSEMENT_ITEM	Reports on expense reimbursement request details from travel and expense. This enables expenses to be reported at a detail level for each reimbursement. It contains the expense description, receipt date, the expense type, unit rate for distance calculations or per diem calculations, payment method, if the expense is reimbursable or not, the calculated amount for the expense, and if the expense was paid for by personal credit card or institution card.
REIMBURSEMENT_ITINERARY	Reports on a travel itinerary associated with a reimbursement. This contains information about a person's or group's plans for traveling with the start and end dates of travel as well as each starting and ending locations. The itinerary may be created at the time of a reimbursement request. This enables reporting to determine where and when a person was at any one point in time while traveling.

Reporting View	Description
REIMBURSEMENT_STATUS_HISTORY	Reports on the various statuses of an expense reimbursement request as it migrates through various stages of completion. This enables reporting to determine how long the life cycle of a reimbursement may take as well as the status history; if it was returned for corrections, denied, approved and when it was paid. This contains each status change, the date it changed, the time in hours since the prior change, and the time in hours from inception to the current status.
TRAVEL_AND_EXPENSE_PROFILE	Reports on current and default information for a travel and expense owner profile. This contains the approver for the expense owner, address type for check payment, email address, profile ID and name, workflow logon if the profile is for an approver, and the total amount reimbursed for the expense owner.

## Reporting Views

Data from your source system database (for example, Student, Human Resources, Finance, etc.) is used to populate Banner ODS composite tables, and can be retrieved in reports using the Banner ODS reporting views. Use the Administrative UI to maintain and view meta data reports for each composite view (data on the source system used as an intermediate step to produce the composite tables and reporting views) and reporting view. The meta data reports enable you to look at the information about the composite or reporting view definition, and the column business definitions by either target, composite or reporting view, or by source administrative system sources.

For additional information on how to view meta data for the composite views or reporting views, refer to the *Administrative User Interface* chapter, [“Composite View Meta Data”](#) section. For additional information on how to maintain meta data for the composite or reporting views, and to maintain sources and source columns, refer to the *Administrative User Interface* chapter, [“Meta Data”](#) section.

## List of Value Views

A list of values (LOV) contains a list of predefined values for a reporting view column in a report. For example, a list of values for Academic Period might contain the values Fall 2006, Spring 2007, and Summer 2007. You use

lists of values in parameters or conditions for a report. When used in parameters or conditions, lists of values enable you to select predefined values rather than enter arbitrary values in a text field.

The Banner ODS has a database schema called ODSLOV that owns the list of value views. Most, but not all, of the views are based on the MGT\_VALIDATION composite table. (At least one view is based on an MGRSDAX rule.) MGT\_VALIDATION is loaded using Oracle Warehouse Builder (OWB) from validation tables (or in some cases static lists of values) in Banner. Validation tables loaded into MGT\_VALIDATION from Banner have been identified as lists of values that have views assigned to them. (Not all the MGT\_VALIDATION validation tables have been created as LOV views.) Each view has the columns TABLE\_NAME, VALUE, and VALUE\_DESC. TABLE\_NAME is the name of Banner validation table. VALUE and VALUE\_DESC are values, or codes, and descriptions for the values. Some of the views also have QUALIFIER, and QUALIFIER\_DESC. QUALIFIER is used to group values by a common attribute. For example, it can be Chart of Accounts, Academic Period or a Banner PIDM. QUALIFIER\_DESC is a description for the QUALIFIER. Qualifier description is only populated when the qualifier is an Academic Period. For example, it can be Chart of Accounts, Academic Period or a Banner PIDM. QUALIFIER\_DESC is a description for the QUALIFIER.

The list of value view provides one place to define the predefined values for a column in reporting views. For example, the LOV\_ACADEMIC\_PERIOD view contains a list of values that is used by Academic Period columns in many reporting views - such as ACADEMIC\_OUTCOME, ACADEMIC\_STUDY, etc. By creating the predefined list in one view and using it for all the columns in the reporting views that require a predefined list of Academic Periods, the Banner ODS provides a simple to understand and use mechanism for creating parameters and conditions. If there were a different list of Academic Periods for every Academic Period column in every reporting view in the Banner ODS, there would be hundreds of different predefined lists of values that would be difficult for end users to understand and information technology departments to maintain.

The list of value view also provides fast access when producing the predefined values. If lists of values were created by selecting distinct values from the reporting views, more rows would be read to produce the list. This can result in unacceptable query times in reports when generating lists for parameter prompts and conditions.

ODSLOV list of value views are used in Self Service Reporting (SSR), the Banner ODS Cognos ReportNet model and Oracle Discoverer End User Layer. How these views are used is described in the SSR and Third Party Reporting Tools chapters.

The following table provides information about the list of value views in the ODSLOV schema.

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_ACADEMIC_PERIOD	STVTERM	STVTERM					
LOV_ACADEMIC_STANDING	STVASTD	STVASTD					
LOV_ACADEMIC_TITLE	ACADEMIC_TITLE	PERBFAC					
LOV_ACADEMIC_YEAR	STVACYR	STVACYR					
LOV_ACCOUNT_ATTRIBUTE_SET	ACCOUNT_SET_CODE	FTVATTS , FTRACTA	X				
LOV_ACCOUNT_ATTRIBUTE_TYPE	ACCOUNT_ATTRIBUTE_TYPE	FTVATTT , FTRACTA	X				
LOV_ACCOUNT_ATTRIBUTE_VALUE	ACCOUNT_ATTRIBUTE_VALUE	FTRATTV , FTRACTA	X				
LOV_ACCOUNT_CLASS	ACCOUNT_CLASS	FTVSDAT	X			X	
LOV_ACCOUNT_LEVEL_1	ACCOUNT_LEVEL_1	FTVACCT	X				
LOV_ACCOUNT_LEVEL_2	ACCOUNT_LEVEL_2	FTVACCT	X				
LOV_ACCOUNT_LEVEL_3	ACCOUNT_LEVEL_3	FTVACCT	X				
LOV_ACCOUNT_LEVEL_4	ACCOUNT_LEVEL_4	FTVACCT	X				
LOV_ACCOUNT_POOL	ACCOUNT_POOL	FTVACCT	X				
LOV_ACCOUNT_TYPE_ATTR_SET	ACCOUNT_TYPE_SET_CODE	FTRATYA , FTVATTS	X				
LOV_ACCOUNT_TYPE_ATTR_TYPE	ACCOUNT_TYPE_ATTR_TYPE	FTVATTT , FTRATYA	X				
LOV_ACCOUNT_TYPE_ATTR_VALUE	ACCOUNT_TYPE_ATTR_VALUE	FTRATTV , FTRATYA	X				
LOV_ACCOUNT_TYPE_LEVEL_1	ACCOUNT_TYPE_LEVEL_1	FTVATYP	X				
LOV_ACCOUNT_TYPE_LEVEL_2	ACCOUNT_TYPE_LEVEL_2	FTVATYP	X				

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_ACTIVITY	STVACTC	STVACTC					
LOV_ACTIVITY_CATEGORY	STVACCG	STVACCG					
LOV_ACTIVITY_TYPE	STVACTP	STVACTP					
LOV_ADDRESS_TYPE	STVATYP	STVATYP					
LOV_ADMISSIONS_ATTRIBUTE	STVATTS	STVATTS					
LOV_ADMISSIONS_POPULATION	STVADMT	STVADMT					
LOV_ADMISSIONS_RATING_TYPE	STVRATP	STVRATP					
LOV_ADVANCEMENT_FUND	ATVFUND	ATVFUND					
LOV_ADVISOR_NAME_LFMI	ADVISOR_NAME_LFMI	SGRADVR					X
LOV_ADVISOR_TYPE	STVADVR	STVADVR					
LOV_AID_FUND	RFRBASE	RFRBASE					
LOV_AID_YEAR	ROBINST	ROBINST					
LOV_APPLICATION_DECISION	STVAPDC	STVAPDC					
LOV_APPLICATION_STATUS	STVAPST	STVAPST					
LOV_ASSIGNMENT_GRADE	ASSIGNMENT_GRADE	NBBPOSN					
LOV_ASSIGNMENT_PAY_ID	PTRPICT	PTRPICT					
LOV_ASSIGNMENT_SALARY_GROUP	NTRSGRP	NTRSGRP					
LOV_AWARD_CATEGORY	STVACAT	STVACAT					
LOV_BENEFIT_CATEGORY	PTRBCAT	PTRBCAT					
LOV_BENEFIT_DEDUCTION	PTRBDCA	PTRBDCA					
LOV_BLOCK_SCHEDULE	STVBLCK	STVBLCK					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_BUDGET	FTVOBUD	FTVOBUD	X				
LOV_BUDGET_GROUP	RTVBGRP	RTVBGRP					
LOV_BUDGET_PHASE	FTVOBPH	FTVOBPH	X				
LOV_BUILDING	STVBLDG	STVBLDG					
LOV_CALENDAR_MONTH	CALENDAR_MONTH	STATIC (01,02,03,04,05,06,07,08,09,10,11,12))					
LOV_CAMPAIGN	AFBCAMP	AFBCAMP					
LOV_CAMPAIGN_TYPE	ATVCMTP	ATVCMTP					
LOV_CAMPUS	STVCAMP	STVCAMP					
LOV_CERTIFICATION	PTRCERT	PTRCERT					
LOV_CHART_OF_ACCOUNTS	FTVCOAS	FTVCOAS				X	
LOV_COHORT	STVCHRT	STVCHRT					
LOV_COLLECTION_AGENCY_NAME	COLLECTION_AGENCY_NAME	TBRCOLC					X
LOV_COLLEGE	STVROLL	STVROLL					
LOV_COMMODITY	FTVCOMM	FTVCOMM				start date term date	
LOV_CONTRACT_NUMBER	CONTRACT_NUMBER	NBRJOBS					
LOV_CONTRACT_TYPE	CONTRACT_TYPE	STATIC (P,S,O)					
LOV_COUNTY	STVCNTY	STVCNTY					
LOV_COURSE_ATTRIBUTE	STVATTR	STVATTR					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_COURSE_IDENTIFICATION	COURSE_IDENTIFICATION	STVTERM, SCBCRSE			X		
LOV_COURSE_REFERENCE_NUMBER	COURSE_REFERENCE_NUMBER	STVTERM, SSBSECT			X		
LOV_CURRENT_TIME_STATUS	STVTMST	STVTMST					
LOV_DEPARTMENT	STVDEPT	STVDEPT					
LOV_DESIGNATION	ADBDESG	ADBDESG					
LOV_DISTRICT_DIVISION	GTVDICD	GTVDICD					
LOV_DIVISION	STVDIVS	STVDIVS					
LOV_DONOR	ATVDONR	ATVDONR					
LOV_EARNINGS	PTREARN	PTREARN					
LOV_EDUCATIONAL_GOAL	STVEGOL	STVEGOL					
LOV_EDUCATIONAL_LEVEL	STVEDLV	STVEDLV					
LOV_EEO_SKILL	PTVESKL	PTVESKL					
LOV_EMPLOYEE_CLASS	PTRECLS	PTRECLS					
LOV_EMPLOYEE_GROUP	PTVEGRP	PTVEGRP					
LOV_EMPLOYEE_STATUS	EMPLOYEE_STATUS	STATIC (A,B,L, F,P,T)					
LOV_EMPLOYER	PTREMPR	PTREMPR					
LOV_EMPLOYER_CATEGORY	ATVJOB	ATVJOB					
LOV_EMPLOYER_INDUSTRIAL_TYPE	ATVSICC	ATVSICC					
LOV_EMPLOYMENT_STATUS	ATVEMPS	ATVEMPS					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_ENROLLMENT_STATUS	STVESTS	STVESTS					
LOV_ETHNICITY	STVETHN	STVETHN					
LOV_FINANCE_ACCOUNT	FTVACCT	FTVACCT	X			X	
LOV_FINANCE_ACCOUNT_TYPE	FTVATYP	FTVATYP	X			X	
LOV_FINANCE_ACTIVITY	FTVACTV	FTVACTV	X			X	
LOV_FINANCE_FUND	FTVFUND	FTVFUND	X			X	
LOV_FINANCE_FUND_TYPE	FTVFTYP	FTVFTYP	X			X	
LOV_FINANCE_LOCATION	FTVLOCN	FTVLOCN	X			X	
LOV_FINANCE_ORGANIZATION	FTVORGN	FTVORGN	X			X	
LOV_FINANCE_PROGRAM	FTVPROG	FTVPROG	X			X	
LOV_FINANCIAL_MANAGER	FINANCIAL_MANAGER	FTVFUND		X			
LOV_FISCAL_PERIOD	FTVFSPD	FTVFSPD	X			start date	
LOV_FISCAL_YEAR	FTVFSYR	FTVFSYR	X			start date	
LOV_FOREIGN_CURRENCY	GTVCURR	GTVCURR				X	
LOV_FUND_ATTRIBUTE_SET	FUND_SET_CODE	FTVATTS , FTRFNDA	X				
LOV_FUND_ATTRIBUTE_TYPE	FUND_ATTRIBUTE_TYPE	FTVATTT , FTRFNDA	X				
LOV_FUND_ATTRIBUTE_VALUE	FUND_ATTRIBUTE_VALUE	FTRATTV , FTRFNDA	X				
LOV_FUND_LEVEL_1	FUND_LEVEL_1	FTVFUND	X				
LOV_FUND_LEVEL_2	FUND_LEVEL_2	FTVFUND	X				
LOV_FUND_LEVEL_3	FUND_LEVEL_3	FTVFUND	X				
LOV_FUND_LEVEL_4	FUND_LEVEL_4	FTVFUND	X				

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_DATE logic	Uses PIDM as Value
LOV_FUND_LEVEL_5	FUND_LEVEL_5	FTVFUND	X				
LOV_FUND_POOL	FUND_POOL	FTVFUND	X				
LOV_FUND_SOURCE	RTVFSRC	RTVFSRC					
LOV_FUND_TYPE	RTVFTYP	RTVFTYP					
LOV_FUND_TYPE_ATTR_SET	FUND_TYPE_SET_CODE	FTVATTS , FTRFTYA	X				
LOV_FUND_TYPE_ATTR_TYPE	FUND_TYPE_ATTR_TYPE	FTVATTT , FTRFTYA	X				
LOV_FUND_TYPE_ATTR_VALUE	FUND_TYPE_ATTR_VALUE	FTRATTV , FT RFTYA	X				
LOV_FUND_TYPE_LEVEL_1	FUND_TYPE_LEVEL_1	FTVATYP	X				
LOV_FUND_TYPE_LEVEL_2	FUND_TYPE_LEVEL_2	FTVATYP	X				
LOV_GENDER	GENDER	STATIC (M, F, N)					
LOV_GEOGRAPHIC_AREA	STVGEOR	STVGEOR					
LOV_GRADE_TYPE	STVGMOD	STVGMOD					
LOV_GRANT	FRBGRNT	FRBGRNT		X			term date
LOV_HOLD	STVHLDD	STVHLDD					
LOV_HR_APPLICATION_STATUS	PTRAPPS	PTRAPPS					
LOV_INCOME_LEVEL	ATVINCM	ATVINCM					
LOV_INSTALLMENT_PLAN	INSTALLMENT_PLAN	TBBISTC					
LOV_INSTRUCTIONAL_METHOD	GTVINSM	GTVINSM					
LOV_INSTRUCTOR_NAME	INSTRUCTOR_NAME	STVTERM , SPRIDEN , SIRASGN			X		X

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_DATE logic	Uses PIDM as Value
LOV_INTERNAL_ACCOUNT_TYPE	INTERNAL_ACCOUNT_TYPE	FTVSDAT	X			X	
LOV_INTERNAL_FUND_TYPE	INTERNAL_FUND_TYPE	FTVSDAT	X			X	
LOV_INTERVIEW_STATUS	STVINTV	STVINTV					
LOV_JOB_LEAVE_CATEGORY	PTRLEAV	PTRLEAV					
LOV_JOB_SUFFIX	JOB_SUFFIX	STATIC (0,00,01,02,03)					
LOV_LEADERSHIP_ROLE	STVLEAD	STVLEAD					
LOV_LEAVE_OF_ABSENCE_REASON	PTRLREA	PTRLREA					
LOV_LEGACY	STVLGCY	STVLGCY					
LOV_LEVEL	STVLEVL	STVLEVL					
LOV_LOCATION_LEVEL_1	LOCATION_LEVEL_1	FTVLOCN	X				
LOV_LOCATION_LEVEL_2	LOCATION_LEVEL_2	FTVLOCN	X				
LOV_LOCATION_LEVEL_3	LOCATION_LEVEL_3	FTVLOCN	X				
LOV_LOCATION_LEVEL_4	LOCATION_LEVEL_4	FTVLOCN	X				
LOV_MAIL	GTVMAIL	GTVMAIL					
LOV_MAJOR	STVMAJR	STVMAJR					
LOV_MARITAL_STATUS	STVMRTL	STVMRTL					
LOV_MEAL_PLAN	STVMRCD	STVMRCD					
LOV_MEETING_TYPE	GTVMTYP	GTVMTYP					
LOV_NATION	STVNATN	STVNATN					
LOV_NATIVE_LANGUAGE	STVLANG	STVLANG					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_ORGANIZATION_ATTR_SET	ORGANIZATION_SET_CODE	F'VATTS , F'TRORGA	X				
LOV_ORGANIZATION_ATTR_TYPE	ORGANIZATION_ATTR_TYPE	F'VATTT , F'TRORGA	X				
LOV_ORGANIZATION_ATTR_VALUE	ORGANIZATION_ATTR_VALUE	F'TRATTV , F'TRORGA	X				
LOV_ORGANIZATION_LEVEL_1	ORGANIZATION_LEVEL_1	F'TVORGN	X				
LOV_ORGANIZATION_LEVEL_2	ORGANIZATION_LEVEL_2	F'TVORGN	X				
LOV_ORGANIZATION_LEVEL_3	ORGANIZATION_LEVEL_3	F'TVORGN	X				
LOV_ORGANIZATION_LEVEL_4	ORGANIZATION_LEVEL_4	F'TVORGN	X				
LOV_ORGANIZATION_LEVEL_5	ORGANIZATION_LEVEL_5	F'TVORGN	X				
LOV_ORGANIZATION_LEVEL_6	ORGANIZATION_LEVEL_6	F'TVORGN	X				
LOV_ORGANIZATION_LEVEL_7	ORGANIZATION_LEVEL_7	F'TVORGN	X				
LOV_ORGANIZATION_POOL	ORGANIZATION_POOL	F'TVORGN	X				
LOV_ORG_FINANCIAL_MANAGER	ORG_FINANCIAL_MANAGER	SPRIDEN , F'TVORGN			X		
LOV_OUTCOME	STVDEGC	STVDEGC					
LOV_OUTCOME_STATUS	STVDEGS	STVDEGS					
LOV_PACKAGING_GROUP	RTVPGRP	RTVPGRP					
LOV_POSITION	NBBPOSN	NBBPOSN					
LOV_POSITION_CHANGE_REASON	PTRJCRE	PTRJCRE					
LOV_POSITION_CLASS	NTRPCLS	NTRPCLS					
LOV_POSITION_DEFERRED_PAY	PTRDFPR	PTRDFPR					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_POSITION_LOCATION	PTRJBLN	PTRJBLN					
LOV_POSITION_STATUS	POSITION_STATUS	STATIC (A,C,F,I)					
LOV_POST_CODE	GTVZIPC	GTVZIPC					
LOV_POST_SECONDARY_SCHOOL	POST_SECONDARY_SCHOOL	STVSBGI					
LOV_PREF_CLASS	PREF_CLAS	APBCONS					
LOV_PRIM_DISABILITY	STVDISA	STVDISA					
LOV_PRINCIPAL_INVESTIGATOR	PRINCIPAL_INVESTIGATOR	FRBGRNT		X			
LOV_PROGRAM	SMRPRLE	SMRPRLE					
LOV_PROGRAM_ATTR_SET	PROGRAM_SET_CODE	F'VATTS, F'TRPRGA	X				
LOV_PROGRAM_ATTR_TYPE	PROGRAM_ATTR_TYPE	F'VATTR, F'TRPRGA	X				
LOV_PROGRAM_ATTR_VALUE	PROGRAM_ATTR_VALUE	F'TRATTV, F'TRPRGA	X				
LOV_PROGRAM_CLASSIFICATION	STVCIPC	STVCIPC					
LOV_PROGRAM_LEVEL_1	PROGRAM_LEVEL_1	F'TVPROG	X				
LOV_PROGRAM_LEVEL_2	PROGRAM_LEVEL_2	F'TVPROG	X				
LOV_PROGRAM_LEVEL_3	PROGRAM_LEVEL_3	F'TVPROG	X				
LOV_PROGRAM_LEVEL_4	PROGRAM_LEVEL_4	F'TVPROG	X				
LOV_PROGRESS_EVALUATION	STVPREV	STVPREV					
LOV_PROJECT	ATVPROJ	ATVPROJ					
LOV_PROSPECT_STATUS	ATVPRST	ATVPRST					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_RATING	ATVRATE	ATVRATE					
LOV_RATING_TYPE	ATVRTGT	ATVRTGT					
LOV_RECEIVABLE_CATEGORY	TTVDCAT	TTVDCAT					
LOV_RECEIVABLE_CONTRACT	RECEIVABLE_CONTRACT	STVTERM, TBBCONT			X		
LOV_RECEIVABLE_DELIQUENCY	TTVDELI	TTVDELI, TBBACCT					
LOV_RECEIVABLE_DETAIL_CODE	TBBDETC	TBBDETC					
LOV_RECEIVABLE_EXEMPTION	RECEIVABLE_EXEMPTION	STVTERM, TBBEXPT			X		
LOV_RECEIVABLE_SOURCE	TTVSRCE	TTVSRCE					
LOV_RECRUITER	STVRECR	STVRECR					
LOV_REGISTRATION_REASON	STVRGRE	STVRGRE					
LOV_REGISTRATION_STATUS	STVRSTS	STVRSTS					
LOV_RESIDENCY	STVRESD	STVRESD					
LOV_REVIEW_TYPE	PTVREVT	PTVREVT					
LOV_SCHEDULE_TYPE	STVSCHD	STVSCHD					
LOV_SECONDARY_SCHOOL	SECONDARY_SCHOOL	STVSBGI					
LOV_SITE	STVSITE	STVSITE					
LOV_SOLICITATION_TYPE	ATVSOLC	ATVSOLC					
LOV_SOLICITOR_TYPE	ATVSOLT	ATVSOLT					
LOV_SOURCE_BACKGROUND	STVSBGI	STVSBGI					
LOV_STATE_PROVINCE	STVSTAT	STVSTAT					

List of Value View Name	Table Name	Table Name	Has Chart of Accounts Qualifier	Has PIDM Qualifier	Has Academic Period Qualifier	Uses EFFECTIVE_DATE and NEXT_CHANGE_D ATE logic	Uses PIDM as Value
LOV_STUDENT_CLASS	STVCLAS	STVCLAS					
LOV_STUDENT_POPULATION	STVSTYP	STVSTYP					
LOV_STUDENT_STATUS	STVSTST	STVSTST					
LOV_SUBJECT	STVSUBJ	STVSUBJ					
LOV_SUB_ACADEMIC_PERIOD	STVPTRM	STVPTRM					
LOV_TERMINATION_REASON	PTRTREA	PTRTREA					
LOV_TEST	STVTESC	STVTESC					
LOV_TEST_RULE	TEST	MGRSDAX_INTERNAL_CODE_GROUP					
LOV_TRACKING_GROUP	RTVTGRP	RTVTGRP					
LOV_VENDOR_TYPE	FTVVVTYP	FTVVVTYP				start date term date	
LOV_VETERAN_CATEGORY	STVVETC	STVVETC					
LOV_VISA	STVVVTYP	STVVVTYP					
LOV_WITHDRAW_REASON	STVWRSN	STVWRSN					
LOV_WORKER_COMPENSATION_CLASS	PTVWKCP	PTVWKCP					



# 6 Self-Service Reporting

Self-Service Reporting (SSR) provides simple, ad hoc access to information within Banner ODS.

SSR is delivered with report templates that provide examples of various common data retrieval needs across your institution. Each report template is based on the functional data relationships set forth in the Business Concept Diagrams found in Banner ODS published meta data. The template design for SSR uses a Filter - List - Detail approach. This approach includes a Search Criteria page where you select the various filters for executing a query, a List page that displays the results of that query, and a Detail Reports page where you access additional information specific to any individual result on the List page.

The information on the List and Detail Reports pages can be viewed online or exported to a .csv file (Microsoft Excel format, for example) or XML file for printing or additional manipulation. The Email icon enables you to send email to everyone on the List page. If you select an individual address from the List page, you can send email to that individual. Optionally, the search criteria may be saved as a Search Rule.

You can save a set of defined search criteria filters for a report template as a search rule for future use. For select templates, you can also save the unique primary identifier(s) for your List page results to Banner ODS as a population to use in custom reports developed with your third party reporting tool.

The following tasks are available to help you create a self-service report:

- View, select and execute search criteria
- View, sort, email or export the List results
- View, sort or export Detail reports
- Create, view, rename, change criteria for, or delete a Search Rule

For a Search Rule, optionally save the unique primary identifier(s) for a result set to Banner ODS using Population Selection

Before you can use the SSR, you *must* set up security. SSR requires authentication and authorization access to your Oracle database security. See the [“Security” on page 6-38](#) section at the end of this chapter for additional information.

## Note

By default, SSR uses Oracle user accounts for authentication and authorization. If you choose to use this default, then SSR will adhere to all Business Profiles and Fine Grain Access Security rules established for your Banner ODS users. Therefore, the data results returned for any SSR report template mirror the accessible data as defined for that SSR/Banner ODS user. If an SSR report template uses a particular Banner ODS

Reporting view to which a user has been denied access via your Oracle Access Controls, then the entire report template is not accessible to the user. ■

## Navigation Quick Reference

---

All SSR web pages use the same basic navigation techniques. The following table describes each navigation feature. You may want to print this page until you become familiar with how to navigate throughout the web pages.

This navigation. . .	Does this . . .
 View this Population Selection icon	Used to maintain Banner ODS populations. <ul style="list-style-type: none"> <li>• When selected from the top right corner of the home page: Opens the View Search Rules window showing all search rules with existing Banner ODS populations, for all templates.</li> <li>• When selected from the View Search Rules window, it opens the Population Detail window for the specified search rule.</li> </ul>
<b>Home</b> link	Returns you to the Home page.
<b>Help</b> link	Opens the help pages.
Breadcrumbs	Located at the top of the page, below the tabs. These indicate the levels you passed through to arrive at the current page. <p><i>Example</i> Home&gt;Student Templates&gt;Advisor’s Student Search Criteria&gt;Advisor’s Student List</p> Select any breadcrumb level to return to that level.
Subject Area tabs at the top	To access a different subject area, select that tab at the top of the page.
Headings that are underlined	On the Search Criteria page, select any headings that are underlined (for example, any of the individual search criteria filter headings) to open the online help window. The help contains either instructional or meta data related information. <p>On the List and Detail Reports pages, any underlined column headings can be selected to resort the results in ascending or descending order.</p>

This navigation. . .	Does this . . .
 Go to Search Rule icon	Select to load, view or maintain an existing search rule. Opens the View Search Rules window showing all existing search rules for the report template in context.
 Save Search Rule and Run Population Selection icon	Select to save a new, or resave an existing search rule. Opens the Save Search Rule window with optional save/refresh Banner ODS population functionality.
<b>Search</b> button	Executes a query based on the selected search criteria filters.
<b>Reset Search</b> button	Resets all search criteria filters to their default state.
<b>Show link</b>	Displays filters for any search criteria category.
<b>Hide link</b>	Hides all sets of grouped search criteria categories.
>	Shows or hides an individual set of grouped search criteria filters for a report template.
<b>Show SQL</b> button	Select to view the actual SQL used to execute a query on the Search Criteria page.  Select from the Detail Reports page to view the actual SQL used to generate the detail reports on the Detail Reports page.  <b>Note:</b> This button only appears if you have been granted access. See the “Customize Parameters” section of the SSR Installation Guide to allow or deny access for all users or individual users.
 View Detail icon	Opens the Detail Reports page for the displayed list results or the Search Rule Detail window.

## Search Criteria Page

The Search Criteria page is the filter portion of a report template. You can use this page to review and select the filters or search criteria on which to report.

Filters are grouped into logical search criteria categories. Each filter label is hyperlinked to Banner ODS meta data providing reporting view and column source information for each filter.

When you access a report template, it opens in its default state with a Search Rule setting of none. You can create a new query by selecting the desired search criteria filters, or load a previously saved query (see [“Search Rules” on page 6-9](#) section). When finished, select the **Search** button to execute the query.

The Search Criteria page retains all defined filters as long as the report template is within a current session (moving between Search Criteria, List and Detail Reports pages). This allows you to easily alter or add search criteria. If you want to execute a different query, select the **Reset Search** button to clear all filter selections and return the Search Criteria page to its default settings or load a different Search Rule.



#### Tip

To select multiple, random values from a list box, select the first value, then hold down the **Ctrl** key while selecting the remaining values.

To select multiple values in sequence from a list box, select the first criteria then hold down the **Shift** key and select the last criteria.

A list box with a (defaulted) value of 'ALL' means the filter is ignored, unless a value/values are selected.

For Range filters, leaving either range blank acts as a wildcard. ■

The following options are available from the Search Criteria pages:

- View and select the desired search criteria filters and execute a query
- Save, load, modify or delete search rules and optionally save, refresh or delete Banner ODS Populations for the template in context (See the [“Search Rules” on page 6-9](#) section)
- View the SQL used to execute a defined query and generate the List Page report

## Recommended and Required Search Criteria

Some report templates include a Recommended Search Criteria category that contains the filters most commonly selected when using a particular template. These may also include one or more 'required' filters. A required filter must be selected to execute a query. Examples of required filters are Academic Periods or Chart of Accounts.



#### Note

Required filters are preceded with an asterisk. ■

## Dependant Search Criteria Filters

Several report templates contain one or more list of values (LOV) filters that must be manually populated after you select a required filter. To load these filters for use in your query, choose your value(s) for the required filter and select the **Populate Search Criteria for...** button. You need to reload these filters any time you change the corresponding

required filter for a new query. See the [“Report Templates” on page 6-18](#) section for additional information.

 **Note**

In their default state, dependant filters will display the following: “Select [required filter name] and Populate.” ■

## List of Values Search Criteria Filters

The list of values (LOV) search criteria filters found in the various SSR reporting templates contain a set of valid values for a corresponding column in a Banner ODS reporting view. These filters are generated from a series of LOV views contained within the ODSLOV schema within Banner ODS. The ODSLOV views obtain their information from Banner ODS composite table called MGT\_VALIDATION, which in turn is populated with validation table values found in your Banner database.

Since these LOV filters are sourced from your Banner validation tables, querying on certain values may produce no results, if those value(s) are not currently associated with any records in your Banner ODS database.

## Show SQL

Select the **Show SQL** button to view the actual SQL used to execute a query on the Search Criteria page, and to display the SQL in a pop-up window. This button only appears if you have been granted access. See the “Customize Parameters” section of the SSR Installation Guide to allow or deny access for all users or individual users.

## List Page

---

The List page shows the results of the query that was executed on the Search Criteria page, and includes a predefined set of information for each result. The following procedures can be performed from this page:

- View and sort the results
- Export the List page report as a .csv file (Microsoft Excel format, for example) or an .xml file (except for the Employee List page)
- Send emails
- Save, load, modify or delete search rules and optionally save, refresh or delete Banner ODS Populations for the template in context (See the [“Search Rules” on page 6-9](#) section)
- Change the ‘Records Per Page’ display setting
- Display the Search Criteria used to generate the List page results

- Access the Detail Reports page

## Export

List results can be saved to format, print or further manipulate in another reporting tool by exporting the results as a .csv file (Microsoft Excel format, for example) or an .xml file. Select the **Excel** icon (for a .csv file) or the **XML** icon (for another application). The File Download window opens. Indicate whether you want to save or open the file.

### Note

Some List page reports contain significantly more columns of information when they are exported to .csv and .xml files than are viewable on the web page. Review the “Report Templates” section for information specific to each template. ■

## Email

The email option enables you to send an email message to any individual on the List page, or to send an email to the entire list. Each option is explained below:

### Individual

To send an email to an individual on the List page, select the email address link in the **Email Address** column for that person. Your local email program opens with the individual’s email address already entered.

### Entire List

To send an email to the entire list, select the envelope icon at the top of the List page. The SSR email utility opens. All recipient email addresses (the individuals on the list report) load into the **Blind Copy** field to ensure that recipients’ cannot see the other email addresses on the distribution list. If you have set up each SSR user as an APEX user, then the user’s email address loads into the **To** field. If you are not using the APEX user accounts, the user must manually enter (their) email address in the **To** field.

## Sort

Any underlined column in the list report can be used to toggle between sorting the results in ascending or descending order. Select the column name by which you want to sort. An up arrow appears if the column is sorting in ascending order. A down arrow appears if the column is sorting in descending order.

## Records Per Page (Display Setting)

The List page displays the total count of all records found for any given query as well as a “Records Per Page” display setting. This setting indicates the maximum number of records to be displayed on the List page.

If the total records returned for a query exceeds the Records Per Page display setting, a sequence of pagination links appear above the List results page. You must select the pagination link to retrieve the (next) set of results. If the total records returned for a query is less than the Records Per Page display setting, all records appear in the List results page.

This feature helps query performance, or more specifically, the amount of time it takes to render the results for HTML display. The delivered default display setting is 100 records per page. This default setting can be changed at the institution level. You may also change the display setting for your current session by selecting the **Records Per Page** hyperlink.

### Note

This display setting applies to the HTML List page only. The .csv/.xml export and Banner ODS Population features save all results, as applicable. ■

## Access Detail Reports

Each report template provides detail reports for all results displayed on the List page. To drill to the Detail Reports page, select  in the left column for any result on the list report.

### Note

For the Finance templates (in addition to the View Detail icon) you can also drill into the respective detail reports by clicking the hyperlinked amount totals in the result set. ■

## Detail Reports Page

---

The Detail Reports page displays all reports for the result selected from the List page. The following procedures can be performed from this page:

- Review the detail reports for the result selected from the List page and select a different detail report
- Export a detail report as a .csv file (Microsoft Excel format, for example)
- View the SQL used to generate the detail reports

## Detail Reports Drop-down List

The Detail Reports page has a drop-down list located at the top of the page. Use this list to select and display any individual detail report.

## Sort

Any underlined column in a detail report can be used to toggle between sorting the results in ascending or descending order. Select the column name by which you want to sort. An up arrow appears if the column is sorting in ascending order. A down arrow appears if the column is sorting in descending order.

## Export

Detail reports can be saved as a .csv file to format, print or further manipulate in another reporting tool. Select the **Excel** icon. The File Download window opens. Indicate whether you want to save or open the file.

### Note

On the Detail Reports page, only reports with an Excel icon can be exported. This option is provided for any report that can display more than one row of information.

Some Detail reports may contain more columns of information when they are exported than are viewable on the web page. Review the [“Report Templates” on page 6-18](#) section for details about each template. ■

## Query Search Criteria

---

Self-Service Reporting (SSR) provides ad hoc access to information within Banner ODS. You can use the report templates to access commonly retrieved data from across your institution. Each report template is based on the functional data relationships set forth in the Business Concept Diagrams found in Banner ODS published meta data.

You can query search criteria on a one time basis or saved as a search rule for future reuse. See [“Search Rules” on page 6-9](#) [“Search Rules” on page 6-9](#) for additional information.

1. Select a business area from the Home page.
2. Select a template.
3. Select your search criteria filters From the Search Criteria page.

The Search Criteria page opens. Use this page to review and select the filters or search criteria on which to report.

For additional information on how to move throughout the SSR pages, see the [“Navigation Quick Reference” on page 6-2](#).

#### 4. Click **Search**.

The List page opens showing the results of the query that was executed on the Search Criteria page. It includes a predefined set of information for each result. From this point, you can continue to:

- View, select and execute search criteria
- View, sort, email or export the List results
- View, sort or export Detail reports

## Search Rules

---

Self-Service Reporting (SSR) enables you to select information that you frequently search on within a template (called search criteria filters), then name and save the filters as a search rule under a user-defined name. This makes it easy to reuse sets of search criteria filters.

### **Note**

You can only access the search rules that are saved under your user name.

You also have the option to indicate whether to create/refresh Banner ODS populations each time you save a search rule. See [“Banner ODS Populations” on page 6-13](#) for additional information.

## Create and Save a Search Rule

Use this procedure to save groups of search criteria that you want to reuse. Search rules can be created from the Search Criteria page or from the List page.

Populations can also be created for search rules. See [“Banner ODS Populations” on page 6-13](#) for additional information on Banner ODS populations.

1. Select a business area from the Home page.
2. Select a template.
3. Select your search criteria filters from the Search Criteria page.

4. Click . (The Save Search Rule and Run Population Selection icon is also available from the List page.)
5. Enter the name of the search rule,
6. (optional) Select the **Create/Refresh Banner ODS Population** check box to save the Banner ODS population for this search rule.

See [“Banner ODS Populations” on page 6-13](#) for additional information on Banner ODS populations.



#### Note

Populations cannot be saved for the Finance templates. ■

7. Click **Save**.

## Load a Search Rule

Use this procedure to load and display a different search rule. Search rules can be loaded from the Search Criteria page or from the List page. (If a population exists for the search rule, then the page can also be accessed using the View this Population Selection icon  on the Home page. See [“Banner ODS Populations” on page 6-13](#) for additional information.)

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Select the search rule to load.

The Search Criteria page returns automatically with the search rule loaded.

## Update a Search Rule

Update a search rule after you have changed or added search criteria filters for an existing rule, and want to save those changes.

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)

4. Select the search rule you want to update.

The screen returns to the Search Criteria page with the rule loaded.

5. Change the search criteria filters.

6. Click . (The Save Search Rule and Run Population Selection icon is also available from the List page.)

The Save a Search Rule window opens.

7. (optional) Select the **Create/Refresh Banner ODS Population** check box. See [“Banner ODS Populations” on page 6-13](#) for additional information on Banner ODS populations.

 **Note**

If the existing search rule has a population saved to Banner ODS, you *must* select the **Create/Refresh Banner ODS Population** check box to update and refresh the population. If this box is not selected for an existing search rule, any previously saved population is deleted. See [“Banner ODS Populations” on page 6-13](#) for additional information on Banner ODS populations. ■

 **Note**

Populations cannot be saved for the Finance templates. ■

8. Click **Save**.

## Save as Another Search Rule (Save As)

Use this procedure to create a new search rule with the same search criteria as an existing search rule, but with a different name.

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)

4. Select the search rule you want to load and save under a different name.

The screen returns to the Search Criteria page with the rule loaded.

5. Click . (The Save Search Rule and Run Population Selection icon is also available from the List page.)

The Save a Search Rule window opens.

6. Enter the new name.
7. (optional) Select the **Create/Refresh Banner ODS Population** check box. See [“Banner ODS Populations” on page 6-13](#) for additional information on Banner ODS populations.

 **Note**

If the existing search rule has a population saved to Banner ODS, you *must* select the **Create/Refresh Banner ODS Population** check box to also save a population for this rule. See [“Banner ODS Populations” on page 6-13](#) for additional information on Banner ODS populations. ■

 **Note**

Populations cannot be saved for the Finance template. ■

8. Click **Save**.

## Rename a Search Rule

Use this procedure to change the name of an existing search rule. You do not have to load the search rule to rename it.

Search rules are renamed from the Search Rule Detail window which can be accessed from the View Search Rules window. If a population exists for the search rule, then the page can also be accessed from the Home page. (See [“Banner ODS Populations” on page 6-13](#) for additional information.)

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available on the List page.)
4. Click  to open the Search Rule Detail window.

 **Note**

If the rule has a population, you can also click the  from the Home page to open the View Search Rules window. See [“Banner ODS Populations” on page 6-13](#) for additional information. ■

5. Enter the new name in the **Rule Name** field.
6. Click **Rename**.

The search rule is saved under the new name.

## Delete a Search Rule

Use this procedure to delete an existing search rule. Search rules are deleted from the View Search Rules window or from the Search Rule Detail window. (If a population exists for the search rule, then the page can also be accessed from the Home page. See [“Banner ODS Populations” on page 6-13](#) for additional information.)

### Note

If you delete a search rule with Banner ODS a Banner ODS population, you also delete the populations. ■

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Use any of the following methods from the View Search Rules window:
  - Click **Delete** to remove the search rule
  - Click  to delete the search rule from the Search Rule Detail window
5. From the corresponding Detail window, click **Delete** and indicate that you want to delete the rule.

The search rule is deleted.

## Banner ODS Populations

---

Banner ODS populations are predefined primary identifier(s) found within List page results that can be saved to Banner ODS to populate reports developed using a third party reporting tool. (See [“Use Populations with Banner ODS” on page 6-17](#) for information on retrieving population detail from the ODS\_Population reporting view in Banner ODS.)

Below is a list of Banner ODS population characteristics:

- Finance report templates cannot save populations.
- Banner ODS population is optional, and can be disabled when SSR is installed
- Populations are associated with a search rule.

- A search rule must be created before you can create a Banner ODS population for that rule. (See [“Create and Save a Search Rule” on page 6-9](#) for additional information.)
- Only one population is allowed per search rule
- Populations can be used for custom reporting with other reporting tools against the Banner ODS.
- Populations cannot be reused within SSR.

## Create Banner ODS Populations

Use this procedure to create populations for a search rule. You can create populations while you are creating the search rule, or from any page or window that contains a  icon.

### *Prerequisites*

A search rule must be created before you can create Banner ODS populations for that rule. See [“Create and Save a Search Rule” on page 6-9](#) for additional information.

### *Create a Population with a Search Rule Loaded*

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Select the search rule to load.

The Search Criteria page returns automatically with the search rule loaded.

5. Click  on the Search Criteria page. (The Save Search Rule and Run Population Selection icon is also available from the List page.)
6. Select the **Create/Refresh Banner ODS Population** check box.
7. Click **Save**.

The population is added to the selected search rule.

### ***Create a Population without a Search Rule Loaded***

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Select  in the **Populations Details** column that corresponds to the search rule for which you want to create a population.

The number next to the  indicates the number of rows of populations available for that search rule. If this number is zero, then a **Create** button displays on the Banner ODS Population Detail window. (If rows are available, then the displayed buttons are **Refresh**, **Delete** and **SQL** display.)

5. Click **Create**.

A population is created for the search rule.

## **Refresh (Update) Banner ODS Populations**

Use this procedure to refresh a population because information has been added or removed (for example, alumni were added). Three possible ways to refresh your populations appear below.

### ***Prerequisites***

At least one population must exist for a search rule. See [“Create Banner ODS Populations” on page 6-14](#) for additional information.

### ***Refresh with a Search Rule Loaded***

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Select the search rule to load.

The Search Criteria page returns automatically with the search rule loaded.

5. Click  on the Search Criteria page. (The Save Search Rule and Run Population Selection icon is also available from the List page.)
6. Select the **Create/Refresh Banner ODS Population** check box.
7. Click **Save**.

The population is refreshed for the selected search rule.

#### ***Refresh without a Search Rule Loaded***

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Click the  that corresponds to the search rule whose population you want to refresh.
5. Click **Refresh**.

The population is refreshed (updated).

#### ***Refresh from the Home page***

1. Click  in the upper right hand corner of the page.
2. Click the  that corresponds to the search rule whose population you want to refresh.
3. Click **Refresh**.

The population is refreshed (updated).

## **Delete a Banner ODS Population**

Use this procedure to delete a population associated with a search rule.

### **Note**

Deleting the population does not delete the associated search rule. ■

Two possible methods of deleting a population appear below.

### ***Delete with a Search Rule not Loaded***

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Click the  that corresponds to the search rule whose population you want to delete.
5. Click **Delete**.

The population is deleted.

### ***Delete from the Home page***

1. Click  in the upper right hand corner of the Home page.
2. Click the  that corresponds to the search rule whose population you want to delete.
3. Click **Delete**.

The population is deleted.

## **Use Populations with Banner ODS**

Banner ODS populations contain the predefined primary identifier(s) for the List page results of any given report template query. These populations can be saved to Banner ODS and used for generating reports developed using a third party reporting tool. (See [“Banner ODS Populations” on page 6-13](#) for additional information.)

Banner ODS contains a schema called SSRMGR. The tables in this schema store search rule and population data. A view in this schema, called ODS\_POPULATION, contains the unique identifiers for each saved population along with the distinguishing characteristics of the corresponding search rule and user.

The search rule parameters below are required to retrieve a population from the ODS\_POPULATION view for reporting purposes. They are used in the sql statement to retrieve the desired population.

- TEMPLATE\_NAME
- RULE\_NAME
- USER\_ID

The **SQL** button in the Banner ODS Population Detail window generates a SQL statement containing these parameters that can be used to retrieve a saved population from the ODS\_POPULATION reporting view.

#### ***Access from the View Search Rules Page***

1. Select a business area from the Home page.
2. Select a template.
3. Click  from the Search Criteria page. (The Go to Search Rule icon is also available from the List page.)
4. Click the  link that corresponds to the search rule whose population you want to select.
5. Click **SQL**.

The unique identifiers for the saved population and the characteristics of the corresponding search rule and user display. You can use this information to create additional reports.

#### ***Access from the Home page***

1. Click  in the upper right hand corner of the Home page.
2. Click the  that corresponds to the search rule whose population you want to select.
3. Click **SQL**.

The unique identifiers for the saved population and the characteristics of the corresponding search rule and user display. You can use this information to create additional reports.

## **Report Templates**

---

This section contains the following information for each delivered report template:

- Search criteria
- List reports
- Detail reports
- Notes

# Accounts Receivable Report Templates

Use the Accounts Receivable report template to obtain reports from Receivable Customer.

## Receivable Customer

Use this report template to:

- Obtain a list of students or organizations and their current balances
- Obtain a list of students or organizations that have transactions that meet a specific transaction category within an academic period
- Determine which students that have holds on their accounts or have bills due within a specific date range
- Obtain a list of students or organizations that have a range of current amounts dues on their account
- Determine current balances of accounts where students are in specific programs, departments, degrees and majors
- Determine the contracts or exemptions with which a student is associated or an organization is associated
- Review all charges and payments on a students or organization's account
- Review the accounting information sent to Finance for all charges, payments, and application of payments for a selected student or organization

## Search Criteria

Required Search filters: At least one Academic Period

Recommended Search filters: Category Detail Code, or Source

## Search Criteria Notes

Certain list of values (LOV) search criteria filters in this template require the selection of one or more Academic Periods to display a specific list of values for that filter. To load these filters to use in your query, choose the desired Academic Period(s) and select the **Populate Search Criteria for Academic Period(s) Selected** button. You need to reload these filters any time you change the Academic Period(s) for a new query.

## List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, current amount due, account balance, delinquency, hold count, non-sufficient funds count, collection agency count and city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the List page for this report template is available using the following:

- Export Options: More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page.
- Banner ODS Population: A population saved for this template includes the distinct Entity UIDs for your query result set.

## Detail Reports

The following detail information can be accessed for any student or organization on the list report as appropriate:

- Current Addresses
- Other Phone Numbers
- Current Internet Address
- Receivable Summary By Category
- Receivable Summary
- Customer Account Details
- Customer Account Detail Accounting
- Customer Accounting Summary
- Application of Payment Detail Accounting
- Application of Payment Detail Accounting Summary
- Receivable Tax Detail History
- Receivable Tax Detail History Summary
- Deposit History
- Deposit History Summary
- Contract History
- Exemption History
- Installment Plan History
- Collection Agency Assignment
- Holds

## Notes

The Receivable Customer reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.

## Advancement Report Templates

Use the Advancement report template to obtain reports from Advancement Person.

### Advancement Person

Use this report template to:

- Locate constituents in a particular geographic area
- Analyze participation or giving trends
- Profile or segment your constituent population

### Search Criteria

Required Search filters: None

Recommended Search filters: None

### List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, spouse name, various constituent indicators, email address and the city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the list page for this report template is available using the following:

- Export Options: More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page. These include the Entity UID and formatted (preferred) mailing address.
- Banner ODS Population: A population saved for this template includes the distinct Entity UIDs for your query result set.

### Detail Reports

Access the following detail information for any individual on the list report:

- Constituent Detail
- Current Addresses

- Other Phone Numbers
- Current Internet Address
- Demographics
- Medical Information
- Veteran Status
- Employment History
- Relationships
- Degree Summary
- Activities and Leadership Roles
- Donor Categories
- Giving History
- Membership
- Mailings
- Exclusions

## Notes

The Advancement Person reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.

## Finance Report Templates

Use the Finance report templates to obtain financial reports from the General Ledger or Operating Ledger.

### General Ledger

Use this report template to:

- Quickly determine if your fund is in balance
- Obtain an asset balance at any fund level
- Obtain fund reports by financial manager or principal investigator
- Create a general ledger report by specific reporting attributes
- Roll up general ledger balances to a higher level within fund and/or account hierarchy

## Search Criteria

Required Search filters: At least one Chart of Accounts and at least one Fiscal Year

Recommended Search filters: Fiscal Period

### Search Criteria Notes

Certain list of values (LOV) search Criteria filters in this template require you to select one or more charts to display a specific list of values for that filter. To load these filters to use in your query, choose the desired Chart(s) and select the **Populate Search Criteria for Chart(s) Selected** button. You need to reload these filters any time you change the Chart(s) for a new query.

When including fund, fund type, account or account type attributes, and not selecting specific attributes as a filter, the lines displayed on the List Page may occur more than once for each unique combination of fund and account. This is based on the number of attributes assigned to each fund, fund type, account, and/or account type within the source system. When the lines are not unique for each fund and account, this affects the total of the amounts displayed in the General Ledger List Summary Report. To avoid duplicate lines, select the specific attributes on which you wish to report.

### Working with Roll Fund or Roll Account Search Criteria

Leave the radio button defaulted to *E* and select a specific level value to report on all funds that report to a specific level fund or fund type, as well as to report on all accounts that report to a specific level account or account type.

If you choose one of the level radio buttons, the list report totals the amounts to that level and displays it at that level. The lower levels no longer display as columns in the list report.

#### Example for level 1

You might choose to list amounts for all level 1 fund types with all their level 2 account types.

If you choose one of the level radio buttons and choose specific fund level values, fund type level values, account level values, or account type level values, the list report displays the selected values with the amounts totaled for that unique combination of selected filters.

#### Example for level 2

You might choose *Restricted* for a fund type level 2. Select the radio button of 1 for Roll Fund Type and choose Roll Account Type Level 2 with the values. This generates a list report of amounts totaled to the Restricted Fund Level 1 for all account type level 2 values.

If you select the roll radio button for any level other than *E*, the fund and account being rolled to will display in the fund and account column in the General Ledger list report.

## List

The General Ledger List report is dynamically built according to selected search criteria to support reporting attributes and roll-up features. This prevents the normal sort feature from being used. Thus, you will not see any underlined columns in the List report for sorting.

This list report provides two sets of results based on the entered search criteria:

**General Ledger List:** Data includes beginning balance, current actual and ending balance for each fund and account or selected levels of fund and account and /or reporting attributes.

**General Ledger List Summary:** Provides a summary of all fund and account amounts displayed in the General Ledger List with a beginning balance, current actual, and an ending balance.

Additional information applicable to the list page for this report template is available using the following:

- **Export Options:** More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page.

Additionally, when a List report is generated using the roll-up feature, more rows are exported with the CSV or XML file than what is viewable online. This is because the online version summarizes information such as chart columns from the general ledger for display on the List page. The data in the export file is not summarized; but instead includes each detail line that meets the queried search criteria.

- **Banner ODS Population:** Not available for finance report templates.

## Detail Reports

These reports provide full access to supporting detail for any general ledger line on the list report:

- **General Ledger Line:** detail report includes one or more detail general ledger lines with report totals. Multiple detail general ledger lines may exist if a search was performed for a report roll-up.
- **Transaction Detail:** report lists information supporting the general ledger line(s). This data includes the fund, organization, account, program, activity, and location as well as field code, journal type, journal description, and source document key information.
- **Transaction Detail Total**

More fields of information are provided with the exportable .csv file detail reports than are viewable on the web page.

## Notes

- The General Ledger reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.
- The only transaction detail lines that currently display are those that directly update the general ledger. Thus, operating ledger transaction detail and encumbrance ledger transaction detail do not display within this report.

## Operating Ledger

Use this report template to:

- Obtain departmental reports by department financial manager
- Obtain reports by financial manager or principal investigator
- Quickly determine departmental budget available balance
- Create a departmental report by reporting attributes
- Roll up operating ledger available balances to a higher level within organization, fund, account, and program and/or location hierarchy
- Obtain a list of all expense or revenue transactions

## Search Criteria

Required Search filters: At least one Chart of Accounts and one Fiscal Year

Recommended Search filters: Fiscal Period

### *Search Criteria Notes*

Certain list of values (LOV) search Criteria filters in this template require the selection of one (or more) Charts to display a specific list of values for that filter. To load these filters to use in your query, choose the desired Chart(s) and select the **Populate Search Criteria for Chart(s) Selected** button. You need to reload these filters any time you change the Chart(s) for a new query.

When including fund, fund type, account, account type, organization, or program attributes, and not selecting specific attributes as a filter, the lines displayed on the List Page may occur more than once for each unique combination of organization, fund, account, program, activity and location. This is based on the number of attributes assigned to each accounting distribution element within the source system. When the lines are not unique for each FOAPAL combination, this affects the total of the amounts displayed in

the Organization Budget Status Summary Report. To avoid duplicate lines, select the specific attributes on which you wish to report.

### **Working with Roll Search Criteria**

Leave the radio button defaulted to *E*, and select a specific level value to report on all organizations that report to a specific level organization, specific level fund or fund type, specific level account or account type, specific level program, as well as to report on all locations that report to a specific level location.

If you select one of the level radio buttons, the list report totals the amounts to that level and displays it at that level. The lower levels no longer display as columns in the list report.

### **Example**

You might choose to list amounts for all level 1 organizations with all their level 2 account types.

If you choose one of the level radio buttons and choose specific organization level values, fund level values, fund type level values, account level values, account type level values, program level values, or location level values, the list report displays the selected values with the amounts totaled for that unique combination of selected filters.

### **Example**

You might choose *Restricted* for a fund type level 2. Select the radio button of *1* for Roll Fund Type and choose *Roll Account Type Level 2* with the values. This generates a list report of amounts totaled to the Restricted Fund Level 1 for all account type level 2 values.

If you select the roll radio button for any level other than *E*, the organization, and/or fund, and/or account, and/or program, and/or location being rolled to displays in their respective columns in the Organization Budget Status list report.

## **List**

This list report provides results based on the entered search criteria:

Organization Budget Status List: Current Period Activity. Year-to-date remaining balance, year-to-date adjusted budget, year-to-date activity, and year-to-date commitments for each accounting distribution or selected levels of the accounting distribution and/or reporting attributes.

Organization Budget Status Summary: Provides a summary by organization only of the chart, fiscal year and period, current period activity, year-to-date remaining balance, year-to-date adjusted budget, year-to-date activity, and year-to-date commitments displayed in the Organization Budget Status List. The report total takes into consideration the normal

balance of the account summarized within the department. If the normal balance of an account is a C for credit, the amount is multiplied by a -1, and then added into the summary total.

**Suppress Zero Activity Detail Report Lines:** To control the number of lines that appear in the Operating Ledger Lines Detail Reports page, select *Yes* from the **Suppress Zero Activity Detail Report Lines** drop-down list. The operating ledger lines that do not have current activity are not listed on the Detail Reports page. If you select *No*, then all operating ledger lines that support the selected List line display regardless of activity. This feature is useful when a List report was requested at a roll-up level, then listing the supporting operating ledger lines on the Detail Reports page. There may be hundreds of departments that had no activity for the period. Reporting to a higher level organization and listing them on the Detail Report page makes it difficult to view the organizations that did have activity.

Additional information applicable to the list page for this report template is available using the following:

- **Export Options:** More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page.

Additionally, when a List report is generated using the roll-up feature, more rows are exported with the CSV or XML file than what is viewable online. This is because the online version summarizes information such as chart columns from the operating ledger for display on the List page. The data in the export file is not summarized; but instead includes each detail line that meets the queried search criteria.

- **Banner ODS Population:** Not available for finance report templates.

**Selecting Current Period Detail or Fiscal Year to-date Detail:** A detail report can only be created for the current period selected by selecting the bolded amount under the **Curr Prd Activity** column. To obtain a detail report of all fiscal periods up to and including the current period, select the bolded amount under the **YTD Remaining Balance** column. If you select the latter amount, the number of lines in your detail report will increase. This is another reason why the **Suppress Zero Activity Detail Report Lines** drop-down list defaults to *Yes*.

 **Note**

The Organization Budget Status List report is dynamically built according to selected search criteria in order to support reporting attributes and roll-up features. This prevents the normal sort feature from being used. Thus, you will not see any underlined columns in the List report for sorting. ■

## Detail Reports

These reports provide full access to supporting detail for any Organization Budget Status line on the list report:

- The Operating Ledger Lines detail report includes one or more detail operating ledger lines with current period activity and report totals. If the normal balance of the line's account is a Credit, the amount displayed is multiplied by a -1. Thus a positive amount may display in the List Report, but that same amount may display as a negative in the Detail Report. This is to ensure the Report Totals are correct by considering the normal balance of the various accounts.
- The Transaction Detail report lists information supporting the operating ledger line(s). This data includes the fund, organization, account, program, activity, and location as well as journal type, journal description, and source document key information. The amount of the transaction is multiplied by a -1 if the normal balance of the account is a Credit. To allow for improved reconciliation between the transaction detail and the operating ledger lines, the field code is broken out into respective amount columns that updated the operating ledger. Thus, if the transaction had a **field\_code** value of *04*, the amount is displayed in the **Curr Prd Encumbrances** column. The breakdown of field code is as follows:
  - 01 = Curr Prd Adopted Budget
  - 02 = Curr Prd Budget Adjustments
  - 03 = Curr Prd Activity
  - 04 = Curr Prd Encumbrances
  - 05 = Curr Prd Budget Reservations
  - 06 = Curr Prd Accumulated Budget
  - 07 = Curr Prd Temporary Budget

## Notes

- The Operating Ledger reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.
- The only transaction detail lines that currently display are those that directly update the operating ledger. Thus, general ledger transaction detail and encumbrance transaction detail do not display within this report. If an Operating Ledger line was selected that did not have any current activity, no transaction detail lines will display.

## Financial Aid Report Templates

Use the Financial Aid report template to obtain Financial Aid Award and Disbursement reports.

### Financial Aid Awards

Use this report template to:

- Determine who has been awarded a specific Financial Aid fund (or group of funds) during a particular Academic Period (or group of Academic Periods)
- Determine who has had financial aid disbursed during a particular academic period (or group of academic periods)
- Determine the status of a particular award (or group of awards) during a particular academic period (or group of academic periods)
- Answer demographic questions about the populations of students awarded financial aid during a particular academic period (or group of academic periods)

## Search Criteria

Required Search filters: At least one Academic Period

Recommended Search filters: Fund, Fund Source Type, Financial Aid Type

## List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, academic period, financial aid information, student status information email address and the city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the list page for this report template is available using the following:

- Export Options: More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page. These include the Entity UID and formatted (preferred) mailing address.
- Banner ODS Population: A population saved for this template will include the distinct combinations of Entity UIDs and Academic Periods for your query result set.

## Detail Reports

The following detail information can be accessed for any student on the list report:

- Current Addresses
- Other Phone Numbers
- Current Internet Address
- Applicant Status
- Award By Person
- Award Disbursement
- Academic Study

- Enrollment Information
- Financial Aid Enrollment
- Academic Information
- Academic Standing
- Satisfactory Academic Progress
- Holds
- Demographics
- Medical Information
- Veteran Status
- International Details

### Notes

The Financial Aid Awards reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.

## Human Resources Report Templates

Use the Human Resources report template to obtain employee reports.

### Employee

Use this report template to:

- Analyze employee demographics
- Download contact, demographic and primary position information about an employee
- Look up detailed information about a particular employee

### Search Criteria

Required Search filters: None

Recommended Search filters: Employee Status, Employee Class, Leave Category, and Benefit Category.

## List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, demographic and employee status information, email address and the city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the list page for this report template is available using the following:

- **Export Options:** More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page.
- **Banner ODS Population:** A population saved for this template will include the distinct Entity UIDs for your query result set.

## Detail Reports

The following detail information can be accessed for any employee on the list report:

- Current Addresses
- Other Phone Numbers
- Current Internet Addresses
- Benefits (Current Year)
- Beneficiaries
- Leave Balances
- Bargaining Units
- Certifications
- Skills
- Tax Deductions (Current Year)
- Review History
- Position History
- Earning History
- Demographics
- Medical Information
- Veteran Status
- International Details
- Employment History

## Notes

- The Human Resources reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.
- Do not use this report template if you expect to retrieve information pertaining to an employee's secondary position(s).

## Student Report Templates

Use the Student report templates to obtain reports relating students to Advisors, students applying for admission or student enrollment.

### Advisor's Students

Use this report template to:

- Retrieve information for specific advisees when meeting with a group of advisees
- Identify a group of students that fit a set of criteria to contact that group of students, such as:
  - Students within an advising type responsibility
  - Students from a geographic region (Nation, State/Province)
  - Students who are international students
  - Students in academic difficulty
  - Students receiving financial assistance
  - Review the assignments made to a group of advisors

### Search Criteria

Required Search filters: At least one Academic Period and the student grouping you wish to see. This will be the group of students currently assigned to an advisor, the group that has never been assigned to an advisor or the group that does not have a current advisor assignment.

Recommended Search filters: Varies based on the group to be reviewed by the advisor.

### Search Criteria Notes

Certain list of values (LOV) search criteria filters in this template require the selection of one (or more) Academic Periods to display a specific list of values for that filter. To load these filters to use in your query, choose the desired Academic Period(s) and select the Populate Search Criteria for Academic Period(s) Selected button. You will need to reload these filters any time you change the Academic Period(s) for a new query.

## List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, academic period, assigned advisor, summary student information, email address and the city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the list page for this report template is available using the following:

- **Export Options:** More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page. These include the Entity UID, formatted (preferred) mailing address and Advisor UID.
- **Banner ODS Population:** A population saved for this template will include the distinct combinations of Entity UIDs and Academic Periods for your query result set.

## Detail Reports

The following detail information can be accessed for any student on the list report:

- Current Addresses
- Other Phone Numbers
- Current Internet Address
- Student Advisor(s)
- Academic Study
- Enrollment Information
- Academic Information
- Academic Standing
- Holds
- Student Courses
- Student Course Grades
- Student Course Attributes
- Student Course Meeting Times
- Demographics
- Medical Information
- Veteran Status
- International Details

- Activities
- Latest Secondary School
- Latest Post Secondary School
- Test Scores
- Employment History

## Notes

- The Advisor's Students reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.
- The Advisor's Students List can be used to download the contact and summary information for the selected group of students advised.
- The Advisor's Students List is designed to retrieve the students being advised for a specified academic period. Therefore, you would not use this to retrieve all the students ever advised by a specific advisor.
- Selection must include one of the following groups of students:
  - Currently assigned to an advisor
  - Never been assigned to an advisor
  - Does not have a current advisor assigned.
- Search Criteria filters in this template require an Academic Period to display a specific list of values for that filter. To load these filters to use in your query, select or change to the desired Academic Period and select the **Populate Search Criteria** button.
- Detail reports display data for the students selected for all academic period independent of the academic period in the selection criteria.

## Admissions Application

Use this report template to:

- Identifying admissions applications that are complete and ready for review
- Monitor application status and review admission application details
- Compile details of applicants matching a set of criteria for further review
- Track admissions application decisions by college and or department

## Search Criteria

Required Search filters: At least one Academic Period

Recommended Search filters: None

## List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, academic period, student level, application complete indicator, program, degree, college, major, department, campus, site, enrolled indicator, latest decision, email address and the city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the list page for this report template is available using the following:

- **Export Options:** More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page. These include the Entity UID and formatted (preferred) mailing address.
- **Banner ODS Population:** A population saved for this template will include the distinct combinations of Entity UIDs and Academic Periods for your query result set.

## Detail Reports

The following detail information can be accessed for any student on the list report:

- Current Addresses
- Other Phone Numbers
- Current Internet Address
- Admissions Application
- Application Academic Study
- Admissions Rating
- Admissions Decisions
- Application Deposit Detail
- Financial Aid Information
- Admissions Attributes
- Admissions Cohorts
- Admissions Requirements
- Additional Information Counts
- Recruitment Information Detail
- Application Additional Information

- Demographics
- Medical Information
- Veteran Status
- International Details
- Latest Secondary School
- Latest Post Secondary School
- Test Scores
- Employment History

## Notes

- The Admission's Application reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.
- Multiple persons at the institution will need to review the data supplied by an applicant for admission and this report template is primarily to pull together for that administrator, reviewer, rater, all the data stored in the system for the applicant into a concise report of the information.
- While multiple academic periods may be used for selection criteria, it is recommended that the admissions applications be looked at for a single academic period at a time. This would correspond to normal application business processing.

## Enrolled Students

Use this report template to:

- Review student enrollments by academic period and program attributes
- Track student course registration activity
- Retrieve a list of students registered in courses with missing scheduling details

## Search Criteria

Required Search filters: At least one Academic Period

Recommended Search filters: None

## Search Criteria Notes

Certain list of values (LOV) search criteria filters in this template require the selection of one (or more) Academic Periods to display a specific list of values for that filter. To load these filters to use in your query, choose the desired Academic Period(s) and select the

Populate Search Criteria for Academic Period(s) Selected button. You will need to reload these filters any time you change the Academic Period(s) for a new query.

The Student Course Search Criteria filters are for course registrations at your institution only. Transfer courses are excluded from queries.

## List

This list report provides one set of results based on the entered search criteria. Data includes ID, name, academic period, sub-academic period, enrollment status, current time status, total credits, enrolled, registered and deceased indicators, email address and the city, state/province, postal code, county, nation, telephone, and address type for a preferred address.

Additional information applicable to the list page for this report template is available using the following:

- **Export Options:** More fields of information are provided with the exportable .csv and .xml files than are viewable on the List page. These include the Entity UID and formatted (preferred) mailing address.
- **Banner ODS Population:** A population saved for this template will include the distinct combinations of Entity UIDs and Academic Periods for your query result set.

## Detail Reports

The following detail information can be accessed for any student on the list report:

- Current Addresses
- Other Phone Numbers
- Current Internet Address
- Academic Study
- Enrollment Information
- Academic Information
- Academic Standing
- Holds
- Student Courses
- Student Course Meeting Times
- Demographics
- Medical Information
- Veteran Status

- International Details

## Notes

- The Student Course reporting template provides a broad array of query opportunities. Performance, or the time it takes to retrieve a list of results, may vary based on the complexity of your query or the size of a potential result set.
- Search Criteria filters in this template require an Academic Period to display a specific list of values for that filter. To load these filters to use in your query, select or change to the desired Academic Period(s) and select the Populate Search Criteria button.
- The Student Course Search Criteria filters and the Registered Courses Detail report do not include transfer course information (STUDENT\_COURSE records where TRANSFER\_COURSE\_IND = Y are excluded).
- Detail reports containing Academic Period based information are for the Academic Period in context for the row selected from the List page.

## Self-Service Reporting Configuration Parameters

---

See the “Customize Parameters” section of the *Self-Service Reporting Installation Guide* for steps on how to set up Administrative User Interface parameters used within SSR.

## Security

---

There are various security options available for SSR. This section discusses authentication and authorization access to your Oracle database. It does not cover security configuration such as firewall placement or securing your application server. Information on those topics is available from Oracle at the Oracle Application Express (APEX) home page:

[http://www.oracle.com/technology/products/database/application\\_express/index.html](http://www.oracle.com/technology/products/database/application_express/index.html).

APEX first performs application authentication. This determines whether a user has access to an application, such as SSR. To determine which components a user has access to, APEX performs authorization.

APEX provides several methods for authentication. You can choose from a series of preconfigured authentication schemes, copy an existing scheme from another APEX application that you have already developed, or create your own custom authentication scheme. A description of the various methods appears below:

- Oracle Account Security

The default SSR security method uses Oracle accounts. If you are already using Oracle accounts for security at your reporting layer, particularly if you have implemented Fine Grained Access in Banner ODS, this option may be the simplest way to implement SSR, and would allow you to bring up SSR without adding to your security maintenance.

- APEX Built-in Authentication

If you have already set up security in Banner ODS for a reporting tool such as Oracle Business Intelligence Discoverer or ReportNet, you may want to establish a similar security scheme for SSR. User accounts for SSR are created and maintained using the APEX Administrative User Interface.

- Use an Existing Security Scheme

If you have already created your own APEX application and have devised your security scheme to govern it, you may be able to use that same security scheme for SSR.

- Create Your Own Custom Authentication Scheme

APEX provides a wizard creates an authentication scheme from scratch.

- Other Security Options

APEX enables you to integrate SSR with an LDAP server or with Oracle's Single Sign-On technology.

For more information on authentication, refer to the *Oracle Application Express User's Guide release 2.2* (Oracle document B28550-01).

## Oracle Account Security

### Authentication

The default SSR security method uses Oracle accounts. If you are already using Oracle accounts for security at your reporting layer, particularly if you have implemented Fine Grained Access in Banner ODS, this may be the simplest way to implement SSR, and would allow you to bring up Self-Service Reporting (SSR) without adding to your security maintenance.

APEX runs on the PL/SQL module of Oracle's Application Server. The PL/SQL module uses a database access descriptor (DAD) and a SQL\*Net connection to log into your Oracle database. The APEX DAD stores a username and password with which to log into the database. However, to use Oracle account security you need to configure the DAD so that it does not store a user name and password. The user is prompted for these values. The user name is then made available in the APEX substitution variable called APP\_USER.

## Authorization

The SSRMGR.SCK\_COMMON package includes the F\_getSSRPermissions function which uses the Oracle User ID to determine which SSR reporting templates that each user is authorized to access.

The function call for F\_getSSRPermissions is located in the SSR application on Page 1 for all report templates, then again in a Branch on each report template page to prevent access to the page via a bookmark or URL manipulation.

### Navigate to Page 1:

1. Log into APEX as an *SSR Workspace Administrator*.
2. Select **Application Builder**.
3. Select the **SSR** application.
4. Navigate to the Page Definition for page 1.
5. The call to F\_getSSRPermissions is located in SET\_PERMISSIONS under *Processes* in the Page Rendering column. Each call to F\_getSSRPermissions sets permission for a single report template. F\_getSSRPermissions calls F\_getSSRPermissionsViewList to retrieve the list of views for which a user must be granted SELECT permission in order to run a given report template.

If you add or delete a report template or add or delete any views for a report template, you *must* change SSRMGR.SCK\_COMMON.F\_getSSRPermissions and/or F\_getSSRPermissionsViewList.

SSR is delivered with scripts which can be used to issue the grants that are required to access each of the SSR report templates. These scripts are located in the `ssr/security` install directory.

### Add Security for a New Report Template

1. Log into APEX as an *SSR Workspace Administrator* using the following URL format as an example:

`http://hostname:port/pls/database_access_descriptor/f?p=4550:1`

- *hostname* is the name of the system where Oracle HTTP Server is installed.
- *port* is the is the port number assigned to Oracle HTTP Server. In a default installation, this number is 7777. For more information, see “Accessing the Oracle Application Express Login Page” in the *APEX Installation Guide*.
- *database\_access\_descriptor* describes how Oracle HTTP Server connects to the database server so that it can fulfill an HTTP request. The default value is *apex*.

- The remainder of the URL indicates to display the login page for a Workspace Administrator.
2. You will now be presented with the APEX Login page. Login using *SSR* as the Workspace and use the administrator ID and password.
  3. Select Application Builder.
  4. Select the SSR application.
  5. Navigate to the Page Definition for page 1.
  6. In the *Items* section, add a new Item for the new report template. For example, an existing security Item is P1\_STU\_ADVISOR\_STUDENT.
  7. In the *Processes* section at the bottom of the Page Rendering column, select **SET\_PERMISSIONS**.
  8. Using the existing code as an example, add the code to set the value of the new Item you just created. Note that permissions are set for each template and for each menu group, i.e., Student, Advancement, etc. and that you need to add code to set the permission for the new report template and add code to set the permission for the menu where your new report template will appear.
  9. Edit the function SSRMGR.SCK\_COMMON. F\_getSSRPermissionsViewList, using the existing code as an example, add the new report template name and associated list of views. Compile SSRMGR.SCK\_COMMON.

### Delete Security for a Report Template

1. To login to SSR and locate page 1, follow the first 5 instructions in the above section, [“Add Security for a New Report Template”](#).
2. In the *Processes* section at the bottom of the Page Rendering column, select **SET\_PERMISSIONS**.
3. Comment out references to the item associated with the report template.
4. Navigate to the Page Definition for the Search Criteria page in the Report Template for which you are modifying security. In the *Branches* section at the bottom of the Page Processing column, delete the security Branch to page 1 from the report template.

### Change the List of Views for a Report Template

1. Edit the function SSRMGR.SCK\_COMMON. F\_getSSRPermissionsViewList.
2. Change the list of views associated with the template(s) you are changing.

3. Compile SSRMGR.SCK\_COMMON.

## APEX Built-in Authentication

If you have already set up security in Banner ODS for a reporting tool such as Oracle Business Intelligence Discoverer or ReportNet, you may want to establish a similar security scheme for SSR. This is accomplished using the APEX Administrative User Interface.

### Change the Authentication Scheme

1. Log into APEX as an *SSR Workspace Administrator* using the following URL format as an example:

`http://hostname:port/pls/database_access_descriptor/f?p=4550:1`

- 1.1. *hostname* is the name of the system where Oracle HTTP Server is installed.
  - 1.2. *port* is the is the port number assigned to Oracle HTTP Server. In a default installation, this number is 7777. For more information, see “Accessing the Oracle Application Express Login Page” in the *APEX Installation Guide*.
  - 1.3. *database\_access\_descriptor* describes how Oracle HTTP Server connects to the database server so that it can fulfill an HTTP request. The default value is *apex*.
  - 1.4. The remainder of the URL indicates to display the login page for a Workspace Administrator.
2. You will now be presented with the APEX Login page. Login using *SSR* as the Workspace and use the administrator ID and password.
  3. Select **Application Builder**.
  4. Select the **SSR** application.
  5. Select **Edit Attributes**.
  6. Select **Security**.
  7. Select **Define Authentication Schemes**.
  8. On the right side of the page, select **Change Current**.
  9. Change the value of **Available Authentication Schemes** to *Application Express*.
  10. On the confirmation page, select **Make Current**.

## Application Express User Account Authentication

APEX performs authentication and authorization using information stored in its own tables. The user accounts are those that are created in the SSR workspace. You can set up user accounts one of two ways.

1. Create a user account through the APEX Administrator interface:
  - 1.1. Login to APEX as an APEX Administrator.
  - 1.2. Select **Manage Workspaces**.
  - 1.3. Select **Manage Developers and Users**.
  - 1.4. Select **Create**.
  - 1.5. Enter the required information and when done select either the **Create** button or the **Create and Create Another** button.
2. Create a user account through the APEX Workspace Administrator interface:
  - 2.1. Login to APEX as an *SSR Workspace Administrator*.
  - 2.2. On the Home page, select the link **Manage Application Express Users** in the Administration box on the right side of the page.
  - 2.3. Select **Create End User**.
  - 2.4. Enter the information except for User Groups and when done select either the **Create** button or the **Create and Create Another** button.

## Create User Groups

Authorization is accomplished by assigning end users to User Groups. Create a User Group for each SSR Report Template:

1. Login to APEX as an *SSR Workspace Administrator*.
2. On the Home page, select the link **Manage Application Express Users** in the Administration box on the right side of the page.
3. Select **Create Group** and create a group for each of the following Report Templates. Be sure to *enter each name exactly as listed below*:
  - Admissions Application
  - Advancement Person
  - Advisor Student Listing
  - Employee

- Enrolled Students
- Financial Aid Awards
- General Ledger
- Operating Ledger
- Receivable Customer

## Application Express User Account Authorization

Authorization is accomplished by assigning end users to User Groups. SSR is delivered with a User Group defined for each Report Template. To assign end users to SSR User Groups:

1. Login to APEX as an *SSR Workspace Administrator*.
2. On the Home page, select the link **Manage Application Express Users** in the Administration box on the right side of the page.
3. Select **Existing Users**.
4. Select a user from the list.
5. Under User Groups, select the appropriate User Groups for that user and select **Apply Changes**.

### Note

Be certain to assign all User Groups to all users who are listed as Developers and/or Workspace Administrators. ■

## Change the Authentication Scheme application item

An APEX application item, or global variable, called F1\_SECURITY\_TYPE has been created to direct the SSR permissions function, SCK\_COMMON.F\_getSSRPermissions, to use either Oracle User Account Security or Application Express User Account Security. To change the value of this item:

1. From the SSR Workspace Administrator Home page, select **Application Builder**.
2. Select the **SSR** application.
3. Select **Shared Components**.
4. In the Logic section, select **Application Items**.
5. At the bottom of the page, select the arrow next to **Existing Application Level Computations**. That will display the list of application items.

6. Select the edit icon in column 1 to edit F1\_SECURITY\_TYPE.
7. In the Computation attribute change the value of *ORACLE* to *APEX* and select **Apply Changes**.

Refer to the *Oracle Application Express User's Guide release 2.0* (Oracle document B16373-01) for additional details.

## Other Security Options

Oracle allows you to configure the SSR application as a partner application with the Single Sign-on (SSO) infrastructure using Oracle Internet Directory (OID). To learn more about this option, visit the Oracle APEX Home Page URL (noted at the beginning of this section), select the link to *How-To's*, then select from the various papers available in the "Security" section.

Any LDAP server can be used for APEX authentication. In the Shared Components section of the SSR application, APEX provides a wizard which allows you to define the access parameters to your LDAP server. The wizard assumes that the server already exists and that it can respond to a SIMPLE\_BIND\_S call for credentials verification. Refer to the same above-mentioned "Security" section at the Oracle Web site.

APEX also allows you to use an existing authentication scheme of your own, or to create a new one. To implement a custom scheme, you must provide a PL/SQL function returning a Boolean value that APEX will execute before processing each page request. As with the setup for an LDAP server, APEX provides a wizard in the Shared Components of the SSR application with which to define a custom authentication scheme.



# Glossary

---

## Attribute

A building block of information within a view. Many of the attributes in a view come directly from fields in the source database. Other attributes are derived from database fields either through calculations or the logic defined in a function.

## Base View

A view of a primary or secondary composite table, which. A base view is used to add fields not extracted from the source database, or ERP, but required for the view, such as counts or other function-based values. In addition, the base view serves to insulate the user from changes to the architecture of the composite tables. Any changes to the underlying table can be handled through the creation of the base view.

The Banner ODS builds all access to data via the base views

## Business Intelligence

A term adopted within the technology industry to represent a broad category of applications for gathering, storing, analyzing, and providing access to data to help users make better business decisions. Applications within a business intelligence environment allow users to monitor the operations and financial soundness of the institution – they may preserve the organization’s fiscal history, display its current state, and forecast future results using business intelligence data.

## Change File

A file that captures and records key information about the updates, additions, and deletions of data in a master file. The creation of the Change File starts the incremental refresh process in the Banner ODS.

## Change Tables

In Banner, Oracle tables that capture key information when data is changed. Change tables drive the incremental refresh of the Banner ODS process. They identify which information needs to be updated in the Banner ODS.

## Cleansing

The process of translating, decoding, or resolving anomalies within source information that resides in Banner Operational Data Store.

## Composite Table

A table within the Banner ODS that groups information from the source system’s database tables to form the foundation from which views will be built.

## **Composite View**

Views within Banner that contain the information that will be extracted into the Banner ODS. The ETL process pulls the information from the composite views into the composite tables of the Banner ODS.

## **Control report**

In the Banner ODS, a report generated after a refresh process that indicates the status of the refresh. The report identifies whether the refresh process was successful, the elapsed time of the refresh, and any errors that might have occurred.

## **CSV**

Comma Separated Values file. CSV is a normal format for files as they are downloaded or exported from an application. A CSV file can be opened and manipulated in common tools like Microsoft Excel.

## **Cube**

A cube is a multidimensional data structure used to store presorted information that has been aggregated based on an underlying data relationship. Data structured in this way can be quickly processed and analyzed, because multiple dimensions can be examined at one time.

## **Customer Support Center**

The Customer Support Center is a centralized support site where you can access support resources for all products.

## **Data**

Recording facts or instructions on a storage medium for communication, retrieval, processing, or presentation.

## **Data Element**

The smallest individual component part of data. A field's literal, technical name.

## **Data Link**

A reference to a remote database, located on a completely different system from the local database.

## **Data Mart**

A subset of a data warehouse that is designed for a particular subject area or branch of the organization's business, such as for the Admissions or Human Resource areas. Data marts are typically built and controlled by a single department in an organization.

## Data Model

A map that displays the data elements that are included in the Banner ODS, and the transition of each data element from its origin in the ERP database to its location in the Banner ODS composite tables and views and Banner EDW star schemata.

## Data Store

Also called Banner Operational Data Store (Banner ODS). A place that stores significant types and pieces of information for an organization, in a format that promotes ease of retrieval and analysis. Banner Operational Data Store (Banner ODS) facilitates operational ad hoc reporting by gathering, transforming, and storing data. The Data Store deals with information that is transactional in nature. It's short-lived, and may be here today and gone tomorrow. See Data Warehouse.

## Data Transformation

The process of converting pieces of raw data into information that is logical, such as by decoding production data and merging information from multiple sources and formats.

## Data Warehouse

Also called Enterprise Data Warehouse (Banner EDW). An informational database that stores data provided and shared by multiple databases. It enables an institution to keep “time slices” of data over time, over history, stored for easy retrieval and comparison. The data warehouse is an extension of the Data Store, which is the primary source of aggregated and detailed data. Partner applications can also be used to feed detailed data into the Banner EDW through the Banner ODS. The data warehouse is separated from the transaction stores, offering scalable performance, product independence and a streamlined extraction process to support the reporting, query or uses of the data warehouse.

Of an Enterprise Data Warehouse (Banner EDW) an institution can ask the question, “How are we doing this month as compared to last month?” See Data Store.

## Denormalized

Describes data that does not conform to any “normalized” form. Normalized data is data in its simplest format, without redundant attributes or keys. Data is normalized for ease when transporting it to another environment, or retrieving it for reporting purposes.

## Dimension

A structural attribute of data that consists of pieces of information of a similar type. A Geography dimension, for example, may contain data about regions, countries, cities, states. A time dimension contains year, month, day and hour members. A multidimensional data structure allows data to be organized and analyzed in a concise, efficient way.

## Dimension Table

A table that contains all the attributes (dimensions) or characteristics that describe observations and their associated measures (related numbers).

Characteristics of the people, places, or things represented in the data are stored in the dimension tables. One row represents a unique combination of the characteristics in a particular dimension table. The unique combination is assigned a surrogate (sequential) key.

## Dynamic Data

Data that is automatically updated every time something changes in the Oracle database.

## Banner EDW (Banner Enterprise Data Warehouse)

See Data Warehouse.

## Enterprise Resource Planning (ERP)

ERP is the term used to describe the transactional system. It's the combination of the major components of these systems (Student, Financial Aid, Human Resource, Finance, and Alumni/Advancement). It provides the core of information for the Banner ODS and the EDW.

## Extract, Transform and Load (ETL)

In managing databases, Extract, Transformation, Load (ETL) refers to three separate functions combined into a single programming tool.

The Extract function reads the data from a specified source database, and extracts a desired subset of data. Next, the Transformation function works with the acquired data, using rules or lookup tables, or creating combinations with other data to convert it to the desired state. Finally, the Load function writes the resulting data (either all of the subset or just the changes) to a target database, which may or may not previously exist.

The ETL process is used to populate Banner Operational Data Store (Banner ODS) from the source database. Another set of ETL processes is used to populate the enterprise data warehouse (Banner EDW) from Banner Operational Data Store (Banner ODS).

## ETL Map Package Parameter

In the Administration tool, a parameter used to group mappings together into a job.

## Facts/Measures

Numbers that are related to the attributes. Facts and measures (the terms are synonymous) generally represent counts, sums or percentages and other ratios. They may be stored and retrieved. They may be calculated from stored measures as the query is executed. Examples of facts/measures are total enrollment, or the number of employees, or the amount of all gifts to the institution.

## Fact Table

A table that contains measures or numerical information used to perform an analysis.

Detailed Fact tables store the most granular level detail in the data warehouse, and support information audit when linked to the source database. Summary Fact tables provide faster responses for queries.

## Fine Grained Access

Terminology used by Oracle to identify how security can be applied to different tables and views. The Banner ODS use fine grained access security to manage user profile access.

## Freeze Process

A process maintained within the Administration tool that allows you to identify what file(s) to capture at a specific moment in time, or “freeze,” and store inside the Banner ODS as new tables for later access. You can use the freeze process to create ad hoc or scheduled “snapshot” database tables.

## Function

A small piece of code that uses specified logic to get information from the source database that isn't stored as a single field. For example, “Age” may not be stored as a field. Using a function that subtracts birth date from today's date and then determines whether the birth month has passed, Age can be provided as an attribute in a view.

The Banner ODS is designed to use functions where practical to calculate values, or determine the location of information within the Presentation Views.

## Grant, Revoke and Privileges

While DDL statements such as Grant and Revoke can't be used directly in PL/SQL, they do have an effect on which SQL statements are legal. In order to Insert or Delete information on an Oracle table, you need permission. Permissions are manipulated via the Grant and Revoke SQL commands.

## Job Killer

Gives you the ability to stop a process while it is running using the JOB KILLER parameter.

## Key Attribute

Attributes that determine the level of information returned by the view. It is important for you to know the level at which information in a view is returned. For example, key attributes can determine whether a view returns one row of information for each person per condition, or simply one row for each person.

## Incremental Refresh

Data in the Banner ODS is updated, or refreshed, at predetermined intervals of time. Only the data that has changed in the source database since the last refresh is updated.

## Information

Data that human beings assimilate and evaluate to solve problems or make decisions.

## Mapping

The activity of associating elements in the source system with their corresponding elements in the Banner ODS. When you run a job (schedule a process via the Administration Tool), it calls the related mappings and loads or updates the data defined by them.

The Banner ODS includes two main categories of mappings:

- **LOAD** mappings: load data from the source system into the Banner ODS. These mapping names have the prefix `LOAD_`.
- **REFRESH** mappings: update the Banner ODS with data that has changed in the source database. Mappings in this category have the prefix `UPDATE_` or `DELETE_`.

Typically, these mappings exist in pairs. To completely refresh the data, run the `DELETE` mapping followed by its associated `UPDATE` mapping.

The Banner ODS is delivered with hundreds of mappings already defined. `LOAD` and `REFRESH` mappings exist for each composite table in the Banner ODS. To make them easier to work with, they are organized into groups by product areas. This gives you the ability to run one job that includes a group of mappings, say all of the Finance-related mappings, at one time. You can also run a single mapping, if desired.

## Master Instance

The database where production data are located. This is also the location where the snapshot logs are run. The master instance is also called the master database or the production database.

## Measure/Fact

See Facts/Measures.

## Meta data

Literally, data about data. Descriptions of what kind of information is stored where, how it is encoded, how it is related to other information, where it comes from, and how it is related to your institution. The information describes data and other structures, such as objects, business rules, and processes.

## Multidimensional Cube

See Cube.

## Multidimensional Database

A Database Management System (DBMS) optimized to support multi-dimensional data.

## Normalize

See Denormalized.

## ODBC

Open Database Connectivity. A product is considered to be ODBC compliant if it allows you to access a relational database in a client/server environment. An example would be using your PC in your office to retrieve information in a database stored in another location.

## Online Analytical Processing (OLAP)

Dynamic, multi-dimensional analysis of historical data which supports activities such as:

- Calculating across dimensions and through hierarchies
- Analyzing trends
- Drilling up and down through hierarchies
- Rotating to change the dimensional orientation

## Banner Operational Data Store (Banner ODS)

See Data Store.

## Banner ODS Instance

The database where all the Banner ODS functions, composite tables, and views are run.

## OLAP

Online Analytical Processing. OLAP enables you to perform multi-dimensional analysis by allowing you to drill up, down, across and through information to see varying levels of detail.

## Oracle Data Dictionary

Oracle stores information about the structure of the database in the Oracle data dictionary. The data itself is located in other areas and the data dictionary describes how the actual data is organized. The dictionary consists of tables and views that you can query in the same way you query any other database tables or views (the views are owned by Oracle user SYS).

## Oracle Warehouse Builder (OWB)

OWB is the ETL tool used to extract data from the ERP and move it to composite tables in the Banner ODS. It is also the tool used to extract the data from the Banner ODS and load it into the Banner EDW.

It is designed to move and transform data, develop and implement data warehouses, perform meta data management, or create and manage Oracle databases and meta data. In addition to its graphical user interface (GUI), Warehouse Builder provides an API in the form of Oracle MetaBase Plus (OMB Plus), where all Warehouse Builder functionality can be accessed using the OMB Scripting Language.

## Package

A collection of functions and/or procedures that are managed and owned by a single object.

## Physical Table

A table where data is actually stored in a database.

## PL/SQL

The 3GL Oracle procedural language extension of SQL. PL/SQL enables you to mix SQL statements with procedural constructs. PL/SQL combines the ease and flexibility of SQL with the procedural functionality of a structured programming language, such as IF...THEN, WHILE, and LOOP. Even when PL/SQL is not stored in the database, applications can send blocks of PL/SQL to the database rather than individual SQL statements, thereby reducing network traffic. With PL/SQL, you can define and execute PL/SQL program units such as procedures, functions, and packages. PL/SQL is interpreted and parsed at runtime; it does not need to be compiled.

## Presentation View

A view that joins together multiple base views to make the information easier to access and report from. The primary purpose of a presentation view is to eliminate the

need to join base views, and add in display defaults when present. The presentation view invokes Oracle's fine grained access to ensure proper access to data by a user.

Presentation View and Reporting View are synonymous terms.

### **Primary Composite Table**

A composite table that manages its stored data using a "unique row per key" format. Typically, these tables are the owners of data, and are supported by secondary composite tables.

### **Privilege: Object vs. System**

An object privilege allows an operation on a particular object (such as a table). A system privilege allows operations on an entire class of objects.

### **Procedure**

A database object that is designed to perform a designated process. A procedure is similar to a function -- it is written using rules that are typically difficult for a report developer to create within a reporting tool. The primary difference between a procedure and a function is that a procedure is used to update data in the database whereas functions can only return values.

The Banner ODS uses procedures within the ETL process of populating the Composite Tables.

### **Reporting Views**

See Presentation View.

### **Relational Online Analytical Processing (ROLAP)**

A form of Online Analytical Processing (OLAP) that performs dynamic multi-dimensional analysis of data stored in a relational database rather than in a multi-dimensional database (which is usually considered the OLAP standard).

### **Role Based Security**

Security provided within the Banner ODS that permits you to control who can access reporting information based on each person's role at the institution. The Banner ODS uses Oracle's fine grained access to implement its security.

### **Secondary Composite Table**

A composite table that manages its information on a "many per key" format. Typically, these tables are used to support primary composites because the data can be associated with a specific value within the primary composite tables.

A secondary composite view is also referred to as a repeating view. It is a building block that contains a defined set of data that has an unlimited number of records in the ERP. It is passed through a display rule filter that slots a limited number of the repeating items for use in reporting. It is usually used in combination with other base composite views, but it may be used alone.

**Slotted View** A view that brings back user-defined information from the source database rather than all information.

## Source Code

The `all_source`, `dba_source`, and `user_source` views contain the source code for stored procedures, functions, packages, and package bodies. Trigger source code is in the `all_triggers`, `dba_triggers`, and `user_triggers` views. If the stored object is wrapped, these views contain the encoded source rather than clear text.

### Note

Within the Banner ODS DED, when you view source code, you see the encoded source. ■

## Star Join

An optimal, denormalized form of organizing data to access a group of people, usually a department. Star joins are usually associated with data marts.

## Star Schema

A standard technique for designing the tables of a data warehouse. It is a collection of related database objects, including logical structures such as tables, views, sequences, stored procedures, synonyms, indexes, clusters, and database links.

Star schemata are made up of fact tables, dimension tables and surrogate or calculate keys.

Fact tables each join to a larger number of independent dimension tables. The tables may be partially denormalized for performance, but most queries still need to join in one or more of the star tables.

A schema is owned by a database user and has the same name as that user; relational schemata are grouped by database user ID.

## Synonym

A renaming of a table reference, similar to an alias for a select list item. A synonym is a data dictionary object and is created by the `CREATE SYNONYM` DDL statement.

## Table

The object within the database where data is stored in a row and column format.

## Translating Code

A code that associates a code in the source database with different code values in the Banner EDW. A translating code can translate one-to-one, or by range. More than one code in the source database can be associated with one code in the Banner EDW.

## Trigger

Triggers are used to populate the change tables which aid in the incremental refresh process.

## View

A grouping of information, also called “logical view.” A view is “logical” because the information in the view is grouped in a logical order, putting related information in the same section of the view. For instance, in the people-related views, you find all the name information together at the beginning of the view, followed by personal, biographical, and demographic information.

Most of the information in a view comes from fields within the source database tables. Some information is calculated from database fields or retrieved using an Oracle function. A single view can include up to 255 pieces of information, called attributes.



# Index

---

## A

---

- access detail reports **6-7**
- accounts receivable SSR report templates **6-19**
- add comments to reporting views **2-148**
- Administrative UI data access **2-53**
- admissions application
  - detail reports **6-35**
  - list **6-35**
  - notes **6-36**
  - search criteria **6-34**
- admissions application SSR report template **6-34**
- advancement person
  - detail reports **6-21**
  - list **6-21**
  - notes **6-22, 6-36**
  - search criteria **6-21**
- advancement SSR report templates **6-21**
- advisor's students **6-32**
  - list **6-33**
  - notes **6-34**
  - search criteria **6-32**

## B

---

- baseline meta data **2-141**
- business profiles **2-20, 2-21, 2-22, 2-24**
  - associate profile with a user **2-25**
  - associate user with a profile **2-26**
  - set up **2-19, 2-20, 2-21, 2-22, 2-23, 2-24**
  - view **2-26**

## C

---

- checks and balances (ODS) **2-98**
- Cognos ReportNet
  - lists of values **4-21, 4-49**
- composite tables **1-33, 1-35**
- control report **2-116**
- control reports **2-115**

- delete **2-116**
  - error messages **2-116**
- create and save a search rule **6-9**
- create ODS populations **6-14**
- create users and PINS **2-3**
- cross reference chart **2-9**
- customized scheduled processes **2-72**

## D

---

- data models
  - ODS **5-1**
  - ODS Entity Relationship Diagrams **5-1**
- delete a search rule **6-13**
- delete an ODS population **6-16**
- delete ODS populations
  - from home page **6-17**
  - with search rule loaded **6-17**
- detail reports
  - access **6-7**
- detail reports page **6-7**
  - drop-down list **6-8**
  - export **6-8**
  - sort **6-8**
- display rule cross reference
  - internal group and internal code **2-9**
- display rule information in published meta data **2-9**
- display rules
  - creating a new rule **2-12**
  - cross reference chart **2-9**
  - duplicate **2-16**
  - reload **2-16**
  - set up **2-6**
  - updating an existing rule **2-15**
- duplicate display rules **2-16**

## E

---

- email
  - entire list **6-6**
  - individual **6-6**

email notification **2-125**  
employee **6-30**  
    detail reports **6-31**  
    list **6-31**  
    notes **6-32**  
    search criteria **6-30**  
enrolled students **6-36**  
    detail reports **6-37**  
    list **6-37**  
    notes **6-38**  
    search criteria **6-36**  
Entity Relationship Diagrams **5-1**  
entity relationship diagrams **5-1**  
    legend **5-2**  
ERD relationship legend **5-2**  
ERDs **5-1**  
error messages **2-116**  
    checks and balances process (ODS)  
        **2-117**  
    freeze process **2-121**  
    publish meta data  
        (PUBLISH\_META\_DATA) **2-122**  
    reconcile (RECONCILE\_JOB,  
        RECONCILE\_SINGLE\_JOB) **2-122**  
ETL control group parameter **2-91**  
ETL map package parameter **2-85**  
ETL slot package parameter **2-90**

## F

---

finance SSR report templates **6-22**  
    general ledger **6-22**  
financial aid awards **6-28**  
financial aid enrollment  
    detail reports **6-29**  
    list **6-29**  
    notes **6-30**  
    search criteria **6-29**  
financial aid SSR report templates **6-28**  
    financial aid awards **6-28**  
freeze data  
    adding a table/view to a data list **2-129**  
    creating a new data list **2-126**  
    editing or deleting a table/view from a  
        data list **2-130**  
    freezing a single ODS table/view **2-131**  
    freezing multiple tables/views at the  
        same time **2-132**

freeze or update ODS recurring data **2-136**

## G

---

general ledger **6-22**  
    detail reports **6-24**  
    expected uses **6-22**  
    notes **6-25**

## H

---

human resources SSR report templates **6-30**  
    employee **6-30**

## I

---

installed process parameter **2-78**  
    adding **2-78**  
    setting up **2-78**

## J

---

job parameter **2-83**

## K

---

kill a job  
    set up **2-68**

## L

---

list of value views **5-67**  
list page **6-5**  
    access detail reports **6-7**  
    email **6-6**  
    export **6-6**  
    sort **6-6**  
lists of values  
    Cognos ReportNet **4-21, 4-48, 4-49**  
load a search rule **6-10**  
local meta data **2-141**

## M

---

mappings  
    add **2-155**

- delete **2-155**
- meta data **2-138**
  - add mappings **2-155**
  - add source column **2-149, 2-153**
  - add source tables to a subject area **2-147, 2-152**
  - add target column **2-149, 2-153**
  - add target views to a subject area **2-147, 2-152**
  - baseline and local **2-141**
  - configure PUBLISH\_LOCATION and VIEW\_URL parameters **2-143**
  - configure VIEW\_URL parameter **2-143**
  - creating **2-141**
  - delete local **2-150, 2-154**
  - delete mappings **2-155**
  - delete source column **2-149, 2-153**
  - delete target column **2-149, 2-153**
  - edit sources **2-146, 2-148, 2-151**
  - edit target **2-146, 2-148, 2-151**
  - parameter set up for publishing reports **2-142**
  - publish for entire subject area **2-158**
  - publish for one source or target **2-159**
  - publish from command line **2-158, 2-159**
  - reporting views **2-171**
  - set up publish preferences **2-141**
  - source and target tables **2-168**
  - source tables **2-164**
  - target tables **2-166**
  - view **2-161**
- metamodel **2-162**

## N

---

- navigation quick reference **6-2**

## O

---

- ODS checks and balances **2-98**
- ODS daily update **2-136**
- ODS data models **5-1**
- ODS display rules **2-6**
- ODS populations **6-13**
  - create **6-14**
  - create with search rule loaded **6-14**
  - create without search rule loaded **6-15**
  - delete **6-16**
  - refresh (update) **6-15**

- refresh from home page **6-16**
- refresh without search rule loaded **6-16**
- use with the ODS **6-17**
- ODS updates
  - specific dates and times **2-137**
- operating ledger **6-25**
  - detail reports **6-27**
  - list **6-26**
  - notes **6-28, 6-30, 6-36**
  - search criteria **6-25**
- oracle account security **6-39**
  - authentication **6-39**
- Oracle warehouse builder runtime audit
  - browser integration **2-123**
  - integration setup **2-123**
  - RAB authentication **2-124**
- OWB mappings and slot packages **2-84**

## P

---

- parameter parameter **2-93**
- parameters
  - change **2-57**
  - create **2-56**
  - delete **2-56, 2-57**
  - ETL control group **2-91**
  - ETL map package **2-85**
  - ETL slot package **2-90**
  - installed process **2-77**
  - job **2-83**
  - maintain ODS **2-55**
  - parameter **2-93**
  - schedule a process **2-76**
  - subprocess **2-79, 2-81**
- password **2-3**
- PIN **2-3**
- policies
  - for a single table **2-50**
  - for all tables **2-49**
- policy management **2-49**
  - set up for a single table **2-50**
  - set up for all tables **2-49**
- populations, see ODS populations **6-13**
- predicates **2-48**
- publishing meta data
  - all reports **2-159**
  - by scheduling a process **2-160**
  - from the Administrative UI **2-158**

from the command line **2-160**

## R

---

reconcile a single ODS table **2-97, 2-99, 2-100**  
reconcile multiple ODS tables **2-97, 2-107**  
records per page display settings **6-7**  
refresh (update) ODS populations **6-15**  
refresh from home page **6-16**  
refresh with search rule loaded **6-15**  
refresh without search rule loaded **6-16**  
rename a search rule **6-12**  
report ODS source change table counts **2-96**  
report templates **6-18**  
reporting  
    self-service **6-1**  
    third party reporting tools **4-1**  
reporting views **2-171, 5-67**  
    add comments **2-148**  
reset Administrative user roles **2-4**  
runtime audit browser integration **2-123**

## S

---

save as another search rule **6-11**  
schedule a process **2-61**  
    customized **2-72**  
    customized scheduled processes **2-72**  
    view and remove **2-64, 2-67**  
schedule a process parameters  
    ETL Control Group **2-91**  
    ETL map package **2-85**  
    ETL slot package **2-90**  
    job **2-83**  
    parameter **2-93**  
    subprocess **2-79, 2-81**  
schedule multiple processes **2-64**  
search criteria  
    dependant filters **6-4**  
    list of values filters **6-5**  
    recommended and required **6-4**  
search criteria page **6-3**  
search rules **6-9**  
    create and save **6-9**  
    load **6-10**  
    rename **6-12**

save as another rule **6-11**  
update **6-10**

security  
    add for a report template **6-40, 6-41**  
    assign rules for each Oracle user **2-35**  
    delete for a report template **6-41**  
    predicates **2-48**  
    requirements **2-28**  
    rules in action **2-50**  
    self-service reporting **6-38**  
security rules  
    turn on and off **2-35**  
self-service reporting **6-1**  
    detail reports page **6-7**  
    list page **6-5**  
    navigation quick reference **6-2**  
    search criteria page **6-3**  
set up the ODS **2-54**  
set up users and PINS **2-3**  
slot packages **2-84**  
slotted tables **1-33, 1-35**  
    updating in the ODS **1-33**  
SSR report template overviews **6-18**  
SSR report templates  
    accounts receivable **6-19**  
    advancement **6-21**  
    finance **6-22**  
    financial aid **6-28**  
    human resources **6-30**  
    receivable customer **6-19**  
    student **6-32**  
student  
    advisor's students **6-32**  
student SSR report templates **6-32**  
subprocess parameter **2-79, 2-81**  
synchronize the ODS **2-54**

## T

---

third party reporting tools (ODS) **3-1**

## U

---

update a search rule **6-10**  
update populations **6-15**  
update/freeze recurring ODS data **2-136**  
use populations with the ODS

- access from the home page **6-18**
- access from view search rules page **6-18**
- user accounts **2-3**
  - user ID **2-3**
- user ID **2-3**
- using populations with the ODS **6-17**
- utilities **2-95**
  - checks and balances (ODS) **2-98**
  - reconcile a single ODS table **2-97, 2-99, 2-100**
  - reconcile multiple ODS tables **2-97, 2-107**
  - report ODS source change table counts **2-96**

## V

---

- view published meta data **2-161**

## W

---

- WebTailor administration **2-200**
- WebTailor and security **2-2**

