



PROJECT TITAN DETAILED DESIGN

Revision History

Version #	Author	Date	Reviewed By	Approved By	Description of Change
0.1	Dan Tarquinio	Mar 17 2010			Initial Draft
0.2	Dan Tarquinio	Apr 21 2010			Updated with additional details and after review with Core User Group
0.3	Dan Tarquinio	May 13 2010			Minor modifications based on Project Team feedback; Added Program Code to Wrkforce; Modifications to Program Code and Project Grant hierarchies to account for Active and Inactive statuses

Document References

Short Name	Document Description	Version	File Name
Requirements Document	Project Titan Requirements Document		
Scope Document	Project Titan Detailed Scope Document		
Project Glossary			

Table of Contents

- Revision History 2
- Document References..... 2
- Introduction 6
 - Document Purpose 6
 - Design Objectives..... 6
 - Design Assumptions..... 6
 - Design Constraints 6
 - Oracle Hyperion Planning Version 11.1.1.3 6
 - Unique Naming within Applications 7
 - Start Year Cannot be Changed 7
 - Terminology 7
- Technical Architecture 8
 - Overall Architecture & Design 8
 - Planning Application Architecture 8
 - Planning Application Design 9
 - Overview 9
 - Master Data Sources 12
 - Automation 19
 - Security 21
- ISPlan Design 22
 - Overview 22
 - Database Structures..... 22
 - Data Loads..... 36
 - Planning Input Forms 37
 - Calculations 38
- BSPlan Design..... 41
 - Overview 41
 - Database Structures..... 41

Data Loads.....	52
Planning Input Forms.....	53
Calculations.....	54
Wrkforce Design.....	54
Overview.....	54
Database Structures.....	55
Data Loads.....	70
Planning Input Forms.....	71
Calculations.....	71
FundPlan Design.....	71
Overview.....	71
Database Structures.....	72
Data Loads.....	81
Planning Input Forms.....	81
Calculations.....	81
FundTxr Design.....	81
Overview.....	81
Database Structures.....	82
Data Loads.....	94
Planning Input Forms.....	95
Calculations.....	95

Introduction

Document Purpose

This document will be used to describe the design of the Oracle Hyperion Planning system to be implemented as part of Project Titan at the University of British Columbia (UBC). The purpose of this document is to convey a vision of the system to be implemented, describe the components and artifacts of the system and provide sufficient detail for the developers of the system to implement it. While end users may find this document helpful in their effort to understand what the system will look like, the primary audience for this document is the development team tasked with the implementation of the system.

Design Objectives

The overall objective of the system design is to deliver the functionality required by the users (as defined in the Requirements Document) within the defined scope (as defined in the Scope Document). In addition to the overall objectives defined in the Requirements and Scope Documents, the system design detailed in this document considers the following objectives:

- The implemented solution should be user-friendly and simple for the end user to understand
- The system should be developed using out-of-the-box Oracle Hyperion Planning functionality, without code customization
- The system should avoid unnecessary complexity and be easy to administer and maintain after implementation

Design Assumptions

In addition to the assumptions provided in the Requirements and Scope Documents, the following assumptions inform the design of the system:

- All financial data will be stated in Canadian Dollars; There are no multi-currency requirements
- The Hyperion Planning Web Interface is assumed to be the primary user interface for data entry
- Offline Planning capabilities within Hyperion Planning will not be deployed, so the design implications caused by the limitations of Offline Planning will not be considered

Design Constraints

The design of the system is constrained by the requirements and scope that are to be addressed. The intent of this section is to summarize the requirements or scope definitions that explicitly constrain design options and identify requirements that are in conflict with one another.

Oracle Hyperion Planning Version 11.1.1.3

The system will be designed based on the features and functionality available in Oracle Hyperion Planning Version 11.1.1.3. A new release (Version 11.1.2) is anticipated to become available during the

build phase of this project. At the time of the writing of this design document, no decision has been made as to whether the system should be developed in the newer version. Therefore, the design assumes Version 11.1.1.3. The availability of certain new features that are known to be included in Version 11.1.2 would impact certain elements of the solution design. Where applicable, the constraints imposed on the design by the unavailability of these features within Version 11.1.1.3 will be noted.

Unique Naming within Applications

Within a Planning Application, all Member Names and Aliases must be unique across all dimensions. Where conflicts are known to occur frequently (e.g. several Accounts and Departments share the same 5 digit string within the PeopleSoft Chart of Accounts) naming conventions are suggested within this design document to ensure uniqueness. These naming conventions seek to maintain as much consistency as possible between the Hyperion system and other existing systems and maximize the usability of the application while ensuring uniqueness within the application. During the build of the applications, additional naming conflicts are likely to arise. If necessary, additional dimension-wide conventions may be adopted, or “one-off” solutions may be implemented to rectify minor conflicts between dimensions.

Start Year Cannot be Changed

The scope of the project states that historical Actual data will only be converted for FY07/08 forward (3 years of historical data). However, the original user requirements called for 5 years of historical data to be loaded. Should the opportunity to add the additional 2 years of data into the scope during the implementation or after go-live, the FiscalYear dimension must be built to start with 2005/06. This must be done at application creation, as the Start Year (or first Year in the application) cannot be changed after application creation. Only future years can be added after application creation.

Terminology

Below is a list of terms used throughout this document to describe elements of the design. This list of terms specifically relates to the Oracle Hyperion system. For other terminology, please refer to the Project Glossary.

- Planning Application – A collection of related Plan Types used to meet a set of planning needs. Plan Types within a Planning Application share certain dimensions, and the security assigned to those dimensions.
- Plan Type – A Plan Type is a distinct set of dimensions within a Planning Application, and equates to a database in Essbase. The structure of the database is defined by an outline.
- Outline – The database structure of a multidimensional database, including all dimensions, members, tags, types, consolidations and mathematical relationships. Data is stored in the database according to the structure defined in the outline.
- Dimension – A data category used to organize business data for storage and retrieval. Dimensions contain hierarchies of related members.

- Members (Master Data) – The discrete components within a dimension. Typically, members within a database structure are referred to as the Master Data.
- Member Name – A member name is typically based on a code or numeric value that represents a member. Examples might include a Department number (e.g. 160400), Account number (e.g. 640001), etc. To ensure uniqueness in member names, it is often necessary to prepend the numeric value with an alpha character representing the dimension that contains the member. (e.g. D160400 or A640001).
- Member Alias – A member alias is typically a textual description of a member. The alias is generally what is displayed on a report. Commonly, the alias will include the member name in order to ensure uniqueness in member aliases. Within a Planning Application, all member names and aliases must be unique. Examples corresponding to the member names shown above include “160400 – Wood Science”, “Office Supplies” or “Office Supplies – A640001”.
- Shared Member – A Shared Member property tag is an efficient way to build multiple instances of a specific member within the structure of a dimension without storing the data for the member twice. Shared Members are used in alternate rollups.
- Alternate Rollups – An alternate rollup is a secondary hierarchy within a dimension, usually consisting of Shared Members. An example of an alternate rollup might be a rollup of UBCO and UBCV Departments separately, while the main hierarchy includes Departments from both campuses.
- Level 0 – This refers to the lowest level of data in a database and the lowest, or leaf, level of a hierarchy within a dimension. Level 0 is often referred to as the “input” level.
- Upper Level – Any member that is not Level 0; a rollup.
- Attribute – An attribute dimension is associated with another standard dimension in an outline, and attribute values are assigned to the members of that standard dimension. Attribute dimensions are used to all additional analytic capability without increasing the database size or system calculation time. Attribute dimension retrievals result in dynamic calculations, which may slow impact retrieval performance.

Technical Architecture

Overall Architecture & Design

Planning Application Architecture

A Planning application may consist of as many as five Plan Types, with three of these Plan Types being fully configurable to meet specific organization planning needs. The other two Plan Types are reserved for the Workforce Planning and Capital Asset Planning modules, if these modules have been licensed. UBC has licensed the Workforce Planning module.

The design of this system will include two Planning applications: UBCPlan and FundPlan. The UBCPlan application will be used for Income Statement, Balance Sheet and Workforce Planning. The FundPlan application will house the GPO Funding Model and will include the ability to allocate and transfer funding between departments.

The applications will contain the Plan Types shown below:

- UBCPlan
 - ISPlan
 - BSPlan
 - Future1
 - Wrkforce
- FundPlan
 - FundPlan
 - FundTxr
 - Future2

The Future* Plan Types listed above will be created as part of this implementation, but will not be configured. The purpose of these Plan Types is to allow for development of additional planning capabilities within these Planning applications, should the need arise in the future. The Plan Types must be created now, as Plan Types cannot be added to a Planning application after initial application creation. As these Plan Types are being created as placeholders, they will include only the dimensionality required to ensure that they are valid and deployable.

Planning Application Design

Overview

UBCPlan - ISPlan

This Plan Type will be used for Income Statement planning for all Departments. All components of Income Statement planning will be done in this Plan Type, except for the planning of Salaries and Benefits and the planning of funding details. These tasks will be performed in other Plan Types described below.

The ISPlan Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Fund
- Department (Entity Dimension)
- Program Code
- Project Grant
- FiscalYear
- Scenario
- Version

UBCPlan - BSPlan

This Plan Type will be used for Balance Sheet planning for all Departments that require the ability to plan the Balance Sheet. All components of Balance Sheet planning will be done in this Plan Type.

The BSPlan Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Fund
- Department (Entity Dimension)
- Project Grant
- FiscalYear
- Scenario
- Version

UBCPlan - Future1

This Plan Type will be created only as a placeholder for possible future development. The dimensionality below represents only that necessary to ensure validation and deployment of the application in total.

The Future1 Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Department (Entity Dimension)
- FiscalYear
- Scenario
- Version

UBCPlan – Wrkforce

This Plan Type will be used for detailed Salary and Benefits planning at the Employee and Position level. This Plan Type will initially be created via the activation of the Workforce Planning module. Aggregated data from the Workforce Planning module will be fed to the ISPlan Plan Type for consolidation into the Income Statement.

The Wrkforce Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Fund
- Department (Entity Dimension)
- Program Code
- Position
- Employee
- Project Grant
- FiscalYear
- Scenario
- Version

FundPlan – FundPlan

This Plan Type will house the GPO Funding Model, and include all of the drivers and calculations needed to execute this model.

The FundPlan Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Department (Entity Dimension)
- FiscalYear
- Scenario
- Version

FundPlan – FundTxr

This Plan Type will be used enable transfers of funding between Departments, while capturing the details necessary to document the funding transfers.

The FundTxr Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Fund
- Line_Item
- Department (Entity Dimension)
- Txr_Dept
- Project Grant
- FiscalYear
- Scenario
- Version

FundPlan – Future2

This Plan Type will be created only as a placeholder for possible future development. The dimensionality below represents only that necessary to ensure validation and deployment of the application in total.

The Future2 Plan Type will consist of the following dimensions:

- Account (Account Dimension)
- Period (Time Dimension)
- Department (Entity Dimension)
- FiscalYear
- Scenario
- Version

Master Data Sources

All automated master data for the Planning applications will be sourced from either PeopleSoft or Position Management and maintained in EPMA. EPMA will provide the functionality to manage and update all hierarchies needed to support the Planning applications.

Account Dimension (UBCPlan)

The Account dimension for the UBCPlan application will be built substantially based on the Hyperion Account tree in PeopleSoft. The “STMT_REV_EXP” segment of the tree will form the core of the Account dimension for the ISPlan Plan Type and the “BALANCE_SHEET” segment will form the core of the Account dimension for the BSPlan Plan Type. The pre-configured Workforce Planning accounts will be built via the initialization of the Workforce Planning module.

Initial Build:

1. Workforce Planning will be initialized to create the Workforce Planning Account members.
2. Top level and default members will be built manually in EPMA.
3. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level accounts in the "ALL_ACCOUNTS" segment of the Hyperion Account tree.
4. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
5. Statistical and Metric Accounts will be created and maintained directly in EPMA.
6. The hierarchy, Account order and level of detail will be confirmed.
7. Planning Accounts will be created and maintained directly in EPMA.
8. Workforce Planning Accounts will be modified and maintained directly in EPMA.

Maintenance:

1. The Planning administrator will be notified of additions or changes to the Hyperion Account tree in PeopleSoft.
2. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level accounts in the "ALL_ACCOUNTS" segment of the Hyperion Account tree.
3. The Planning administrator will verify that the changes to the hierarchy are properly reflected in EPMA.
4. New Statistical and Metric Accounts will be created and maintained directly in EPMA.
5. New Planning Accounts will be created and maintained directly in EPMA.
6. New Workforce Planning Accounts will be created and maintained directly in EPMA.

Account Dimension (FundPlan)

The Account dimension for the FundPlan application will be built and maintained manually in EPMA.

Fund Dimension

The Fund dimension for both the UBCPlan and FundPlan applications will be built substantially based on the Hyperion Fund Code tree in PeopleSoft, but will contain only the level of detail needed for planning and forecasting activities. In addition, to the hierarchy represented in the Hyperion Fund Code tree, alternate hierarchies may be built to group fund codes (e.g. Operating Funds).

Initial Build:

1. Top level and default members will be built manually in EPMA.
2. Members representing Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets will be built manually in EPMA.
3. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level Fund dimension members in the Hyperion Fund Code tree.
4. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
5. The hierarchy will be confirmed.
6. Any required alternate hierarchies (e.g. Operating Funds) will be built directly in EPMA.

Maintenance:

1. The Planning administrator will be notified of additions or changes to the Hyperion Fund Code tree in PeopleSoft.
2. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level Fund dimension members in the Hyperion Fund Code tree.
3. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
4. The Planning administrator will verify that the changes to the hierarchy are properly reflected in EPMA.
5. Any required alternate hierarchies (e.g. Operating Funds) will be built directly in EPMA.

Line_Item Dimension (FundPlan)

The Line_Item dimension for the FundPlan application will be built and maintained manually in EPMA.

Department Dimension

The Department dimension for both the UBCPlan and FundPlan applications will be built substantially based on the ALL_DEPTIDS tree in PeopleSoft. In addition, to the full hierarchy represented in the ALL_DEPTIDS tree, the ALL_DEPTIDS_VAN and ALL_DEPTIDS_UBCO will be included as alternate hierarchies within the dimension.

Initial Build:

1. Workforce will be initialized before the dimension is renamed from Entity to Department.
2. Top level and default members will be built manually in EPMA.
3. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level departments in the ALL_DEPTIDS tree, ALL_DEPTIDS_VAN tree and ALL_DEPTIDS_UBCO tree.
4. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
5. The hierarchy will be confirmed.

Maintenance:

1. The Planning administrator will be notified of additions or changes to the ALL_DEPTIDS tree in PeopleSoft.
2. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level departments in the ALL_DEPTIDS tree, ALL_DEPTIDS_VAN tree and ALL_DEPTIDS_UBCO tree.
3. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
4. The Planning administrator will verify that the changes to the hierarchy are properly reflected in EPMA.
5. If required, additional alternate hierarchies (e.g. representing planned future changes to the University structure) will be created and maintained directly in EPMA.

Txr_Dept Dimension (FundPlan)

The Txr_Dept dimension for the FundPlan application will be built substantially based on the ALL_DEPTIDS tree in PeopleSoft. This dimension will essentially be a replica of the Department dimension, but will not contain the alternate hierarchies that exist in the Department dimension.

Initial Build:

1. Top level and default members will be built manually in EPMA.
2. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level departments in the ALL_DEPTIDS tree.
3. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
4. The hierarchy will be confirmed.

Maintenance:

1. The Planning administrator will be notified of additions or changes to the ALL_DEPTIDS tree in PeopleSoft.
2. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 and upper level departments in the ALL_DEPTIDS tree.
3. Member names and aliases will be written to the EPMA interface tables and imported into EPMA.
4. The Planning administrator will verify that the changes to the hierarchy are properly reflected in EPMA.

Program Code Dimension (UBCPlan)

The Program Code dimension for the UBCPlan application will be built based on the Program Code table in PeopleSoft.

Initial Build:

1. The top level and default members will be built manually in EPMA.
2. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 Program Codes in the Program Code table.
3. Member names and aliases will be written to the EPMA interface tables and imported into EPMA as children of "ACTIVE_PROGRAMS" or "INACTIVE_PROGRAMS", depending on whether the Program Code is flagged as Active or Inactive in PeopleSoft.

Maintenance:

1. Member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 Program Codes in the Program Code table.
2. Member names and aliases will be written to the EPMA interface tables and imported into EPMA as children of "ACTIVE_PROGRAMS" or "INACTIVE_PROGRAMS", depending on whether the Program Code is flagged as Active or Inactive in PeopleSoft.

Project Grant Dimension

The Project Grant dimension for both the UBCPlan and FundPlan applications will be built substantially based on the Project Grant table in PeopleSoft. In the PS_PROJECT_STATUS table, a Level 0 Department is assigned in the DEPTID field identifying the Department to which the Project Grant belongs. If this field is null, the Project Grant is used by multiple Departments. If a DEPTID exists, the Project Grant will be built as a descendant of the Department PG Parent. If the DEPTID is null, the Project Grant will be built as a descendant of the "PG_SHARED" member. Under each Department PG Parent, there will be an Active and Inactive Department PG Parent. Level 0 Project Grants will be built as children of the Active or Inactive parent based on the Active/Inactive status flag in PeopleSoft.

Initial Build:

1. The top level, default and "PG_SHARED" members will be built manually in EPMA.
2. All Departments that exist as DEPTIDs in the PS_PROJECT_STATUS field will be built as children of "ALL_PGS" and siblings of "PG_DEFAULT".
3. An Active and Inactive Department PG Parent will be built as children of each Department PG Parent.
4. DEPTID Active or Inactive parents, member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 Project Grants.
5. DEPTID Active or Inactive parents, member names and aliases will be written to the EPMA interface tables and imported into EPMA.

Maintenance:

1. All Departments that exist as DEPTIDs in the PS_PROJECT_STATUS field will be built as children of "ALL_PGS" and siblings of "PG_DEFAULT".
2. An Active and Inactive Department PG Parent will be built as children of each Department PG Parent.
3. DEPTID Active or Inactive parents, member names and aliases will be sourced from PeopleSoft via ODI for all Level 0 Project Grants.
4. DEPTID Active or Inactive parents, member names and aliases will be written to the EPMA interface tables and imported into EPMA.

Position Dimension (UBCPlan)

The Position dimension for the UBCPlan application will be built based on the Position table in the Position Management system. It is assumed that Positions will be assigned to a primary Department parent, and that each Position will be a child of that Department parent in the Position dimension.

Initial Build:

1. The top level and "No Position" members will be built manually in EPMA.
2. All Departments that exist as Department Parents will be built as children of "ALL_POSITIONS".
3. Department Parents, member names (Position Numbers) and aliases (Position Descriptions) will be sourced from Position Management via ODI for all Level 0 Positions.
4. Department Parents, member names and aliases will be written to the EPMA interface tables and imported into EPMA.

Maintenance:

1. All Departments that exist as Department Parents will be built as children of "ALL_POSITIONS".
2. Department Parents, member names and aliases will be sourced from Position Management via ODI for all Level 0 Positions.
3. Department Parents, member names and aliases will be written to the EPMA interface tables and imported into EPMA.

Employee Dimension (UBCPlan)

The Employee dimension for the UBCPlan application will be built initially via the initialization of the Workforce Planning module, but will substantially based on the Employee table in the Position Management system. The Employee list will be sorted alphabetically based on Last Name, and each Employee will be created under and alphabetic parent (e.g. "A_Employees", "B_Employees", etc.).

Initial Build:

1. The top level and default members will be built via Workforce Planning initialization.
2. The alphabetic parent members will be built manually in EPMA.
3. Member names (Employee Numbers) and aliases (Employee Names) will be sourced from Position Management via ODI for all Level 0 Positions.
4. Alphabetic Parents, member names and aliases will be written to the EPMA interface tables and imported into EPMA.

Maintenance:

1. Member names (Employee Numbers) and aliases (Employee Names) will be sourced from Position Management via ODI for all Level 0 Positions.
2. Alphabetic Parents, member names and aliases will be written to the EPMA interface tables and imported into EPMA.

Period Dimension

The Period dimension for both the UBCPlan and FundPlan applications will be built and maintained manually.

FiscalYear Dimension

The FiscalYear dimension for both the UBCPlan and FundPlan applications will be built and maintained manually. However, Workforce Planning should be initialized in the UBCPlan application before the dimension name is changed from “Year” to “FiscalYear”.

Scenario Dimension (UBCPlan)

The Scenario dimension for the UBCPlan application will be built and maintained manually.

Scenario Dimension (FundPlan)

The Scenario dimension for the FundPlan application will be built and maintained manually.

Version Dimension (UBCPlan)

The Version dimension for the UBCPlan application will be built and maintained manually.

Version Dimension (FundPlan)

The Version dimension for the FundPlan application will be built and maintained manually.

Automation

The following automation processes will exist within the implemented system:

Update Account Dimension (UBCPlan) – On Demand

Account updates from PeopleSoft are expected to be infrequent, and will be executed upon notice of changes to the Hyperion Account tree in PeopleSoft. See the **Master Data Sources** - Account Dimension (UBCPlan) section above for details on this update process.

Update Department Dimension – On Demand

Department dimension changes from PeopleSoft are expected to be infrequent, and will be executed upon notice of changes to the ALL_DEPTIDS tree in PeopleSoft. See the **Master Data Sources** - Department Dimension section above for details on this update process. Since the Trx_Dept dimension will be a replica of the Department dimension, the Trx_Dept Dimension update should be executed whenever a Department dimension update is executed.

Update Trx_Dept Dimension – On Demand

This dimension will be updated at the same times as the Department dimension. See the **Master Data Sources** - Trx_Dept Dimension (FundPlan) section above for details on this update process.

Update Fund Dimension – On Demand

Fund dimension changes from PeopleSoft are expected to be infrequent, and will be executed upon notice of changes to the Hyperion Fund tree in PeopleSoft. See the **Master Data Sources** - Fund Dimension section above for details on this update process.

Update Project Grant Dimension – Nightly

The Project Grant dimension is expected to change frequently as new Project Grants are added to PeopleSoft frequently. Updates to this dimension must occur nightly to ensure that all Actual data can be successfully loaded. See the **Master Data Sources** - Project Grant Dimension section above for details on this update process.

Update Program Code Dimension – Nightly

The Program Code dimension is expected to change frequently as new Program Codes can be added to PeopleSoft at any time. Updates to this dimension must occur nightly to ensure that all Actual data can be successfully loaded. See the **Master Data Sources** - Program Code Dimension (UBCPlan) section above for details on this update process.

Update Position Dimension – Monthly

The Position dimension is expected to be updated in the UBCPlan application on a monthly basis in order to support a monthly refresh of data from this system. See the **Master Data Sources** - Position Dimension (UBCPlan) section above for details on this update process.

Update Employee Dimension – Monthly

The Employee dimension is expected to be updated in the UBCPlan application on a monthly basis in order to support a monthly refresh of data from this system. See the **Master Data Sources - Employee Dimension (UBCPlan)** section above for details on this update process.

Load Actual Data to ISPlan – Nightly

Actual data will be loaded from PeopleSoft on a nightly basis. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **ISPlan Design - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Load Actual Data to BSPlan – Nightly

Actual data will be loaded from PeopleSoft on a nightly basis. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **Error! Reference source not found. - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Load Research Funding Data to FundTxr – On Demand

Research Data will be loaded periodically from RISE to the FundTxr Plan Type to support planning and forecasting activities. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **FundTxr Design - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Load Funding Data to FundTxr – Monthly

Funding Data will be refreshed from the Central Ledger to the FundTxr Plan Type on a monthly basis. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **FundTxr Design - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Load Commitments Data to ISPlan – Monthly

Commitments data will be refreshed from the Commitment Control Ledger to the ISPlan Plan Type on a monthly basis. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **ISPlan Design - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Load Funding Driver Data to FundPlan – On Demand

Funding Driver Data for the new Budget Model will be loaded to the FundPlan application as needed. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **FundPlan Design - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Cost Driver Data to ISPlan – TBD

Required Cost Driver data has not yet been finalized, so details and schedule must still be finalized.

Load Position Management Data to Wrkforce – Monthly

Position details will be refreshed from the Position Management system to the Wrkforce Plan Type on a monthly basis. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **Error! Reference source not found. - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Load Actual Payroll Data to Wrkforce – Monthly

Actual payroll data by Employee will be loaded from HRMS to the Wrkforce Plan Type on a monthly basis. Prior to loading this data, all necessary dimension updates must be completed and the UBCPlan and FundPlan applications must be deployed. See the **Error! Reference source not found. - Error! Reference source not found. - Error! Reference source not found.** section below for details on this load process.

Export Approved Plan – On Demand

The Approved Plan must be exported from the ISPlan Plan Type and will be loaded to the PeopleSoft Planning Ledger.

Export Transfer Details – Monthly

Transfer Details will be exported from the FundTxr Plan Type on a monthly basis and will be loaded to the Central Ledger.

Backup/Defragmentation – Nightly

Essbase databases should be backed up on a nightly basis and the databases should be defragmented on a nightly basis. The following general process should be followed:

1. A Level 0 export shall be taken from all Planning/Essbase databases
2. All Essbase objects (Outlines, Calc Scripts, Report Scripts, Load Rules) for all Planning/Essbase databases will be copied
3. Essbase security and configuration files will be copied
4. Essbase database will be restructured
5. All of the above data is copied to a backup location on a rolling 7 day basis

Security

Within each application, the dimensions that will have security applied are designated in the design sections below. A security matrix will be developed to determine the security assignments to be made to each user and group.

ISPlan Design

Overview

The ISPlan Plan Type will serve as the consolidation point for all Planning sub-models (Wrkforce and FundPlan), as well as the main model for planning and forecasting Income Statement detail for UBC. This Plan Type will provide reporting on various versions of the Plan and Forecast, and will provide Actual data for comparative analysis as well.

Database Structures

Following is a detailed description of the dimensions and structures in the ISPlan Plan Type:

Account Dimension

The Account dimension, which will be tagged as the Accounts dimension (allowing for Expense Reporting and Time Balance tagging) will mainly be based on the PeopleSoft Chart of Accounts, but will contain only the level of detail required to support planning and forecasting activities. In addition, the dimension will contain certain calculated metrics and statistical accounts required for reporting or to support calculations. Security will be defined along the Account dimension.

The “ALL_ACCOUNTS” segment of the Account dimension will be based on the PeopleSoft Hyperion Accounts tree, and should only include “STMT_REV_EXP”. The “STATS”, “METRICS” and “PLAN_ACCTS” segments of the hierarchy will be built manually in EPMA. The finalization of the members in these segments is dependent on the finalization of reporting requirements.

Within the “ALL_ACCOUNTS” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the account description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft, with no modifications. The alias will be based on the account description from PeopleSoft, with the member name appended to the end of the alias. Within the “ALL_ACCOUNTS” segment, all upper level members shall be Dynamically Calculated, and all Level 0 members shall be Stored.

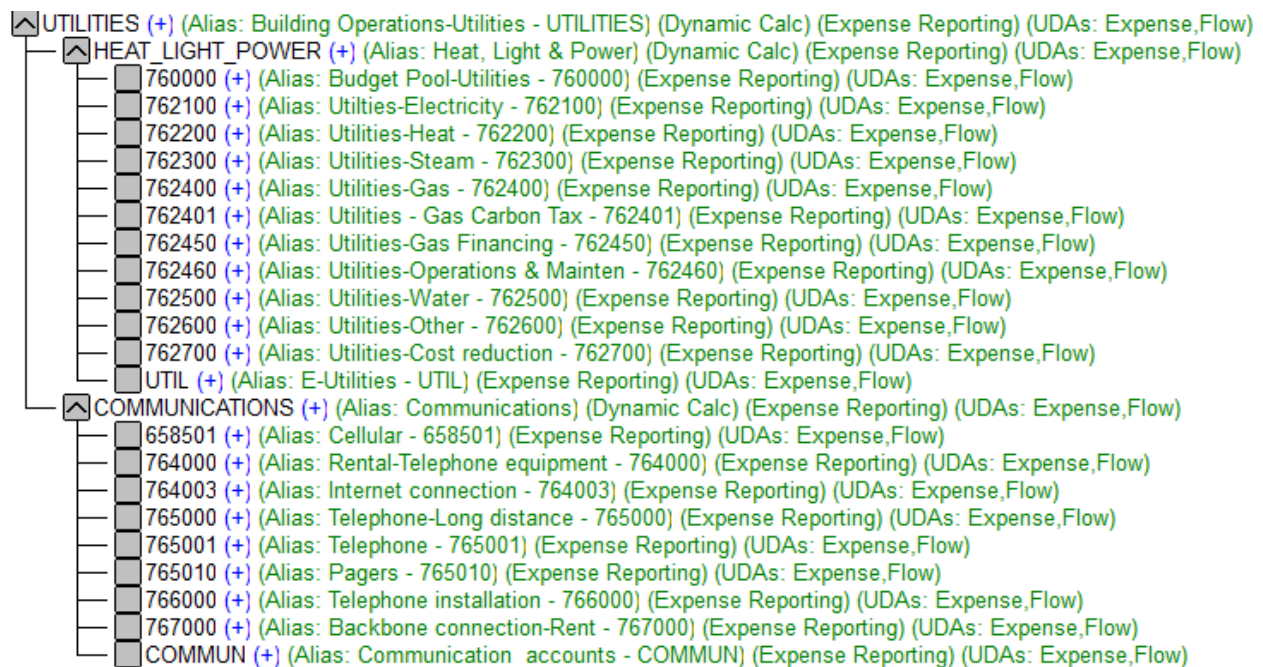
Following are key attributes of the Account dimension within the ISPlan Plan Type:

- Data Storage – Dense
- Size – Approximately 525 stored members
- Update Frequency – Updated infrequently, when new accounts required for planning and forecasting purposes are added to the Hyperion Accounts tree in PeopleSoft

The screen shot below shows a high level representation of the expected structure of the Accounts dimension in this Plan Type:



The “UTILITIES” segment of the hierarchy is shown below as an example of the naming conventions and storage properties to be used in the hierarchy:



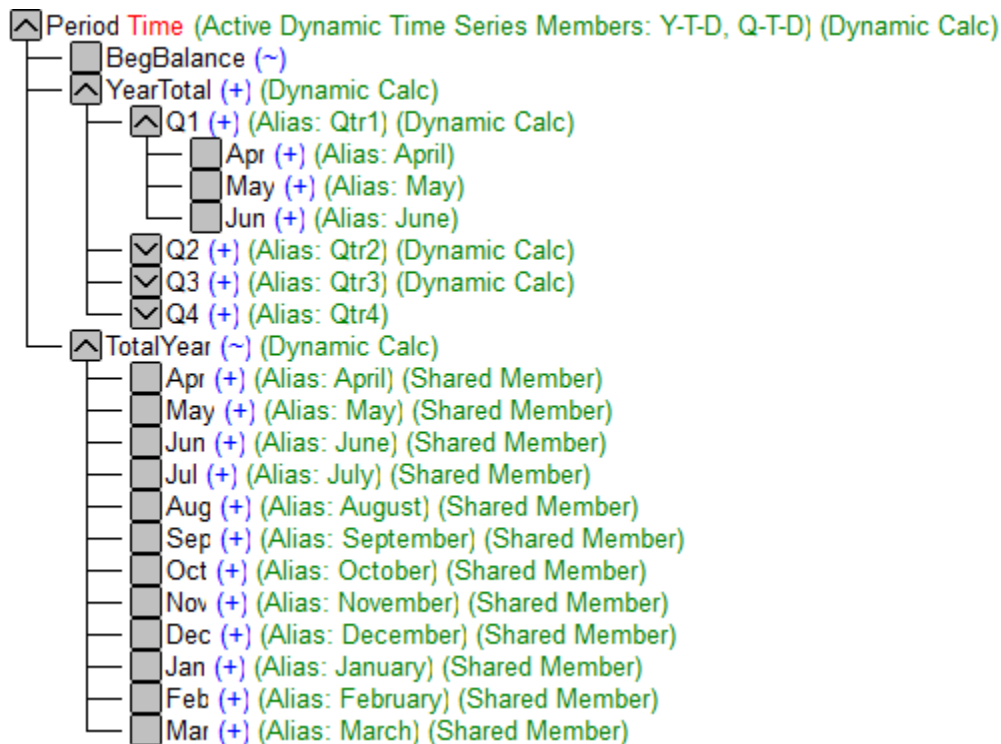
Period Dimension

The Period dimension, which will be tagged as the Time dimension (allowing for Y-T-D and Time Balance calculations) will consist of Months rolling up to Quarters, and Quarters rolling up to “YearTotal”. An additional hierarchy will be created with all 12 months rolling up directly to “TotalYear”. This additional shared hierarchy will allow for input into the “TotalYear” member on forms where the Quarters are not desired. All upper level members of this dimension will be Dynamically Calculated. This dimension will have the same structure in all Plan Types. Security is not defined along the Period dimension. Dynamic Times Series (DTS) functionality will be enabled to allow for Y-T-D and Q-T-D reporting. Security is not defined along the Period Dimension.

Following are key attributes of the Period dimension within the ISPlan Plan Type:

- Data Storage – Dense
- Size – Approximately 13 stored members
- Update Frequency – Not Updated

The screen shot below represents the expected structure of the Period dimension in this Plan Type:



Fund Dimension

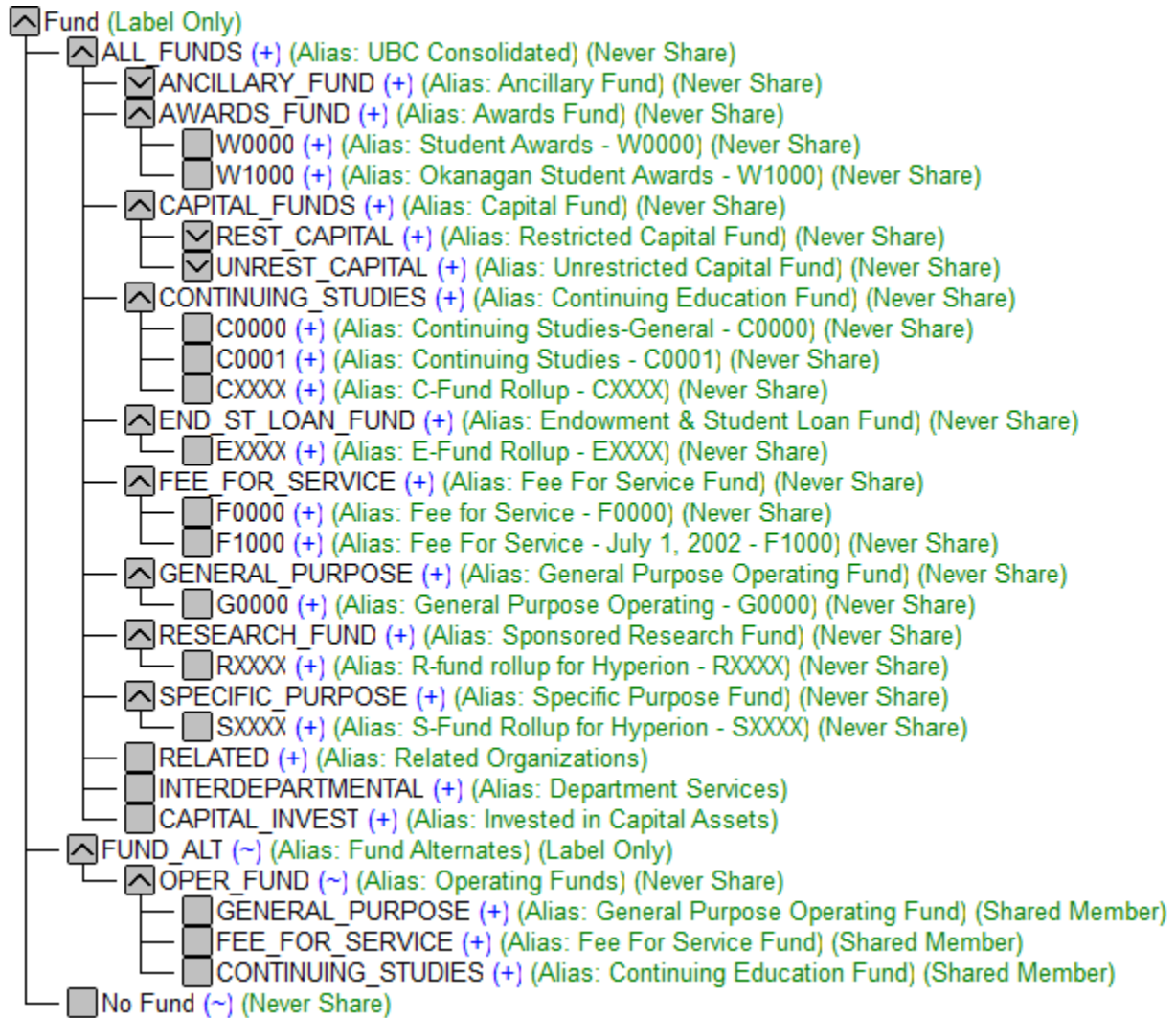
The Fund dimension will be based on the PeopleSoft Fund Codes tree, but will consist only of those fund codes or summarized representations of fund codes required for planning and forecasting purposes. Additionally, members representing Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets will also be included. The “ALL_FUNDS” segment of the Fund dimension will be based on the PeopleSoft Hyperion Fund Codes tree. The “No Fund” , Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets members will be built manually in EPMA.

Within the “ALL_FUNDS” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft, with no modifications. This consists of a 5 character string, with the first character representing the Fund Group. The alias will be based on the description from PeopleSoft, with the member name appended to the end of the alias. All members in this dimension, with the exception of the top level “Fund” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Fund Dimension.

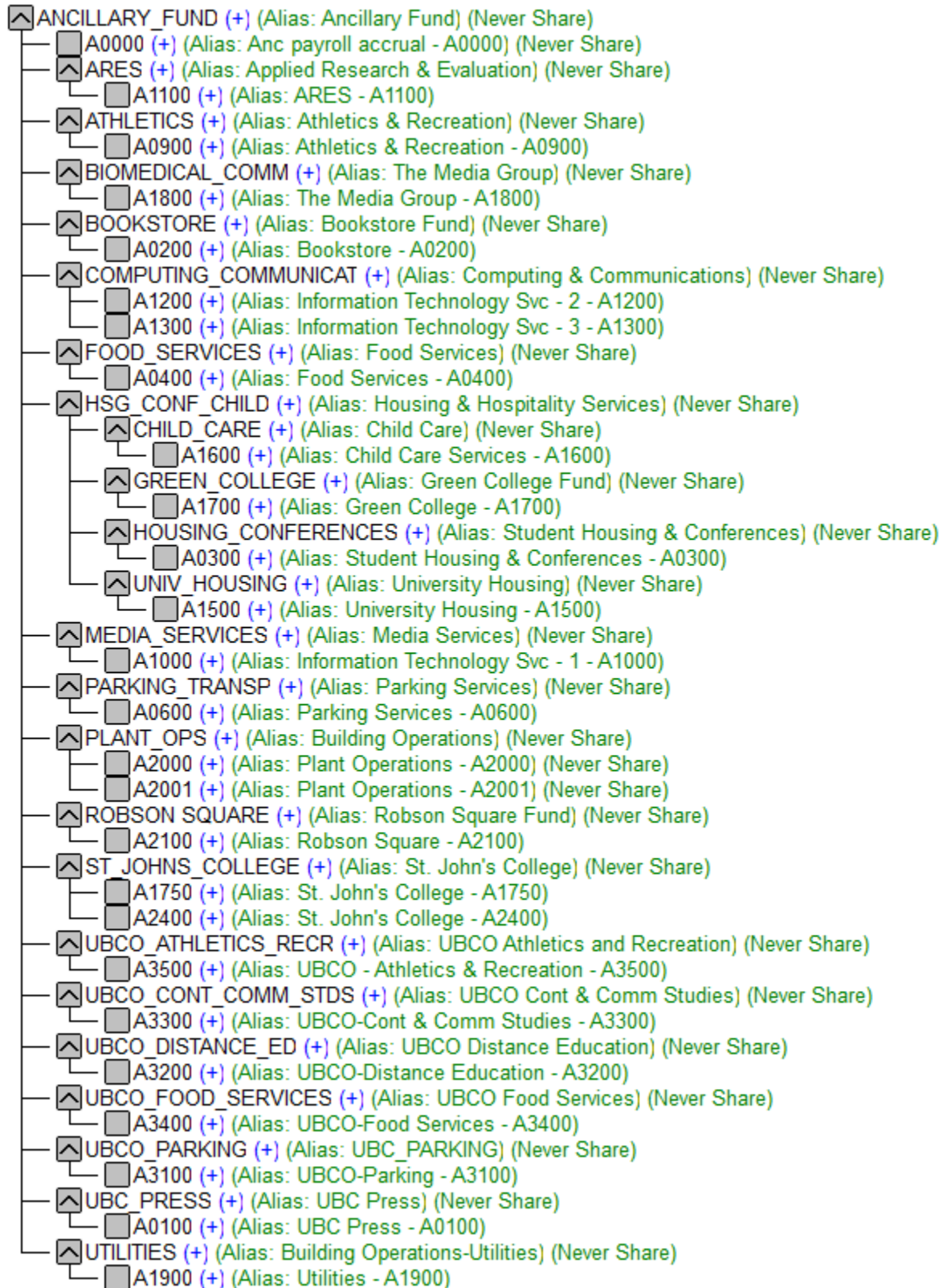
Following are key attributes of the Fund dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 75 stored members
- Update Frequency – Updated infrequently, when new funds required for planning and forecasting purposes are added to the Hyperion Fund Codes tree in PeopleSoft

The screen shot below shows a high level representation of the expected structure of the Fund dimension in this Plan Type:



While most detailed funds will be summarized within the Fund dimension as shown above, all Ancillary Funds will be included, as shown in the screen shot below:



Department Dimension

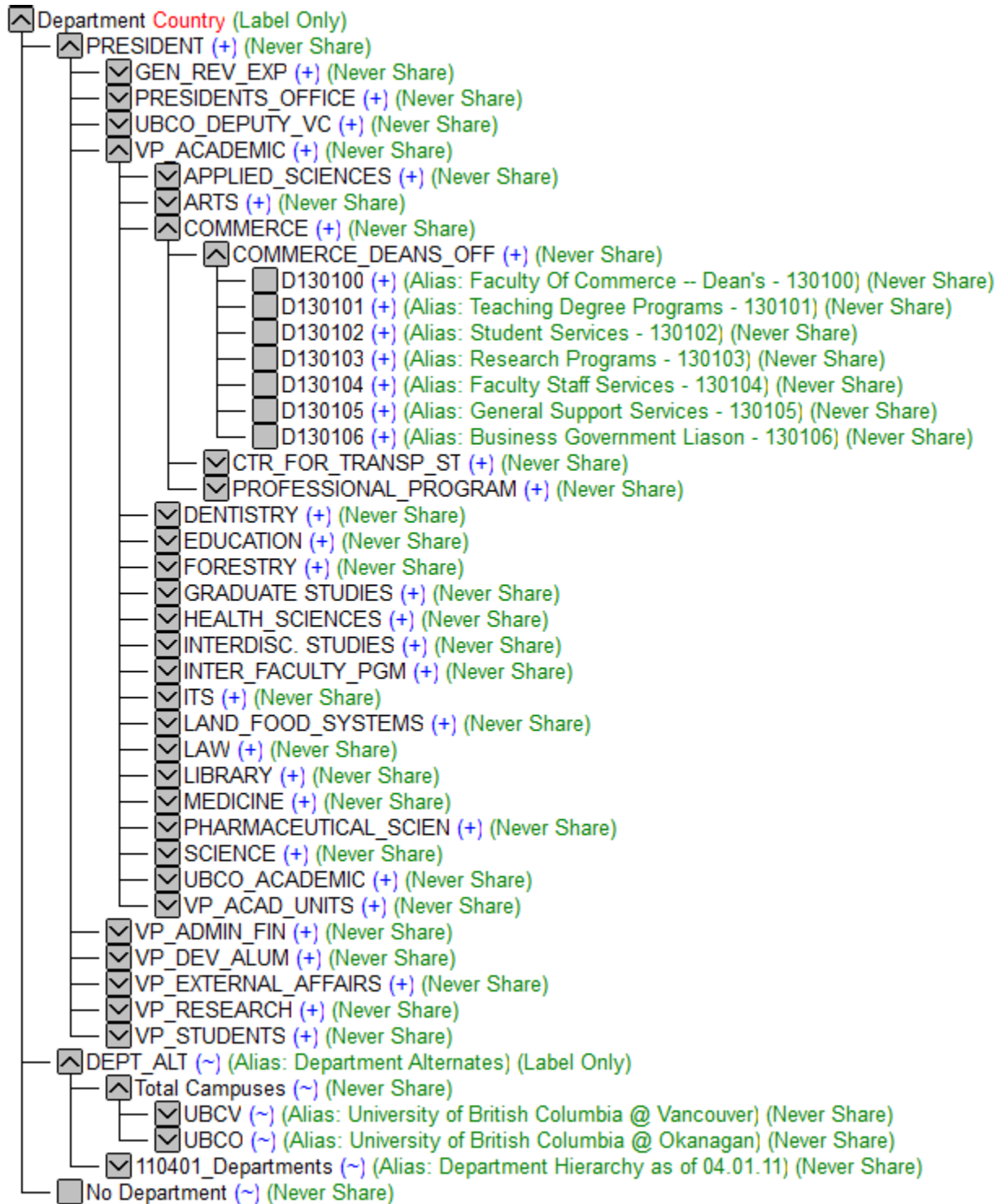
The Department dimension will be based on the PeopleSoft ALL_DEPTIDS tree, with alternate hierarchies for the ALL_DEPTIDS_VAN and ALL_DEPTIDS_UBCO. Additional alternate hierarchies could be developed to represent planned future changes to the University structure or for other reporting needs. The Department dimension will be the Entity dimension, so the hierarchy as defined will control the Workflow/Approval process. The “PRESIDENT”, “UBCV” and “UBCO” segments of the hierarchy will be based on PeopleSoft. The “No Department” member and any additional alternate hierarchies required will be built manually in EPMA.

Within the “PRESIDENT” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft (which consists of a 6 digit string), prepended with a “D”. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Department” member, and the “DEPT_ALT” member shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will be defined along the Department dimension.

Following are key attributes of the Department dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 1,300 stored members
- Update Frequency – Updated infrequently, when new Departments are added in PeopleSoft or reorganizations of the Department structure occur

The screen shot below represents a high level view of the expected structure of the Department dimension in this Plan Type:



Program Code Dimension

The Program Code dimension will be based on the PeopleSoft Program Code table. The descendants of "ACTIVE_PROGRAMS", with the exception of "PC_DEFAULT", will be based on PeopleSoft. The "ALL_PROGRAMS", "PC_DEFAULT" and "No Program Code" members will be built manually in EPMA.

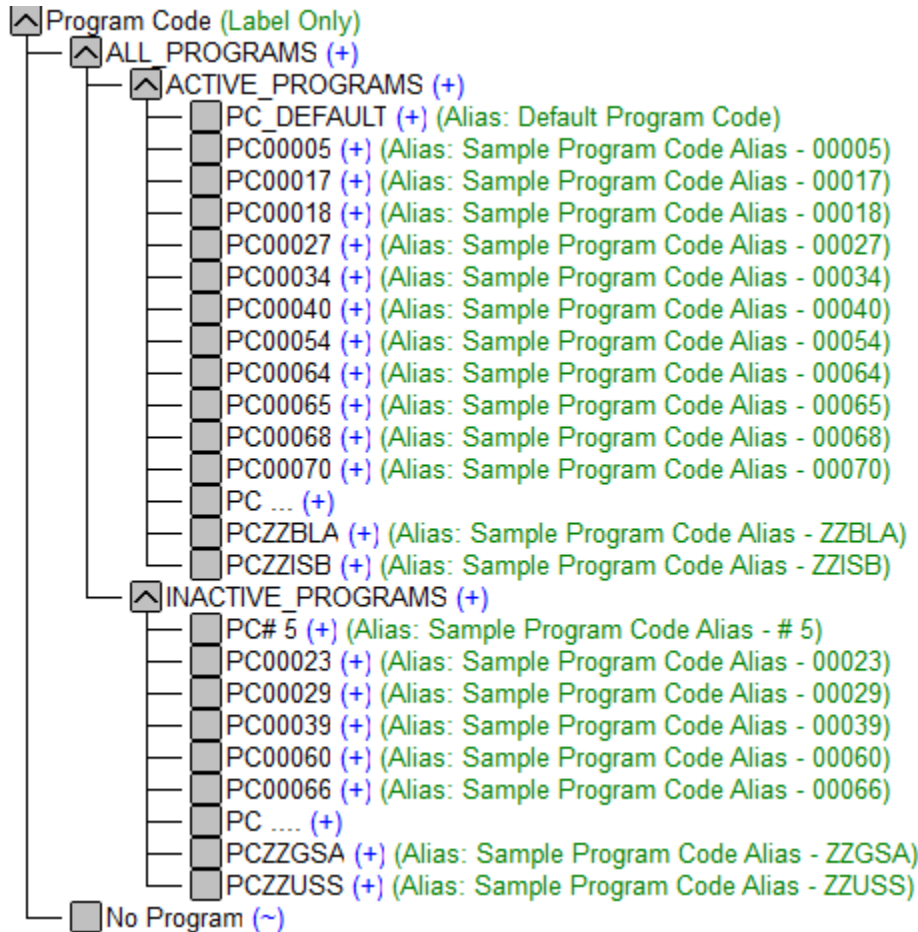
The member names for descendants of "ALL_PROGRAMS", all of which are Level 0 members, will be based on the member names from PeopleSoft, and prepended with a "PC". In general, the member name from PeopleSoft consists of a 5 character string, but there are many exceptions. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All members in this dimension shall be Stored. Security will be defined along the Program Code dimension. Planners will be granted Write access to the descendants of "ACTIVE_PROGRAMS" and Read Access to the descendants of "INACTIVE_PROGRAMS".

Aggregation performance considerations may cause a need to add additional hierarchy to this dimension. After a representative data set is loaded to the database, aggregation performance will be tested to determine whether additional hierarchy should be created.

Following are key attributes of the Program Code dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 5,000 stored members
- Update Frequency – Updated frequently, as new Program Codes can be added to PeopleSoft at any time

The screen shot below represents a high level view of the expected structure of the Program Code dimension in this Plan Type:



Project Grant Dimension

The Project Grant dimension will be based on the PeopleSoft Project Grant table. The children of “ALL_PGS”, with the exception of “PG_DEFAULT” and “PG_SHARED”, will be based on the DEPTID assigned in the PS_PROJECT_STATUS table in PeopleSoft. The “ALL_PGS”, “PG_DEFAULT”, “PG_SHARED” and “No Project Grant” members will be built manually in EPMA.

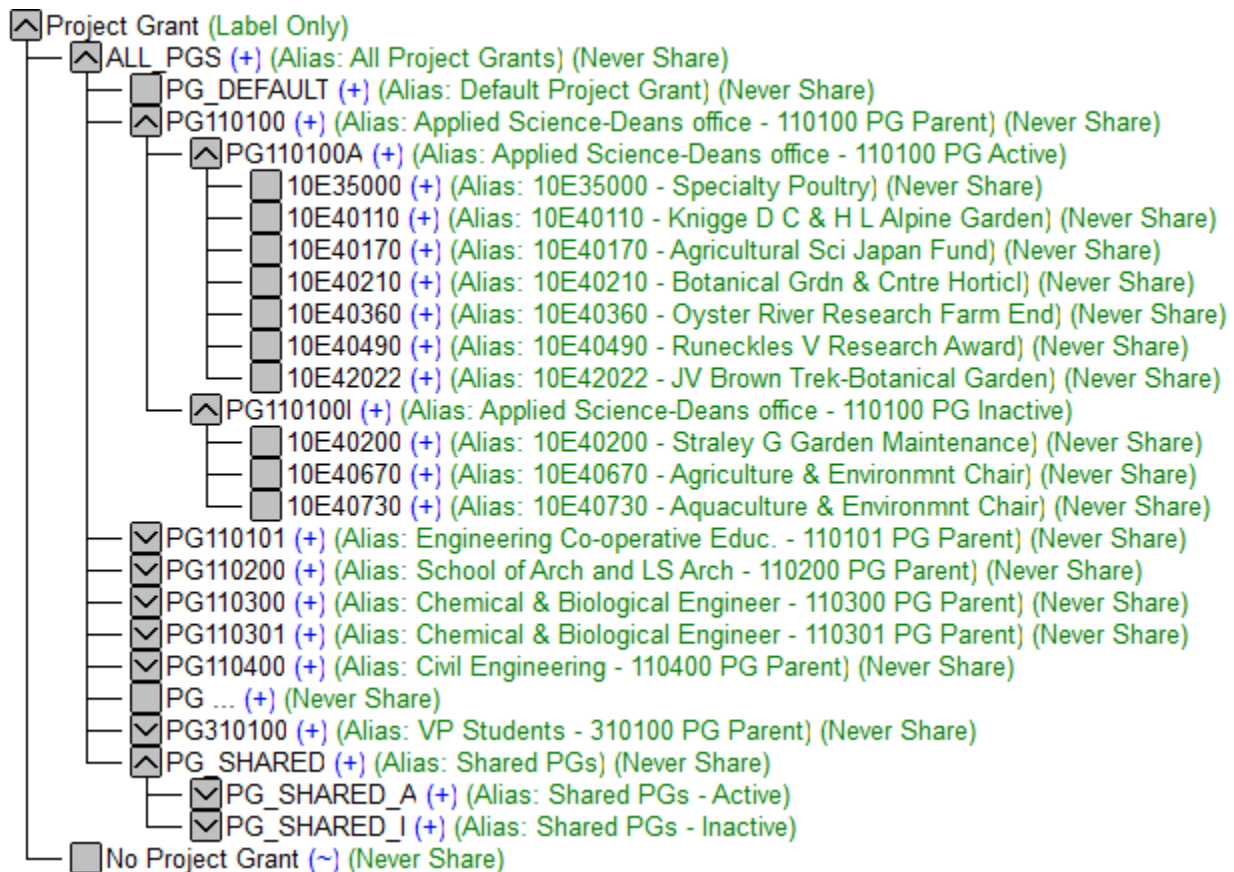
The member names for the children of “ALL_PGS” that are based on the DEPTID field will be based on the PeopleSoft member names prepended with “PG”. The aliases for these members will be based on the PeopleSoft description of the Department, appended with the member name and “PG Parent”. Under each of these DEPTID parents, an Active and Inactive parent will be created to segregate Active Project Grants from Inactive Project Grants. These levels of hierarchy exist in the Project Grant dimension to aid in the assignment of security. Security assignments within the dimension will be essential to limiting the number of PGs that users will see when budgeting, which will enhance the usability of the system.

The Level 0 descendants of “ALL_PGS” will be based on the member names from the PeopleSoft Project Grant table, with no modification. In general, the member name from PeopleSoft consists of an 8 character string, with the first two positions indicating the upper level Department with which the Project Grant is associated and the third position indicating the Fund Group with which the Project Grant is associated. The alias for these members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Project Grant” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. As noted above, security will be defined along the Project Grant dimension.

Following are key attributes of the Project Grant dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 25,000 stored members
- Update Frequency – Updated frequently, as new Project Grants are frequently added to PeopleSoft

The screen shot below represents a high level view of the expected structure of the Project Grant dimension in this Plan Type:



FiscalYear Dimension

As the word “Year” is reserved by Essbase for Dynamic Time Series functionality, the standard “Year” dimension name cannot be used if Dynamic Time Series will be used. Dynamic Time Series will be enabled in the application, so the dimension name will be changed to “FiscalYear”.

As noted in the Design Constraints section above, the application must be built with 2005/06 as the Start Year should the opportunity to load 5 years of historical data arise prior to go-live. The application will go-live for the 2011/12 plan, and the ability to create a 5 year forecast is required. Therefore, the End Year in the application at go-live will be 2015/16.

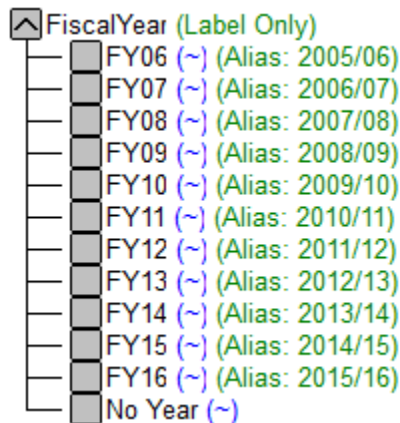
Planning does not offer flexibility in the member naming conventions within this dimension, so the FiscalYear dimension members will be named FY06, FY07, etc. Aliases will be used to incorporate UBC Fiscal Year naming standards.

The FiscalYear member will be set to Label Only, so when a user has not selected a member from the FiscalYear dimension, the data for the first stored member (FY06 in this case) will be displayed. The Level 0 members can be set to Never Share or Stored. Security is not applied along the FiscalYear dimension.

Following are key attributes of the FiscalYear dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 12 stored members
- Update Frequency – Updated annually with an additional year

The screen shot below represents the expected structure of the FiscalYear dimension in this Plan Type:



Scenario Dimension

The Scenario dimension will be used to store the data related to different Planning cycles, store Actual data and calculate Variances.

The “Scenario” member shall be set to Label Only, so when a user has not selected a member from the Scenario dimension, the data for the first stored member (Forecast in this case) will be displayed. The other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

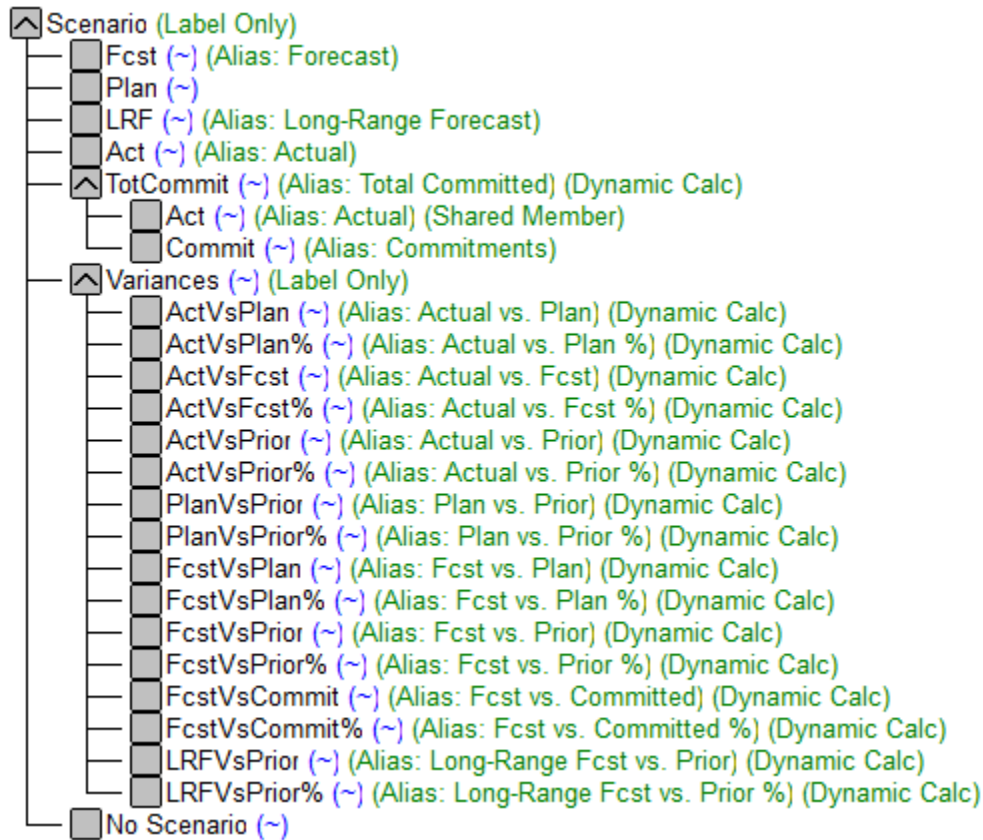
Following are key attributes of the Scenario dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 5 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Scenario member that will be configured in Planning:

	Start Year	Start Mo.	End Year	End Mo.	Data Storage	Two-Pass	Incl. Beg. Bal.	Process Management
Forecast	FY10	Oct	FY10	Mar	Store	N/A	Yes	No
Plan	FY11	Apr	FY11	Mar	Store	N/A	Yes	Yes
Long-Range Forecast	FY11	Apr	FY15	Mar	Store	N/A	Yes	No
Actual	FY06	Apr	FY06	May	Store	N/A	Yes	No
Total Committed	FY06	Apr	FY06	May	Dynamic Calc	N/A	Yes	No
Commitments	FY06	Apr	FY06	May	Store	N/A	Yes	No
All Variances	FY06	Apr	FY06	May	Dynamic Calc	Yes	Yes	No
No Scenario	FY06	Apr	FY06	May	Store	N/A	Yes	No

The screen shot below represents the expected structure of the Scenario dimension in this Plan Type:



Version Dimension

The Version dimension will be used to store different versions of plans and forecasts. The main Version for user input will be the Working Version. This is the only Version that will be enabled for Process Management, and users will be granted Write access to this member. The Approved Version will be used to store the Actual data and the final Approved Plan. Draft1 and Draft2 will be used to store different Plan or Forecast iterations (e.g. “What-If” iterations) that are created by the users. This Version may also be used to load Balance Sheet Actual activity before the system calculates balances for storage in the Approved Version. Variance Versions may be added to allow comparisons between Versions should the need for these comparisons arise.

The “Version” member shall be set to Label Only, and the other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

Following are key attributes of the Version dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 6 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Version member that will be configured in Planning:

Member	Type	Data Storage	Two-Pass	Process Management
Approved	Standard Bottom Up	Store	N/A	No
Working	Standard Bottom Up	Store	N/A	Yes
Draft1	Standard Bottom Up	Store	N/A	No
Draft2	Standard Bottom Up	Store	N/A	No
No Version	Standard Bottom Up	Store	N/A	No

The screen shot below represents the expected structure of the Version dimension in this Plan Type:



Data Loads

Following is a description of the data load processes that will feed data to the ISPlan Plan Type:

Actual Data from PeopleSoft

Actual Data from the General Ledger will be loaded to the ISPlan Plan Type. This data will be loaded to the Plan Type on a nightly basis. All data will be loaded to Level 0 dimension members in the Plan Type.

The table below describes the level of detail and locations within the database that will be populated by this data load:

Dimension	Value	Comment
Account	Level 0 Value for Account	Descendants of "STMT_REV_EXP"
Period	Current Actual Month	Designated via Substitution Variable &CurGLMth
Fund	Level 0 Value for Fund	
Department	Level 0 Value for Department	
Program Code	Level 0 Value for Program Code	
Project Grant	Level 0 Value for Project Grant	
FiscalYear	Current Actual Year	Designated via Substitution Variable &CurGLYr
Scenario	Act	Constant
Version	Approved	Constant

The Substitution Variable &CurGLMth will need to be maintained on a monthly basis. After the final General Ledger data load for a month occurs, the variable's value should be shifted to the next month. The Substitution Variable &CurGLYr will need to be maintained on an annual basis, shifting to the next year as the &CurGLMth variable shifts from "Mar" to "Apr".

As data will be loaded on a nightly basis, a calculation will need to be executed prior to the data load to clear any data in the Current Actual Month that may have previously been loaded.

Commitments Data from PeopleSoft

Future Commitments data from the Commitment Control Ledger will be loaded to the ISPlan Plan Type. This data will be loaded to the Plan Type on a monthly basis. All data will be loaded to Level 0 dimension members in the Plan Type.

The table below describes the level of detail and locations within the database that will be populated by this data load:

Dimension	Value	Comment
Account	Level 0 Value for Account	Descendants of "STMT_REV_EXP"
Period	Current Actual Month through end of year (Mar)	Current Actual Month would be designated via Substitution Variable &CurGLMth
Fund	Level 0 Value for Fund	
Department	Level 0 Value for Department	
Program Code	Level 0 Value for Program Code	
Project Grant	Level 0 Value for Project Grant	
FiscalYear	Current Actual Year	Designated via Substitution Variable &CurGLYr
Scenario	Commit	Constant
Version	Approved	Constant

Commitment data that has previously been loaded should be cleared via calculation prior to the execution of this data load.

Cost Driver Data

The cost driver data that is required in the application has not yet been defined. This section will be completed upon the final determination of the data required.

Planning Input Forms

The ISPlan Plan Type will require several forms to support the input and consolidation of the Income Statement. In general, separate but similar forms will be required to support the different Planning processes (Plan, Forecast and Long Range Forecast). The currently defined required forms include:

- General Expense Input
- Account Input (with History)
- Project Grants and Programs by Department, Account and Fund
- Consolidated Income Statement Review
- Research Grant Review
- Below the line items
- Investment Income Revenue
- Sales and Service

- Related Organizations Input
- Consolidation Review (Capital)
- Depreciation
- Amortization of Deferred Capital Contributions
- Endowment Expenditures

The sections below provide a representative sample of the different form definition methodologies that will be used to develop the Planning Input Forms listed above:

General Expense Input - Fcst

This form provides a mechanism for general expense input for the Forecast Scenario by Department, Fund, Project Grant and Program Code.

The screen shot below presents a mock-up of the anticipated form (yellow cells represent cells available for user input):

	Select Department	Select Fund	Select Project Grant	Select Program Code	Approved Plan				Working Forecast		Working Fcst vs. Plan		Working Total Committed		Working Fcst vs. Committed	
	TotalYear	April	May	June	...	March	- TotalYear	TotalYear	TotalYear	TotalYear	TotalYear	TotalYear	TotalYear	TotalYear	TotalYear	TotalYear
Field trips-Public carrier - 611000	50	5	4	7	5	5	61	(11)	19	(42)						
Travel expenses - 620000	90	7	8	7	8	8	94	(4)	34	(60)						
Relocation & moving - 634000	30	4	3	3	3	3	37	(7)	7	(30)						
E-Travel - TRAVL	60	5	5	5	5	5	60		30	(30)						
Travel	230	21	20	22	21	21	252	(22)	90	(162)						

Users will select members from the Page dimensions and then enter values for each Account. If users would like to perform a detailed build-up of values, native Supporting Detail functionality will be used.

Calculations

The following sections provide a description of the Calculation Manager Rules and other calculation logic required for the ISPlan Plan Type.

Clear Current Actual Month

A Rule is required to clear data in the Current Actual Month prior to the load of Actual data from the General Ledger.

The table below describes the region within the database that must be cleared:

Dimension	Value
Account	Descendants of "STMT_REV_EXP"
Period	&CurGLMth
Fund	Level 0 Descendants of "ALL_FUNDS"
Department	Level 0 Descendants of "PRESIDENT"
Program Code	Level 0 Descendants of "ALL_PROGRAMS"
Project Grant	Level 0 Descendants of "ALL_PGS"
FiscalYear	&CurGLYr
Scenario	Act

Dimension	Value
Version	Approved

Clear System Commitments

A Rule is required to clear previously loaded Commitment data prior to the load of Commitment data from the Commitment Control Ledger.

The table below describes the region within the database that must be cleared:

Dimension	Value
Account	Descendants of "STMT_REV_EXP"
Period	Level 0 Descendants of "YearTotal"
Fund	Level 0 Descendants of "ALL_FUNDS"
Department	Level 0 Descendants of "PRESIDENT"
Program Code	Level 0 Descendants of "ALL_PROGRAMS"
Project Grant	Level 0 Descendants of "ALL_PGS"
FiscalYear	&CurGLYr
Scenario	Fcst
Version	Load

Apply Historical Average

A Rule will be created to allow users to set plan or forecast values based on other data in the database.

This Rule will:

- Allow users to select the Account, Department, Fund, Project Grant and Program Code to modify
- Allow users to select the range of Periods within a FiscalYear to modify
- Select a Scenario, Version, FiscalYear and range of Periods & FiscalYears to base the average on
- Select whether to apply historical seasonality or set all values equal to a single average
- Input an adjustment % to be applied to the average

Apply Historical Seasonality

A Rule will be created to allow users to apply historical seasonality to the values that they have planned or forecast, without changing the total value planned.

This Rule will:

- Allow users to select the Account, Department, Fund, Project Grant and Program Code to modify
- Allow users to select the range of Periods within a FiscalYear to modify
- Select a Scenario, Version and FiscalYear to spread based on

Calculate Research Revenue and Expense

A Rule will be used to calculate revenue and expenses at approximately the Faculty level after the Research Grants have been reviewed by the Faculties.

User Aggregation

A Rule will be used to allow users to select the areas in the database (to which they have access) and aggregate the database for those areas.

Revenue Calculations

A Rule will be needed to perform the following Central Budget Office revenue calculations:

- Calculate Investment Income based on a percentage input member applied to the Working Capital balance
- Calculate Research Overhead based on a percentage input member applied to the total Research Grants
- Calculate Ancillary Revenue based on a percentage input member applied to the total Ancillary Revenue
- Calculate Endowment Overhead Revenue and offsetting Overhead Expense based on a percentage input applied to an input assumed Market Value

Capital Consolidation

A Rule will be used to offset values that have been entered by users for Capital items into a central department ("810100"). Rule will also move any Capital values input in non-Capital Funds into a Capital Fund.

Related Org Consolidation

A Rule will be used to collapse the contribution of Related Organizations into one line on the Income Statement.

Interdepartmental Services Elimination

A Rule will be used to eliminate Interdepartmental Services values.

Depreciation/Amortization Calculations

A Rule will be used to calculate depreciation based on the Net Book Value by Asset Type along with input Useful Lives by Asset Type.

Spreadback

A Rule will be created to allow users to spread an input value across all PGs and Program Codes for a Level 0 Department, Account and Fund. Users will have the ability to spread based on the values in any Account and any Scenario->Version->FiscalYear->Period combination.

Generate Forecast

A Rule will be created to allow users to automatically update the Forecast based on YTD Actuals and set the Forecast equal to the Approved Plan.

This Rule will:

- Import Actual data into the selected Version of the Forecast
- Import updated System Commitments into the selected Version of the Forecast
- Calculate the variance versus the Approved Plan or any other stored Scenario and Version
- Eliminate the Variance by forcing the Total Year to match the selected Scenario and Version
- The user will have the option to spread the calculated variance evenly or proportionately, or to plug the variance into one Period
- The user will also have the option to allow or disallow negative values (if negative values are disallowed, variances will remain)

BSPlan Design

Overview

The BSPlan Plan Type will serve as the main model for planning and forecasting the Balance Sheet for those Departments that are required to plan a Balance Sheet. This Plan Type will provide reporting on various versions of the Plan and Forecast, and will provide Actual data for comparative analysis as well.

Database Structures

Following is a detailed description of the dimensions and structures in the BSPlan Plan Type:

Account Dimension

The Account dimension, which will be tagged as the Accounts dimension (allowing for Expense Reporting and Time Balance tagging) will mainly be based on the PeopleSoft Chart of Accounts, but will contain only the level of detail required to support planning and forecasting activities. Security will be defined along the Account dimension.

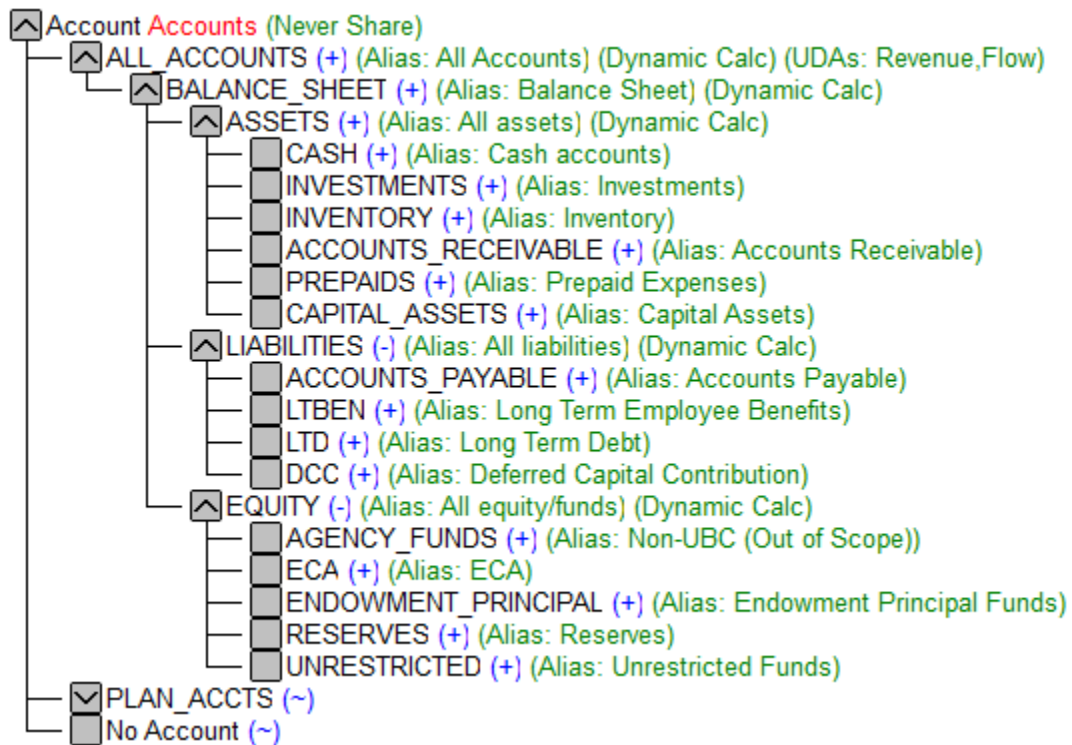
The “ALL_ACCOUNTS” segment of the Account dimension will be based on the PeopleSoft Hyperion Accounts tree, and should only include “BALANCE_SHEET” and its descendants. The “PLAN_ACCTS” segment of the hierarchy will be built manually in EPMA. The finalization of the members in these segments is dependent on the finalization of reporting requirements.

Within the “BALANCE_SHEET” segment, the naming convention shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The Balance Sheet will be planned at a very high level, so detailed numeric accounts will not be included in this hierarchy. Within the “ALL_ACCOUNTS” segment, all upper level members shall be Dynamically Calculated, and all Level 0 members shall be Stored.

Following are key attributes of the Account dimension within the BSPlan Plan Type:

- Data Storage – Dense
- Size – Approximately 25 stored members
- Update Frequency – Updated infrequently, when new accounts required for planning and forecasting purposes are added to the Hyperion Accounts tree in PeopleSoft

The screen shot below shows the expected structure of the Accounts dimension in this Plan Type:



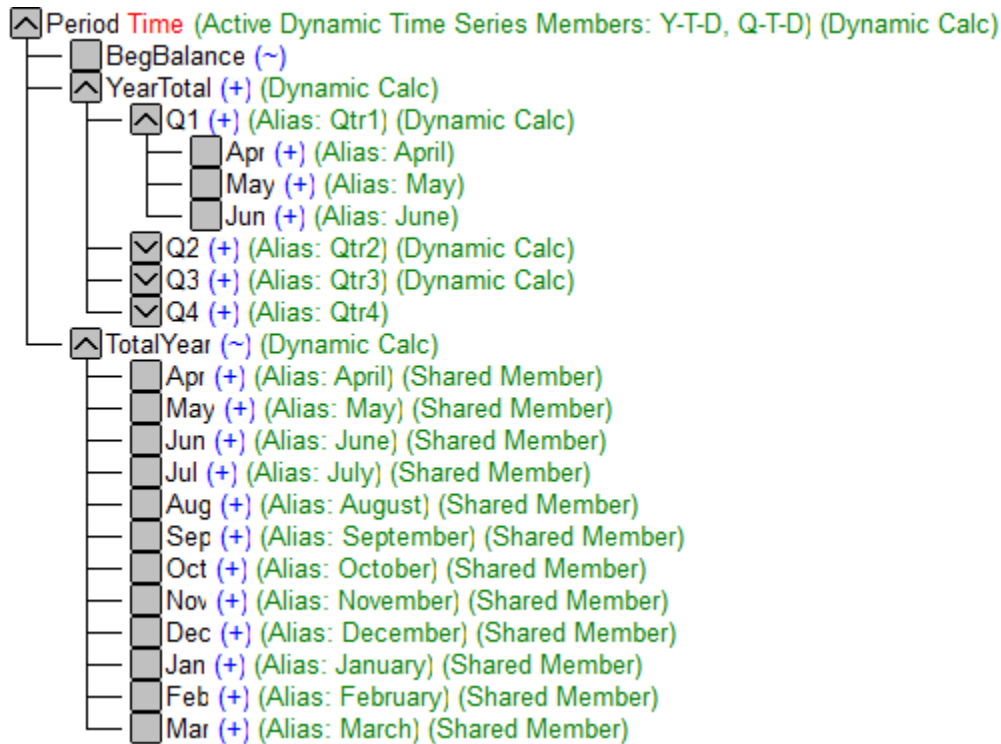
Period Dimension

The Period dimension, which will be tagged as the Time dimension (allowing for Y-T-D and Time Balance calculations) will consist of Months rolling up to Quarters, and Quarters rolling up to “YearTotal”. An additional hierarchy will be created with all 12 months rolling up directly to “TotalYear”. This additional shared hierarchy will allow for input into the “TotalYear” member on forms where the Quarters are not desired. All upper level members of this dimension will be Dynamically Calculated. This dimension will have the same structure in all Plan Types. Security is not defined along the Period dimension. Dynamic Times Series (DTS) functionality will be enabled to allow for Y-T-D and Q-T-D reporting. Security is not defined along the Period Dimension.

Following are key attributes of the Period dimension within the BSPlan Plan Type:

- Data Storage – Dense
- Size – Approximately 13 stored members
- Update Frequency – Not Updated

The screen shot below represents the expected structure of the Period dimension in this Plan Type:



Fund Dimension

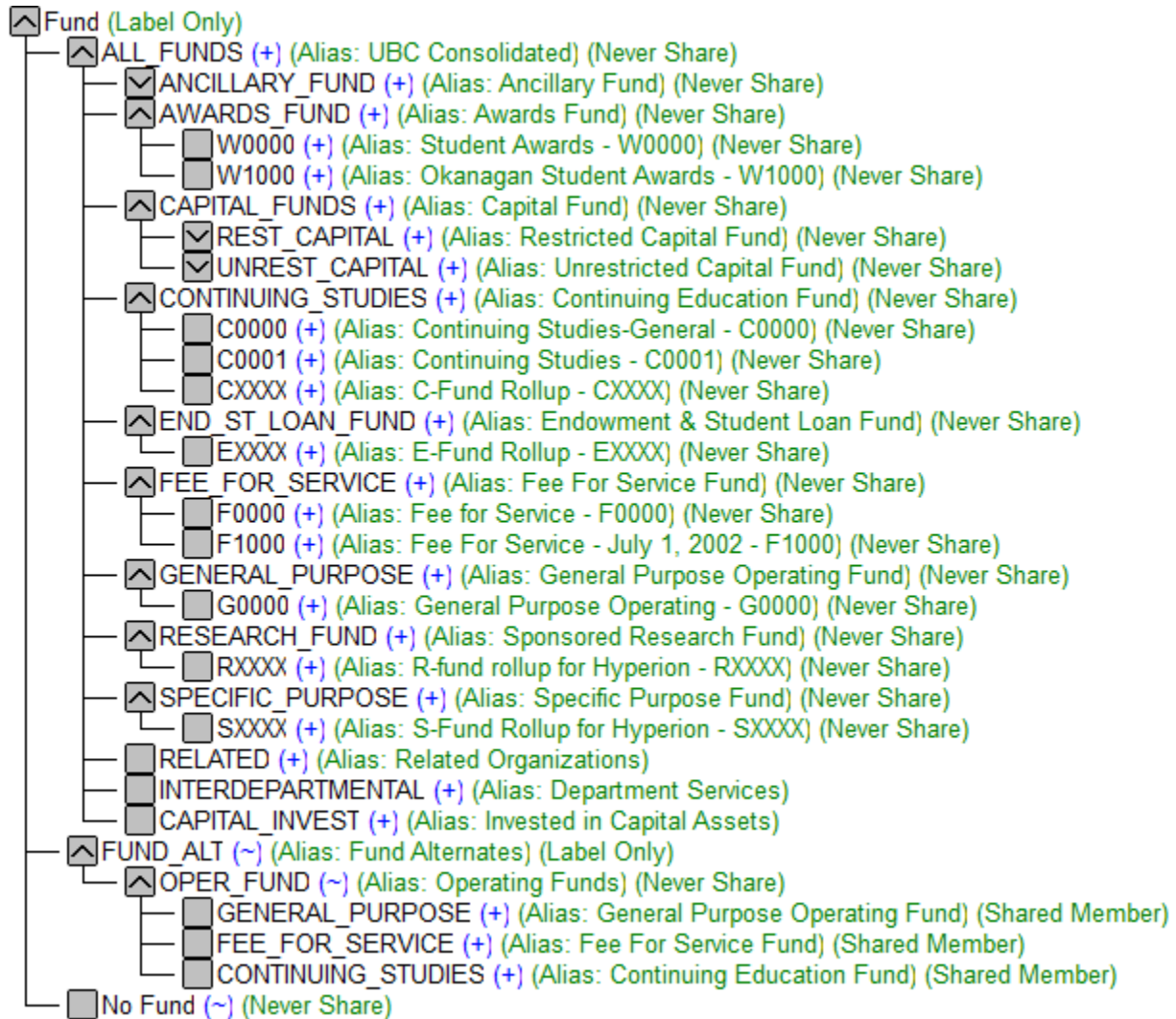
The Fund dimension will be based on the PeopleSoft Fund Codes tree, but will consist only of those fund codes or summarized representations of fund codes required for planning and forecasting purposes. Additionally, members representing Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets will also be included. The “ALL_FUNDS” segment of the Fund dimension will be based on the PeopleSoft Hyperion Fund Codes tree. The “No Fund” , Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets members will be built manually in EPMA.

Within the “ALL_FUNDS” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft, with no modifications. This consists of a 5 character string, with the first character representing the Fund Group. The alias will be based on the description from PeopleSoft, with the member name appended to the end of the alias. All members in this dimension, with the exception of the top level “Fund” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Fund Dimension.

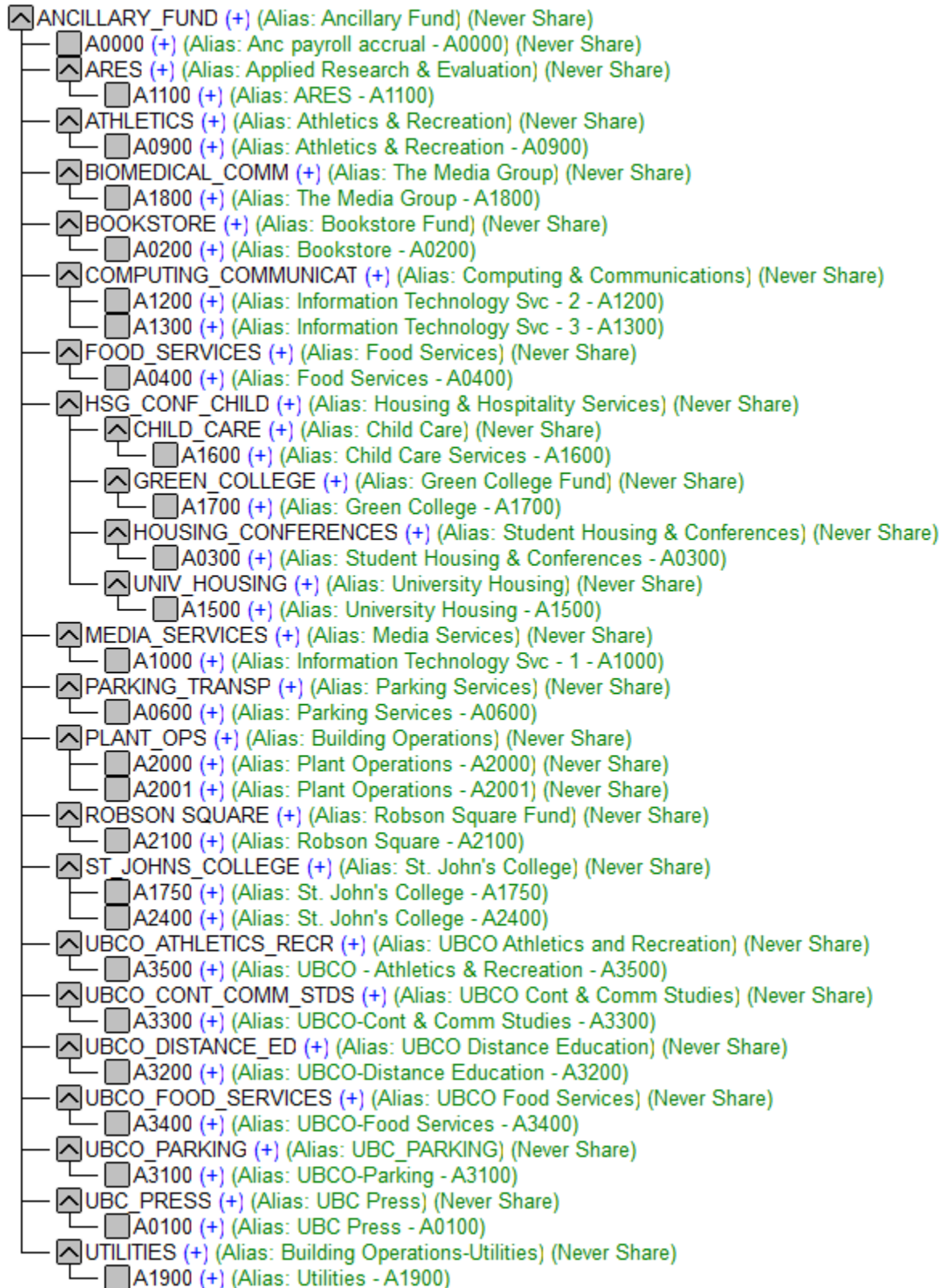
Following are key attributes of the Fund dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 75 stored members
- Update Frequency – Updated infrequently, when new funds required for planning and forecasting purposes are added to the Hyperion Fund Codes tree in PeopleSoft

The screen shot below shows a high level representation of the expected structure of the Fund dimension in this Plan Type:



While most detailed funds will be summarized within the Fund dimension as shown above, all Ancillary Funds will be included, as shown in the screen shot below:



Department Dimension

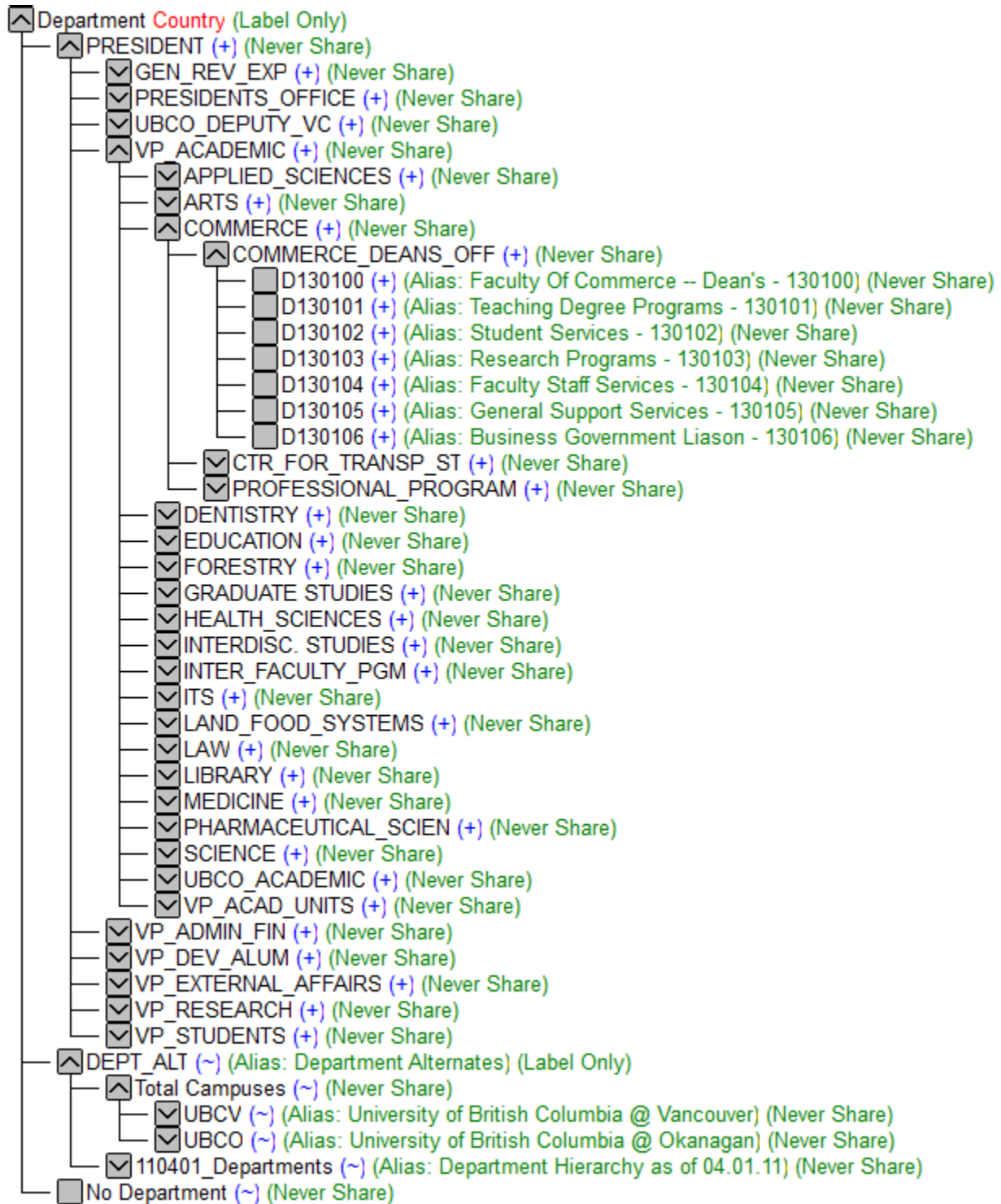
The Department dimension will be based on the PeopleSoft ALL_DEPTIDS tree, with alternate hierarchies for the ALL_DEPTIDS_VAN and ALL_DEPTIDS_UBCO. Additional alternate hierarchies could be developed to represent planned future changes to the University structure or for other reporting needs. The Department dimension will be the Entity dimension, so the hierarchy as defined will control the Workflow/Approval process. The “PRESIDENT”, “UBCV” and “UBCO” segments of the hierarchy will be based on PeopleSoft. The “No Department” member and any additional alternate hierarchies required will be built manually in EPMA.

Within the “PRESIDENT” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft (which consists of a 6 digit string), prepended with a “D”. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Department” member, and the “DEPT_ALT” member shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will be defined along the Department dimension.

Following are key attributes of the Department dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 1,300 stored members
- Update Frequency – Updated infrequently, when new Departments are added in PeopleSoft or reorganizations of the Department structure occur

The screen shot below represents a high level view of the expected structure of the Department dimension in this Plan Type:



Project Grant Dimension

The Project Grant dimension will be based on the PeopleSoft Project Grant table. The children of “ALL_PGS”, with the exception of “PG_DEFAULT” and “PG_SHARED”, will be based on the DEPTID assigned in the PS_PROJECT_STATUS table in PeopleSoft. The “ALL_PGS”, “PG_DEFAULT”, “PG_SHARED” and “No Project Grant” members will be built manually in EPMA.

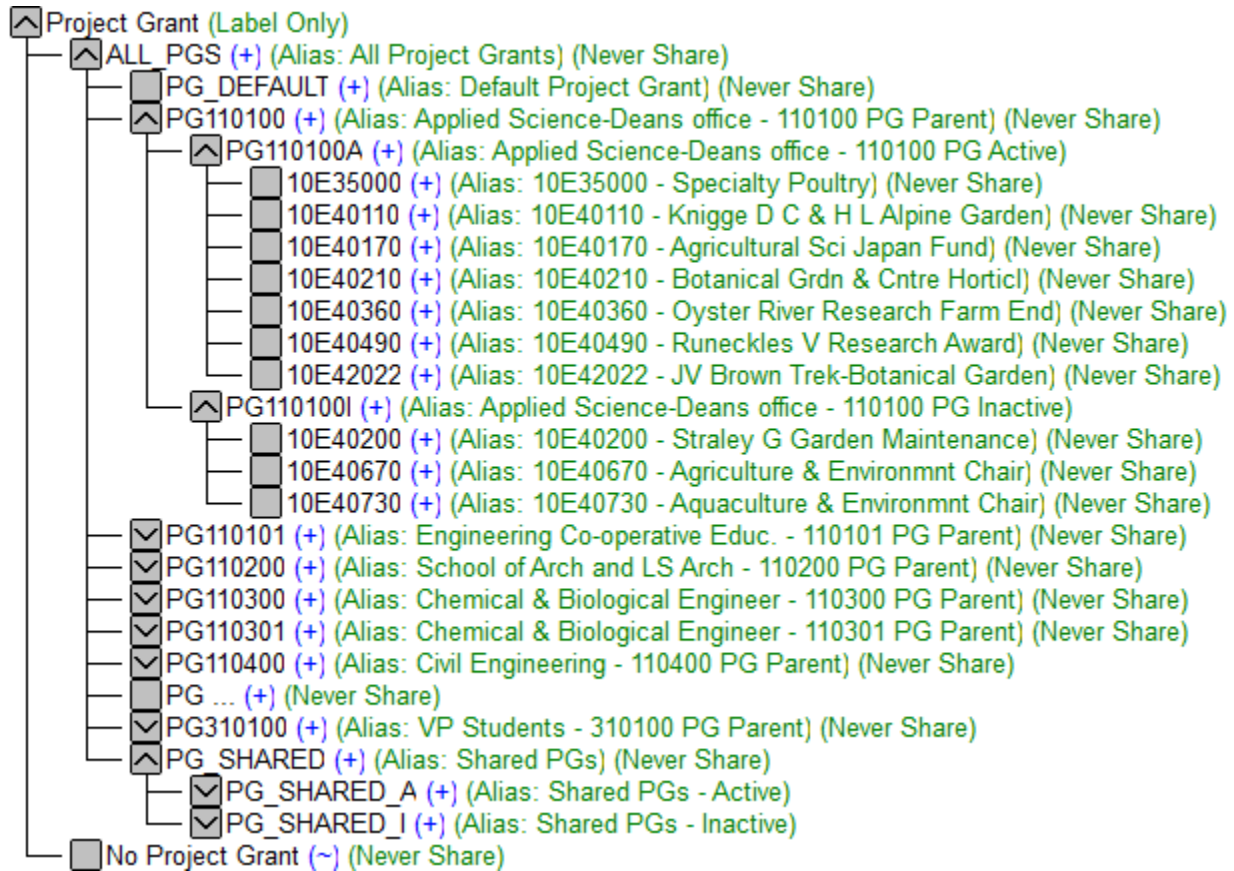
The member names for the children of “ALL_PGS” that are based on the DEPTID field will be based on the PeopleSoft member names prepended with “PG”. The aliases for these members will be based on the PeopleSoft description of the Department, appended with the member name and “PG Parent”. Under each of these DEPTID parents, an Active and Inactive parent will be created to segregate Active Project Grants from Inactive Project Grants. These levels of hierarchy exist in the Project Grant dimension to aid in the assignment of security. Security assignments within the dimension will be essential to limiting the number of PGs that users will see when budgeting, which will enhance the usability of the system.

The Level 0 descendants of “ALL_PGS” will be based on the member names from the PeopleSoft Project Grant table, with no modification. In general, the member name from PeopleSoft consists of an 8 character string, with the first two positions indicating the upper level Department with which the Project Grant is associated and the third position indicating the Fund Group with which the Project Grant is associated. The alias for these members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Project Grant” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. As noted above, security will be defined along the Project Grant dimension.

Following are key attributes of the Project Grant dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 25,000 stored members
- Update Frequency – Updated frequently, as new Project Grants are frequently added to PeopleSoft

The screen shot below represents a high level view of the expected structure of the Project Grant dimension in this Plan Type:



FiscalYear Dimension

As the word “Year” is reserved by Essbase for Dynamic Time Series functionality, the standard “Year” dimension name cannot be used if Dynamic Time Series will be used. Dynamic Time Series will be enabled in the application, so the dimension name will be changed to “FiscalYear”.

As noted in the Design Constraints section above, the application must be built with 2005/06 as the Start Year should the opportunity to load 5 years of historical data arise prior to go-live. The application will go-live for the 2011/12 plan, and the ability to create a 5 year forecast is required. Therefore, the End Year in the application at go-live will be 2015/16.

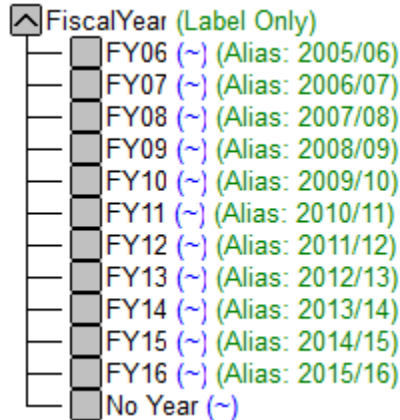
Planning does not offer flexibility in the member naming conventions within this dimension, so the FiscalYear dimension members will be named FY06, FY07, etc. Aliases will be used to incorporate UBC Fiscal Year naming standards.

The FiscalYear member will be set to Label Only, so when a user has not selected a member from the FiscalYear dimension, the data for the first stored member (FY06 in this case) will be displayed. The Level 0 members can be set to Never Share or Stored. Security is not applied along the FiscalYear dimension.

Following are key attributes of the FiscalYear dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 12 stored members
- Update Frequency – Updated annually with an additional year

The screen shot below represents the expected structure of the FiscalYear dimension in this Plan Type:



Scenario Dimension

The Scenario dimension will be used to store the data related to different Planning cycles, store Actual data and calculate Variances.

The “Scenario” member shall be set to Label Only, so when a user has not selected a member from the Scenario dimension, the data for the first stored member (Forecast in this case) will be displayed. The other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

Following are key attributes of the Scenario dimension within the BSPlan Plan Type:

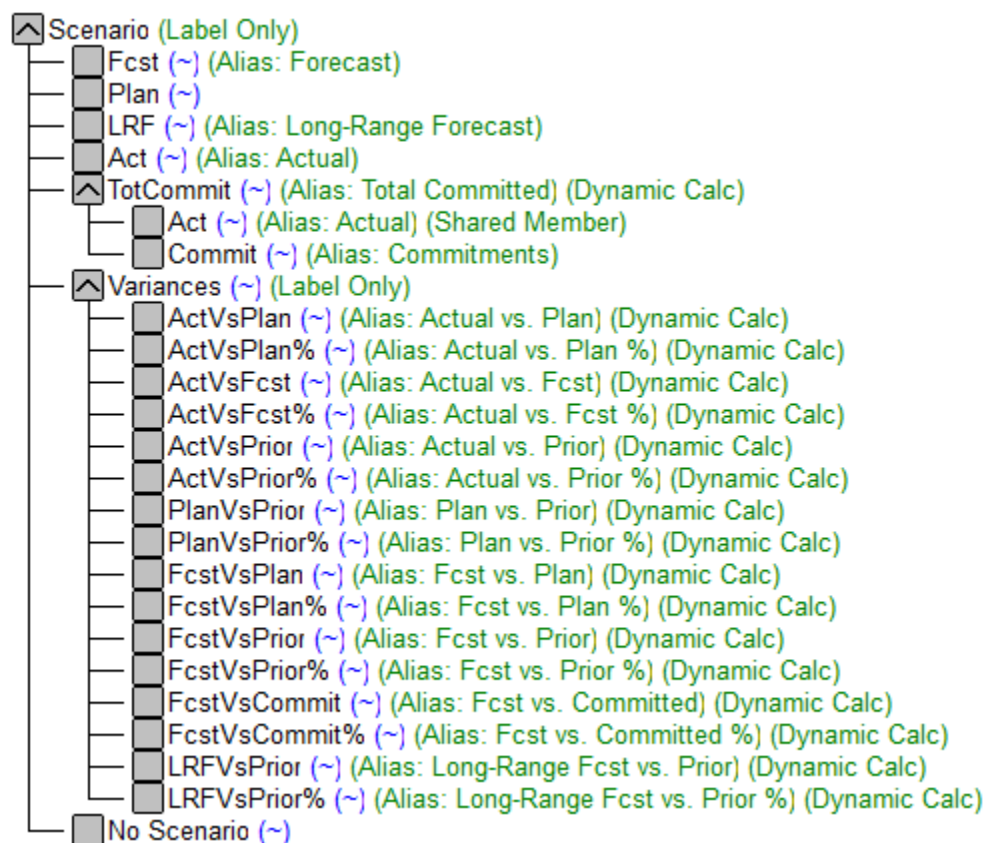
- Data Storage – Sparse
- Size – Approximately 5 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Scenario member that will be configured in Planning:

	Start Year	Start Mo.	End Year	End Mo.	Data Storage	Two-Pass	Incl. Beg. Bal.	Process Management
Forecast	FY10	Oct	FY10	Mar	Store	N/A	Yes	No
Plan	FY11	Apr	FY11	Mar	Store	N/A	Yes	Yes
Long-Range Forecast	FY11	Apr	FY15	Mar	Store	N/A	Yes	No
Actual	FY06	Apr	FY06	May	Store	N/A	Yes	No
Total Committed	FY06	Apr	FY06	May	Dynamic Calc	N/A	Yes	No
Commitments	FY06	Apr	FY06	May	Store	N/A	Yes	No
All Variances	FY06	Apr	FY06	May	Dynamic Calc	Yes	Yes	No

	Start Year	Start Mo.	End Year	End Mo.	Data Storage	Two-Pass	Incl. Beg. Bal.	Process Management
No Scenario	FY06	Apr	FY06	May	Store	N/A	Yes	No

The screen shot below represents the expected structure of the Scenario dimension in this Plan Type:



Version Dimension

The Version dimension will be used to store different versions of plans and forecasts. The main Version for user input will be the Working Version. This is the only Version that will be enabled for Process Management, and users will be granted Write access to this member. The Approved Version will be used to store the Actual data and the final Approved Plan. Draft1 and Draft2 will be used to store different Plan or Forecast iterations (e.g. “What-If” iterations) that are created by the users. This Version may also be used to load Balance Sheet Actual activity before the system calculates balances for storage in the Approved Version. Variance Versions may be added to allow comparisons between Versions should the need for these comparisons arise.

The “Version” member shall be set to Label Only, and the other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

Following are key attributes of the Version dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 6 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Version member that will be configured in Planning:

Member	Type	Data Storage	Two-Pass	Process Management
Approved	Standard Bottom Up	Store	N/A	No
Working	Standard Bottom Up	Store	N/A	Yes
Draft1	Standard Bottom Up	Store	N/A	No
Draft2	Standard Bottom Up	Store	N/A	No
No Version	Standard Bottom Up	Store	N/A	No

The screen shot below represents the expected structure of the Version dimension in this Plan Type:



Data Loads

Following is a description of the data load processes that will feed data to the BSPlan Plan Type:

Actual Data from PeopleSoft

Actual Data from the General Ledger will be loaded to the BSPlan Plan Type. This data will be loaded to the Plan Type on a nightly basis. All data will be loaded to Level 0 dimension members in the Plan Type.

The table below describes the level of detail and locations within the database that will be populated by this data load:

Dimension	Value	Comment
Account	Level 0 Value for Account	Descendants of "BALANCE_SHEET"
Period	Current Actual Month	Designated via Substitution Variable &CurGLMth
Fund	Level 0 Value for Fund	
Department	Level 0 Value for Department	
Program Code	Level 0 Value for Program Code	
Project Grant	Level 0 Value for Project Grant	
FiscalYear	Current Actual Year	Designated via Substitution Variable &CurGLYr
Scenario	Act	Constant
Version	Approved	Constant

The Substitution Variable &CurGLMth will need to be maintained on a monthly basis. After the final General Ledger data load for a month occurs, the variable’s value should be shifted to the next month. The Substitution Variable &CurGLYr will need to be maintained on an annual basis, shifting to the next year as the &CurGLMth variable shifts from “Mar” to “Apr”.

As data will be loaded on a nightly basis, a calculation will need to be executed prior to the data load to clear any data in the Current Actual Month that may have previously been loaded.

Planning Input Forms

The BSPlan Plan Type will require at least one form to support the input and consolidation of the Balance Sheet. In general, separate but similar forms will be required to support the different Planning processes (Plan, Forecast and Long Range Forecast). The currently defined required form(s) include:

- Balance Sheet Input

The sections below provide a representative sample of the different form definition methodologies that will be used to develop the Planning Input Forms listed above:

Balance Sheet Input - Fcst

This form provides a mechanism for balance sheet input for the Forecast Scenario by Department, Fund and Project Grant.

The screen shot below presents a mock-up of the anticipated form (yellow cells represent cells available for user input):

	Select Department	Select Fund	Select Version			
	April	May	June	...	December	- TotalYear
Cash accounts	1,322,752	1,272,371	1,270,225	1,225,110	1,199,224	1,199,224
Investments	387,073	394,814	402,710	398,683	406,657	406,657
Inventory	21,845	21,627	22,060	21,839	22,276	22,276
Accounts Receivable	874	891	882	900	918	918
Prepaid Expenses	1,748	1,731	1,766	1,784	1,766	1,766
Capital Assets	1,121,858	1,133,077	1,155,739	1,190,411	1,214,219	1,214,219
All assets	2,856,150	2,824,511	2,853,382	2,838,727	2,845,060	2,845,060
Accounts Payable	4,370	4,455	4,410	4,500	4,590	4,590
Long Term Employee Benefits	14,858	14,709	15,003	14,853	15,150	15,150
Long Term Debt	560,929	549,710	538,716	527,942	517,383	517,383
Deferred Capital Contribution	29,716	29,419	29,125	29,708	29,411	29,411
All liabilities	609,873	598,293	587,254	577,003	566,534	566,534
Non-UBC	20,450	20,450	20,450	20,450	20,450	20,450
ECA	6,509	6,639	6,573	6,704	6,838	6,838
Endowment Principal Funds	1,234,044	1,221,704	1,246,138	1,233,677	1,258,351	1,258,351
Reserves	200,342	200,342	200,342	200,342	200,342	200,342
Unrestricted Funds	784,932	777,083	792,625	800,551	792,545	792,545
All equity/funds	2,246,277	2,226,218	2,266,128	2,261,724	2,278,526	2,278,526
Balance Sheet						

Users will select members from the Page dimensions and then enter values for each Account. Values will be entered as balances, rather than activity. If users would like to perform a detailed build-up of values, native Supporting Detail functionality will be used.

Calculations

The following sections provide a description of the Calculation Manager Rules and other calculation logic required for the BSPlan Plan Type.

Clear Current Actual Month

A Rule is required to clear data in the Current Actual Month prior to the load of Actual data from the General Ledger.

The table below describes the region within the database that must be cleared:

Dimension	Value
Account	Descendants of "BALANCE_SHEET"
Period	&CurGLMth
Fund	Level 0 Descendants of "ALL_FUNDS"
Department	Level 0 Descendants of "PRESIDENT"
Project Grant	Level 0 Descendants of "ALL_PGS"
FiscalYear	&CurGLYr
Scenario	Act
Version	Approved

User Aggregation

A Rule will be used to allow users to select the areas in the database (to which they have access) and aggregate the database for those areas.

Wrkforce Design

Overview

The Wrkforce Plan Type will be used to calculate detailed Salary and Benefit expenses by employee and position. The design of this Plan Type is based on the pre-packaged Workforce Planning module from Oracle, but it will be configured to meet UBC's requirements for Salary and Benefit Planning. This Plan Type will provide reporting on various versions of the Plan and Forecast, and will provide Actual data for comparative analysis as well.

Database Structures

Following is a detailed description of the dimensions and structures in the Wrkforce Plan Type:

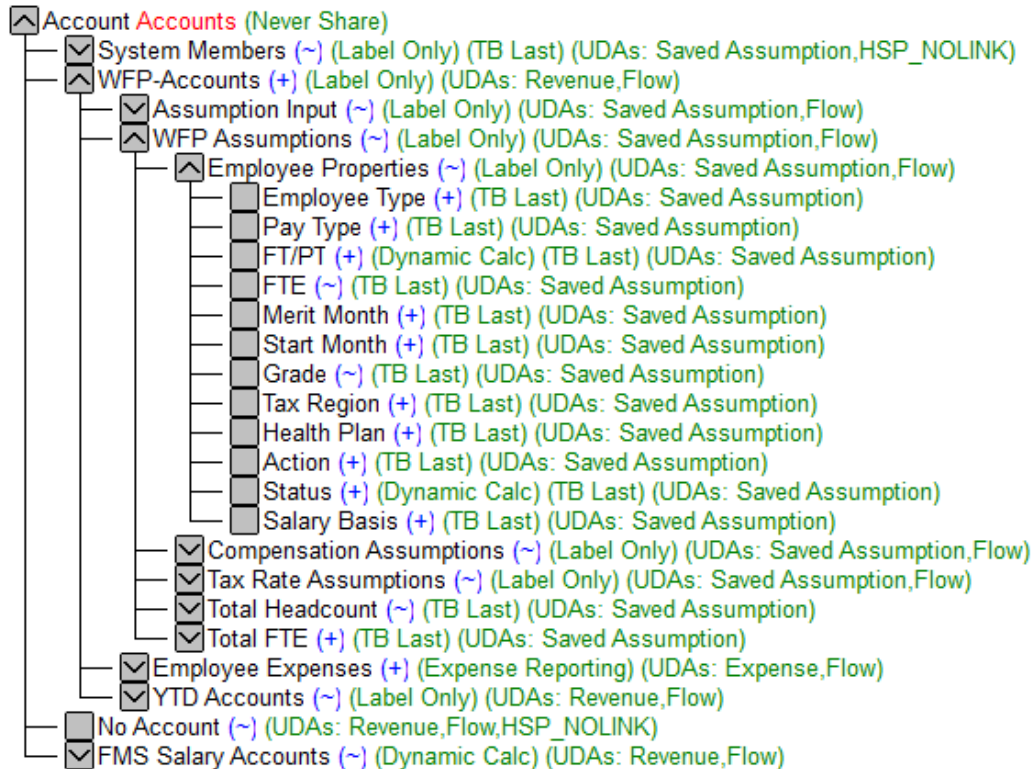
Account Dimension

The Account dimension, which will be tagged as the Accounts dimension (allowing for Expense Reporting and Time Balance tagging) will initially be built via the initialization of the Wrkforce Planning module. After initialization and deployment, the dimension will be modified to conform to UBC's specific Workforce Planning requirements. Security will be defined along the Account dimension.

Following are key attributes of the Account dimension within the Wrkforce Plan Type:

- Data Storage – Dense
- Size – Approximately 125 stored members
- Update Frequency – Updated infrequently

The screen shot below shows the expected structure of the Accounts dimension in this Plan Type:



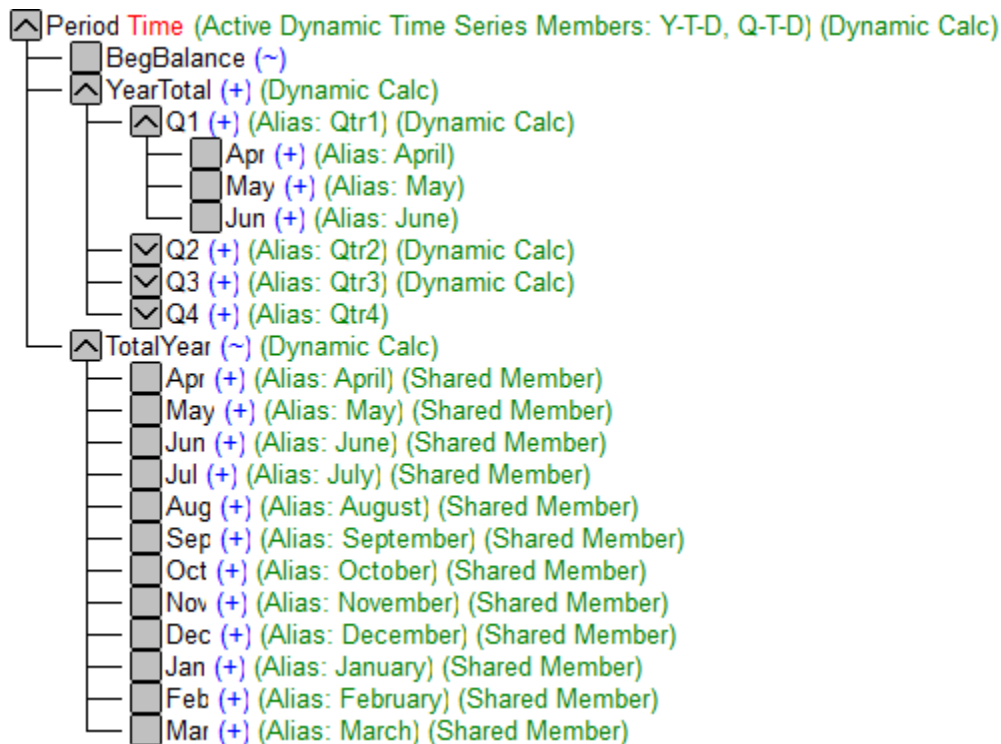
Period Dimension

The Period dimension, which will be tagged as the Time dimension (allowing for Y-T-D and Time Balance calculations) will consist of Months rolling up to Quarters, and Quarters rolling up to “YearTotal”. An additional hierarchy will be created with all 12 months rolling up directly to “TotalYear”. This additional shared hierarchy will allow for input into the “TotalYear” member on forms where the Quarters are not desired. All upper level members of this dimension will be Dynamically Calculated. This dimension will have the same structure in all Plan Types. Security is not defined along the Period dimension. Dynamic Times Series (DTS) functionality will be enabled to allow for Y-T-D and Q-T-D reporting. Security is not defined along the Period Dimension.

Following are key attributes of the Period dimension within the Wrkforce Plan Type:

- Data Storage – Dense
- Size – Approximately 13 stored members
- Update Frequency – Not Updated

The screen shot below represents the expected structure of the Period dimension in this Plan Type:



Fund Dimension

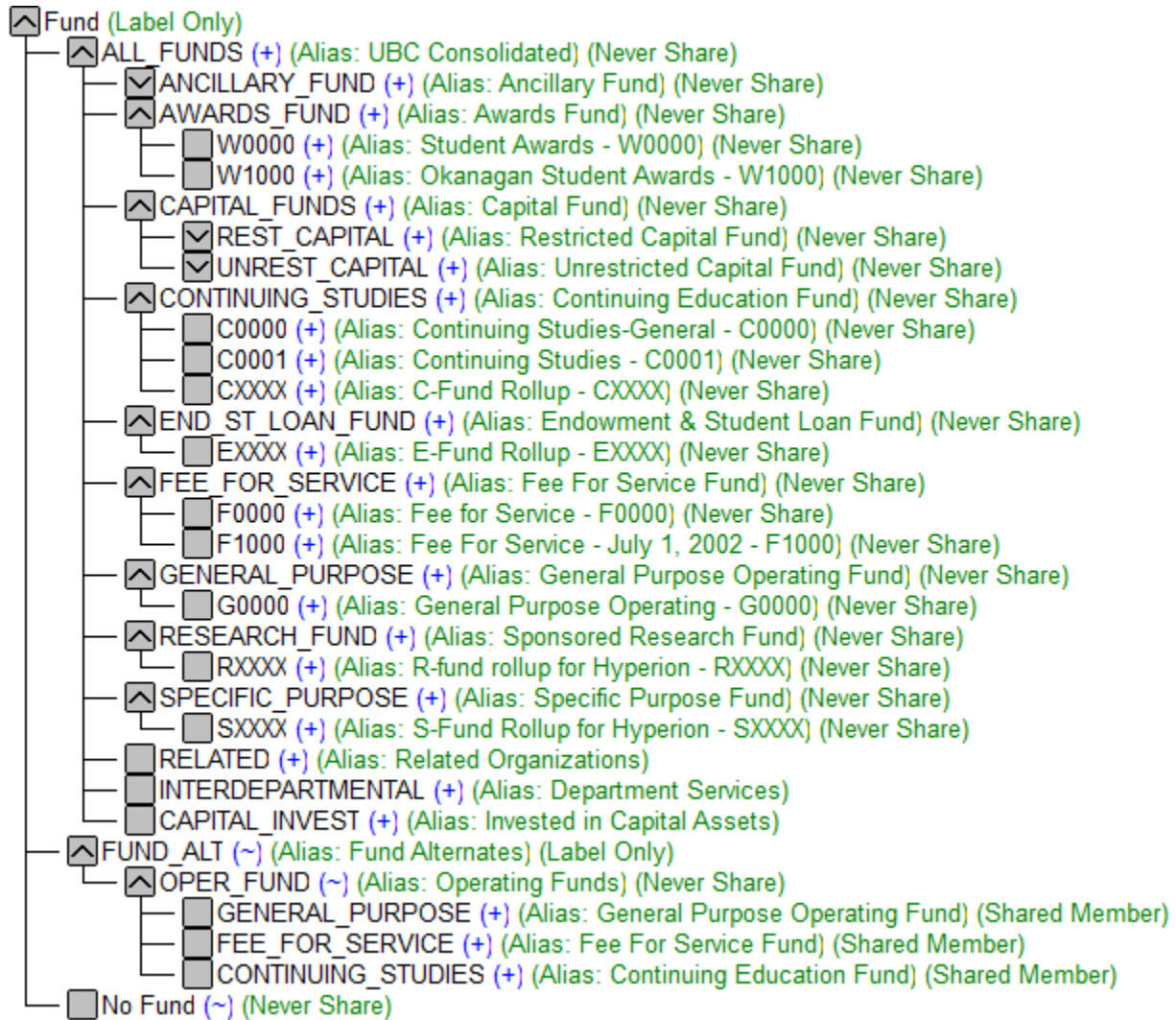
The Fund dimension will be based on the PeopleSoft Fund Codes tree, but will consist only of those fund codes or summarized representations of fund codes required for planning and forecasting purposes. Additionally, members representing Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets will also be included. The “ALL_FUNDS” segment of the Fund dimension will be based on the PeopleSoft Hyperion Fund Codes tree. The “No Fund” , Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets members will be built manually in EPMA.

Within the “ALL_FUNDS” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft, with no modifications. This consists of a 5 character string, with the first character representing the Fund Group. The alias will be based on the description from PeopleSoft, with the member name appended to the end of the alias. All members in this dimension, with the exception of the top level “Fund” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Fund Dimension.

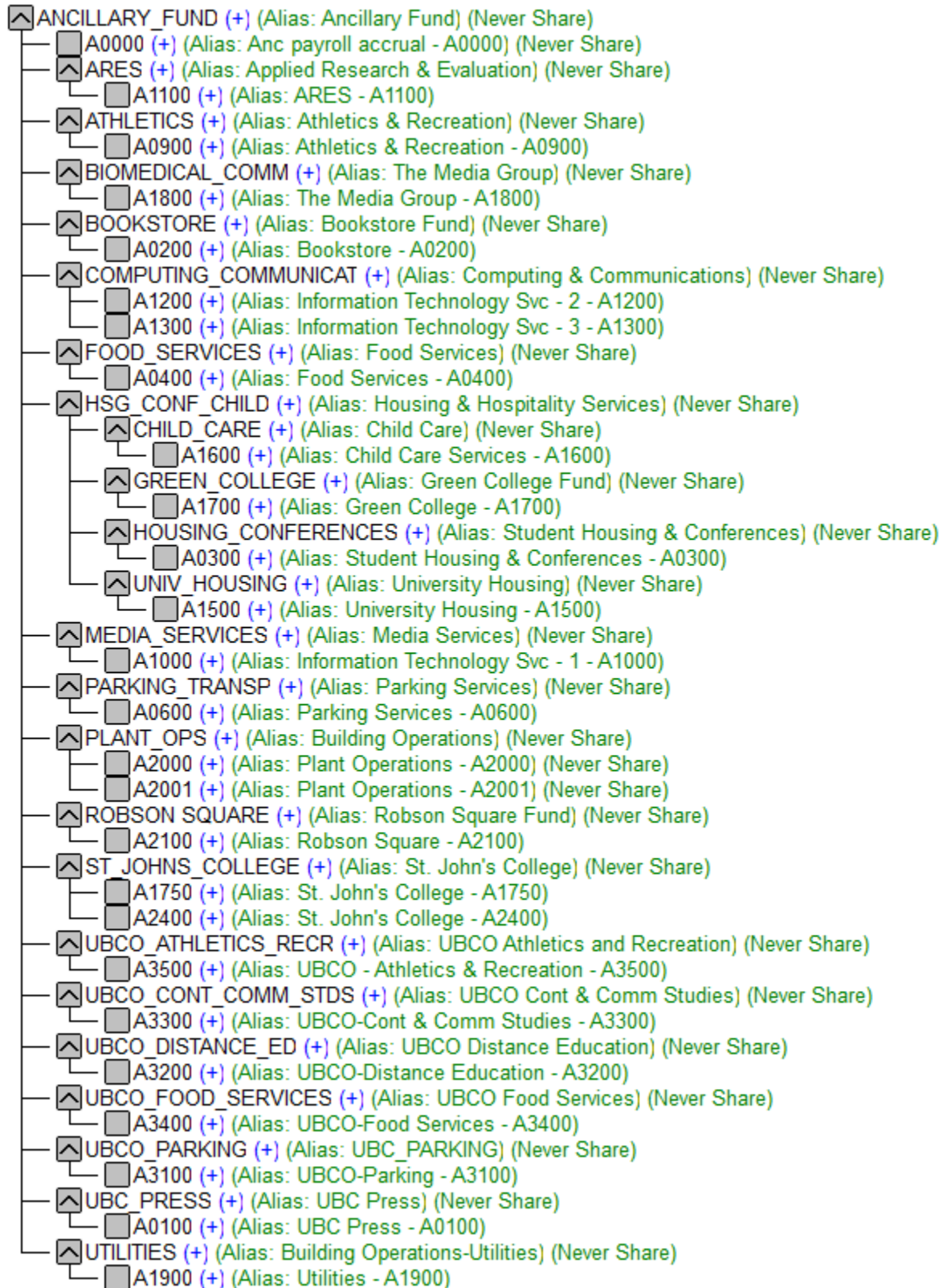
Following are key attributes of the Fund dimension within the Wrkforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 75 stored members
- Update Frequency – Updated infrequently, when new funds required for planning and forecasting purposes are added to the Hyperion Fund Codes tree in PeopleSoft

The screen shot below shows a high level representation of the expected structure of the Fund dimension in this Plan Type:



While most detailed funds will be summarized within the Fund dimension as shown above, all Ancillary Funds will be included, as shown in the screen shot below:



Department Dimension

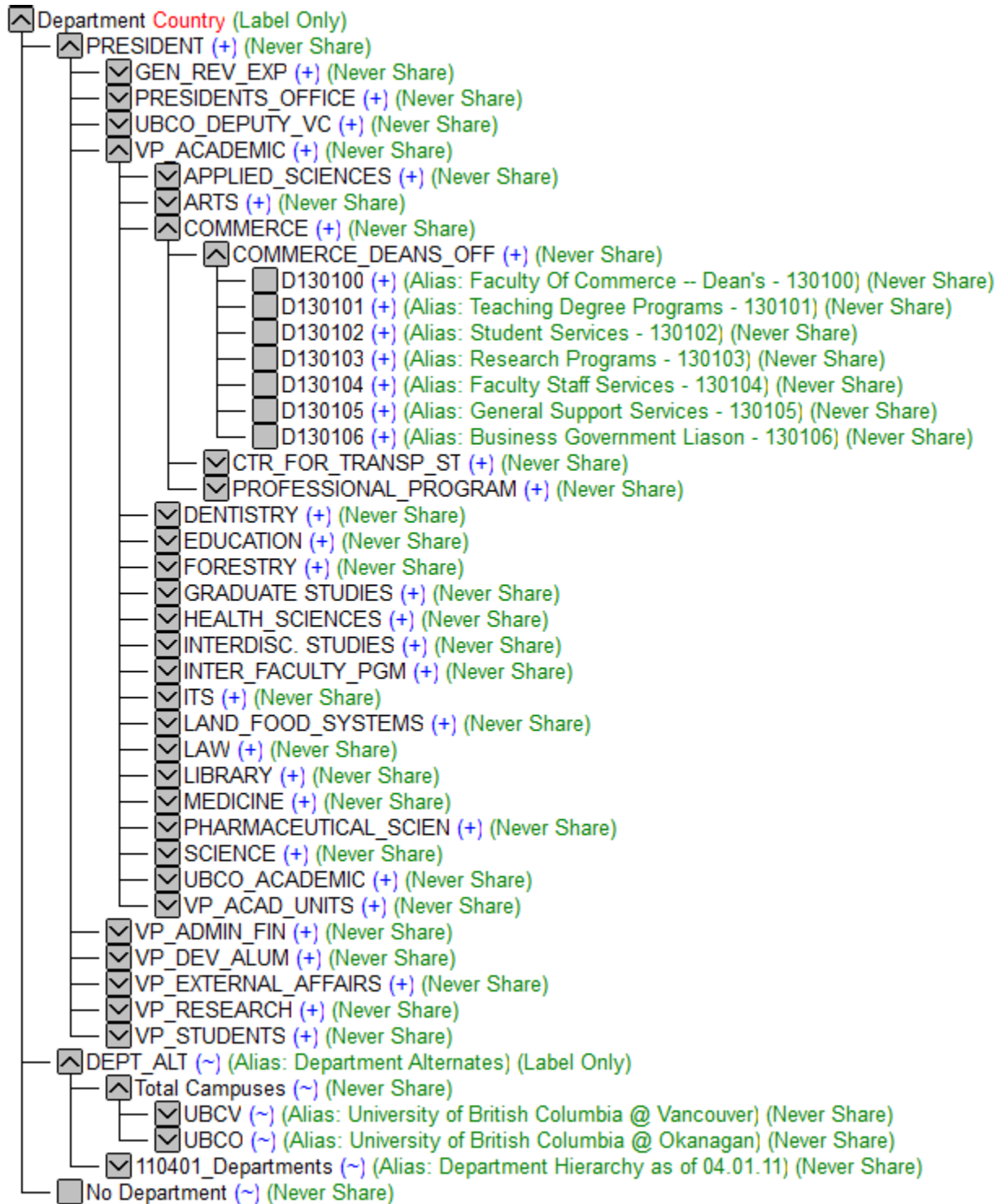
The Department dimension will be based on the PeopleSoft ALL_DEPTIDS tree, with alternate hierarchies for the ALL_DEPTIDS_VAN and ALL_DEPTIDS_UBCO. Additional alternate hierarchies could be developed to represent planned future changes to the University structure or for other reporting needs. The Department dimension will be the Entity dimension, so the hierarchy as defined will control the Workflow/Approval process. The “PRESIDENT”, “UBCV” and “UBCO” segments of the hierarchy will be based on PeopleSoft. The “No Department” member and any additional alternate hierarchies required will be built manually in EPMA.

Within the “PRESIDENT” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft (which consists of a 6 digit string), prepended with a “D”. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Department” member, and the “DEPT_ALT” member shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will be defined along the Department dimension.

Following are key attributes of the Department dimension within the Workforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 1,300 stored members
- Update Frequency – Updated infrequently, when new Departments are added in PeopleSoft or reorganizations of the Department structure occur

The screen shot below represents a high level view of the expected structure of the Department dimension in this Plan Type:



Program Code Dimension

The Program Code dimension will be based on the PeopleSoft Program Code table. The descendants of "ACTIVE_PROGRAMS", with the exception of "PC_DEFAULT", will be based on PeopleSoft. The "ALL_PROGRAMS", "PC_DEFAULT" and "No Program Code" members will be built manually in EPMA.

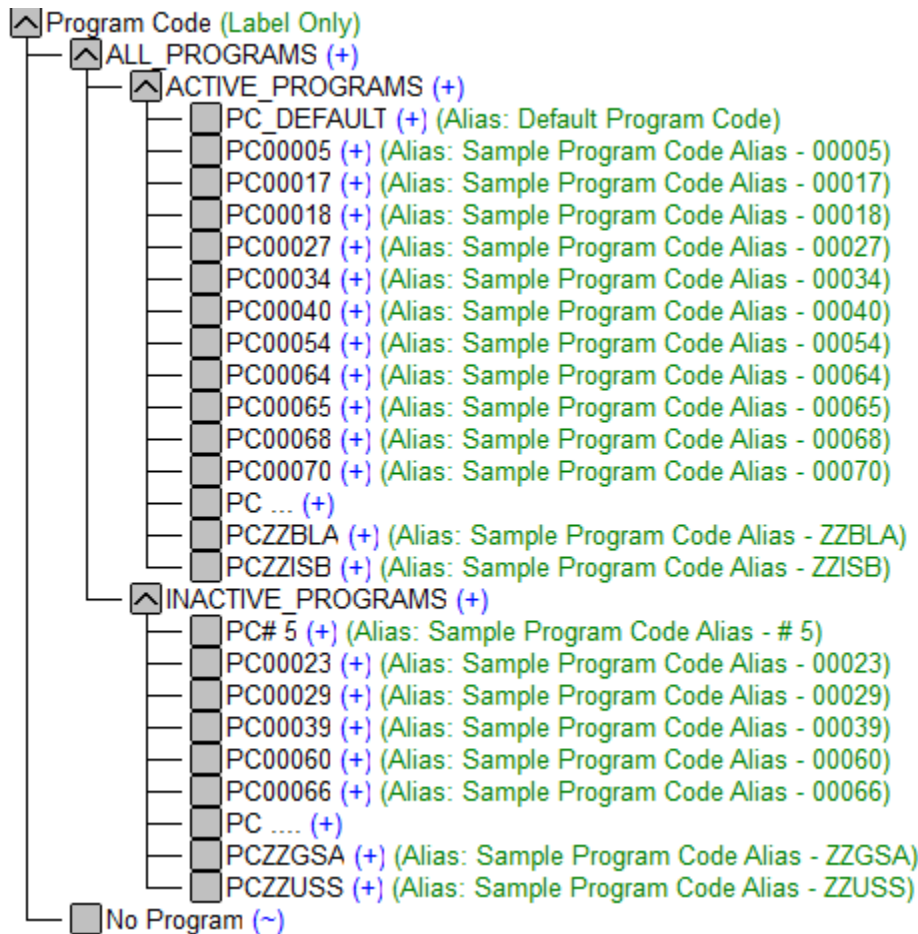
The member names for descendants of "ALL_PROGRAMS", all of which are Level 0 members, will be based on the member names from PeopleSoft, and prepended with a "PC". In general, the member name from PeopleSoft consists of a 5 character string, but there are many exceptions. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All members in this dimension shall be Stored. Security will be defined along the Program Code dimension. Planners will be granted Write access to the descendants of "ACTIVE_PROGRAMS" and Read Access to the descendants of "INACTIVE_PROGRAMS".

Aggregation performance considerations may cause a need to add additional hierarchy to this dimension. After a representative data set is loaded to the database, aggregation performance will be tested to determine whether additional hierarchy should be created.

Following are key attributes of the Program Code dimension within the ISPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 5,000 stored members
- Update Frequency – Updated frequently, as new Program Codes can be added to PeopleSoft at any time

The screen shot below represents a high level view of the expected structure of the Program Code dimension in this Plan Type:



Position Dimension

The Position dimension will be based on the Position table in the Position Management system. The children of “ALL_POSITIONS” will be based on the Department Parent assigned in the Position Management system. The “ALL_POSITIONS” and “No Position” members will be built manually in EPMA.

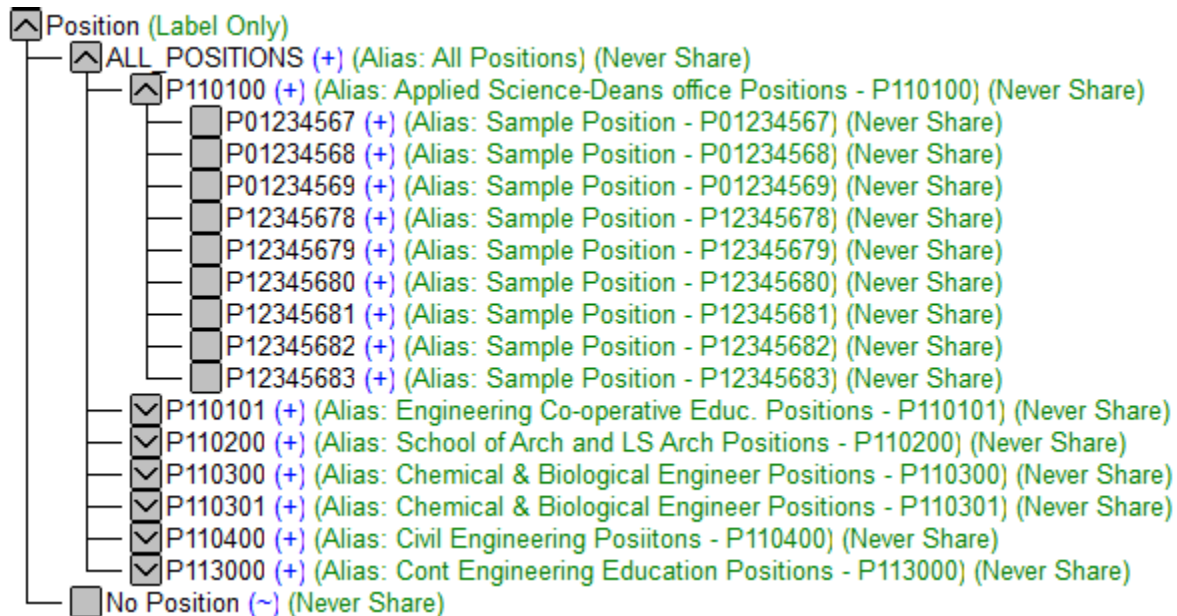
The member names for the children of “ALL_POSITIONS” that are based on the Department Parent will be based on the PeopleSoft member names prepended with “P”. The aliases for these members will be based on the PeopleSoft description of the Department, appended with “Positions” and the member name.

The Level 0 descendants of “ALL_POSITIONS” will be based on the member names from the Position Management system’s Position table, prepended with a “P”. The alias for these members will consist of the Position description from the Position Management system with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Position” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Position dimension.

Following are key attributes of the Position dimension within the Wrkforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 7,000 stored members
- Update Frequency – Updated monthly, as Position Management data will be refreshed monthly

The screen shot below represents a high level view of the expected structure of the Position dimension in this Plan Type:



Employee Dimension

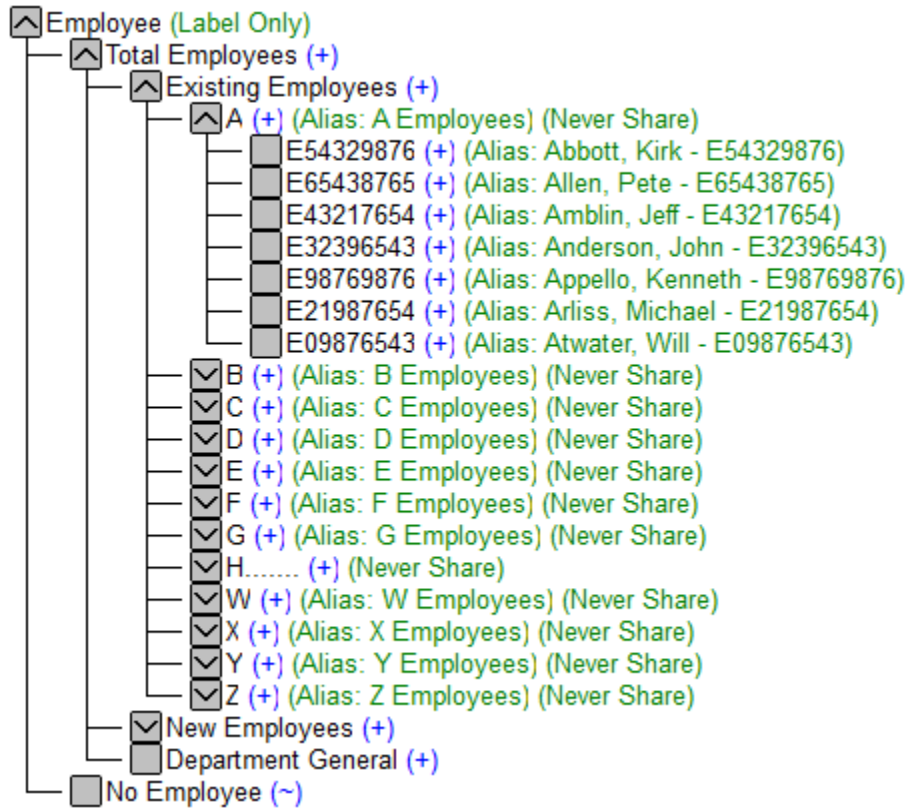
The Employee dimension will be based substantially on the Employee table in the Position Management system. The children of “Existing Employees” will be alphabetic parents for the Level 0 Employees.

The Level 0 descendants of “Total Employees” will be based on the Employee numbers from the Position Management system’s Employee table, prepended with a “E”. The alias for these members will consist of the Employee Name from the Position Management system with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Employee” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Employee dimension.

Following are key attributes of the Employee dimension within the Wrkforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 7,000 stored members
- Update Frequency – Updated monthly, as Position Management data will be refreshed monthly

The screen shot below represents a high level view of the expected structure of the Employee dimension in this Plan Type:



Project Grant Dimension

The Project Grant dimension will be based on the PeopleSoft Project Grant table. The children of “ALL_PGS”, with the exception of “PG_DEFAULT” and “PG_SHARED”, will be based on the DEPTID assigned in the PS_PROJECT_STATUS table in PeopleSoft. The “ALL_PGS”, “PG_DEFAULT”, “PG_SHARED” and “No Project Grant” members will be built manually in EPMA.

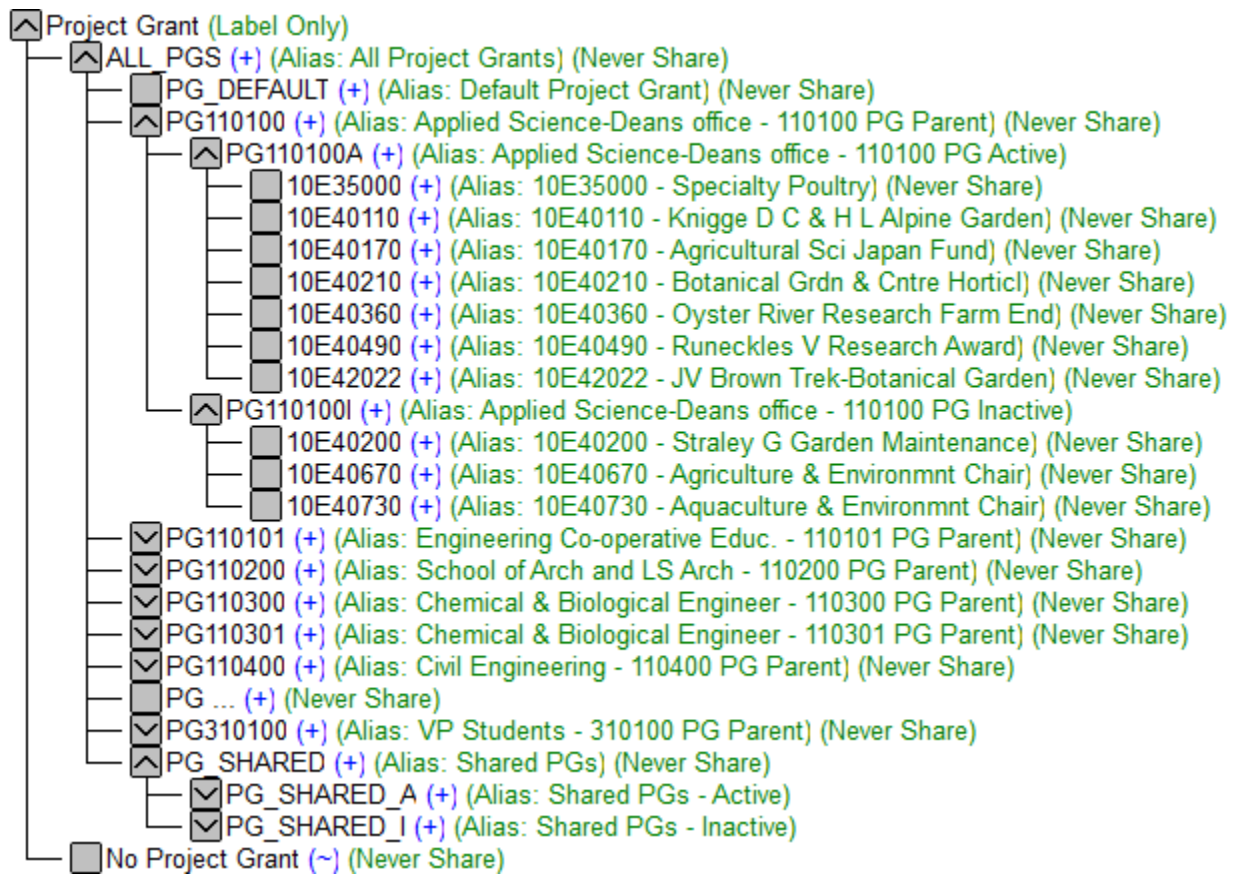
The member names for the children of “ALL_PGS” that are based on the DEPTID field will be based on the PeopleSoft member names prepended with “PG”. The aliases for these members will be based on the PeopleSoft description of the Department, appended with the member name and “PG Parent”. Under each of these DEPTID parents, an Active and Inactive parent will be created to segregate Active Project Grants from Inactive Project Grants. These levels of hierarchy exist in the Project Grant dimension to aid in the assignment of security. Security assignments within the dimension will be essential to limiting the number of PGs that users will see when budgeting, which will enhance the usability of the system.

The Level 0 descendants of “ALL_PGS” will be based on the member names from the PeopleSoft Project Grant table, with no modification. In general, the member name from PeopleSoft consists of an 8 character string, with the first two positions indicating the upper level Department with which the Project Grant is associated and the third position indicating the Fund Group with which the Project Grant is associated. The alias for these members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Project Grant” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. As noted above, security will be defined along the Project Grant dimension.

Following are key attributes of the Project Grant dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 25,000 stored members
- Update Frequency – Updated frequently, as new Project Grants are frequently added to PeopleSoft

The screen shot below represents a high level view of the expected structure of the Project Grant dimension in this Plan Type:



FiscalYear Dimension

As the word “Year” is reserved by Essbase for Dynamic Time Series functionality, the standard “Year” dimension name cannot be used if Dynamic Time Series will be used. Dynamic Time Series will be enabled in the application, so the dimension name will be changed to “FiscalYear”.

As noted in the Design Constraints section above, the application must be built with 2005/06 as the Start Year should the opportunity to load 5 years of historical data arise prior to go-live. The application will go-live for the 2011/12 plan, and the ability to create a 5 year forecast is required. Therefore, the End Year in the application at go-live will be 2015/16.

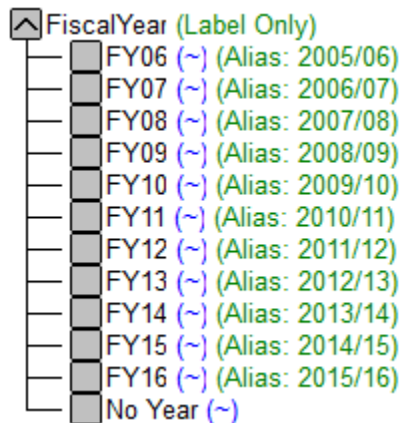
Planning does not offer flexibility in the member naming conventions within this dimension, so the FiscalYear dimension members will be named FY06, FY07, etc. Aliases will be used to incorporate UBC Fiscal Year naming standards.

The FiscalYear member will be set to Label Only, so when a user has not selected a member from the FiscalYear dimension, the data for the first stored member (FY06 in this case) will be displayed. The Level 0 members can be set to Never Share or Stored. Security is not applied along the FiscalYear dimension.

Following are key attributes of the FiscalYear dimension within the Wrkforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 12 stored members
- Update Frequency – Updated annually with an additional year

The screen shot below represents the expected structure of the FiscalYear dimension in this Plan Type:



Scenario Dimension

The Scenario dimension will be used to store the data related to different Planning cycles, store Actual data and calculate Variances.

The “Scenario” member shall be set to Label Only, so when a user has not selected a member from the Scenario dimension, the data for the first stored member (Forecast in this case) will be displayed. The other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

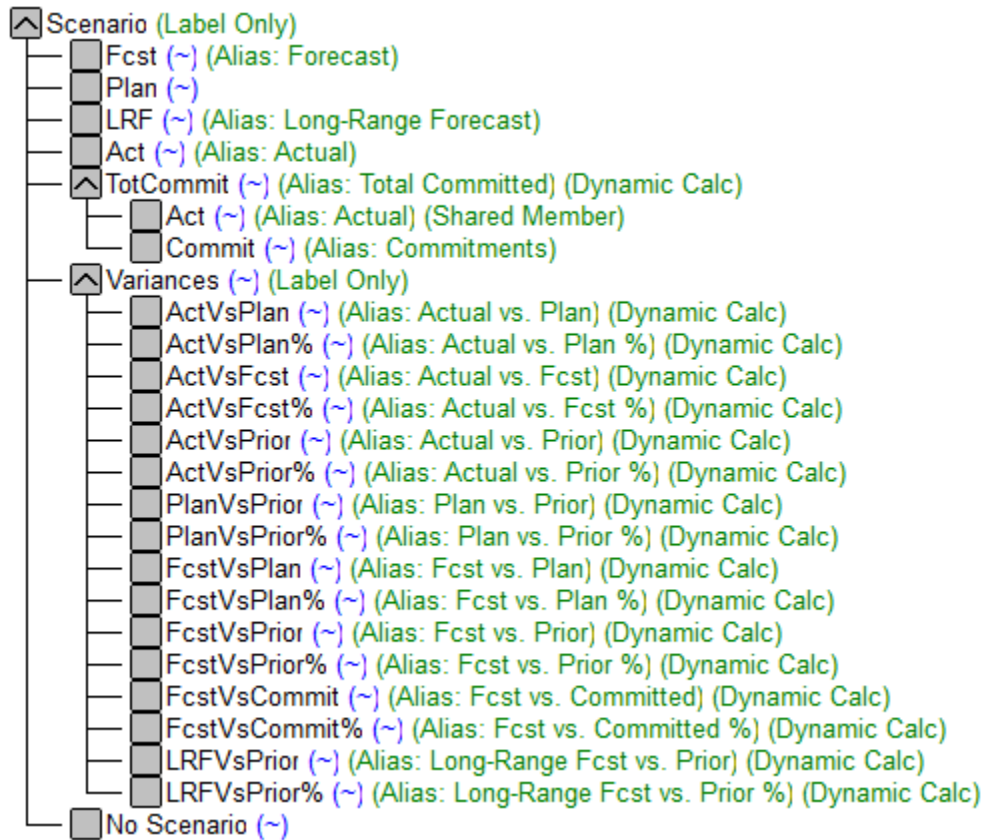
Following are key attributes of the Scenario dimension within the Wrkforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 5 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Scenario member that will be configured in Planning:

	Start Year	Start Mo.	End Year	End Mo.	Data Storage	Two-Pass	Incl. Beg. Bal.	Process Management
Forecast	FY10	Oct	FY10	Mar	Store	N/A	Yes	No
Plan	FY11	Apr	FY11	Mar	Store	N/A	Yes	Yes
Long-Range Forecast	FY11	Apr	FY15	Mar	Store	N/A	Yes	No
Actual	FY06	Apr	FY06	May	Store	N/A	Yes	No
Total Committed	FY06	Apr	FY06	May	Dynamic Calc	N/A	Yes	No
Commitments	FY06	Apr	FY06	May	Store	N/A	Yes	No
All Variances	FY06	Apr	FY06	May	Dynamic Calc	Yes	Yes	No
No Scenario	FY06	Apr	FY06	May	Store	N/A	Yes	No

The screen shot below represents the expected structure of the Scenario dimension in this Plan Type:



Version Dimension

The Version dimension will be used to store different versions of plans and forecasts. The main Version for user input will be the Working Version. This is the only Version that will be enabled for Process Management, and users will be granted Write access to this member. The Approved Version will be used to store the Actual data and the final Approved Plan. Draft1 and Draft2 will be used to store different Plan or Forecast iterations (e.g. “What-If” iterations) that are created by the users. This Version may also be used to load Balance Sheet Actual activity before the system calculates balances for storage in the Approved Version. Variance Versions may be added to allow comparisons between Versions should the need for these comparisons arise.

The “Version” member shall be set to Label Only, and the other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

Following are key attributes of the Version dimension within the Wrkforce Plan Type:

- Data Storage – Sparse
- Size – Approximately 6 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Version member that will be configured in Planning:

Member	Type	Data Storage	Two-Pass	Process Management
Approved	Standard Bottom Up	Store	N/A	No
Working	Standard Bottom Up	Store	N/A	Yes
Draft1	Standard Bottom Up	Store	N/A	No
Draft2	Standard Bottom Up	Store	N/A	No
No Version	Standard Bottom Up	Store	N/A	No

The screen shot below represents the expected structure of the Version dimension in this Plan Type:



Data Loads

Following is a description of the data load processes that will feed data to the Wrkforce Plan Type:

Position Management Data

Position Management Data will be loaded to the Wrkforce Plan Type. This data will be loaded to the Plan Type on a monthly basis. All data will be loaded to Level 0 dimension members in the Plan Type.

Actual Payroll Data from HRMS

Actual Payroll Data from HRMS will be loaded to the Wrkforce Plan Type. This data will be loaded to the Plan Type on a monthly basis. All data will be loaded to Level 0 dimension members in the Plan Type.

Planning Input Forms

The Planning Input Forms in the Wrkforce Plan Type will be created via the initialization of the Workforce Planning module. These forms will be modified to conform to the requirements of UBC. The currently defined form(s) include:

- Manage Existing Employee Status
- Manage Existing Employees
- New Hires
- Compensation Adjustments
- Employee Expenses Summary
- Headcount and FTE
- Employee Information
- Begin vs End
- Forecast vs Plan
- Department Level Expenses
- Department Level Headcount
- Benefit Rates

Calculations

The Calculations required for the Wrkforce Plan Type will initially be created via the initialization of the Workforce Planning module, and will be configured to conform to UBC's requirements. The currently defined calculations include:

- Add TBH Hourly
- Add TBH Salary
- Change Employee Status
- CopyProps
- Plan Departure
- Reconcile TBH
- Remove TBH
- Rollup

FundPlan Design

Overview

The FundPlan Plan Type will be used to calculate the GPO Funding Model.

Database Structures

Following is a detailed description of the dimensions and structures in the FundPlan Plan Type:

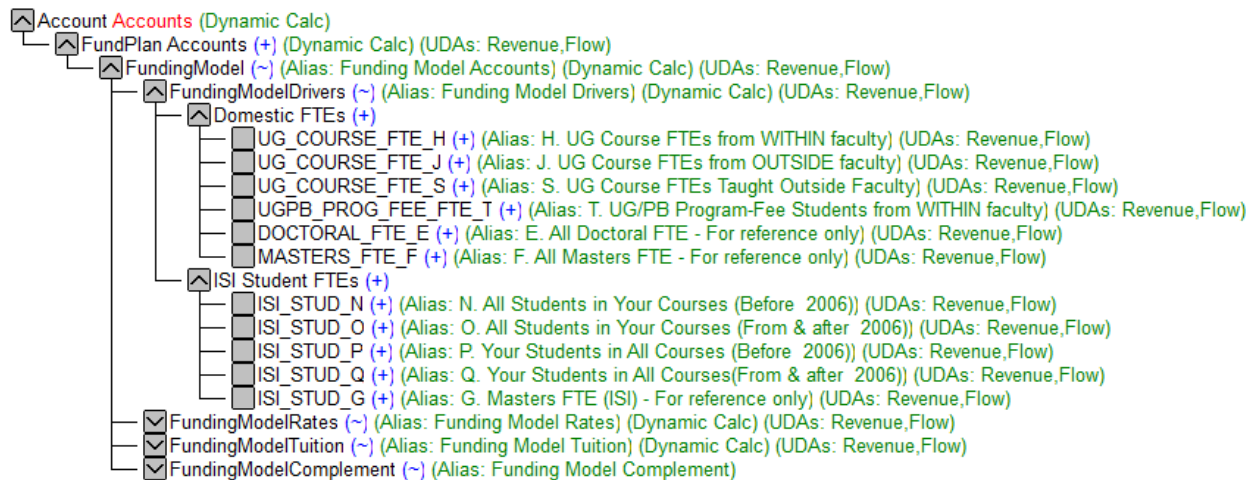
Account Dimension

The Account dimension, which will be tagged as the Accounts dimension (allowing for Expense Reporting and Time Balance tagging) will be built manually with the drivers, rates and calculated values needed for the GPO Funding model. Security will be defined along the Account dimension.

Following are key attributes of the Account dimension within the FundPlan Plan Type:

- Data Storage – Dense
- Size – Approximately 50 stored members
- Update Frequency – Updated infrequently

The screen shot below shows the expected structure of the Accounts dimension in this Plan Type:



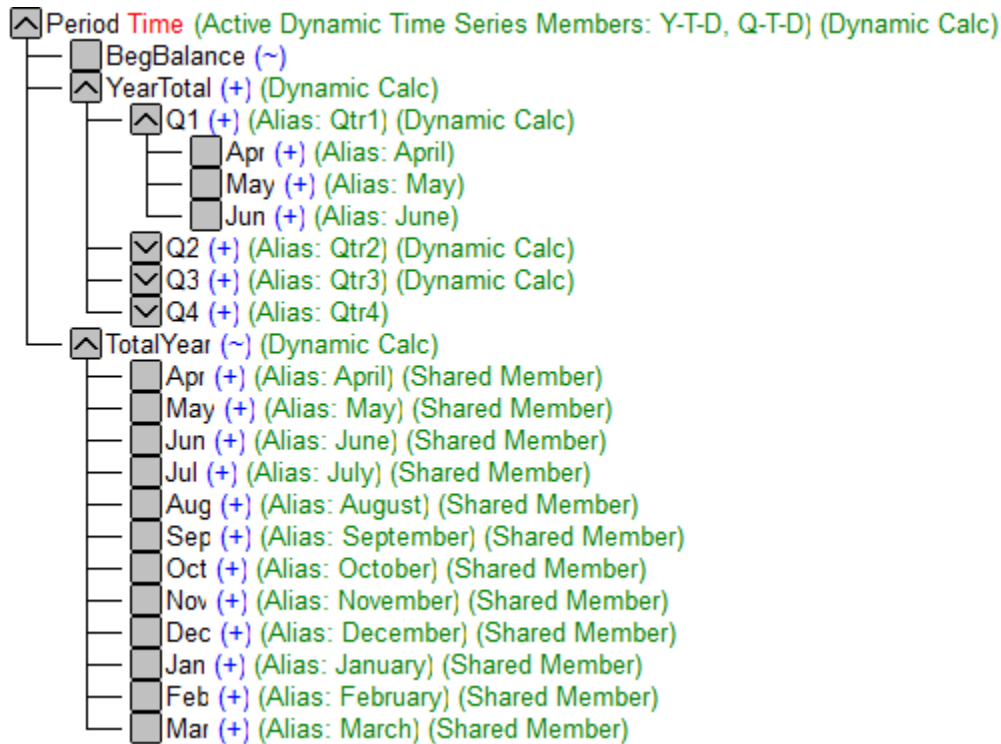
Period Dimension

The Period dimension, which will be tagged as the Time dimension (allowing for Y-T-D and Time Balance calculations) will consist of Months rolling up to Quarters, and Quarters rolling up to “YearTotal”. An additional hierarchy will be created with all 12 months rolling up directly to “TotalYear”. This additional shared hierarchy will allow for input into the “TotalYear” member on forms where the Quarters are not desired. All upper level members of this dimension will be Dynamically Calculated. This dimension will have the same structure in all Plan Types. Security is not defined along the Period dimension. Dynamic Times Series (DTS) functionality will be enabled to allow for Y-T-D and Q-T-D reporting. Security is not defined along the Period Dimension.

Following are key attributes of the Period dimension within the FundPlan Plan Type:

- Data Storage – Dense
- Size – Approximately 13 stored members
- Update Frequency – Not Updated

The screen shot below represents the expected structure of the Period dimension in this Plan Type:



Fund Dimension

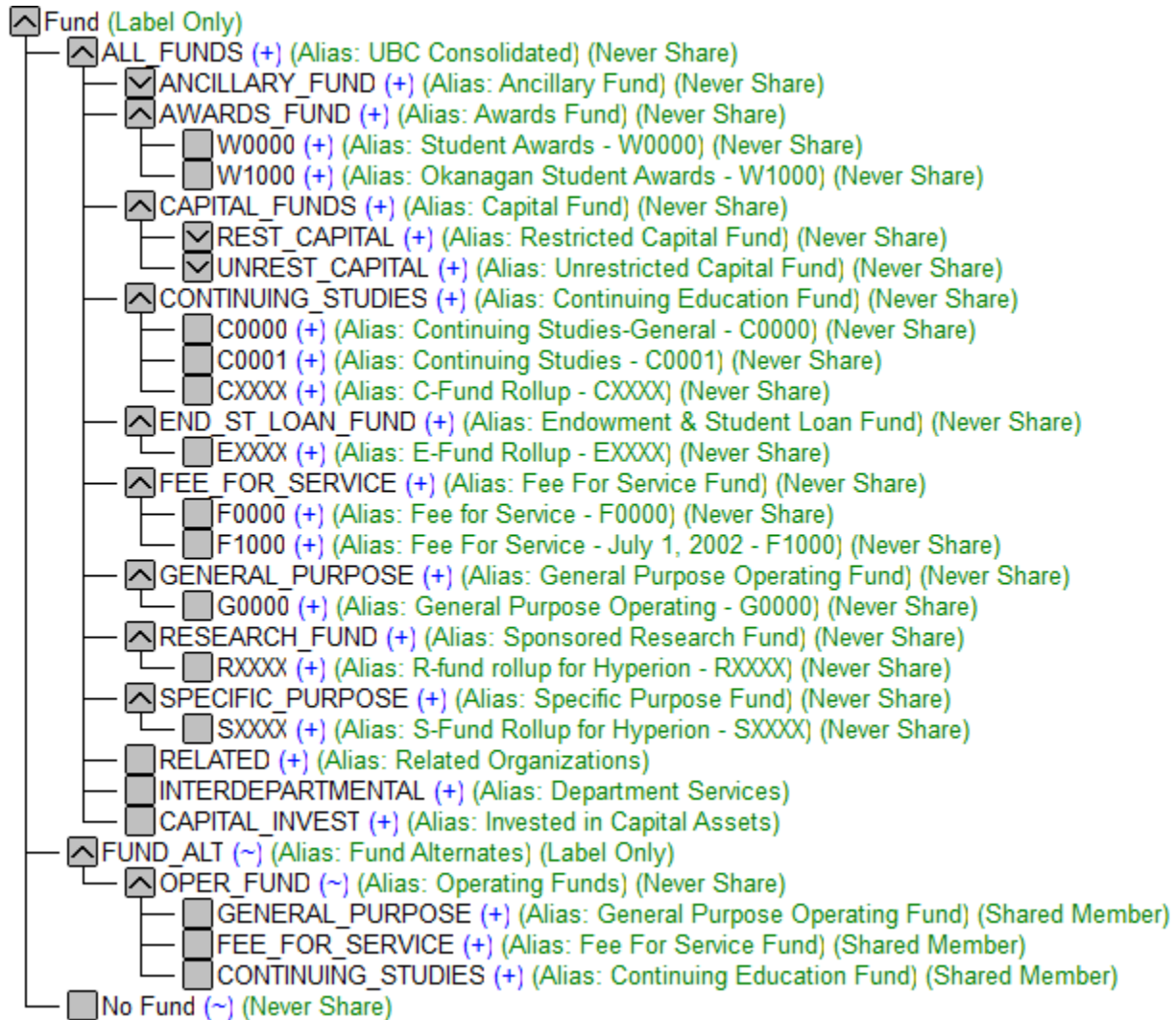
The Fund dimension will be based on the PeopleSoft Fund Codes tree, but will consist only of those fund codes or summarized representations of fund codes required for planning and forecasting purposes. Additionally, members representing Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets will also be included. The “ALL_FUNDS” segment of the Fund dimension will be based on the PeopleSoft Hyperion Fund Codes tree. The “No Fund” , Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets members will be built manually in EPMA.

Within the “ALL_FUNDS” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft, with no modifications. This consists of a 5 character string, with the first character representing the Fund Group. The alias will be based on the description from PeopleSoft, with the member name appended to the end of the alias. All members in this dimension, with the exception of the top level “Fund” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Fund Dimension.

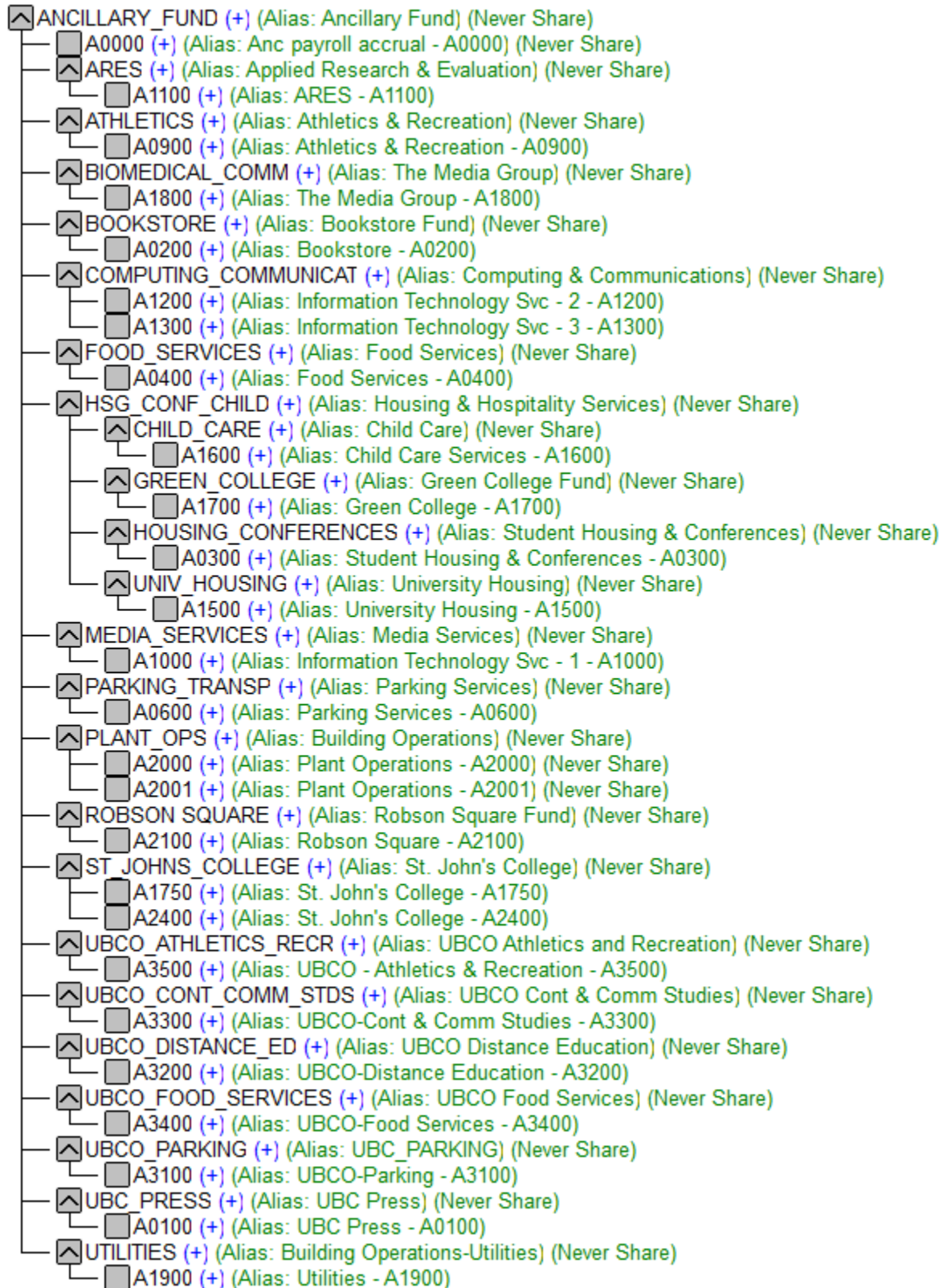
Following are key attributes of the Fund dimension within the FundPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 75 stored members
- Update Frequency – Updated infrequently, when new funds required for planning and forecasting purposes are added to the Hyperion Fund Codes tree in PeopleSoft

The screen shot below shows a high level representation of the expected structure of the Fund dimension in this Plan Type:



While most detailed funds will be summarized within the Fund dimension as shown above, all Ancillary Funds will be included, as shown in the screen shot below:



Department Dimension

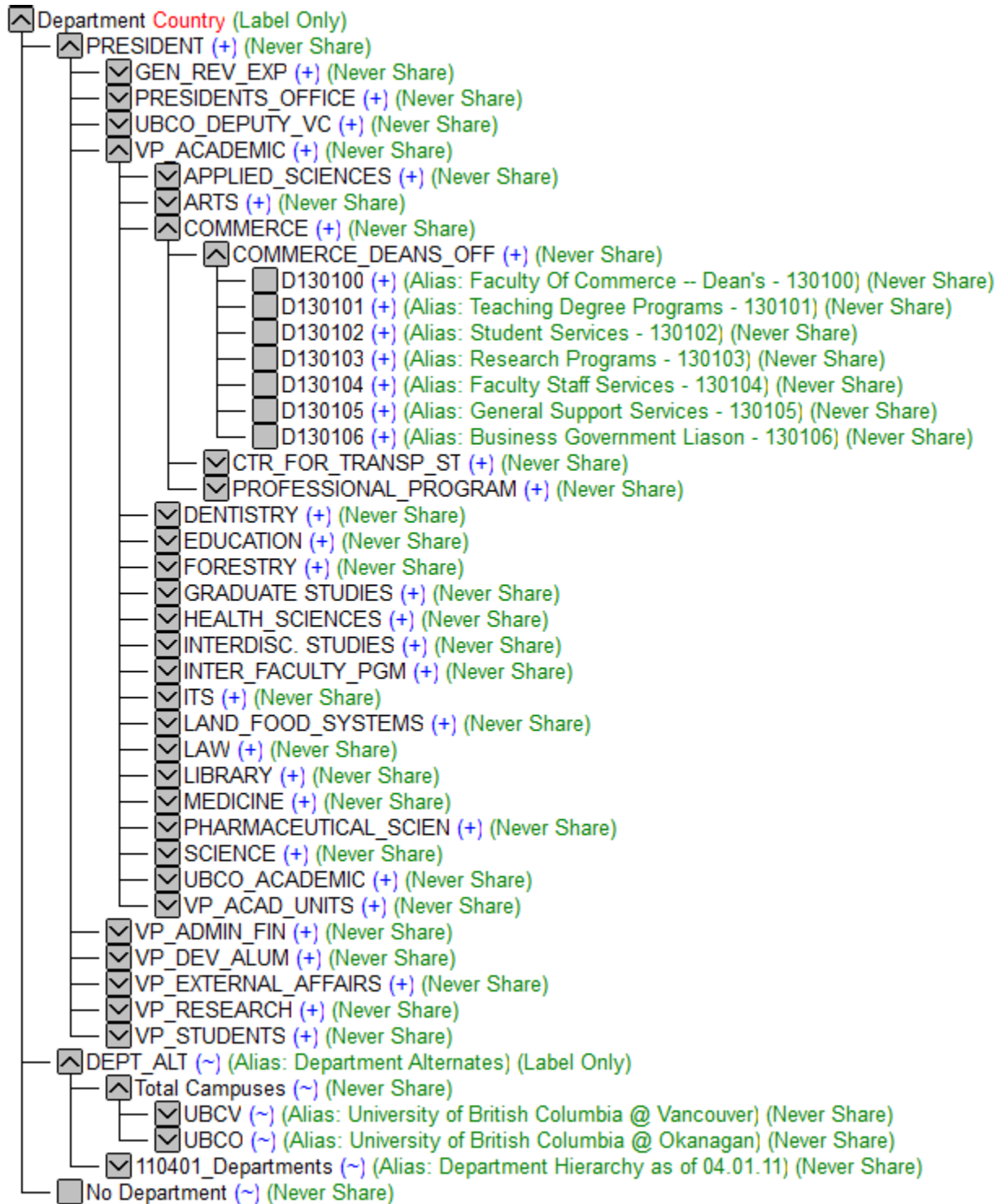
The Department dimension will be based on the PeopleSoft ALL_DEPTIDS tree, with alternate hierarchies for the ALL_DEPTIDS_VAN and ALL_DEPTIDS_UBCO. Additional alternate hierarchies could be developed to represent planned future changes to the University structure or for other reporting needs. The Department dimension will be the Entity dimension, so the hierarchy as defined will control the Workflow/Approval process. The “PRESIDENT”, “UBCV” and “UBCO” segments of the hierarchy will be based on PeopleSoft. The “No Department” member and any additional alternate hierarchies required will be built manually in EPMA.

Within the “PRESIDENT” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft (which consists of a 6 digit string), prepended with a “D”. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Department” member, and the “DEPT_ALT” member shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will be defined along the Department dimension.

Following are key attributes of the Department dimension within the FundPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 1,300 stored members
- Update Frequency – Updated infrequently, when new Departments are added in PeopleSoft or reorganizations of the Department structure occur

The screen shot below represents a high level view of the expected structure of the Department dimension in this Plan Type:



FiscalYear Dimension

As the word “Year” is reserved by Essbase for Dynamic Time Series functionality, the standard “Year” dimension name cannot be used if Dynamic Time Series will be used. Dynamic Time Series will be enabled in the application, so the dimension name will be changed to “FiscalYear”.

As noted in the Design Constraints section above, the application must be built with 2005/06 as the Start Year should the opportunity to load 5 years of historical data arise prior to go-live. The application will go-live for the 2011/12 plan, and the ability to create a 5 year forecast is required. Therefore, the End Year in the application at go-live will be 2015/16.

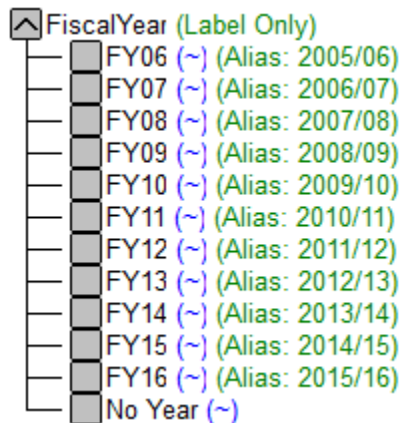
Planning does not offer flexibility in the member naming conventions within this dimension, so the FiscalYear dimension members will be named FY06, FY07, etc. Aliases will be used to incorporate UBC Fiscal Year naming standards.

The FiscalYear member will be set to Label Only, so when a user has not selected a member from the FiscalYear dimension, the data for the first stored member (FY06 in this case) will be displayed. The Level 0 members can be set to Never Share or Stored. Security is not applied along the FiscalYear dimension.

Following are key attributes of the FiscalYear dimension within the FundPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 12 stored members
- Update Frequency – Updated annually with an additional year

The screen shot below represents the expected structure of the FiscalYear dimension in this Plan Type:



Scenario Dimension

The Scenario dimension will be used to store the data related to different Planning cycles, store Actual data and calculate Variances.

The “Scenario” member shall be set to Label Only, so when a user has not selected a member from the Scenario dimension, the data for the first stored member (Forecast in this case) will be displayed. The other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

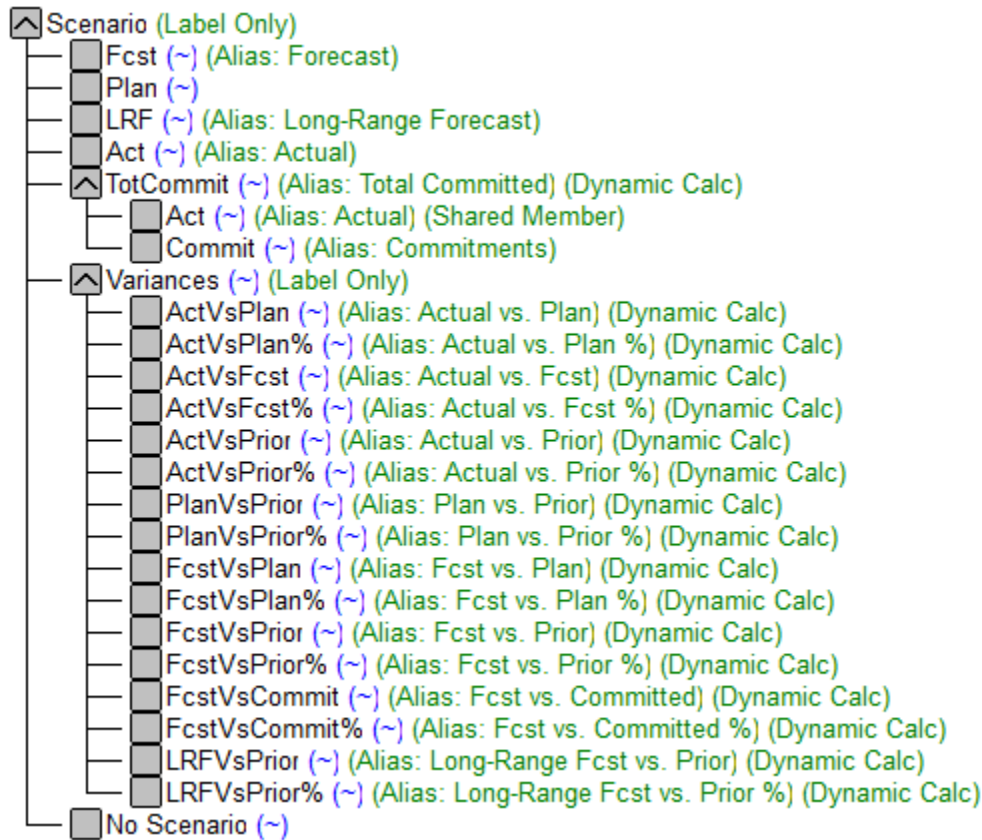
Following are key attributes of the Scenario dimension within the FundPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 5 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Scenario member that will be configured in Planning:

	Start Year	Start Mo.	End Year	End Mo.	Data Storage	Two-Pass	Incl. Beg. Bal.	Process Management
Forecast	FY10	Oct	FY10	Mar	Store	N/A	Yes	No
Plan	FY11	Apr	FY11	Mar	Store	N/A	Yes	Yes
Long-Range Forecast	FY11	Apr	FY15	Mar	Store	N/A	Yes	No
Actual	FY06	Apr	FY06	May	Store	N/A	Yes	No
Total Committed	FY06	Apr	FY06	May	Dynamic Calc	N/A	Yes	No
Commitments	FY06	Apr	FY06	May	Store	N/A	Yes	No
All Variances	FY06	Apr	FY06	May	Dynamic Calc	Yes	Yes	No
No Scenario	FY06	Apr	FY06	May	Store	N/A	Yes	No

The screen shot below represents the expected structure of the Scenario dimension in this Plan Type:



Version Dimension

The Version dimension will be used to store different versions of plans and forecasts. The main Version for user input will be the Working Version. This is the only Version that will be enabled for Process Management, and users will be granted Write access to this member. The Approved Version will be used to store the Actual data and the final Approved Plan. Draft1 and Draft2 will be used to store different Plan or Forecast iterations (e.g. “What-If” iterations) that are created by the users. This Version may also be used to load Balance Sheet Actual activity before the system calculates balances for storage in the Approved Version. Variance Versions may be added to allow comparisons between Versions should the need for these comparisons arise.

The “Version” member shall be set to Label Only, and the other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

Following are key attributes of the Version dimension within the FundPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 6 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Version member that will be configured in Planning:

Member	Type	Data Storage	Two-Pass	Process Management
Approved	Standard Bottom Up	Store	N/A	No
Working	Standard Bottom Up	Store	N/A	Yes
Draft1	Standard Bottom Up	Store	N/A	No
Draft2	Standard Bottom Up	Store	N/A	No
No Version	Standard Bottom Up	Store	N/A	No

The screen shot below represents the expected structure of the Version dimension in this Plan Type:



Data Loads

Following is a description of the data load processes that will feed data to the FundPlan Plan Type:

Funding Driver Data from SIS

Funding driver data will be loaded to the FundPlan Plan Type. This data will be loaded to the Plan Type on demand as needed. All data will be loaded to Level 0 dimension members in the Plan Type.

Planning Input Forms

The Planning Input Forms in the FundPlan Plan Type will be used to input and review driver values and rates to recalculate the Funding Model. The currently defined form(s) include:

- GPO Funding Model

Calculations

The Calculations required for the FundPlan Plan Type are those needed to calculate the GPO Funding Model, and the aggregate the values in the database. The currently defined calculations include:

- GPO Funding Model Calculation
- User Aggregation

FundTxr Design

Overview

The FundTxr Plan Type will be used to allocate funding and to transfer funding between Departments. This Plan Type will provide reporting on various versions of the Plan and Forecast, and will provide Actual data for comparative analysis as well.

Database Structures

Following is a detailed description of the dimensions and structures in the FundTxr Plan Type:

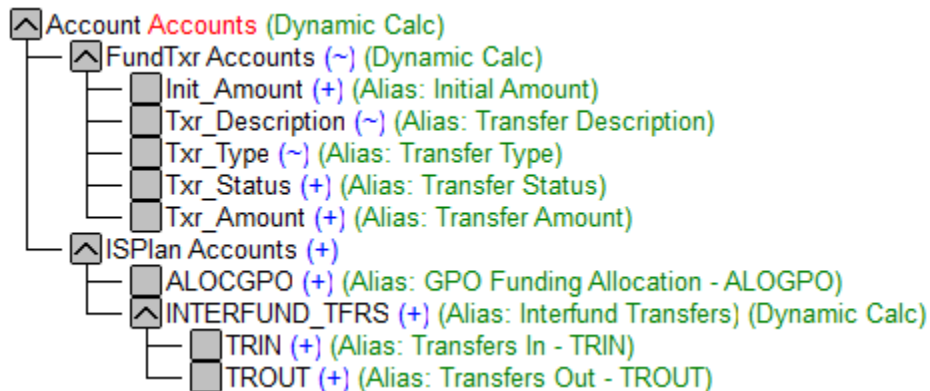
Account Dimension

The Account dimension, which will be tagged as the Accounts dimension (allowing for Expense Reporting and Time Balance tagging) will be built to allow for the assignment of descriptions, statuses and values to allocations and transfers. The accounts that will be impacted in the ISPlan Plan Type will also be included. Security will be defined along the Account dimension.

Following are key attributes of the Account dimension within the FundTxr Plan Type:

- Data Storage – Dense
- Size – Approximately 10 stored members
- Update Frequency – Updated infrequently

The screen shot below shows the expected structure of the Accounts dimension in this Plan Type:



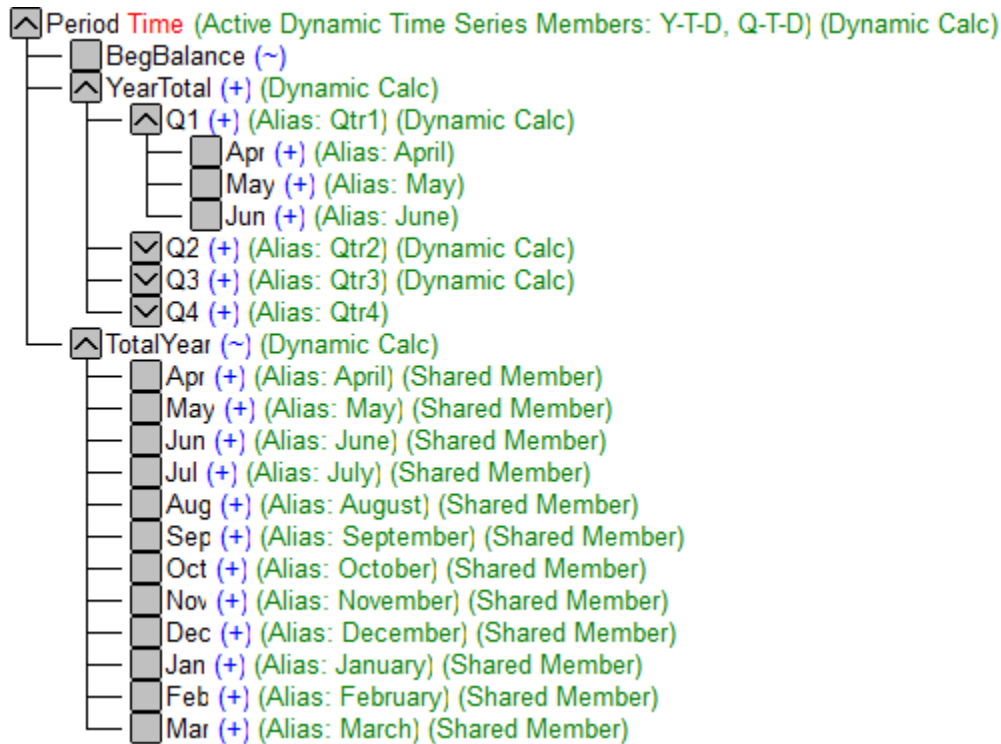
Period Dimension

The Period dimension, which will be tagged as the Time dimension (allowing for Y-T-D and Time Balance calculations) will consist of Months rolling up to Quarters, and Quarters rolling up to “YearTotal”. An additional hierarchy will be created with all 12 months rolling up directly to “TotalYear”. This additional shared hierarchy will allow for input into the “TotalYear” member on forms where the Quarters are not desired. All upper level members of this dimension will be Dynamically Calculated. This dimension will have the same structure in all Plan Types. Security is not defined along the Period dimension. Dynamic Times Series (DTS) functionality will be enabled to allow for Y-T-D and Q-T-D reporting. Security is not defined along the Period Dimension.

Following are key attributes of the Period dimension within the FundTxr Plan Type:

- Data Storage – Dense
- Size – Approximately 13 stored members
- Update Frequency – Not Updated

The screen shot below represents the expected structure of the Period dimension in this Plan Type:



Fund Dimension

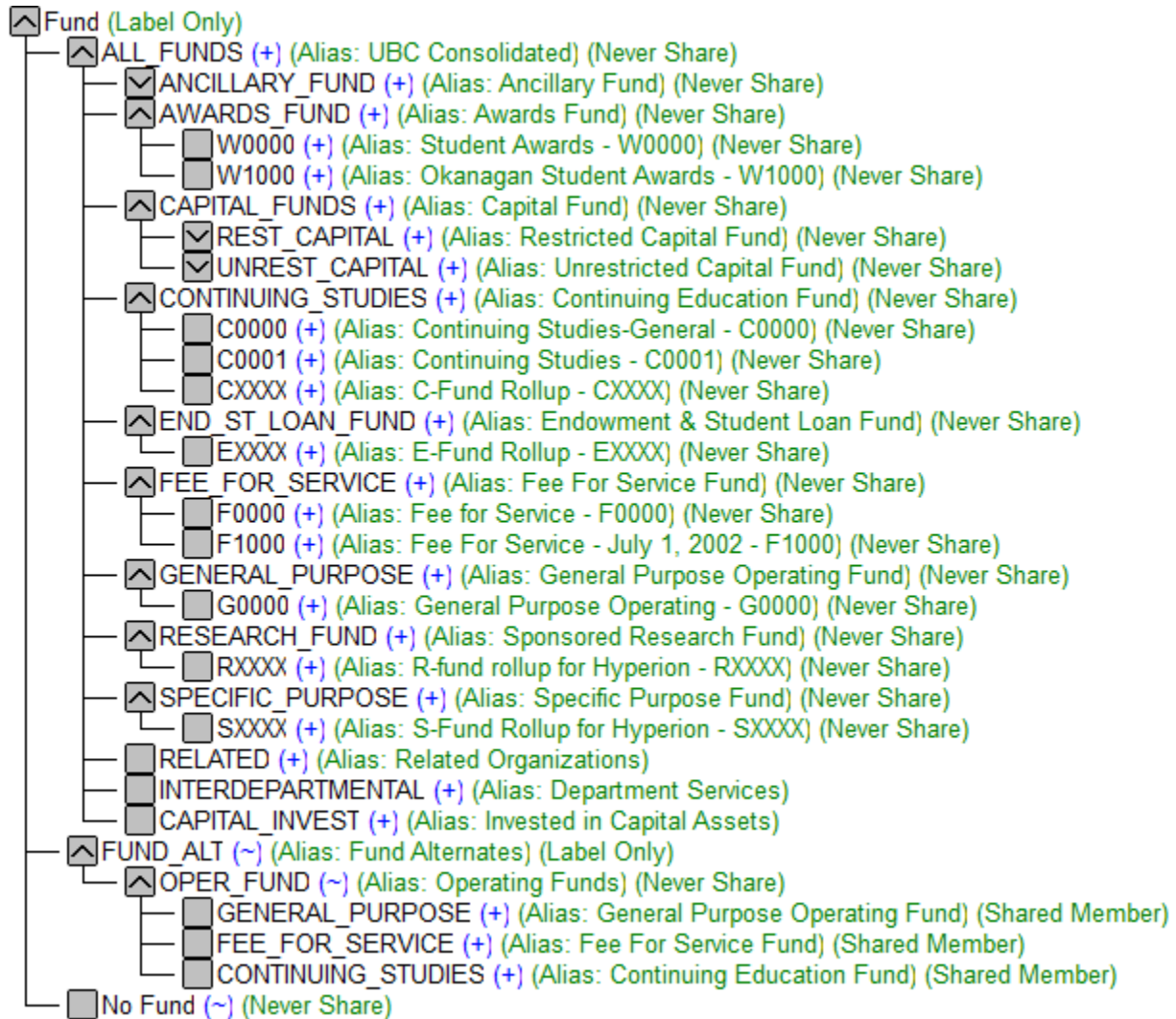
The Fund dimension will be based on the PeopleSoft Fund Codes tree, but will consist only of those fund codes or summarized representations of fund codes required for planning and forecasting purposes. Additionally, members representing Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets will also be included. The “ALL_FUNDS” segment of the Fund dimension will be based on the PeopleSoft Hyperion Fund Codes tree. The “No Fund”, Related Organizations, Interdepartmental Eliminations and Invested in Capital Assets members will be built manually in EPMA.

Within the “ALL_FUNDS” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft, with no modifications. This consists of a 5 character string, with the first character representing the Fund Group. The alias will be based on the description from PeopleSoft, with the member name appended to the end of the alias. All members in this dimension, with the exception of the top level “Fund” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Fund Dimension.

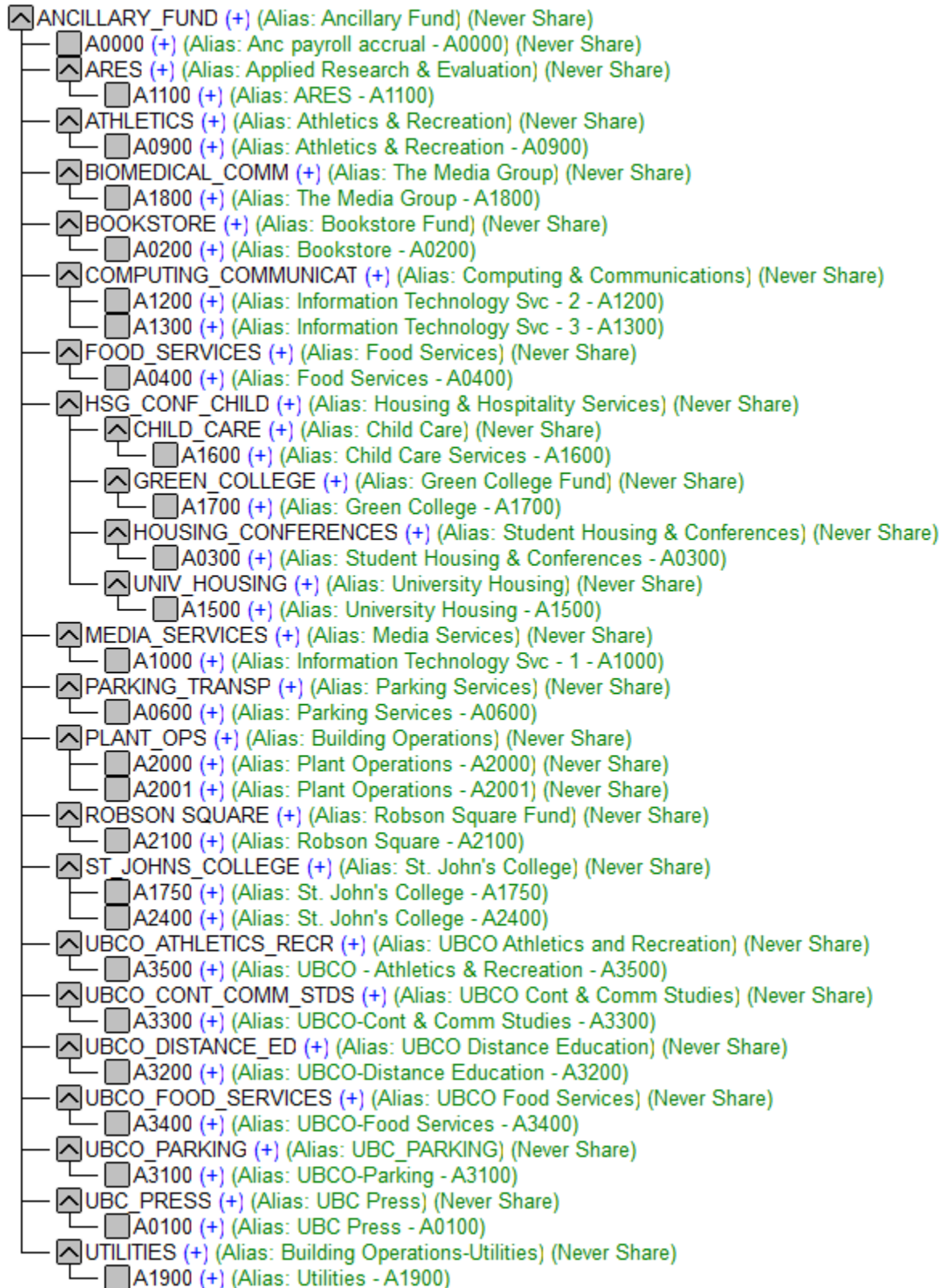
Following are key attributes of the Fund dimension within the FundTxr Plan Type:

- Data Storage – Sparse
- Size – Approximately 75 stored members
- Update Frequency – Updated infrequently, when new funds required for planning and forecasting purposes are added to the Hyperion Fund Codes tree in PeopleSoft

The screen shot below shows a high level representation of the expected structure of the Fund dimension in this Plan Type:



While most detailed funds will be summarized within the Fund dimension as shown above, all Ancillary Funds will be included, as shown in the screen shot below:



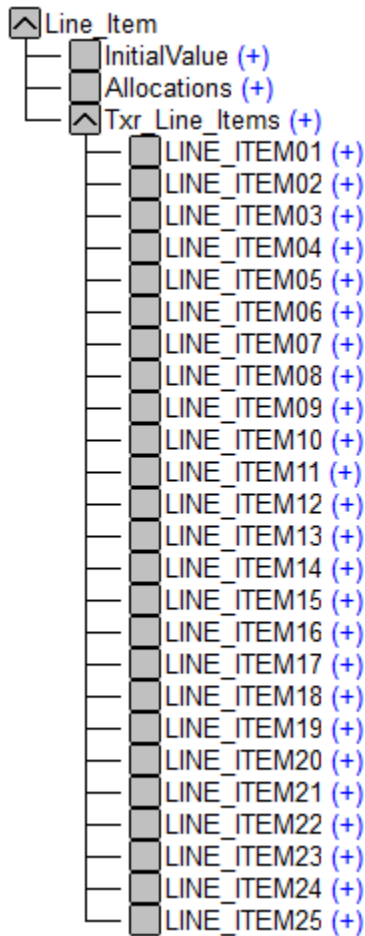
Line_Item Dimension

The Line_Item dimension will be used to store the transactional detail of transfers. Security will not be defined along the Period Dimension.

Following are key attributes of the Line_Item dimension within the FundTxr Plan Type:

- Data Storage – Dense
- Size – Approximately 30 stored members
- Update Frequency – Not Updated, unless additional Line Items are needed to support a high volume of transfers

The screen shot below represents the expected structure of the Line_Item dimension in this Plan Type:



Department Dimension

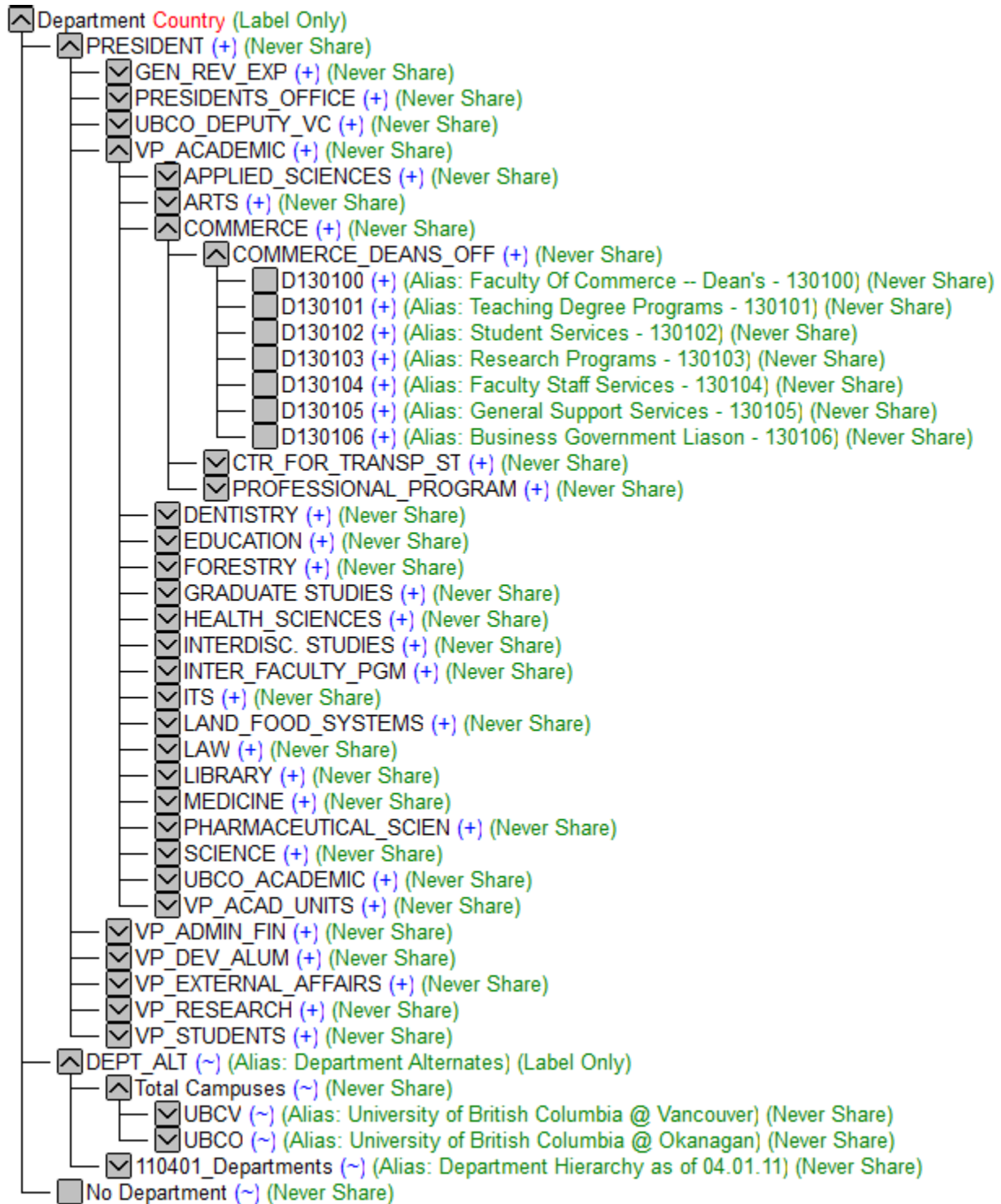
The Department dimension will be based on the PeopleSoft ALL_DEPTIDS tree, with alternate hierarchies for the ALL_DEPTIDS_VAN and ALL_DEPTIDS_UBCO. Additional alternate hierarchies could be developed to represent planned future changes to the University structure or for other reporting needs. The Department dimension will be the Entity dimension, so the hierarchy as defined will control the Workflow/Approval process. The “PRESIDENT”, “UBCV” and “UBCO” segments of the hierarchy will be based on PeopleSoft. The “No Department” member and any additional alternate hierarchies required will be built manually in EPMA.

Within the “PRESIDENT” segment, the naming convention for upper level members shall be that the member names will be based on the member names from PeopleSoft, with no modifications. The alias shall be based on the description from PeopleSoft, with no modification. For Level 0 members, the member name will be based on the member names from PeopleSoft (which consists of a 6 digit string), prepended with a “D”. The alias for Level 0 members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Department” member, and the “DEPT_ALT” member shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will be defined along the Department dimension.

Following are key attributes of the Department dimension within the FundTxr Plan Type:

- Data Storage – Sparse
- Size – Approximately 1,300 stored members
- Update Frequency – Updated infrequently, when new Departments are added in PeopleSoft or reorganizations of the Department structure occur

The screen shot below represents a high level view of the expected structure of the Department dimension in this Plan Type:



Txr_Dept Dimension

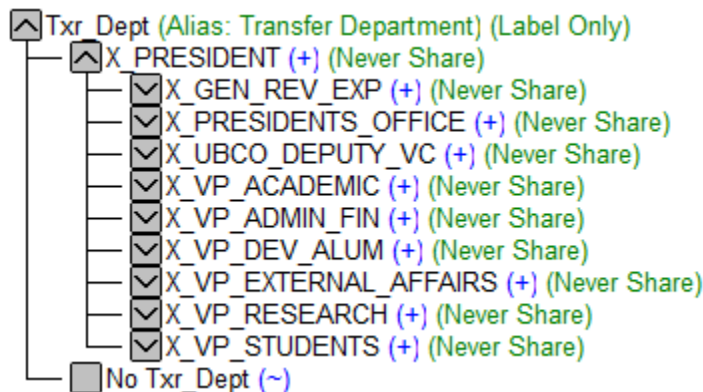
The Txr_Dept dimension will be a replica of the Department dimension, but will not include the alternate hierarchies included in the Department dimension.

All members except the top level “Txr_Dept” member and the “No Txr_Dept” member will be prepended with “X_”. The representation of the member name within the alias will also be prepended with “X_”. All upper level members in this dimension, with the exception of the top level “Txr_Dept” member shall be set to Never Share. Level 0 members can be set to Never Share or Stored. Security will not be defined along the Txr_Dept dimension.

Following are key attributes of the Txr_Dept dimension within the FundTxr Plan Type:

- Data Storage – Sparse
- Size – Approximately 1,300 stored members
- Update Frequency – Updated infrequently, when new Departments are added in PeopleSoft or reorganizations of the Department structure occur

The screen shot below represents a high level view of the expected structure of the Txr_Dept dimension in this Plan Type:



Project Grant Dimension

The Project Grant dimension will be based on the PeopleSoft Project Grant table. The children of “ALL_PGS”, with the exception of “PG_DEFAULT” and “PG_SHARED”, will be based on the DEPTID assigned in the PS_PROJECT_STATUS table in PeopleSoft. The “ALL_PGS”, “PG_DEFAULT”, “PG_SHARED” and “No Project Grant” members will be built manually in EPMA.

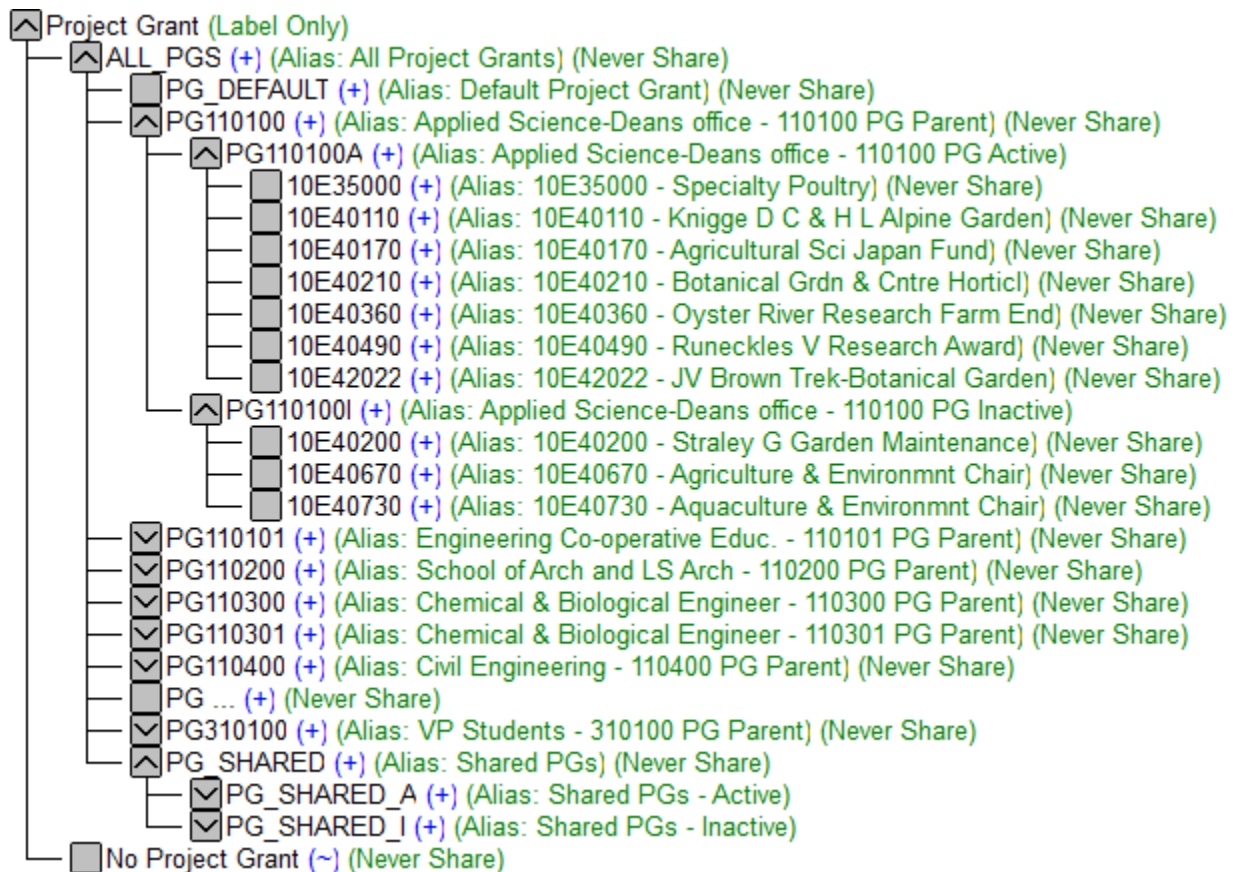
The member names for the children of “ALL_PGS” that are based on the DEPTID field will be based on the PeopleSoft member names prepended with “PG”. The aliases for these members will be based on the PeopleSoft description of the Department, appended with the member name and “PG Parent”. Under each of these DEPTID parents, an Active and Inactive parent will be created to segregate Active Project Grants from Inactive Project Grants. These levels of hierarchy exist in the Project Grant dimension to aid in the assignment of security. Security assignments within the dimension will be essential to limiting the number of PGs that users will see when budgeting, which will enhance the usability of the system.

The Level 0 descendants of “ALL_PGS” will be based on the member names from the PeopleSoft Project Grant table, with no modification. In general, the member name from PeopleSoft consists of an 8 character string, with the first two positions indicating the upper level Department with which the Project Grant is associated and the third position indicating the Fund Group with which the Project Grant is associated. The alias for these members will consist of the description from PeopleSoft with the member name appended to the end of the alias. All upper level members in this dimension, with the exception of the top level “Project Grant” member, shall be set to Never Share. Level 0 members can be set to Never Share or Stored. As noted above, security will be defined along the Project Grant dimension.

Following are key attributes of the Project Grant dimension within the BSPlan Plan Type:

- Data Storage – Sparse
- Size – Approximately 25,000 stored members
- Update Frequency – Updated frequently, as new Project Grants are frequently added to PeopleSoft

The screen shot below represents a high level view of the expected structure of the Project Grant dimension in this Plan Type:



FiscalYear Dimension

As the word “Year” is reserved by Essbase for Dynamic Time Series functionality, the standard “Year” dimension name cannot be used if Dynamic Time Series will be used. Dynamic Time Series will be enabled in the application, so the dimension name will be changed to “FiscalYear”.

As noted in the Design Constraints section above, the application must be built with 2005/06 as the Start Year should the opportunity to load 5 years of historical data arise prior to go-live. The application will go-live for the 2011/12 plan, and the ability to create a 5 year forecast is required. Therefore, the End Year in the application at go-live will be 2015/16.

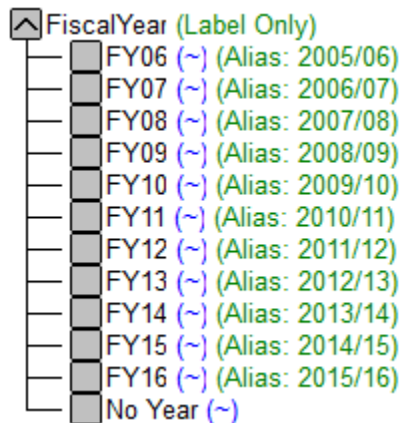
Planning does not offer flexibility in the member naming conventions within this dimension, so the FiscalYear dimension members will be named FY06, FY07, etc. Aliases will be used to incorporate UBC Fiscal Year naming standards.

The FiscalYear member will be set to Label Only, so when a user has not selected a member from the FiscalYear dimension, the data for the first stored member (FY06 in this case) will be displayed. The Level 0 members can be set to Never Share or Stored. Security is not applied along the FiscalYear dimension.

Following are key attributes of the FiscalYear dimension within the FundTxr Plan Type:

- Data Storage – Sparse
- Size – Approximately 12 stored members
- Update Frequency – Updated annually with an additional year

The screen shot below represents the expected structure of the FiscalYear dimension in this Plan Type:



Scenario Dimension

The Scenario dimension will be used to store the data related to different Planning cycles, store Actual data and calculate Variances.

The “Scenario” member shall be set to Label Only, so when a user has not selected a member from the Scenario dimension, the data for the first stored member (Forecast in this case) will be displayed. The other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

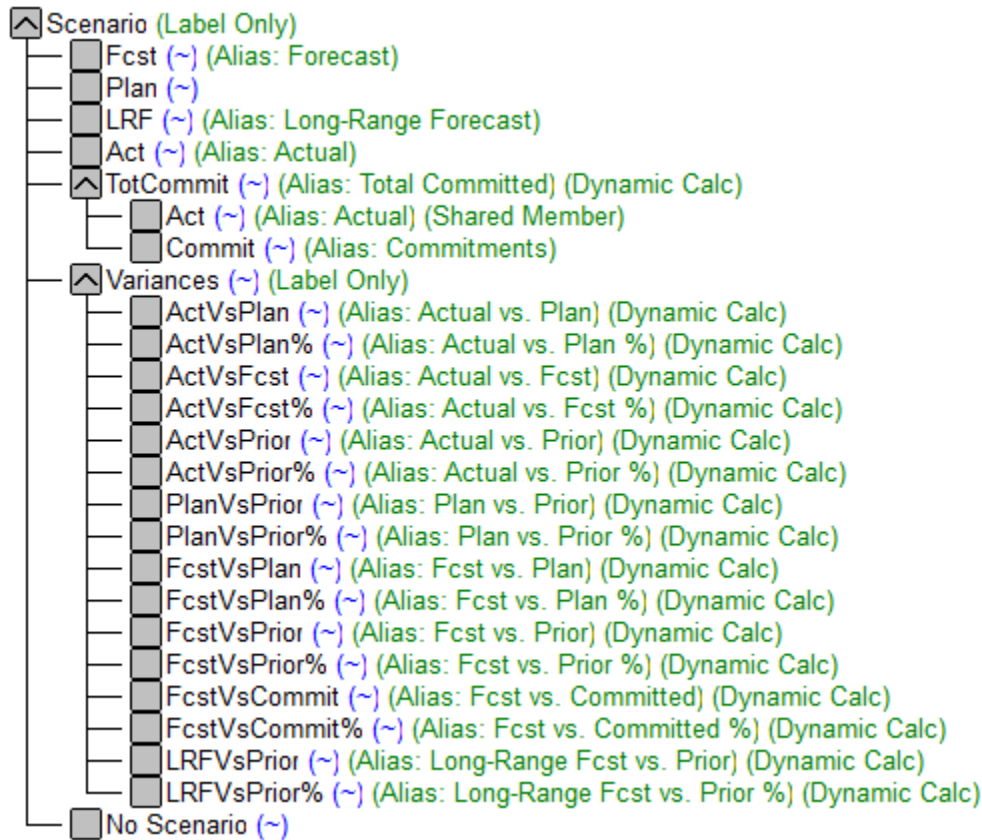
Following are key attributes of the Scenario dimension within the FundTxr Plan Type:

- Data Storage – Sparse
- Size – Approximately 5 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Scenario member that will be configured in Planning:

	Start Year	Start Mo.	End Year	End Mo.	Data Storage	Two-Pass	Incl. Beg. Bal.	Process Management
Forecast	FY10	Oct	FY10	Mar	Store	N/A	Yes	No
Plan	FY11	Apr	FY11	Mar	Store	N/A	Yes	Yes
Long-Range Forecast	FY11	Apr	FY15	Mar	Store	N/A	Yes	No
Actual	FY06	Apr	FY06	May	Store	N/A	Yes	No
Total Committed	FY06	Apr	FY06	May	Dynamic Calc	N/A	Yes	No
Commitments	FY06	Apr	FY06	May	Store	N/A	Yes	No
All Variances	FY06	Apr	FY06	May	Dynamic Calc	Yes	Yes	No
No Scenario	FY06	Apr	FY06	May	Store	N/A	Yes	No

The screen shot below represents the expected structure of the Scenario dimension in this Plan Type:



Version Dimension

The Version dimension will be used to store different versions of plans and forecasts. The main Version for user input will be the Working Version. This is the only Version that will be enabled for Process Management, and users will be granted Write access to this member. The Approved Version will be used to store the Actual data and the final Approved Plan. Draft1 and Draft2 will be used to store different Plan or Forecast iterations (e.g. “What-If” iterations) that are created by the users. This Version may also be used to load Balance Sheet Actual activity before the system calculates balances for storage in the Approved Version. Variance Versions may be added to allow comparisons between Versions should the need for these comparisons arise.

The “Version” member shall be set to Label Only, and the other members of this dimension can be set to Never Share or Stored. Security will be applied along this dimension.

Following are key attributes of the Version dimension within the FundTxr Plan Type:

- Data Storage – Sparse
- Size – Approximately 6 stored members
- Update Frequency – Not updated

The table below indicates the settings for each Version member that will be configured in Planning:

Member	Type	Data Storage	Two-Pass	Process Management
Approved	Standard Bottom Up	Store	N/A	No
Working	Standard Bottom Up	Store	N/A	Yes
Draft1	Standard Bottom Up	Store	N/A	No
Draft2	Standard Bottom Up	Store	N/A	No
No Version	Standard Bottom Up	Store	N/A	No

The screen shot below represents the expected structure of the Version dimension in this Plan Type:



Data Loads

Following is a description of the data load processes that will feed data to the FundTxr Plan Type:

Research Funding from RISE

Research Data will be loaded from RISE to the FundTxr Plan Type to support planning and forecasting activities. This data will be loaded on demand as needed.

Funding Data from Central Ledger

Funding Data will be refreshed from the Central Ledger to the FundTxr Plan Type. This data will be loaded on a monthly basis.

Planning Input Forms

The Planning Input Forms in the FundTxr Plan Type will be used to review Transfer Details, input allocation values and clear Pending Transfers. The currently defined form(s) include:

- Transfer Details
- Allocation Input
- Pending Transfers

Calculations

The Calculations required for the FundTxr Plan Type will be used to enable allocation of funding data and to execute transfers. The currently defined calculations include:

- Allocate Funding based on History
- Transfer
- Request Transfer
- Accept/Reject Transfer
- User Aggregation