

PBCS 101

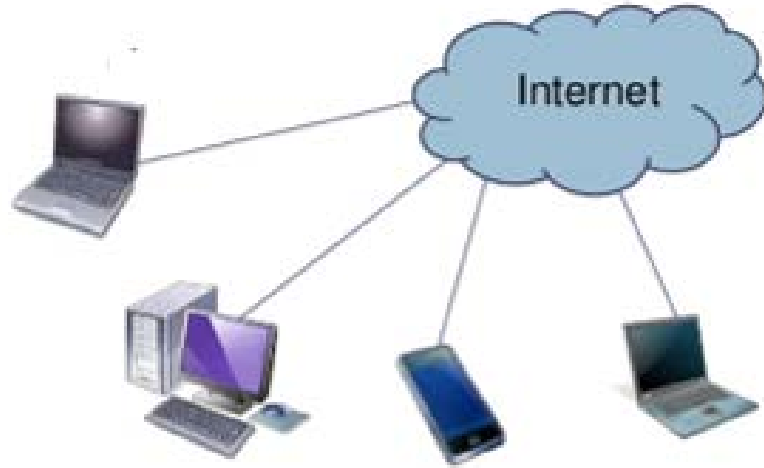
April 2018



Agenda

- What is Cloud Computing?
- What is a Pod?
- Pod Welcome Screen
- What is a Webform?
- What are Business Rules/Calculations?
- What is a Dimension?
- Configure vs Customize
- Alternate Tree Example
- Roles: Power User vs End User

PBCS 101: What is Cloud Computing?

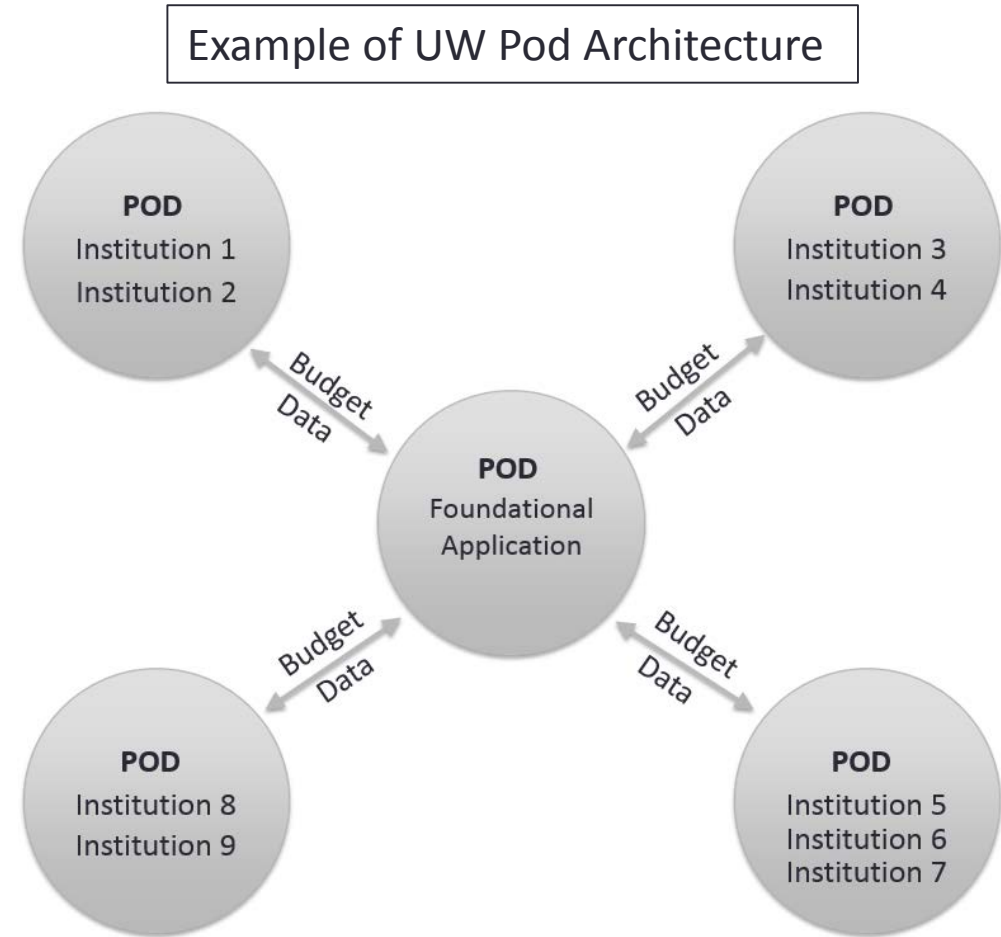


- A basic definition of cloud computing is the use of the Internet for the tasks you perform on your computer. The “cloud” represents the Internet.

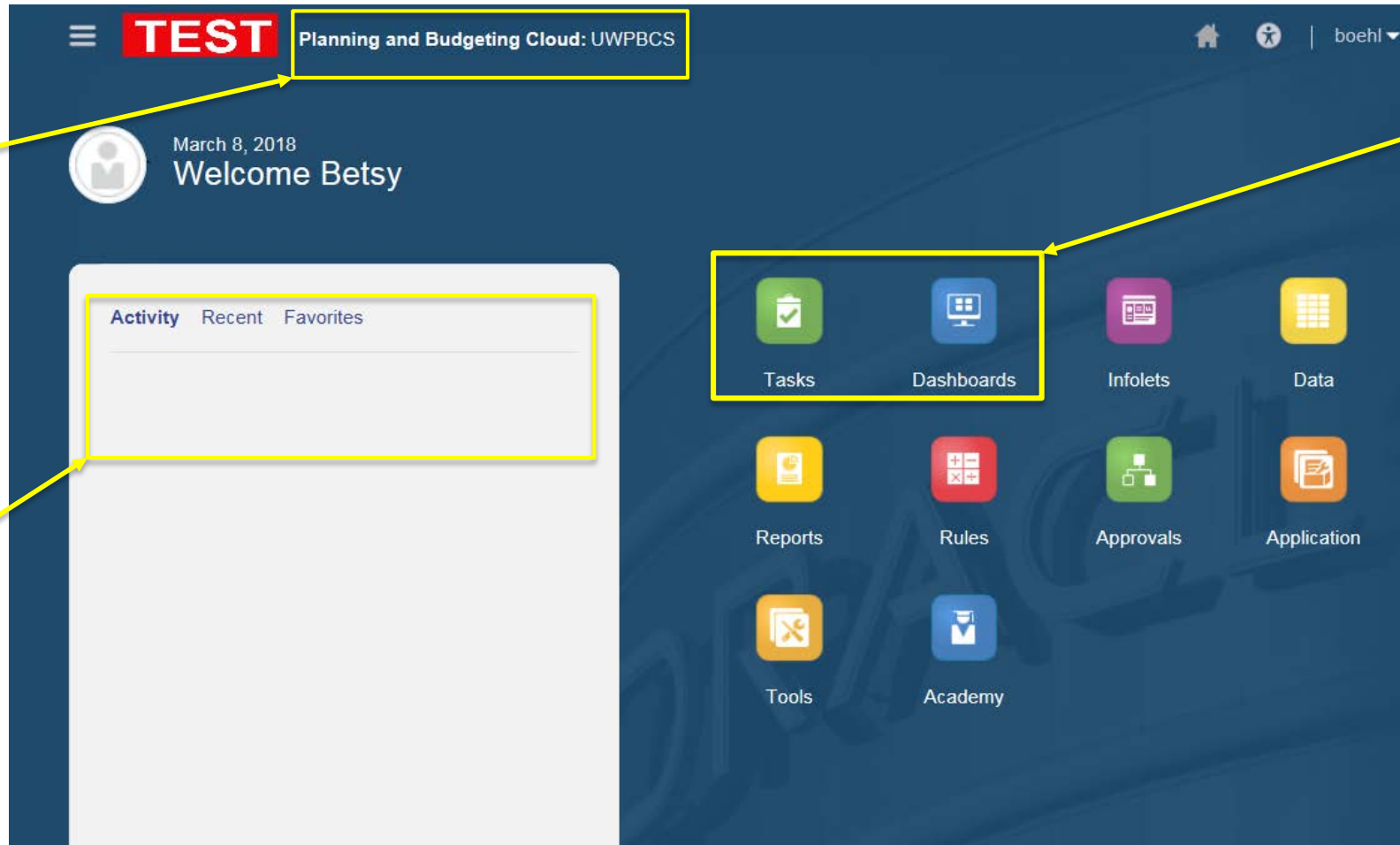
- The physical infrastructure, or servers, of the Cloud (aka “the Internet”) are located in data centers around the U.S.
- Regular patches and upgrades now happen automatically, on a schedule.
- No longer necessary to buy or maintain on-site servers.
- Users and work groups can access all of their information anytime, from anywhere in the world.

PBCS 101: What is a Pod?

- A database in the Cloud with a web address. All interaction with the Pod will be through a Browser.
Ex: The internet location www.facebook.com is an application/database that exists in the Cloud.
- Institution Pods will be built with the same architecture as the Foundational Pod, but the contents of the Pod can be configured to the preferences of the institutions included each Pod.
- The Foundational Pod receives and consolidates Budget data from all the Institution Pods.



PBCS 101: Pod Welcome Screen



This is the name of the Pod you are logged into

This is the area where your tasks will show up

These are called Tiles ... they are used to navigate to different areas of the Pod

PBCS 101: What is a Webform?

Purpose:

- A formatted, purpose built database 'view' for inputting or reviewing data.
- Structured similar to an Excel 'grid' view (rows, columns, page drop-down menus).
- Represents the main way in which users work with the tool

Scope of Use:

Can (and should) be used for a wide variety of purposes:

- Data input (lots of features)
- Data analysis (even more features, can be used as mock reporting)
- Workspace dashboards (combined input with immediate analysis)
- Execute Calculations

Sample Form ⓘ

🔍 | 👤 | **Actions** ▾ **Save** **Refresh** **Close**

Department	Program	Fund	Project					
My Department	Student Services	F_101	Project					
					2016	2017	2018	2018
					Actual	Actual	Budget	Estimated Actuals
								Variance
S&E BUDGET								
Travel Budget								
Non-Training Budget								
2100 - Travel-Employee-In State Bus								
2101 - Travel Empl Reportable Meals								
2115 - Travel-Employee-In State-Conf								
2120 - Travel-Employee-Out State-Bus								
2126 - Travel-Employee-Foreign Bus								
2127 - Travel-Employee-Foreign Conf								
2130 - Travel-Employee-Out State-Conf								
2140 - Travel-Fleet Charges-Vehicles								
2145 - Travel-Fleet Charges-Aircraft								
2146 - Mvng/Temp Ldg Emp-Taxable								
2147 - Moving Exp Emp-Non Taxable								
2164 - Travel-Team Travel								
2165 - Travel-Board Members								

PBCS 101: What are Business Rules and Calculations?

Purpose:

- Used to execute purpose built calculations and processes via commands.
- Can be extremely specific or have a broader purpose

Scope of Use:

Can (and should) be used for a wide variety of purposes:

- Data input validation (prevent human errors)
- Calculation of output (inputs drive outputs)
- Administrative tasks
- Data auditing
- Data management (maintaining data sets, scenarios/versions)



PBCS 101: What is a Dimension?

Dimension:

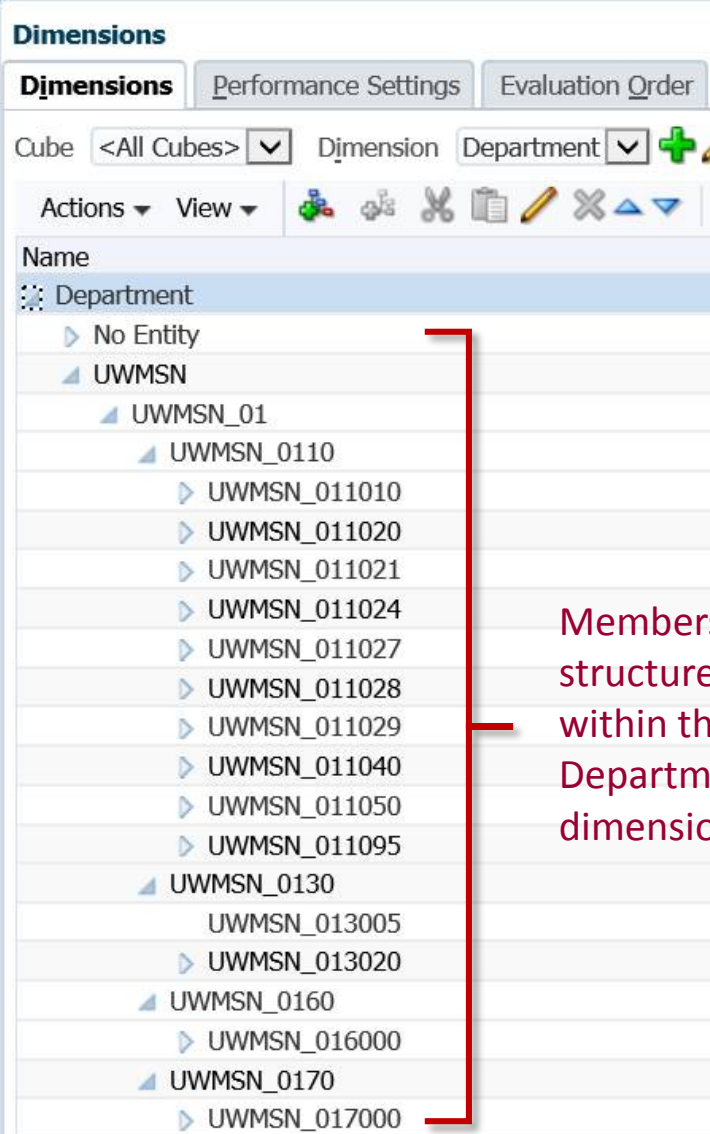
A Dimension is the hierarchical organization of a segment(s) of your GL Chart string. Similar to your CoA trees within SFS or WISER.

Example: Units, Divisions, Departments and Sub-Departments roll up to create the Department dimension.

Member:

Dimension Members are used to identify a data item's position and description within a dimension, and they in turn make up a Dimension

Example: Sub-Department UWMSN_011010 is a member within the Department Dimension



The screenshot shows the 'Dimensions' tool interface. At the top, there are tabs for 'Dimensions', 'Performance Settings', and 'Evaluation Order'. Below the tabs, there are dropdown menus for 'Cube' (set to '<All Cubes>') and 'Dimension' (set to 'Department'). A toolbar with various icons is visible. The main area displays a hierarchical tree structure for the 'Department' dimension. The tree starts with 'Department' and branches into 'No Entity', 'UWMSN', 'UWMSN_01', 'UWMSN_0110', and 'UWMSN_0130'. Under 'UWMSN_0110', there is a list of members: UWMSN_011010, UWMSN_011020, UWMSN_011021, UWMSN_011024, UWMSN_011027, UWMSN_011028, UWMSN_011029, UWMSN_011040, UWMSN_011050, UWMSN_011095, UWMSN_013005, UWMSN_013020, UWMSN_0160, UWMSN_016000, UWMSN_0170, and UWMSN_017000. A red bracket on the right side of the tree highlights the members under 'UWMSN_0110', with a red arrow pointing to the text 'Members structured within the Department dimension.'

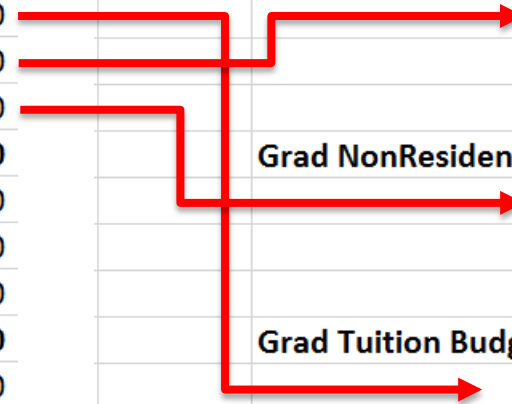
PBCS 101: Alternate Tree Example

Budget CoA Tree

Grad Tuition Revenue (SFS Tree Rollup)		187,500
Grad Tuition - Semester 1		75,000
<i>Grad Tuition - Semester 1 Budget</i>		50,000
9113 - Sem 1-Resident-Graduate		15,000
9117 - Sem 1-NonResident-Graduate		10,000
Grad Tuition - Semester 2		75,000
<i>Grad Tuition - Semester 2 Budget</i>		50,000
9123 - Sem 2-Resident-Graduate		15,000
9127 - Sem 2-NonResident-Graduate		10,000
Grad Tuition - Summer		37,500
<i>Grad Tuition - Summer Budget</i>		25,000
9103 - Summer-Resident-Grad		7,500
9107 - Summer-NonResident-Graduate		5,000

Alternate (Campus) CoA Tree

Grad Tuition - By Residency (Alternate Tree Rollup)		187,500
Grad Resident Tuition		37,500
9113 - Sem 1-Resident-Graduate		15,000
9123 - Sem 2-Resident-Graduate		15,000
9103 - Summer-Resident-Grad		7,500
Grad NonResident Tuition		25,000
9117 - Sem 1-NonResident-Graduate		10,000
9127 - Sem 2-NonResident-Graduate		10,000
9107 - Summer-NonResident-Graduate		5,000
Grad Tuition Budget		125,000
<i>Grad Tuition - Semester 1 Budget</i>		50,000
<i>Grad Tuition - Semester 2 Budget</i>		50,000
<i>Grad Tuition - Summer Budget</i>		25,000



Budget CoA Shared Trees will be built in PBCS Foundational Pod

PBCS 101: Configure vs Customize

Create Application: Details

*** Configure Example**

Back Cancel

General **Details** Customize Review

Planning Frequency

- Monthly
- Weekly
- Quarterly
- Custom

Start and End year: 2016 to 2027

First Month of Fiscal Year: July

Fiscal Year Start Date: Same Calendar Year

Weekly Distribution: Even Distribution

Dimensions

Dimensions Performance Settings Evaluation Order

Cube: <All Cubes> Dimension: Period Search: Name

Base Period: 12 Months First Fiscal Year: 2016

Actions View

- Period
 - BegBalance
 - YearTotal
 - Q1
 - Jul
 - Aug
 - Sep
 - Q2
 - Oct
 - Nov
 - Dec
 - Q3
 - Jan
 - Feb
 - Mar
 - Q4
 - Apr
 - May
 - Jun

Structure is created and configured during initialization of the application and then maintained automatically by the system

Create Application: Details

*** Customize Example**

Back Cancel

General **Details** Customize Review

Planning Frequency

- Monthly
- Weekly
- Quarterly
- Custom

Start and End year: 2016 to 2027

Rolling Forecast: Enable 1 Period Duration

Dimensions

Dimensions Performance Settings Evaluation Order

Cube: <All Cubes> Dimension: Period Search: Name

Actions View

- Period
 - BegBalance
 - YearTotal
 - Day_1
 - Day_2
 - Day_3
 - Day_4
 - Day_5
 - Day_6
 - Day_7
 - Day_8
 - Day_9
 - Day_10

Custom structure is created manually and requires CUSTOM SCRIPTING to operate and maintain

```
/* 5 case statements exist for datefactor
1: start and end dates are fully inclusive of current month
2: start and end dates are fully exclusive of current month
3: start date is within current month, end date is inclusive of current month
4: start date is inclusive of current month, end date is within current month
5: both start and end dates are within current month
*/
/* 1: start and end dates are fully inclusive of current month */
IF(@startdate<="FirstdayinMonth"+1 AND @enddate>="FirstdayinMonth"+@daysinMonth)
datefactor=1;
/* 2: start and end dates are fully exclusive of current month */
ELSEIF(@startdate="FirstdayinMonth"+@daysinMonth OR @enddate<="FirstdayinMonth")
datefactor=#missing;
/* 3: start date is within current month, end date is inclusive of current month */
ELSEIF(@startdate="FirstdayinMonth"+1 AND @enddate="FirstdayinMonth"+@daysinMonth)
datefactor=(@daysinMonth-(@startdate-@INT(@startdate/100)+1)/@daysinMonth);
/* 4: start date is inclusive of current month, end date is within current month */
ELSEIF(@startdate<="FirstdayinMonth" AND @enddate<="FirstdayinMonth"+@daysinMonth)
datefactor=(@enddate-@INT(@enddate/100)+1)/@daysinMonth;
/* 5: both start and end dates are within current month */
ELSEIF(@startdate="FirstdayinMonth" AND @enddate<="FirstdayinMonth"+@daysinMonth)
datefactor=((@enddate-@INT(@enddate/100)+1)-(@startdate-@INT(@startdate/100)+1)/@daysinMonth);
ENDIF;
```

PBCS 101: User Types

Power Users

Responsibilities During Project Phases:

- Keep Up to Date on Project Communications
- Provide Feedback on during Design Roadshow
- Hands-On Participation for Build Review Session
- Participate in UAT Testing
- Assist Project Team with writing Test Scenarios
- UAT Tester
- Assist Project Team with Data Validation for your Institution
- Assist Change Management Team as End User Training Resource
- Learn Smart View Tool

Responsibilities Once PBCS is Live:

- Input Global Rates for use in Budget Cycle
- Assist End Users with questions (if needed)
- Enter Budget Data for 2019-20 Budget Cycle into the new tool
- Give Feedback on the Tool to the Support Team
- Build Ad-Hoc reports with SmartView

End Users

Responsibilities During Project Phases:

- Keep Up to Date on Project Communications
- Participate in End User Training Sessions
- Ask Questions at any time to your Campus Representative on the Institutional Advisory Group.

Responsibilities Once PBCS is Live:

- Enter Budget Data for 2019-20 Budget Cycle in the new tool
- Run Reports to analyze Budget Data
- Give feedback on the Tool to Institutional Advisory Group Representative

QUESTIONS?