

Planning Analytics

Financials solution accelerator

Summary

This document details how the Financials solution accelerator for IBM Planning Analytics with Watson is designed. The solution is available on the IBM Accelerator Catalog. From there, it can be downloaded and uploaded into another environment for use as a starting point for building a more comprehensive Financial Reporting and Planning model.

This documentation will cover:

- The model purpose and how to it is intended to be used
- Overview of how the model is designed
- Model architecture - inflow and outflows
- Purpose and design principles for each cube, dimension, process and chore
- What rules have been applied
- How to update the model with your own data
- Suggestions on how to build upon it

Table of Contents

Model Purpose 4

How to Access the Model 4

Overview of How the Model is Designed 5

Model Architecture 6

 Cubes Overview 6

 Cube Design 7

Dimension Design 8

 Account 8

 Account Type 10

 Company 11

 Cost Centre 13

 Currency 15

Measures Financials	16
Measures Relative Time.....	17
Parameter Name	18
Parameter Value	18
Period	19
Relative Time	22
Version	24
Processes	26
Chores.....	28
Rules	29
Financials Cube	29
Relative Time Cube.....	30
Parameter Cube.....	30
Updating the Model with Your Own Data	31
Summary	31
Task List.....	31
Step-by-Step walk through	32
Clear existing dummy data in the Financials cube	32
Download the data source text files from File Manager.....	32
Populate the files with your own data where applicable	32
Upload the new data source text files to File Manager	33
Run the processes to import the data (text files)	34
Review imported figures within Planning Analytics.....	36
Establish a connection in Planning Analytics to the data source (where relevant).....	37
Create database table.....	38
Populate the database table with data	38
Update process ODBC connection details and database query	39
Run the processes to import the data (ODBC)	39
Schedule the chores to import the Financials data, if relevant.....	40
Re-populate the Relative Time cube	40
Schedule the Chore to SaveDataAll.....	40
Decommission any processes not used to prevent erroneous database updates.....	41

Suggestions for building upon the design 42

- Adding more dimensions to the Financials cube 42
- Front end application in Apps and Plans 44
- Adding currency conversion 45
- Adding in driver-based planning for integrated business planning 45
- Adding Cash Flow reporting and planning 46
- Ability to add journal adjustments..... 46
- Ability to map codes using in local finance systems to a set of group codes used in dimensions within Planning Analytics..... 47
- Validation of data prior to loading it into the Financials cube - prevents loading of data under certain criteria 48

Model Purpose

This model is intended to be used for Financial Reporting and Planning. It covers P&L and Balance sheet only. It combines actual and planning data (i.e. budget and forecasting). It features the follows:

- Multi-company group structures
- Rolling forecast and budgeting
- Relative time comparison (e.g. this year vs last year, YTD/Balance)
- Version comparison (e.g. actual vs budget)
- Provision for foreign currency conversion
- Planning data collection - manual input or file upload

The solution is intended to be a basic financial model in which you upload your own data into it using the pre-existing data upload templates. The model and upload templates contain dummy data that is intended to be used as an example/guide of how to structure the data.

From there, some basic planning and reporting can be achieved. The model is intended to be used as a fast start but can be built upon to include more advanced features. Whilst there is provision for foreign currency version in terms of data structures, the model omits the foreign currency rules but could be added to customize the solution.

How to Access the Model

Navigate to this page on the [IBM Accelerator Catalog](#). Download the zip file. Extract the zip and upload to your own environment to import the model into a Planning Analytics database.

Overview of How the Model is Designed

The model contains a main cube called 'Financials'. It contains the actual and plan data for the P&L and Balance Sheet. There are two other cubes which support the rules against this cube. 'Relative Time' cube is used to support the relative time rules. 'Parameter' is used to store central parameters which govern how the model works. The parameter 'Latest Actual Period' is used by the Financials cube rules to dynamically reference the actuals within the Working Forecast. It also is used to prevent any rules against future months within the 'Actual' to improve performance of the model.

It is intended that any forecast or budget data is entered against the Working Forecast for any year or month in the future. Once the forecast or budget cycle is complete and plan is signed off, the relevant year within Working Forecast would then be copied into either budget or the relevant forecast (e.g. 3+9, 6+6) as a static copy of the data. This static copy would be reported against within financial reports.

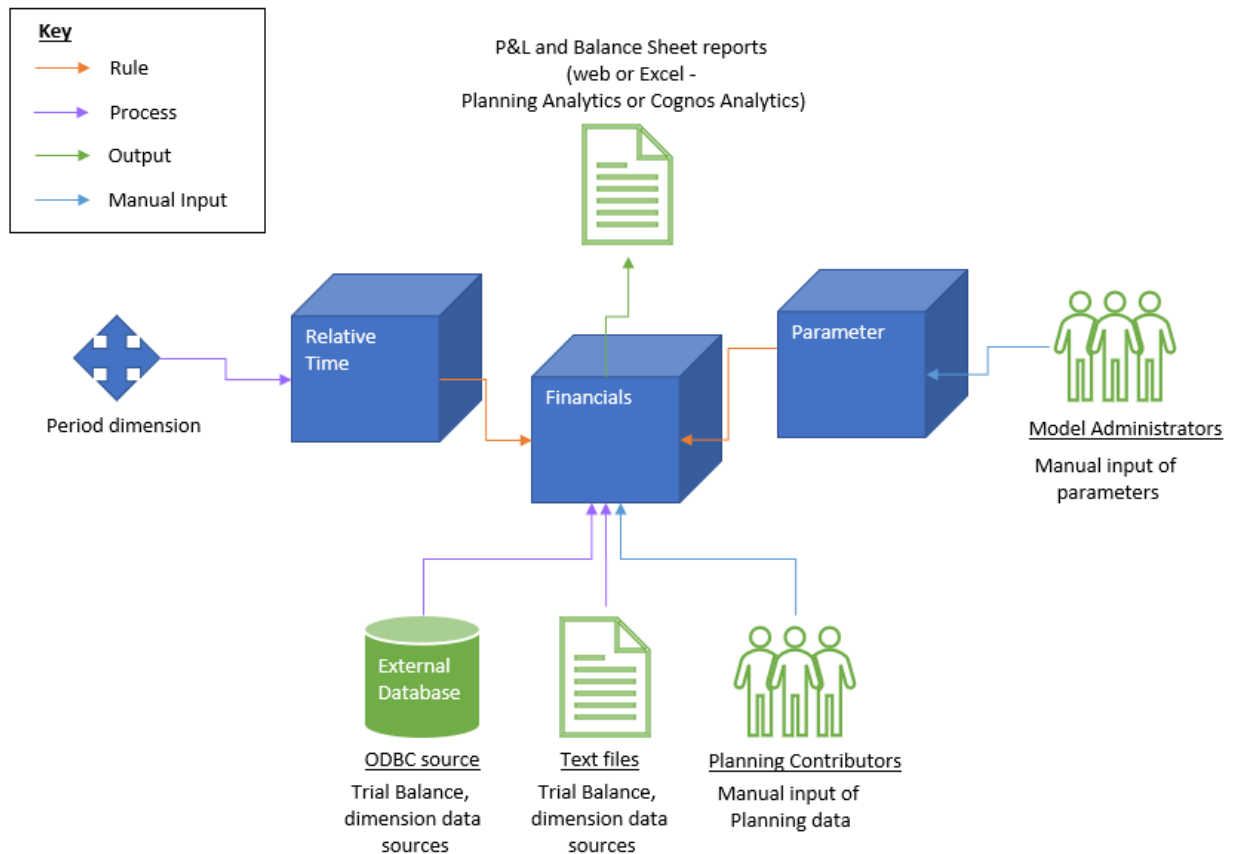
There are two options for uploading the trial balance data into the model. One option is via a text file, the other option is ODBC. Where dimensions are imported via a process, this is done using text files. The text files are included in 'data' folder behind the model. These can be opened, populated and re-uploaded into File Manager within Workbench before rerunning the processes to update the dimension designs. The ODBC option will require updating the example ODBC details during setup to the ones where the business data is stored.

There is also a template process which can be used to speed up creating new dimensions from a process.

The model rules and processes are commented with business logic to explain the scripting in business language. If any scripting is updated, it is recommended that the commentary is also updated. Any new rules or processes are encouraged to follow the same methodology.

Model Architecture

This architecture diagram describes the inflow and outflows that interact with the cubes in the model. Data is either manually input, imported via a process, or dynamically connected via a rule. The output is a set of monthly reports containing the P&L and Balance Sheet. This can be within Planning Analytics or extended into Cognos Analytics where there are more advanced distribution methods available such as bursting.

















Cubes Overview





Cube Name	Contains...	Purpose
Financials	P&L and Balance Sheet data by period for Actual and Plan (Budget and Forecasts)	Financial reporting and planning for P&L and Balance Sheet
Relative Time	A list of all periods plus a lookup of other periods relative to it, for example the same period last year, the same period 12 months prior, next period, etc	Support relative time rules throughout the model (e.g. Period LY, YTD, YTD LY etc).
Parameter	A list of parameters (mixture of text and numeric) centrally stored which will be reference by rules, processes and other model objects	Support central governance of the model. Within the Working Forecast it determines up to which month includes actuals.

Cube Design

A list of the cubes and their dimensions:

- ▼  Financials
 - ▼  Dimensions
 - ▶  Relative Time
 - ▶  Period
 - ▶  Version
 - ▶  Currency
 - ▶  Account
 - ▶  Company
 - ▶  Cost Centre
 - ▶  Measures Financials

- ▼  Parameter
 - ▼  Dimensions
 - ▶  Parameter Value
 - ▶  Parameter Name

- ▼  Relative Time
 - ▼  Dimensions
 - ▶  Period
 - ▶  Measures Relative Time

Dimension Design

Account

Contains...	Purpose
P&L and Balance Sheet accounts used by the group. The P&L is embedded within the Balance Sheet structure to ensure that they are in line. The total of all accounts adds up to the 'Trial Balance' total. This should be zero.	P&L and Balance sheet reporting within the Financials cube

The member ID is a code. The caption is an alias containing the description. Each member must have a unique description. Account type is an attribute assigned to each member (see Account Type dimension for more details). This drives the logic in populating the 'BS or P&L' attribute. The BS or P&L (Balance Sheet or Profit and Loss) attribute is used to help with the YTD calculation. 'YTD' in Relative Time dimension is balance (all time to date) for Balance Sheet accounts, and Year to Date (reset the balance every period 1) for Profit and Loss accounts. Account Type is also assigned at the consolidated levels as the 'Reporting Amount' measure (from the 'Measures Financials') dimension uses this to convert the Accounting amount to Reporting Amount with the correct signage. Rate Type is used to assist with the currency conversion from Local Currency to Group Currency. The other attributes are iterations of the code and description combined or separate and useful for display purposes.

Account	BS or P&L (Text)	Account Ty... (Text)	Desc and C... (Alias)	Rate Type (Text)	Desc (Text)	Code (Alias)	Code and D... (Alias)	Caption (Alias)
B00000 - Trial Balance	BS	A	Trial Balanc...	B	Trial Balance	B00000	B00000 - Tr...	Trial Balance
B29999 - Total Assets	BS	A	Total Assets...	B	Total Assets	B29999	B29999 - T...	Total Assets
B19099 - Non-Current Assets	BS	A	Non-Curren...	B	Non-Curren...	B19099	B19099 - N...	Non-Curren...
B29099 - Current Assets	BS	A	Current Ass...	B	Current Ass...	B29099	B29099 - C...	Current Ass...
B99999 - Equity and Liabilities	BS	L	Equity and ...	B	Equity and ...	B99999	B99999 - E...	Equity and ...
B39999 - Liabilities	BS	L	Liabilities - ...	B	Liabilities	B39999	B39999 - Li...	Liabilities
B69999 - Equity	BS	E	Equity - B6...	E	Equity	B69999	B69999 - E...	Equity
B69099 - Equity - Group	BS	E	Equity - Gro...	U	Equity - Gro...	B69099	B69099 - E...	Equity - Gro...
B60099 - Share capital	BS	E	Share capit...	E	Share capital	B60099	B60099 - S...	Share capital
B61000 - Share premium reserve	BS	E	Share prem...	U	Share prem...	B61000	B61000 - S...	Share prem...
B62010 - Capital redemption reserve	BS	E	Capital red...	E	Capital red...	B62010	B62010 - C...	Capital red...
B62013 - Revaluation reserve	BS	E	Revaluation...	E	Revaluation...	B62013	B62013 - R...	Revaluation...
B62015 - Other reserves	BS	E	Other reser...	E	Other reser...	B62015	B62015 - O...	Other reser...
B62016 - Foreign exchange reserve	BS	E	Foreign exc...	E	Foreign exc...	B62016	B62016 - F...	Foreign exc...
B62018 - Rounding diff	BS	E	Rounding di...	M	Rounding diff	B62018	B62018 - R...	Rounding diff
B62019 - Exchange difference on equity	BS	E	Exchange d...	U	Exchange d...	B62019	B62019 - E...	Exchange d...
B65000 - Retained earnings-result	BS	E	Retained e...	U	Retained e...	B65000	B65000 - R...	Retained e...
B65000300 - Ret Earnings / OB	BS	E	Ret Earning...	C	Ret Earning...	B65000300	B65000300...	Ret Earning...
B65000310 - Balance Sheet Result / Additions	BS	E	Balance Sh...	V	Balance Sh...	B65000310	B65000310...	Balance Sh...
P99999 - Profit			Profit - P99...	U	Profit	P99999	P99999 - Pr...	Profit

This dimension is built from the following process: DIM_Account

Here is the data source column layout:

	A	B	C	D	E	F	G
1	Child	Child Desc	Parent 1	Parent 2	Parent 3	Account Type	Rate Type
2	B00000	Trial Balance				A	B
3	B10000	Goodwill	B10099			A	B
4	B10010	Trade-marks & licenses	B10099			A	B
5	B10020	Software licences	B10099			A	B
6	B10030	Other intangibles	B10099			A	B
7	B10099	Intangible Assets	B19099			A	B
8	B11000	Land and buildings	B11099			A	B
9	B11010	Plant & machinery	B11099			A	B
10	B11020	Motor vehicles	B11099			A	B

- **Child:** reflects the member ID. It comprises a mixture of leaf level and consolidated level members. Every member in the hierarchy should appear in this column.
- **Child Desc:** the description of the member ID in the child column. This drives the caption.
- **Parent 1:** the parent of the member in the child column. Should be in member ID format. If it is a root member (e.g. Trial Balance) it will not have a parent.
- **Parent 2:** optional second parent of the member in the child column. A member should not appear twice under the 'trial balance' root member otherwise it will double up the numbers. Instead, parent 2 should be used as a way of creating an alternate roll up in the same hierarchy.
- **Parent 3:** similar to parent 2 but option for a third parent.
- **Account Type:** populated with a letter describing the account type. Those in the trial balance will be either A, L, I, C or E. See 'Account Type' section below. If there are KPI related accounts such as Gross Margin %, Operating Profit %, these can be assigned a different letter or no letter.
- **Rate Type:** populated with a letter describing the rate type. Typically, Balance Sheet accounts will have a rate type of closing rate, and profit and loss accounts will have a rate type of average rate. The letters can be chosen upon set up. These are not used in the model but are intended to be used when currency translation rules are implemented.

Account Type

Contains...	Purpose
The list of account types referenced in the 'Account' dimension. Contains: A, L, I, C, E. Respectively, this means Asset, Liability, Income, Cost, Equity.	Standalone dimension used as a lookup to determine if an account is P&L or Balance Sheet and the multiplication factor to calculate figures in accounting vs reporting signage

The screenshot shows the 'Account Type' dimension editor. It features a search bar with 'Find members' and a table with the following data:

Account Type	Accounting to Reporting Signage MF (Number)	BS or P&L (Text)
A	1	BS
L	1	BS
I	-1	P&L
C	-1	P&L
E	1	BS

This dimension is maintained manually via the dimension editor. The 'Accounting to Reporting Signage MF' attribute is used within rules in the Financials cube, specifically the rule that calculates Reporting Amount (# Financials.2.001 - Reporting Amount). The 'BS or P&L' attribute is used in the process DIM_Account to populate the 'BS or P&L' attribute in the Account dimension.

Company

Contains...	Purpose
<p>List of entities within a group structure along with sub totals that aggregate them up. Can contain group adjustment companies which can be used as a method of entering data against a sub total level.</p> <p>Can contain more than one structure/roll up but each structure must add up to a single root.</p>	<p>P&L and Balance sheet reporting within the Financials cube at company level.</p>

Company	Currency (Text)	Desc (Text)	Code (Alias)	Desc and C... (Alias)	Code and D... (Alias)	Caption (Alias)
ME0000 - Management Structure	GBP	Manageme...	ME0000	Manageme...	ME0000 - ...	ME0000 - ...
ME1000 - EMEA	GBP	EMEA	ME1000	EMEA - ME...	ME1000 - E...	ME1000 - E...
1110 - UK	GBP	UK	1110	UK - 1110	1110 - UK	1110 - UK
1111 - Ireland	EUR	Ireland	1111	Ireland - 1111	1111 - Ireland	1111 - Ireland
1112 - France	EUR	France	1112	France - 1112	1112 - France	1112 - France
ME1000AJ - EMEA - Group Adj	GBP	EMEA - Gro...	ME1000AJ	EMEA - Gro...	ME1000AJ...	ME1000AJ...
ME2000 - Americas	USD	Americas	ME2000	Americas - ...	ME2000 - A...	ME2000 - A...
2110 - USA	USD	USA	2110	USA - 2110	2110 - USA	2110 - USA
2111 - Canada	CAD	Canada	2111	Canada - 2...	2111 - Can...	2111 - Can...
ME2000AJ - Americas - Group Adj	USD	Americas - ...	ME2000AJ	Americas - ...	ME2000AJ...	ME2000AJ...
ME3000 - APAC	JPY	APAC	ME3000	APAC - ME...	ME3000 - A...	ME3000 - A...
3110 - Japan	JPY	Japan	3110	Japan - 3110	3110 - Japan	3110 - Japan
3111 - China	CNY	China	3111	China - 3111	3111 - China	3111 - China
ME3000AJ - APAC - Group Adj	JPY	APAC - Gro...	ME3000AJ	APAC - Gro...	ME3000AJ...	ME3000AJ...
ME0000AJ - Management Structure - Group Adj	GBP	Manageme...	ME0000AJ	Manageme...	ME0000AJ...	ME0000AJ...

The member ID is the code, and the caption is an alias containing the code and description. Despite the example structure being balanced, it can be unbalanced. The Currency attribute is not referenced in the model. It is intended to be used when currency translation rules are implemented. The other attributes are iterations of the code and description combined or separate and useful for display purposes.

The dimension is built from a process called 'DIM_Company'. There is another process called 'DIM_Company (Wizard)' which is an example of a process that used the dimension building wizard to populate it.

Here is the data source column layout for 'DIM_Company':

	A	B	C	D	E	F
1	Child	Child Desc	Parent 1	Parent 2	Parent 3	Currency
2		1110 UK	ME1000			GBP
3		1111 Ireland	ME1000			EUR
4		1112 France	ME1000			EUR
5		2110 USA	ME2000			USD
6		2111 Canada	ME2000			CAD
7		3110 Japan	ME3000			JPY
8		3111 China	ME3000			CNY
9	ME1000	EMEA	ME0000			GBP
10	ME2000	Americas	ME0000			USD
11	ME3000	APAC	ME0000			JPY
12	ME0000	Management Structure				GBP
13	ME1000AJ	EMEA - Group Adj	ME1000			GBP
14	ME2000AJ	Americas - Group Adj	ME2000			USD
15	ME3000AJ	APAC - Group Adj	ME3000			JPY
16	ME0000AJ	Management Structure - Group Adj	ME0000			GBP

- **Child:** reflects the member ID. It comprises a mixture of leaf level and consolidated level members. Every member in the hierarchy should appear in this column.
- **Child Desc:** the description of the member ID in the child column. This drives the caption.
- **Parent 1:** the parent of the member in the child column. Should be in member ID format. If it is a root member (e.g. Management Structure) it will not have a parent.
- **Parent 2:** optional second parent of the member in the child column. A member should not appear twice under the root member otherwise it will double up the numbers. Instead, parent 2 should be used as a way of creating an alternate roll up in the same hierarchy.
- **Parent 3:** similar to parent 2 but option for a third parent.
- **Currency:** The local currency of the company.

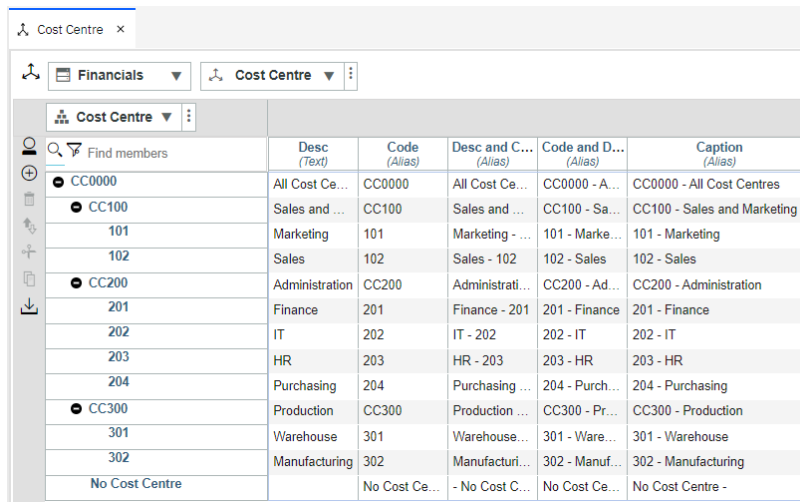
Here is the data source column layout for 'DIM_Company (wizard)'.

	A	B	C	D	E	F	G
1	Child	Desc	Parent	Currency	Code and De	Desc and Cod	Code
2	1110	UK	ME1000	GBP	1110 - UK	UK - 1110	1110
3	1111	Ireland	ME1000	EUR	1111 - Ireland	Ireland - 1111	1111
4	1112	France	ME1000	EUR	1112 - France	France - 1112	1112
5	2110	USA	ME2000	USD	2110 - USA	USA - 2110	2110
6	2111	Canada	ME2000	CAD	2111 - Canada	Canada - 2111	2111
7	3110	Japan	ME3000	JPY	3110 - Japan	Japan - 3110	3110
8	3111	China	ME3000	CNY	3111 - China	China - 3111	3111
9	ME1000	EMEA	ME0000	GBP	ME1000 - EM	EMEA - ME10	ME1000
10	ME2000	Americas	ME0000	USD	ME2000 - Am	Americas - M	ME2000
11	ME3000	APAC	ME0000	JPY	ME3000 - AP	APAC - ME30	ME3000
12	ME0000	Management Structu	ME0000	GBP	ME0000 - Ma	Management	ME0000

The layout is similar but only has one Parent field and the attribute values for 'Code and Desc', 'Desc and Code' and 'Code' are created in the data source rather than created by the script in the process. This is due to the inability to concatenate/calculate fields in the wizard.

Cost Centre

Contains...	Purpose
List of cost centres with sub totals that aggregate them up. Can contain more than one structure /roll up but each structure must add up to a single root.	P&L reporting within the Financials cube at cost centre level.



The screenshot shows the SAP Cost Centre configuration table. The table has columns for Description, Code, and various combinations of Description and Code. The rows are organized into hierarchical groups: CC0000 (All Cost Centres), CC100 (Sales and Marketing), CC200 (Administration), and CC300 (Production). Each group contains sub-centres with their respective codes and descriptions.

	Desc (Text)	Code (Alias)	Desc and C... (Alias)	Code and D... (Alias)	Caption (Alias)
● CC0000	All Cost Ce...	CC0000	All Cost Ce...	CC0000 - A...	CC0000 - All Cost Centres
● CC100	Sales and ...	CC100	Sales and ...	CC100 - Sa...	CC100 - Sales and Marketing
101	Marketing	101	Marketing - ...	101 - Marke...	101 - Marketing
102	Sales	102	Sales - 102	102 - Sales	102 - Sales
● CC200	Administration	CC200	Administrati...	CC200 - Ad...	CC200 - Administration
201	Finance	201	Finance - 201	201 - Finance	201 - Finance
202	IT	202	IT - 202	202 - IT	202 - IT
203	HR	203	HR - 203	203 - HR	203 - HR
204	Purchasing	204	Purchasing ...	204 - Purch...	204 - Purchasing
● CC300	Production	CC300	Production ...	CC300 - Pr...	CC300 - Production
301	Warehouse	301	Warehouse...	301 - Ware...	301 - Warehouse
302	Manufacturing	302	Manufacturi...	302 - Manuf...	302 - Manufacturing
No Cost Centre		No Cost Ce...	- No Cost C...	No Cost Ce...	No Cost Centre -

The member ID is the code, and the caption is the code and description. The member 'No Cost Centre' is advised to be included as not all financial values are cut by cost centre (e.g. Balance Sheet). So, when uploading a trial balance, the Cost Centre field may be blank for certain rows. In this case, the process [DATA_Financials (Import Trial Balance)] maps these blank values to 'No Cost Centre'. The related extract of script from this process is below:

```
#####
*# IMPORT DATA
#####

# Declare elements of cell intersection to import into

sDim1Element = 'Period';
sDim2Element = vPeriod;
sDim3Element = pVersion;
sDim4Element = 'LC';
sDim5Element = vAccount;
sDim6Element = vCompany;
sDim7Element = IF(vCostCentre@='', 'No Cost Centre', vCostCentre);
sDim8Element = 'Amount';
```

The other attributes are iterations of the code and description combined or separate and useful for display purposes.

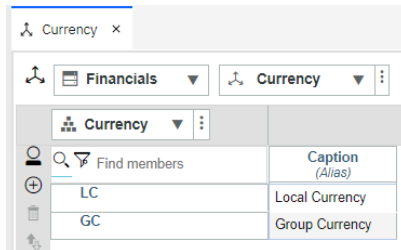
The process that builds the Cost Centre dimension is 'DIM_Cost Centre'. Here is the data source column layout:

	A	B	C	D	E
1	Child	Child Desc	Parent 1	Parent 2	Parent 3
2	CC0000	All Cost Centres			
3		101 Marketing	CC100		
4		102 Sales	CC100		
5		201 Finance	CC200		
6		202 IT	CC200		
7		203 HR	CC200		
8		204 Purchasing	CC200		
9		301 Warehouse	CC300		
10		302 Manufacturing	CC300		
11	CC100	Sales and Marketing	CC0000		
12	CC200	Administration	CC0000		
13	CC300	Production	CC0000		
14	No Cost Centre		CC0000		

- **Child:** reflects the member ID. It comprises a mixture of leaf level and consolidated level members. Every member in the hierarchy should appear in this column.
- **Child Desc:** the description of the member ID in the child column. This drives the caption.
- **Parent 1:** the parent of the member in the child column. Should be in member ID format. If it is a root member (e.g. All Cost Centres) it will not have a parent.
- **Parent 2:** optional second parent of the member in the child column. A member should not appear twice under the root member otherwise it will double up the numbers. Instead, parent 2 should be used as a way of creating an alternate roll up in the same hierarchy.
- **Parent 3:** similar to parent 2 but option for a third parent.

Currency

Contains...	Purpose
Contains two elements: local currency and group currency	Reporting figures for the local Currency of each company as well Group Currency.



The member ID is the code, and caption is an alias containing the description. This dimension is maintained manually via the dimension editor. Local currency is where all the Trial Balance figures are imported into. Group currency reflects the local currency values converted into Group Currency. There are no rules that populate Group Currency but it is intended to be used when currency translation rules are implemented.

Measures Financials

Contains...	Purpose
The measures used in the Financials cube including figures in accounting signage (Amount), and in reporting signage (Reporting Amount). Also contains 'Comment' used for adding commentary to figures	P&L and Balance Sheet reporting in accounting and reporting signage, and adding commentary to figures.

Find members	Format (Text)	Caption (Alias)
Reporting Amount	d,##0,	Reporting Signage
Amount	d,##0,	Accounting Signage
Comment		

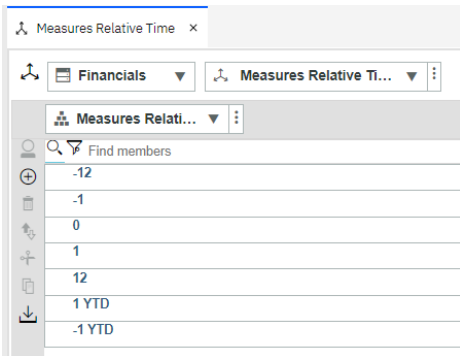
This dimension is maintained manually via the dimension editor. The caption displays a slightly different name to distinguish more clearly the purpose of 'Amount' and 'Reporting Amount'. Both are formatted to zero decimal places. But, of course, this could be changed. The figures are stored against 'Amount'. Reporting Amount is rule-calculated and converts the signage of 'Amount' into a signage more suited to reporting. This is because figures within a trial balance are often in a different signage to that used in reporting. The attribute 'Accounting to Reporting Signage MF' in the 'Account Type' dimension is used to calculate the 'Reporting Amount'. Here is an extract of the script that calculates Reporting Amount:

```
# Financials.2.001 - Reporting Amount
# Logic: Amount * Accounting to Reporting Signage MF (Multiplication Factor) - attribute on Account Type dimension
['Reporting Amount']=C:
    ['Amount']*ATTRN('Account Type',ATTRS('Account',!Account,'Account Type'),'Accounting to Reporting Signage MF');
```

The 'Reporting Amount' is a parent of 'Amount' because that way it avoids having to feed the calculation which improves the performance and efficiency of the model. As a result, the rule is a C level rule.

Measures Relative Time

Contains...	Purpose
The measures used in the Relative Time cube. Each measure represents a time period relative to a list of periods. E.g. -1 means 1 period ago, 12 means 12 periods into the future etc.	Assist in relative time rules



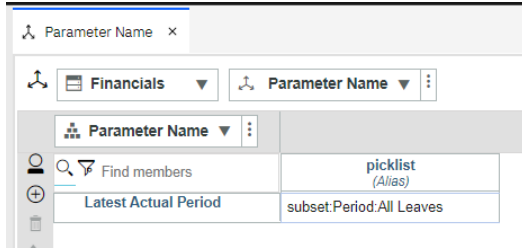
For context, the relative time cube looks as follows:

Period	-12	-1	0	1	12	1 YTD	-1 YTD
▼ All Years							
▼ 2029							
▼ 2029H2							
▼ 2029Q4							
202912			202912	203001	203012		
▼ 2030							
▼ 2030H1							
▼ 2030Q1							
203001		202912	203001	203002	203101	203002	
203002		203001	203002	203003	203102	203003	203001
203003		203002	203003	203004	203103	203004	203002
▼ 2030Q2							
203004		203003	203004	203005	203104	203005	203003
203005		203004	203005	203006	203105	203006	203004
203006		203005	203006	203007	203106	203007	203005
▼ 2030H2							
▼ 2030Q3							
203007		203006	203007	203008	203107	203008	203006
203008		203007	203008	203009	203108	203009	203007
203009		203008	203009	203010	203109	203010	203008
▼ 2030Q4							
203010		203009	203010	203011	203110	203011	203009
203011		203010	203011	203012	203111	203012	203010
203012	202912	203011	203012	203101	203112		203011

This dimension is maintained manually via the dimension editor. The members have a string element type. 1 YTD and -1 YTD are similar to 1 and -1 but the last period is blank in 1 YTD and the first period is blank in -1 YTD. This ensures that the P&L balances reset every period 1 and that the feeders don't feed into the next financial year.

Parameter Name

Contains...	Purpose
Contains a list of parameters used throughout the model to drive how the model works (e.g. Latest Actual Period). The element type (text or numeric) drives whether it's a string or numeric parameter	Used in the Parameter cube to assist in rules that drive central changes throughout the model

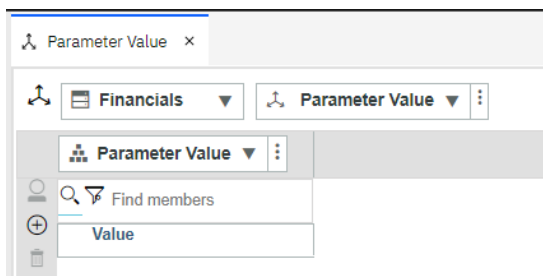


There is currently only 1 parameter in Parameter cube, and that is 'Latest Actual Period'. This is a string element type. Attribute 'picklist' is used to ensure that the value is selected from a picklist. The picklist represents the leaf level of the Period dimension. It should be the member ID only and not the caption because it is used in rules in the Financials cube that rely on it being a numeric naming convention. The Parameter Name dimension is intended to include additional global parameters in the future if the model is customized with additional functionality.

This dimension is maintained manually via the dimension editor.

Parameter Value

Contains...	Purpose
Contains one member 'Value' which is numeric. Parameter values in the parameter name cube are held against this member	Used in the Parameter cube to provide a second dimension which is required for building a cube

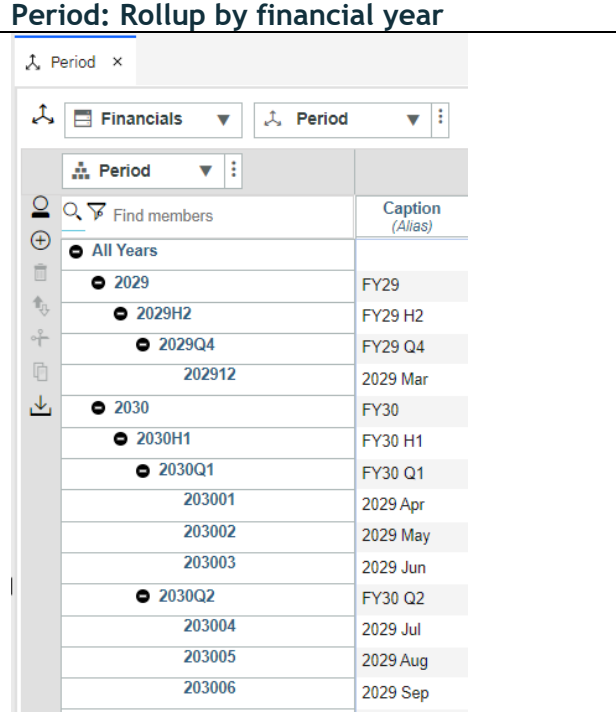
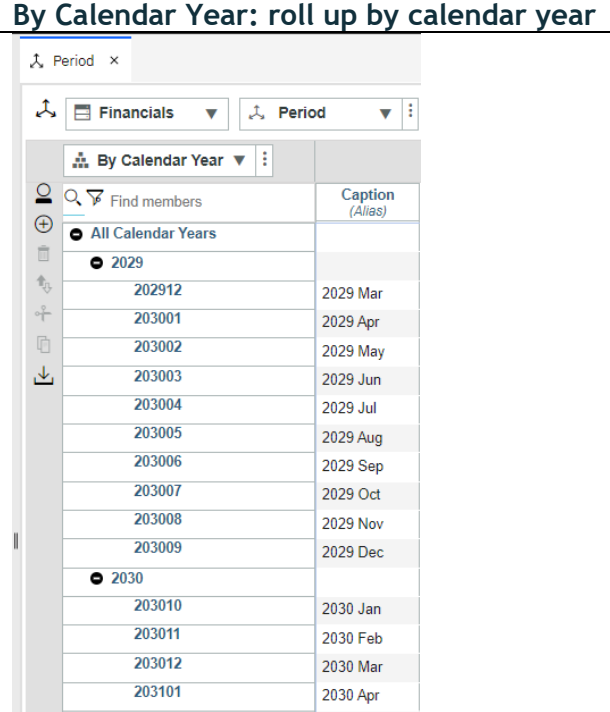


This dimension is maintained manually via the dimension editor.

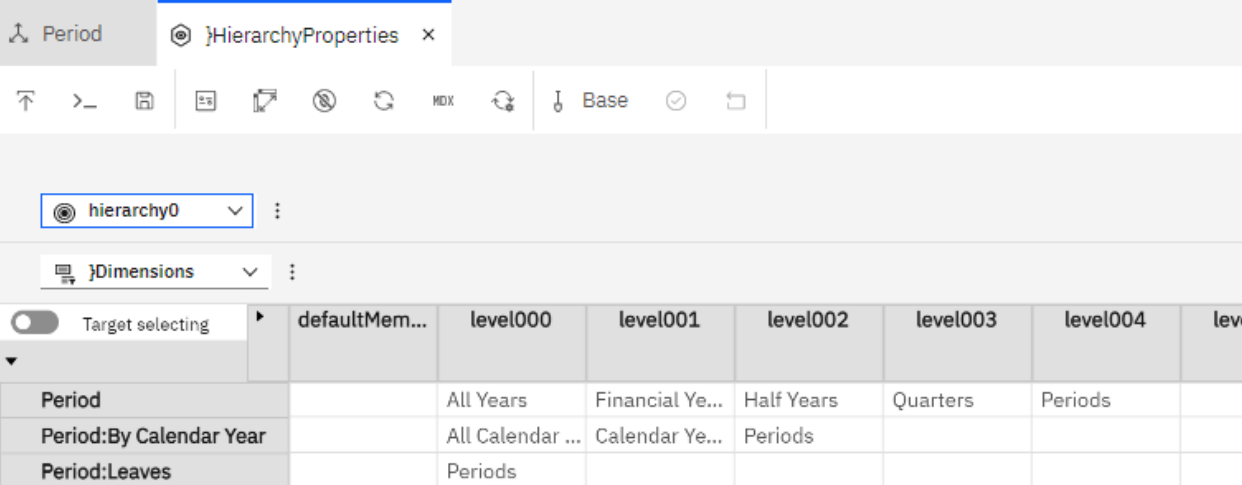
Period

Contains...	Purpose
List of periods in a continuous time format. Two hierarchies, one for Financial Year, and one for Calendar Year. Financial year roll up also includes half year and quarter	Used in cubes that split data by period.

The two hierarchies in this dimension appears as follows:

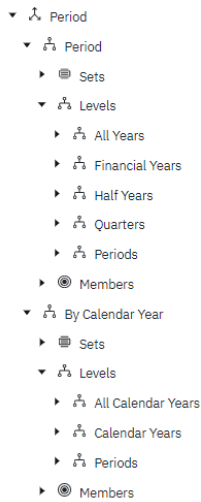
Period: Rollup by financial year	By Calendar Year: roll up by calendar year
	
Levels: All years > financial year > half year > quarter > period	Levels: All calendar years > calendar year > period

The level names for this cube have been set in the }HierarchyProperties control cube



	defaultMem...	level000	level001	level002	level003	level004	lev
Period		All Years	Financial Ye...	Half Years	Quarters	Periods	
Period:By Calendar Year		All Calendar ...	Calendar Ye...	Periods			
Period:Leaves		Periods					

This is what drives the level names in the Database pane and in Cognos Analytics, if used.



The period dimension should start with period 12 of the year prior to the model needs to start from. E.g. if the model starts in year 2030, it should include period 202912 as the opening period. This period 12 will receipt the opening balances otherwise the balance sheet balances will not be correct.

The dimension is updated from the process: DIM_Period. Here is the data source column layout:

	A	B	C	D	E	F	G	H	I	J
1	Period	Quarter	Half	Fin Year	Cal Year	Month Na	Period De	Qtr Desc	Half Desc	Year Desc
2	202912	2029Q4	2029H2	2029	2029	Mar	2029 Mar	FY29 Q4	FY29 H2	FY29
3	203001	2030Q1	2030H1	2030	2029	Apr	2029 Apr	FY30 Q1	FY30 H1	FY30
4	203002	2030Q1	2030H1	2030	2029	May	2029 May	FY30 Q1	FY30 H1	FY30
5	203003	2030Q1	2030H1	2030	2029	Jun	2029 Jun	FY30 Q1	FY30 H1	FY30
6	203004	2030Q2	2030H1	2030	2029	Jul	2029 Jul	FY30 Q2	FY30 H1	FY30
7	203005	2030Q2	2030H1	2030	2029	Aug	2029 Aug	FY30 Q2	FY30 H1	FY30
8	203006	2030Q2	2030H1	2030	2029	Sep	2029 Sep	FY30 Q2	FY30 H1	FY30
9	203007	2030Q3	2030H2	2030	2029	Oct	2029 Oct	FY30 Q3	FY30 H2	FY30
10	203008	2030Q3	2030H2	2030	2029	Nov	2029 Nov	FY30 Q3	FY30 H2	FY30
11	203009	2030Q3	2030H2	2030	2029	Dec	2029 Dec	FY30 Q3	FY30 H2	FY30
12	203010	2030Q4	2030H2	2030	2030	Jan	2030 Jan	FY30 Q4	FY30 H2	FY30
13	203011	2030Q4	2030H2	2030	2030	Feb	2030 Feb	FY30 Q4	FY30 H2	FY30
14	203012	2030Q4	2030H2	2030	2030	Mar	2030 Mar	FY30 Q4	FY30 H2	FY30

The Excel version of this text file is stored within the model within File Manager. It is called 'Period dimension - with formulas.xlsx'. This has Excel formulas that speed up the creation of the data source. It can be downloaded from within Modeling Workbench and updated for the years and periods required by the model before saving as a txt.

File manager

Financials

<input type="checkbox"/>	Name
<input type="checkbox"/>	Account Dimension Import File.txt
<input type="checkbox"/>	Company Dimension for Wizard.txt
<input type="checkbox"/>	Company Dimension Import File.txt
<input type="checkbox"/>	Cost Centre Dimension Import File.txt
<input type="checkbox"/>	Financials Actuals 202912.txt
<input type="checkbox"/>	Financials Actuals 2030.txt
<input type="checkbox"/>	Period dimension - with formulas.xlsx
<input type="checkbox"/>	Period dimension Import File.txt
<input type="checkbox"/>	Relative Time Dimension Import File.txt
<input type="checkbox"/>	Template Dimension Import File.txt
<input type="checkbox"/>	Version Dimension Import File.txt

- **Period:** Represents the member ID of the periods. Based on Financial Year and Period number in YYYYPP format. Period number is not the calendar month number. Period number 1 reflects the start month of the financial year. YYYY is not the calendar year, it is the financial year. If the financial year spans over two financial years, it is often the year ending that is selected. E.g. 2030/31 would have financial year member ID of 2031. However, it is flexible to cater for financial year starting if that is the policy within the business. The naming convention should not deviate from the YYYYPP format. It is designed to be a numeric naming convention so that rules using Greater Than or Less Than formulae are compatible and easy to author.
- **Quarter:** Represents the member ID of the financial quarters. YYYYQQ format. YYYY should be financial year to keep consistency of naming convention throughout the hierarchy. QQ being Q1, Q2, Q3, Q4 which are each of the 4 quarters of the year.
- **Half:** Represents the member ID of the financial half years. YYYYHH format. YYYY should be financial year to keep consistency of naming convention throughout the hierarchy. HH being H1, H2 which are the two halves of the financial year.
- **Fin Year:** Represents the member ID of the financial years. YYYY format which is the financial year. Should be the same YYYY used in the Period, Half and Quarter columns.
- **Cal Year:** Represents the member ID and the caption of the calendar year. YYYY format. This may be a different year to Fin Year particularly if the financial year spans across two calendar years. Cal Year should represent the calendar year that the period falls into.
- **Month Name:** Used to form the Period Description. Represents the month of the period in 3 letter format. E.g. Jan, Feb, Mar.
- **Period Desc:** Represents the caption of the period.
- **Quarter Desc:** Represents the caption of the Financial Quarter.
- **Half Desc:** Represents the caption of the Financial Half Year.
- **Year Desc:** Represents the caption of the Financial Year.

Relative Time

Contains...	Purpose
Contains the members that split monthly, YTD, last year comparatives, etc. Also includes variances and variance %.	Used in cubes that require comparison of monthly numbers with other time periods relative to it.

	Balance Sheet Caption (Alias)	Caption (Alias)	Format (Text)
● Period LY Var %	Movement LY Var %		d#.00%
● Period LY Var	Movement LY Var		
Period	Movement		
Period LY	Movement LY		
● YTD LY Var %	Balance LY Var %		d#.00%
● YTD LY Var	Balance LY Var		
YTD	Balance		
YTD LY	Balance LY		

The member 'Period' stores the figures and represents monthly numbers across P&L and Balance Sheet. Monthly numbers are commonly referred to as 'Movement' in balance sheet. Similarly, YTD is a P&L concept and represents an aggregate of the monthly figures beginning at period 1 up to the period intersecting the YTD member within the same year. For Balance Sheet, the aggregated number does not reset every year like P&L. Instead, the balance is carried forward from year to year, which effectively represents all time to date. In Balance Sheet terms, this is known as the 'Balance'. YTD member is shared across P&L and Balance sheet calculated with a different formula for each. The 'Balance Sheet Caption' applies Balance Sheet terminology (i.e. Movement and Balance).

The caption attribute is left blank but could be populated to vary the name of the member ID (e.g. 'Period' could be changed to 'Monthly' via the caption).

The Variance %'s are formatted with as a percentage to 2 decimal places.

The variance members (Period LY Var, YTD LY Var) are calculated using weights. The weight of Period LY and YTD LY are set to -1 to ensure that the variance subtracts those numbers. This is why the variances appear as a parent. The variance %s are calculated using rules. However, they appear as a parent to avoid the need to apply a feeder because the components that make up the feeder are its children. Less feeders means better performance, database size and efficiency within the model design.

The dimension is built from a process called 'DIM_Relative Time'. The data source for this looks as follows:

	A	B	C	D	E	F	G	H	I	J
1	Child	Child Desc	Parent 1	Parent 2	Parent 3	Balance Sheet Caption	Weight Parent 1	Weight Parent 2	Weight Parent 3	
2	Period		Period LY Var			Movement				
3	Period LY		Period LY Var			Movement LY	-1			
4	Period LY Var		Period LY Var %			Movement LY Var	0			
5	Period LY Var %					Movement LY Var %				
6	YTD		YTD LY Var			Balance				
7	YTD LY		YTD LY Var			Balance LY	-1			
8	YTD LY Var		YTD LY Var %			Balance LY Var	0			
9	YTD LY Var %					Balance LY Var %				
10										
11										

- **Child:** reflects the member ID. It comprises a mixture of leaf level and consolidated level members. Every member in the hierarchy should appear in this column.
- **Child Desc:** the description of the member ID in the child column. This drives the caption attribute.
- **Parent 1:** the parent of the member in the child column. Should be in member ID format. If it is a root member (e.g. Period LY Var %) it will not have a parent.
- **Parent 2:** optional second parent of the member in the child column. A member should not appear twice under the root member otherwise it will double up the numbers. Instead, parent 2 should be used as a way of creating an alternate roll up in the same hierarchy.
- **Parent 3:** similar to parent 2 but option for a third parent.
- **Balance Sheet Caption:** terminology used for balance sheet reporting.
- **Weight Parent 1:** the weight of the member in the child column. It reflects the weight that is applied to drive the summation method for Parent 1.
- **Weight Parent 2:** the weight of the member in the child column. It reflects the weight that is applied to drive the summation method for Parent 2.
- **Weight Parent 3:** the weight of the member in the child column. It reflects the weight that is applied to drive the summation method for Parent 3.

Version

Contains...	Purpose
Contains members that split out actual and plan figures. Working Forecast contains a blend of actuals and plan. This is copied into one of the other planned members (e.g. Budget, Forecast 3+9 etc). Also includes variances and variance %.	Used in cubes that require comparison of actual vs planned numbers as well as input against a working forecast.

Find members	Short Caption (Text)	Caption (Alias)	Format (Text)
F3	3+9	3+9 Forecast	
F6	6+6	6+6 Forecast	
F9	9+3	9+3 Forecast	
● AC v BU %	Act v Bud %	Actual vs Budget Var %	d#.00%
● AC v BU	Act v Bud	Actual vs Budget Var	
AC	Act	Actual	
BU	Bud	Budget	
● BU v F0 %	Bud v Working FC %	Budget vs Working Forecast Var %	d#.00%
● BU v F0	Bud v Working FC	Budget vs Working Forecast	
BU	Bud	Budget	
F0	Working FC	Working Forecast	

The member ID is a code and the Caption is an attribute containing the member description. There is a second alias which is used for a shortened caption which is useful in reporting when column space is limited. The captions can be updated to reflect the terminology used by the business. The variance percentages are formatted as percentage to 2 decimal places.

The working forecast (F0) can be used as a method of inputting the latest forecast information for any future year or period at any point in time. Once a budgeting or forecasting cycle is complete, this can be copied as static data into one of the other planning versions such as F3, F6, F9 or BU and treated as the final signed off version available for reporting. There are currently 3 quarterly forecasts, but this could be adapted to include all other monthly forecasts to allow for a rolling forecast approach. Each forecast or budget can include as many periods as desired, e.g. F3 could include all 12 periods which are 3 months of actuals and 9 months of forecast. Or F3 could include 18 periods which are 3 months of actuals and 12 months of forecast. It would simply mean copying 12 or 18 months from F0 to F3 upon completion of the forecasting cycle.

Similar to the Relative Time dimension, the version variances (AC v BU, BU v F0) are calculated using weights. The version being subtracted has a weight of -1. That's why the variances are a parent of the components of its calculation. The variance %s are calculated using rules. However, they appear as a parent to avoid the need to apply a feeder because

the components that make up the feeder are its children. Less feeders means better performance, database size and efficiency within the model design.

F0 is a mixture of manual input and rules driven. The rules on F0:

- There is a rule displaying Actual up to and including the 'Latest Actual Period' defined in the parameter cube.
- Could later include additional rules for driver-based planning calculations if the model was extended. These rules would only apply to F0 as narrowing down the area of the rules helps with efficiency within the model. The results of the rules driven numbers would be copied as static data into the relevant version (e.g. F3, BU).

Processes

Process Name	Purpose	Input source & Output target	Template upload file
ADMIN_Clear Cube Data	Clears out all data within a cube, whichever cube is specified in the parameter.	None	None
ADMIN_RefreshMDX Hierarchy	Refresh updated level names in Cognos Analytics from those defined in the }HierarchyProperties cube	None	None
ADMIN_SaveDataAll	Commits data, held in memory, to disk. Unless this runs, the data imported/input into the cubes (and attributes) are not stored on disk. Data which is only held in memory and not stored on disk is not included in backups. Such data could also be lost if a server restart is performed. Data must be committed to disk on a regular basis to ensure the data is properly saved.	None	None
DATA_Financials (Import Trial Balance)	Imports monthly trial balance numbers into the Financials cube for a given Version, Period (or group of periods), and company (or group of companies) in local currency.	Input = text file. Output = Financials cube	Financials Actuals 202912.txt
DATA_Financials (Import Trial Balance) - ODBC	Same as the process 'DATA_Financials (Import Trial Balance)' but the data source is an ODBC table rather than a text file. Omits any parameter or scripting related to data source files.	Input = ODBC Output = Financials cube	
DATA_Relative Time	Automatically populates the Relative Time cube so that no manual population is required	Input = Period dimension, leaf level. Output = Relative Time cube	
DIM_Account	Populates the Account dimension. References the Account Type dimension to determine some of the attributes.	Input = text file. Output = Account dimension	Account Dimension Import File.txt
DIM_Company	Populates the Company dimension.	Input = text file. Output = Company dimension	Company Dimension Import File.txt
DIM_Company (Wizard)	Populates the Company dimension - was created using the wizard. Has a different text file template. In the	Input = text file. Output =	Company Dimension for Wizard.txt

	non-wizard version (above), the script concatenates columns to produce the attribute values. In the wizard, the concatenation is performed in the text file.	Company dimension	
DIM_Cost Centre	Populates the Cost Centre dimension	Input = text file. Output = Cost Centre dimension	Cost Centre Dimension Import File.txt
DIM_Period	Populates the Period dimension for all hierarchies.	Input = text file. Output = Period dimension	Period dimension Import File.txt
DIM_Relative Time	Populates the Relative Time dimension	Input = text file. Output = Relative Time dimension	Relative Time Dimension Import File.txt
DIM_Version	Populate the Version dimension	Input = text file. Output = Version dimension	Version Dimension Import File.txt
TEMPLATE_Create Dimension (From File)	A process that can be used to fast track creating a dimension. It should be opened, then saved as a different name e.g. DIM_Customer, and then update the value against pDimension on the Parameter tab to reflect the dimension that needs to be created (e.g. Customer). The data source file and scripting would need updating as appropriate.	Input = text file. Output = none. pDimension is intentionally left blank so that it doesn't populate a dimension if it was run accidently.	

Chores

Chore Name	Purpose	Frequency
Build all dimensions	Recreates all dimensions based on latest updates to the data source. Based on processes which use a text file upload. Currently used ad hoc. Many data sources will be updated manually and not on a regular basis. Therefore, it is more likely that individual processes will be run rather than all dimension building processes as a group. This could be adapted to include only those dimensions which come from a dynamically updating data source and scheduled to run daily or as required.	None set.
Import Trial Balance Daily	Runs the process 'DATA_Financials (Import Trial Balance) - ODBC' which imports data into a cube on a daily basis. Uses the ODBC import file with the assumption that the data source can dynamically update on a frequent basis. The period would need updating manually to the latest actual period or the solution could be customized to reflect more dynamic approach. The parameters need populating before this chore can be run.	Daily at 11pm. Not scheduled. If intended to be used, the 'scheduled' field would need to be activated.
Import Trial Balance Every 30 minutes	Similar to the above, except it runs on a more frequent basis - every 30 minutes. Intended to be scheduled only during busy periods such as month end when a frequently updated picture is desired. Scheduled should be inactive outside of that to avoid putting too much performance overhead on the system which may cause slow down.	15 minutes and 45 minutes past the hour. Caution needs to be applied whether activating this schedule would cause conflict with the daily chore. The daily chore could be deactivated when this is in use.
SaveDataAll Every 2 hours	Runs the 'ADMIN_SaveDataAll' process every 2 hours to recommit data regularly to disk. 2 hours is a suggested starting point for a frequency. The frequency should reflect how frequent data is updated in the database. So, this could be set to whatever frequency makes sense.	Every two hours starting at 11.30pm (i.e also runs at 2-hour intervals after that). Not scheduled. This should be set to scheduled upon first set up.

Rules

Financials Cube

Here is a table of contents of the rules in the Financials cube.

```
#####  
# CONTENTS  
#####  
##### 1. Rule Exclusions & Rules Precedence #####  
# Financials.1.002 - Don't calculate actuals beyond current month  
# Financials.1.001 - F0 Actuals Portion  
#  
##### 2. Measures #####  
# Financials.2.001 - Reporting Amount  
#  
##### 3. Relative Time #####  
# Financials.3.001 - Period LY  
# Financials.3.002 - Period LY Var %  
# Financials.3.003 - YTD LY  
# Financials.3.004 - YTD LY Var %  
# Financials.3.005 - YTD (or Balance)  
#  
##### 4. Version #####  
# Financials.4.001 - AC v BU %  
# Financials.4.002 - BU v F0 %  
#  
#####
```

The rules are grouped into 4 sections.

- **1. Rule Exclusions & Rules Precedence:** The first section is reserved for rules that need to appear at the top so that they take precedence over other rules that conflict with it. This also includes defining areas of the cube that will preclude rules.
- **2. Measures:** This covers rules applied to the measures dimension.
- **3. Relative Time:** This covers rules applied to the relative time dimension.
- **4. Version:** this covers rules applied to the version dimension.

Each section has a rule ID (e.g. Financials.1.002) comprised of the cube name, the section number and a sequential number which represents the order in which it the rule was added to that section. Note: the ID numbers don't need to be in alphanumeric order. This is just an ID number that can be used to uniquely identify a rule and its corresponding feeder (which share the same ID). Example below shows a rule and corresponding feeder with the same rule ID:

Rule:

```
# Financials.3.001 - Period LY  
# Logic: Fetch same value last year from 'Period Element'. Use -12 period reference in Relative Time cube. Only apply  
to Actual and Working FC as others don't make sense to compare plan against last years plan.  
['Period LY', {'AC', 'F0'}]=N:  
DB('Financials', 'Period', DB('Relative Time', !Period, '-12'), !Version, !Currency, !Account, !Company, !'Cost  
Centre', !'Measures Financials' );
```

Feeder:

```
# Financials.3.001 - Period LY  
['Period', {'AC', 'F0'}]=>  
DB('Financials', 'Period LY', DB('Relative Time', !Period, '12'), !Version, !Currency, !Account, !Company, !'Cost  
Centre', !'Measures Financials' );
```

Each rule has an unhashed section below the rule ID called 'Logic' that describes in business language what the rule is trying to achieve. This is pitched at an audience that do not understand rule scripting and gives them an idea of how the rule works.

Relative Time Cube

There are no rules on the Relative Time cube. It is populated purely from the 'DATA_Relative Time' process.

Parameter Cube

There are no rules on the Parameter cube. It is populated purely from manual data entry.

Updating the Model with Your Own Data

Summary

Prior to importing new data into the model, the dummy data within the Financials cube will need clearing out. To update the model with your own data, the data source text files will need to be updated and then run the processes to rebuild the dimensions and import the data. If using 'DATA_Financials (Import Trial Balance) - ODBC' then there are additional steps to create a data source table that aligns with the data source template columns required. A data source connection will have to be established first, then the process will need to be updated with ODBC connection detail and revised SQL query. After that, the process can be run to import the data.

Once dimensions and financial data have been updated imported, the data can be reconciled against the month end accounts and the appropriate reports can be created.

Task List

Actions: loading text files

- ✓ Clear existing dummy data in the Financials cube
- ✓ Download the data source text files from File Manager
- ✓ Populate the files with your own data where applicable
- ✓ Upload the new data source text files to File Manager
- ✓ Run the processes to import the data
- ✓ Review imported figures within Planning Analytics
- ✓ Create reports

Actions: loading external database table (trial balance) via ODBC (if required)

- ✓ Establish a connection in Planning Analytics to the external data source /database
- ✓ Create database table
- ✓ Populate the database table with data
- ✓ Update process ODBC connection details and database query
- ✓ Run the processes to import the data
- ✓ Schedule the chores to import the Financials data, if relevant

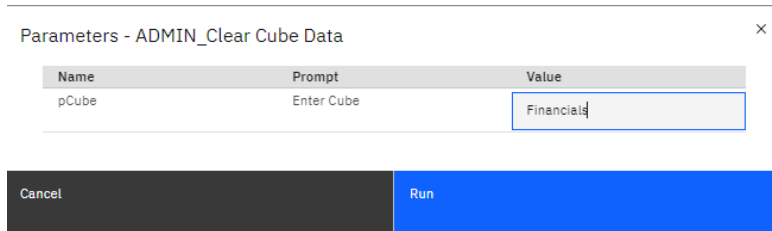
Actions: additional tasks

- ✓ Re-populate the Relative Time cube
- ✓ Schedule the Chore to SaveDataAll
- ✓ Decommission any processes not used to prevent erroneous database updates

Step-by-Step walk through

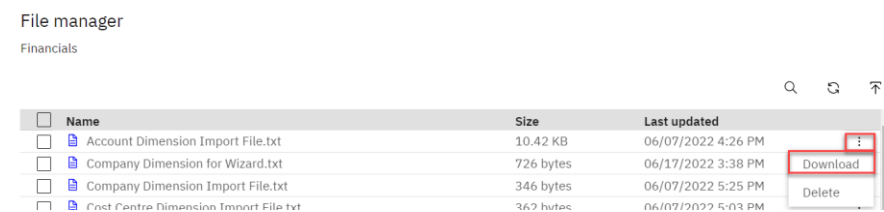
Clear existing dummy data in the Financials cube

- Execute the 'ADMIN_Cube Cube Data' process and put 'Financials' into the pCube parameter



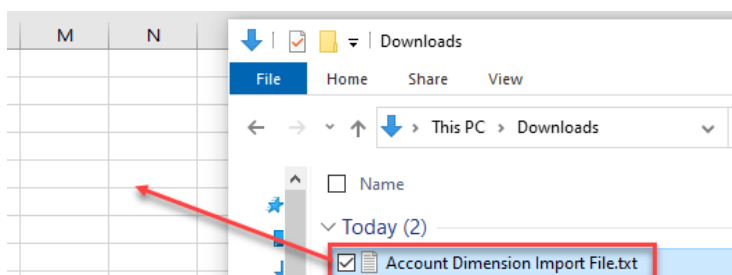
Download the data source text files from File Manager

- Log into PAW > open a Modeling Workbench
- Right click on the Financials database > File Manager
- Click on the tree dots next to the file you want to populate. Click download



Populate the files with your own data where applicable

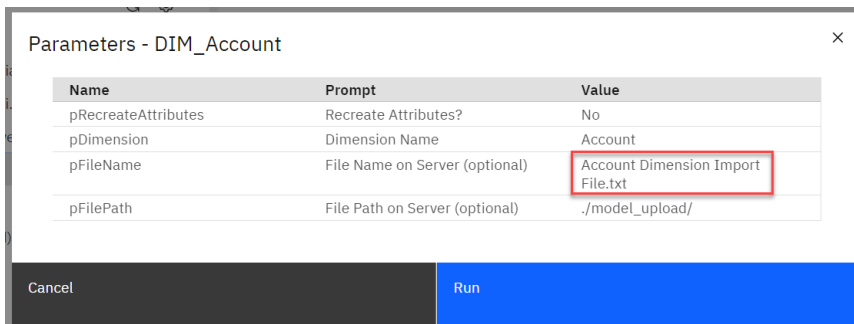
- Open Excel and select a blank workbook
- Click and drag the downloaded file from File Explorer onto the Excel workbook



- The file should open in Excel and be available to edit and replace with your own data structures.

	A	B	C	D	E	F	G
1	Child	Child Desc	Parent 1	Parent 2	Parent 3	Account Type	Rate Type
2	B00000	Trial Balance				A	B
3	B10000	Goodwill	B10099			A	B
4	B10010	Trade-marks & licenses	B10099			A	B
5	B10020	Software licences	B10099			A	B
6	B10030	Other intangibles	B10099			A	B
7	B10099	Intangible Assets	B19099			A	B
8	B11000	Land and buildings	B11099			A	B
9	B11010	Plant & machinery	B11099			A	B
10	B11020	Motor vehicles	B11099			A	B

- When saving the file, it does not matter if the file has the same name as the original or not because when running the process you can override the default file on specified on the data source tab.



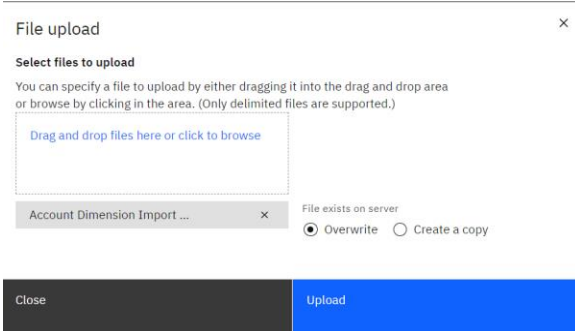
- At a minimum, the files which are expected to be populated with your own data include:
 - o **Account dimension:** Account Dimension Import File.txt
 - o **Company dimension:** Either 'Company Dimension Import File.txt' for DIM_Company process or 'Company Dimension for Wizard.txt' for DIM_Company (Wizard) process
 - o **Centre dimension:** Cost Centre Dimension Import File.txt
 - o **Period dimension:** Period dimension Import File.txt (use 'Period dimension - with formulas.xlsx' to quickly update the template before saving as a txt file)
 - o **Financials actuals:** Financials Actuals 202912.txt - if uploading text file and not using ODBC equivalent process.
 - Note: You can upload an entire year or all years in the same file, split by period. But you must specify which consolidated member represents the sum of the periods contained in the file in the process parameter.
- The other dimensions include default options which are generic and may be sufficient as a starting point. But in the future, these dimensions may be adapted to extend and customize the design, however further rules would be required to support this:
 - o **Relative Time dimension:** Relative Time Dimension Import File.txt
 - o **Version dimension:** Version Dimension Import File.txt

Upload the new data source text files to File Manager

- Log into PAW > open a Modeling Workbench
- Right click on the Financials database > File Manager
- Click the upload button



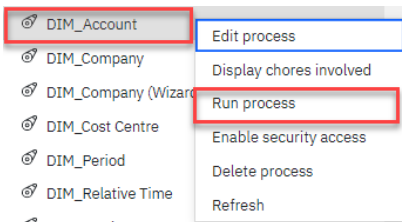
- Select the file to upload. If the same named file already exists, you can choose to overwrite or create a copy. It is advisable to store a copy of the original files somewhere if you intend to overwrite.



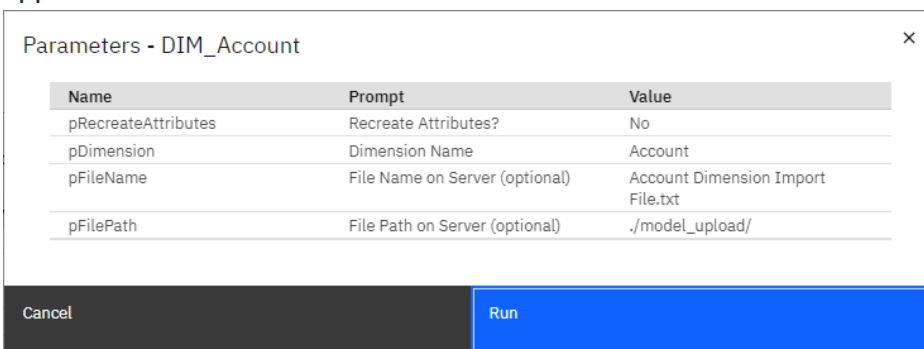
- Click Upload then Close.

Run the processes to import the data (text files)

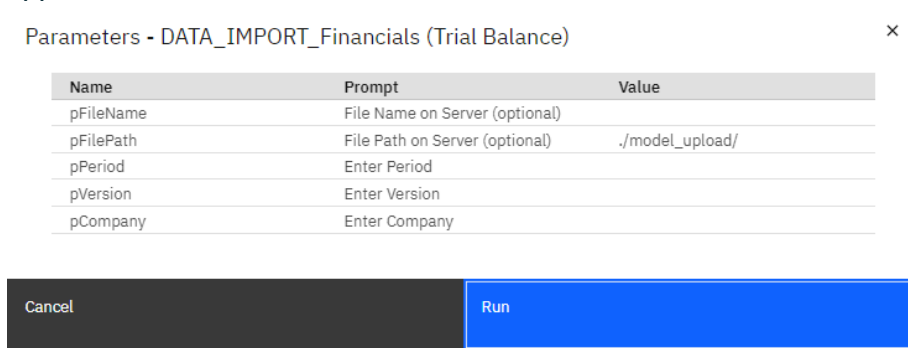
- Log into PAW > open a Modeling Workbench
- Expand the Financials database > expand Processes
- Right click on the process which relates to the text file you want to import > Run process.



- See processes section of this document to locate which upload file relates to which process.
- At a minimum, the processes which are expected to execute to import your own data include:
 - o **Account dimension:** DIM_Account
 - o **Company dimension:** DIM_Company or DIM_Company (Wizard)
 - o **Cost Centre dimension:** DIM_Cost Centre
 - o **Period dimension:** DIM_Period
 - o **Financials Actuals:** DATA_Financials (Import Trial Balance) - if uploading text file and not using ODBC equivalent process
- When running processes which import the dimensions prompt page like this will appear:



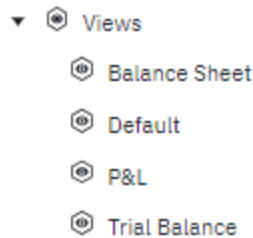
- The parameters can be populated as follows:
 - o **pRecreateAttributes:** setting this to 'yes' will delete and recreate the attributes prior to populating them. Setting this to 'no' will update the attribute values without deleting them first. Typically, this could be set to 'Yes' when a caption is assigned to one member ID and it needs to be assigned to a different member ID. Without setting this to 'yes' the process may error saying that the attribute values are not unique.
 - o **pDimension:** Leave the default value as is
 - o **pFileName:** Make sure the file name reflects the one you are wanting to load. Include the file extension. File extension should be txt.
 - o **pFilePath:** By uploading the file into File Manager, the file automatically goes into model_upload folder. This could be updated if you have dropped the file into a different location within the file share. The single dot at the start reflects the 'data' folder where the model objects are stored. Putting two dots at the start reflects a level above the data folder (i.e. at the same level as the data folder).
- When running the process to import Financials actuals, a prompt page like this will appear:



- Open the dimensions that have been updated and check the results. The processes can be run multiple times with a revised data source to achieve the dimension design required.
- The parameters can be populated as follows:
 - o **pPeriod, pVersion, pCompany:** you can specify the member ID or member alias/caption (e.g. AC or Actual). You can also specify a consolidated member or a leaf level member.
 - o **pFileName:** Make sure the file name reflects the one you are wanting to load. Include the file extension.
 - o **pFilePath:** By uploading the file into File Manager, the file automatically goes into model_upload folder. This could be updated if you have dropped the file into a different location within the file share.

Review imported figures within Planning Analytics

- Open a view on the Financials cube and check the results. The processes can be run multiple times with a revised data source to achieve the figures required.
- There are 4 predefined views which can assist with analysis:



- o **Balance Sheet:** Shows balance sheet lines by period. Shows Balances. Uses 'Balance Sheet' set which is dynamically created from the Account dimension where the 'BS or PL' attribute is 'BS'. Excludes leaf level and root level members.
 - o **P&L:** Shows P&L lines by period. Shows period movements. Uses 'P&L' set which is dynamically created from the Account dimension where the 'BS or PL' attribute is 'P&L'.
 - o **Trial Balance:** Shows all leaf level accounts plus the 'Trial Balance' root member by period. Shows period movements. Uses 'Trial Balance' set which is dynamically created from the Account dimension from the Trial Balance root member and its leaf level members.
 - o **Default:** a cube view that can be customized to reflect a commonly used view of the data. It is a data source within the process 'DATA_Financials (Copy Working Forecast)' but its not used in the process as a temporary view is assigned to it upon run time. However its used to produce the variables required for the process.
- The figures in the cube may not be aligned with the data source if the following occurs:
 - o Errors existed on the processes which prevented some data source lines from importing. Review any error messages and resolve.
 - o There are some members in the dimension which don't sit under the root member e.g. a cost centre that doesn't roll up into the 'All Cost Centres' total. Review the dimension to analyse if there are any orphaned members and correct the parentage.
 - o Some members appear in a roll up more than once which incorrectly multiplies the numbers. E.g. one cost centre appears twice under the same 'All Centres' root element. Review the dimension to analyse if there are duplicate members in one roll up and correct the parentage.

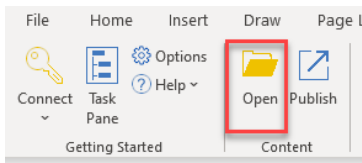
Create Reports

- Install Planning Analytics for Excel (PAfE) by downloading it from Planning Analytics Workspace > Administration > Excel and Customizations.
- Follow the installation guide on the same page.
- There is an existing PAfE report 'Profit and Loss' in the Reports folder

The screenshot shows the 'Profit and Loss' report in the Planning Analytics for Excel interface. The report is displayed in an Excel spreadsheet format with columns for 'MONTH' and 'YEAR TO DATE'. The data is as follows:

	MONTH						YEAR TO DATE							
	Actual	Budget	vs Budget	Forecast	vs Forecast	Prior Yr	Actual	Budget	vs Budget	Forecast	vs Forecast	Prior Yr	vs Prior Yr	
Revenue	500,000	0	500,000	500,000	-500,000	0	500,000	1,910,000	0	1,910,000	1,910,000	-1,910,000	0	1,910,000
Cost of sales	-200,000	0	-200,000	-200,000	200,000	0	-200,000	-568,000	0	-568,000	-568,000	568,000	0	-568,000
Offset P/L	0	0	0	0	0	0	0	-20,000	0	-20,000	-20,000	20,000	0	-20,000
Gross profit	300,000	0	300,000	300,000	-300,000	0	300,000	1,322,000	0	1,322,000	1,322,000	-1,322,000	0	1,322,000
Administrative expenses	0	0	0	0	0	0	20,000	0	20,000	20,000	-20,000	0	20,000	
Profit from operations	300,000	0	300,000	300,000	-300,000	0	300,000	1,342,000	0	1,342,000	1,342,000	-1,342,000	0	1,342,000
Translation difference	0	0	0	0	0	0	40,000	0	40,000	40,000	-40,000	0	40,000	
Share of profits - equity accounted associates	0	0	0	0	0	0	0	0	0	0	0	0	0	
Share of loss - equity accounted associates	0	0	0	0	0	0	40,000	0	40,000	40,000	-40,000	0	40,000	
Profit before tax	300,000	0	300,000	300,000	-300,000	0	300,000	1,422,000	0	1,422,000	1,422,000	-1,422,000	0	1,422,000
Profit from continuing operations	300,000	0	300,000	300,000	-300,000	0	300,000	1,422,000	0	1,422,000	1,422,000	-1,422,000	0	1,422,000
Profit/(loss) before minority share	300,000	0	300,000	300,000	-300,000	0	300,000	1,422,000	0	1,422,000	1,422,000	-1,422,000	0	1,422,000
Minority share on Profit/(loss)	-30,000	0	-30,000	-30,000	30,000	0	-30,000	-298,000	0	-298,000	-298,000	298,000	0	-298,000
Profit	270,000	0	270,000	270,000	-270,000	0	270,000	1,124,000	0	1,124,000	1,124,000	-1,124,000	0	1,124,000

- It is likely the report will be blank or error once your own data is imported into the model because the periods, accounts, and company structure, which feature in the report design, will have changed. But it can be used as a starting point for speeding up creating your own report, if desired.
- To open this report, go to 'open' on the PAfE ribbon. Navigate to Financials > Reports > Profit and Loss > Open.



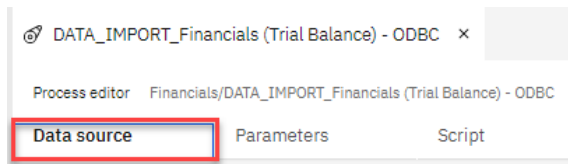
Establish a connection in Planning Analytics to the data source (where relevant)

- Check if the database you connecting to is supported by IBM.
 - o This link contains the default data source connections (ODBC drivers) available: <https://www.ibm.com/docs/en/planning-analytics/2.0.0?topic=connections-default-data-source>
 - o This link contains a list of additional ODBC drivers that can be supplied to IBM along with details or how to request additional ones that are listed: <https://www.ibm.com/docs/en/planning-analytics/2.0.0?topic=connections-customer-provided-data-sources-odbc-drivers>

- Where the secure gateway client is required, review its documentation: <https://www.ibm.com/docs/en/planning-analytics/2.0.0?topic=workspace-administer-secure-gateway-cloud-only>
- Using the link (and sub-links) above:
 - o Create a secure gateway
 - o Install the gateway client
 - o Add a data source to a secure gateway
 - o Use the secure gateway

Create database table

- Log into PAW > open a Modeling Workbench > expand the Financials database > expand Processes
- The process 'DATA_Financials (Import Trial Balance) - ODBC' is exactly the same as the process 'DATA_Financials (Import Trial Balance)' except it uses an ODBC connection instead of an upload file. That means the column layout of the data source is identical between both processes.
- Right click on the process 'DATA_Financials (Import Trial Balance) - ODBC' > Select 'Edit Process' > Click on the 'Data source' tab



- The variables section contains the Name and Type of variables that should be in the data source and in which order they should appear.

Preview title	Sample data	Variable name	Type
		vPeriod	String
		vCompany	String
		vAccount	String
		vCostCentre	String
		vAmount	Numeric

- Construct a table in the external database that conforms to this design. The table name can be different to the one used in the process because later the query will be updated.

Populate the database table with data

- Populate the table containing the Trial Balance actuals.
- For simplicity, it could just contain the data pertaining to the period which is being loaded. That way the SQL query will be straight forward.
- If you wish the table to include all years data and want the process to filter the data for the period required, the SQL query and process will need to be customized to

parameterize the period being queried - tying that to the period defined on the parameter tab. Here is an example of the script customized to filter by the period defined in pPeriod:

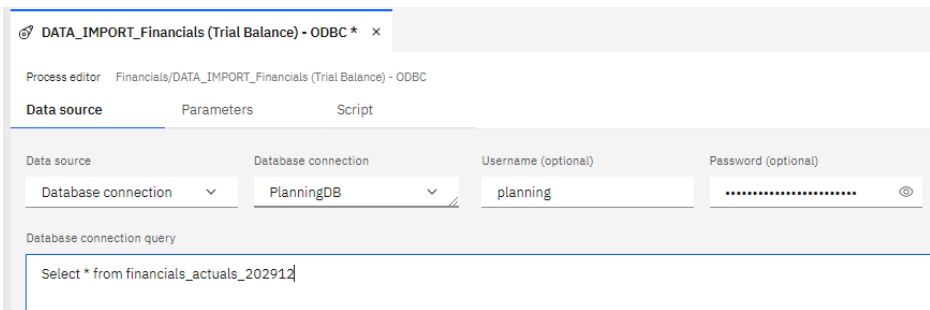
```
Database connection query

Select * from financials_actuals_202912
where period = '?pPeriod?'
```

This does mean that pPeriod could only be populated with a period in YYYYPP format as that is the format used in the database tables. If you choose not to apply this customization, i.e. the table only contains the data required to upload, then you can specify an aggregate of periods to load e.g. a year, a half, a quarter.

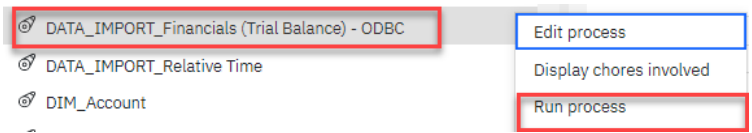
Update process ODBC connection details and database query

- Go to Modeling Workbench > expand Financials database > expand processes
- Right click on the process 'DATA_Financials (Import Trial Balance) - ODBC' > Select 'Edit Process' > Click on the 'Data source' tab
- Update the following:
 - o **Database connection:** select the new data source that you set up.
 - o **Username and password:** either update them on this screen or centrally in the 'ODBC settings' menu. [Learn more](#)
 - o **Database connection query:** update the query as appropriate referencing the table name created earlier.



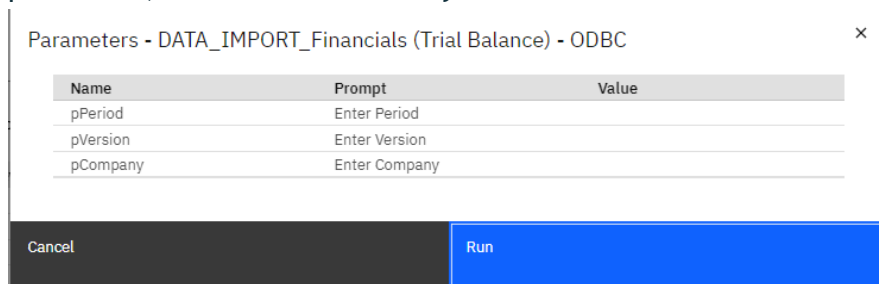
Run the processes to import the data (ODBC)

- Go to Modeling Workbench > expand Financials database > expand processes.
- Right click on the process 'DATA_Financials (Import Trial Balance) - ODBC' > Run process



- Before it executes, the parameter page will appear. Enter the period, version and company. You can specify member ID or the member alias/caption (e.g. AC or Actual)

will be sufficient. You can also specify a consolidated member or a leaf level member. However, as mentioned previously, if you've customized the SQL to filter by a parameter, it will need to be only leaf level members.



- Open a view on the Financials cube and check the results. The process can be run multiple times with a revised data source to achieve the figures required.

Schedule the chores to import the Financials data, if relevant

- The process 'DATA_Financials (Import Trial Balance) - ODBC' can be scheduled to run using the chores that are already built. This can be useful if the external table automatically updates with new transactions every day or regularly throughout the day, and visibility of the latest data is required within Planning Analytics on a regular basis.
- There are two chores which can be scheduled which contain the same process but have a different frequency:
 - o Import Trial Balance Daily
 - o Import Trial Balance Every 30 minutes.
- More information on these chores can be found in the Chores section of this document
- If they are required, edit the chores > adapt the frequency and time if desired and then active the schedule.

Re-populate the Relative Time cube

- After the Period dimension has been updated, the relative time cube will need to be re-populated.
- Go to Modeling Workbench > expand Financials database > expand processes > execute the process 'DATA_Relative Time'
- Open a view on the Relative Time cube to check the results.

Schedule the Chore to SaveDataAll

- Now that new data has been populated into various cubes (Financials, Relative Time, attribute cubes) the data stored in memory needs to be committed to the server to ensure that it is included in backups and will not lose the data if the server is restarted during a maintenance window.
- Run the process now to commit the data and then schedule the chore which runs this process regularly.

- Go to Modeling Workbench > expand Financials database > expand processes > execute the process 'ADMIN_SaveDataAll'
- Right click on the chore 'SaveDataAll Every 2 hours' > Adapt the frequency and time if desired and then active the schedule

Decommission any processes not used to prevent erroneous database updates

- Some of the processes in the model may not be required going forwards. For example, if 'DIM_Company' is used to update the Company dimension, then 'DIM_Company (Wizard)' will not be required. In fact, 'DIM_Company (Wizard)' may still be pointing to the template file containing dummy companies. In this example, running the wizard process may revert the changes you made to the Company dimension and incorrectly delete data along with it.
- To avoid any confusion, it is worth deleting or archiving processes which aren't required. If you know you will never use the process, then delete may sufficient. But if you may use it in the future, it could be worth storing a copy of the process. For example, in the first phase you may be uploading the trial balance via a text file but have plans to implement an ODBC approach much later.
- To archive processes:
 - Run the ADMIN_SaveDataAll process
 - Stop the database
 - Navigate to the data folder behind the database.
 - Locate the appropriate redundant process (has .pro file extension) and cut and paste it into a different location outside the data folder where it can be easily retrieved when needed.
 - Start the database

Suggestions for building upon the design

The model can be extended and development to add different capabilities to the solution. This would involve building additional objects within the model (i.e. cubes, processes, dimensions, rules) and collecting additional data. Some potential ways in which it can be extended include:

- Adding more dimensions to the Financials cube
- Adding a front-end application design using Apps and Plans and workflow
- Adding currency conversion
- Adding in driver-based planning for integrated business planning
- Adding Cash Flow reporting and planning
- Ability to add journal adjustments
- Ability to map codes using in local finance systems to a set of group codes used in dimensions within Planning Analytics
- Validation of data prior to loading it into the Financials cube - prevents loading of data under certain criteria

Each of these areas will be explored in more depth to give some ideas on how to get started.

Adding more dimensions to the Financials cube

During the initial build, it may be required to change the dimensionality of the cube to add additional dimensions specific to the business using it. One example of a typical additional dimension to the Financials cube might be 'Project'. This is best done before the trial balance is loaded, so be sure to determine what dimensionality is required upfront at the start of the model development. To add extra dimensions to the cube, here is a guide on how this can be done

- Within the data folder copy the 'Financials.rux' file and the 'Financials}vues' folder. This represents the rules file behind the Financials cube and the Views on the Financials cube. These will be restored later after the cube has had the dimension added
- Create the new dimension(s) to be added to the Financials cube
- Delete Financials cube
- Recreate the Financials cube, adding in the extra dimension. It is advised that you do not put the new dimension last as the 'Measures Financials' dimension should be last to allow string input against the 'Comment' element. **Keep the dimensional order largely the same order as before to minimize the amount of rework on scripting.** See 'Cube Design' section for a screenshot of existing dimension order,
- Run the ADMIN_SaveDataAll process to commit data to disk as the next step involves stopping the database.

- Stop the database. Paste the 'Financials.rux' file and the 'Financials}vues' back into the data folder. Say 'yes' to overwriting any existing objects with the same name. Start the database again.
- Open the Financials.rux file and update any rules that specify the dimension order so it includes the additional dimension. For example, here is an existing rule specifying each of the dimensions in the Financials cube in the order they appear in the cube:

```
# Financials.3.001 - Period LY
# Logic: Fetch same value last year from 'Period Element'. Use -12 period reference in Relative Time cube. Only apply to Actual and Working FC as others
['Period LY',{!AC,!F0}]=N:
  DB('Financials', 'Period', DB('Relative Time',!Period,-12'), !Version, !Currency, !Account, !Company, !'Cost Centre', !'Measures Financials' );
```

Now there is a new dimension, this script would need updating to include a new argument for the extra dimension. If Project dimension was added after the Cost Centre dimension, the latter part of the script would read: *!'Cost Centre', !Project, !'Measures Financials')*.

- Update the data source files in File Manager or the ODBC table to include the new dimension.
- Open up the following processes:
 - o DATA_Financials (Import Trial Balance)
 - o DATA_Financials (Import Trial Balance) - ODBC
- Refresh the data source on the data source tab so the new data source column comes through. Rename the variables to align with the new order as appropriate.
- Update the script in the 'Data' section to include the new dimension. This is the script to edit:

```
#Section Data
1 #****Begin: Generated Statements***
2 #****End: Generated Statements****
3
4 #####
5 # IMPORT DATA
6 #####
7
8 # Declare elements of cell intersection to import into
9
10 sDim1Element = 'Period';
11 sDim2Element = vPeriod;
12 sDim3Element = pVersion;
13 sDim4Element = 'LC';
14 sDim5Element = vAccount;
15 sDim6Element = vCompany;
16 sDim7Element = IF(vCostCentre@='', 'No Cost Centre', vCostCentre);
17 sDim8Element = 'Amount';
18
19 nValue = vAmount;
20
21 # Validate if can import into the cell
22
23 IF(CellIsUpdateable( sCubeTarget, sDim1Element, sDim2Element, sDim3Element, sDim4Element, sDim5Element, sDim6Element, sDim7Element, sDim8Element )=1);
24
25 # Import the data into the cube
26
27 CellIncrementN( nValue, sCubeTarget, sDim1Element, sDim2Element, sDim3Element, sDim4Element, sDim5Element, sDim6Element, sDim7Element, sDim8Element );
28
29
30 ENDIF;
```

For example, if Project dimension was added in the position after the Cost Centre dimension, the revised script would appear as follows:

```

#Section Data
1 #***Begin: Generated Statements***
2 #***End: Generated Statements***
3
4 #####
5 # IMPORT DATA
6 #####
7
8 # Declare elements of cell intersection to import into
9
10 sDim1Element = 'Period';
11 sDim2Element = vPeriod;
12 sDim3Element = pVersion;
13 sDim4Element = 'LC';
14 sDim5Element = vAccount;
15 sDim6Element = vCompany;
16 sDim7Element = IF(vCostCentre='', 'No Cost Centre', vCostCentre);
17 sDim8Element = vProject;
18 sDim9Element = 'Amount';
19
20 nValue = vAmount;
21
22 # Validate if can import into the cell
23
24 IF(CellsIsUpdateable( sCubeTarget, sDim1Element, sDim2Element, sDim3Element, sDim4Element, sDim5Element, sDim6Element, sDim7Element, sDim8Element, sDim9Element )=1);
25
26 # Import the data into the cube
27
28 CellIncrementN( nValue, sCubeTarget, sDim1Element, sDim2Element, sDim3Element, sDim4Element, sDim5Element, sDim6Element, sDim7Element, sDim8Element, sDim9Element );
29
30
31 ENDF;
~

```

- Save the processes with the new updates and close.
- Open up the cube views and sense check if any layout changes need to be applied. Save the views if any are updated.

Front end application in Apps and Plans

Apps and Plans let you organize Planning Analytics content (books, views, websheets) into containers which address a particular business process. These containers can either be an ‘Application’ or a ‘Plan’ and are given a name that describes what business process it fulfils.

- **Application:** can group books into sections. Each section may have a description against it describing what this section does. Different sections can be hidden from specific audiences where the content isn’t relevant.
- **Plan:** can group books into steps. Each step may have a description against it describing what this section does. It is intended to be a workflow such as a budgeting or forecasting process. Each step may have a start and end date, dependencies and ability to ‘submit’ each step for approval and rejection.

Learn more about Apps and plans here: <https://www.ibm.com/docs/en/planning-analytics/2.0.0?topic=workspace-use-applications-plans-organize-work>

Suggested apps and plans include:

- Plan called ‘**Financials Actuals**’. Purpose: Importing the trial balance and reconcile the actuals data ready for month end reporting.
- Application called ‘**Financials Administration**’. Purpose: Updating dimensions, updating central parameters and other administrative tasks such as SaveDataAll, assigning security, publishing the Working Forecast into a static plan so its available in reporting.
- Plan called ‘**P&L Plan**’. Purpose: Updating the planning data for the profit and loss
- Plan called ‘**Balance Sheet Planning**’. Purpose: Updating the planning data for the Balance Sheet.

Adding currency conversion

Currency conversion rules need adding in order to determine the Group Currency calculation. Group currency member is located in the 'Currency' dimension within the Financials cube.

This would involve the following:

- Creating a Currency cube which holds FX rates by rate type, version and period.
- Aligning the rate types with those defined against each account in the Account dimension ('Rate Type' attribute).
- Putting on currency conversion rules for 'Group Currency' member in the Financials cube. This needs to account for:
 - o Differences in Currency conversion between P&L and Balance Sheet. The Balance Sheet balances often converts at closing rate. The movement is difference between balances across this month and prior month.
 - o Accounts which are converted at historic rate. These are group currency values that are explicitly defined rather than calculated.
 - o Accounts which use a blended rate which is gathered from conversion of supplementary information.
 - o Accounts which calculate the forex difference, ensuring the group currency trial balance adds up to zero
 - o Accounts which carry forward the balance from the prior year, such as retained earnings.
- Loading Group Currency values, rather than calculating them, for the opening balances. Opening balances are stored in the first period in the Period dimension. In the default model this is period 202912. For example, P&L Group Currency YTD in period 12 can often be an aggregation of 12 periods of group currency values, each one converted at a different rate. As a result, the opening balance group currency can't be calculated using a rate against local currency. Instead, it's aggregation of data converted at different rates. Therefore, it is best to load the group currency values for the opening period and the remaining periods would be rules calculated.

Adding in driver-based planning for integrated business planning

The financials plan is intended to be collected in the Working Forecast (F0) against each of the P&L and Balance Sheet accounts. However, some of the planning data which is inputted comes from detailed workings to derive the figure which needs to be input. These workings may be designed within a spreadsheet and then copy and pasted into the appropriate P&L or Balance Sheet code. However, the model could be extended to replicate the detailed workings in Planning Analytics within different cubes that sit alongside the Financials cube. These driver-based (detailed workings) cubes would receipt input of detailed data and calculate the final result required for the financials plan. The final result could be

dynamically linked to the appropriate GL account in the Financials cube. This means, as soon as driver-based assumptions have been updated, the P&L and Balance Sheet plan are automatically updated in line with it.

Typical driver-based additions may include:

- Staff costs
- Creditors
- Debtors
- Stock
- Gross Margin
- Overheads

Adding Cash Flow reporting and planning

The model covers P&L and Balance Sheet, but this could be extended to also cover Cash Flow providing an integrated reporting and planning solution where all three are inter-connected. This would involve mapping the relevant P&L and Balance Sheet lines into a Cash Flow structure alongside additional information that isn't in the Trial Balance such as additions and disposals.

The main additional cubes required could be as follows:

- **Cash Flow:** contains the cash flow actuals and plan, mapping each of the lines to P&L, Balance Sheet or Supplementary Information. Contains movements and balance by month. Allows input of non-cash adjustments on cash flow each line.
- **Cash Flow Supplementary Info:** collects supplementary information used in the Cash Flow statement. This information provides a more detailed breakdown of one of the accounts the Trial Balance. The information collected must reconcile back to the movements on the account it relates to. Contains actuals and plan and reconciliation differences. Reconciliations include:
 - o Reconciling calculated net cash total against the Cash on the Balance Sheet
 - o Reconciling the supplementary information to the GL code it is linked to.

Ability to add journal adjustments

The Financials cube is made up of a mixture of different sources and adjustments. In the main this comprises:

- **Trial Balance** from financial ledger which includes journals.
- **Journal adjustments** outside the financial ledger to align with different reporting views such as group reporting, US GAAP reporting, IFRS reporting, local reporting etc.

The latter can be collected within the model and layered over the top of the Trial Balance from the finance ledger to produce the different reporting views required. This would mean having different reporting layers that aggregate up into different subtotals.

These different reporting layers and their components could be split out within the Measures Financials dimension.

The main additional cubes required could be as follows:

- **Journals Entry:** A cube that is used to collect detailed journal information at line level detail. The “data” collected in the cube would be a mixture of string and numeric including Account, Cost Centre, Amount, Journal line comment, for a given period, company and journal number.

The information above would be collected and posted into the Financials cube against the relevant journal layer.

Ability to map codes using in local finance systems to a set of group codes used in dimensions within Planning Analytics

Financial reporting and planning often means collecting information from various companies within a group. Each of these businesses might supply their financial information from differing source systems/ERP systems. Each of those source systems may use different codes and these must be aggregate into a group chart of accounts.

Typically, the group chart of accounts uses a different set of coding convention (e.g. Account code) to those found in any source system to allow it to be system agnostic. This future-proofs the solution when a company changes ERP system or merges/acquires another business that uses a system with different account codes. Often the Group Accounts will be at a summarized level than that found in a GL system.

If the Account dimension is designed with a group chart of accounts, then the local chart of accounts being imported will need a mapping into it. To facilitate this mapping, the main additional cubes required could be as follows:

- **Local Mapping Entry:** Data entry cube for entering a local code and its corresponding group code at line level detail. The “data” collected would be string elements for the entering the local and group codes, for a given dimension (e.g. account, cost centre)
- **Local Mapping:** Takes data from the ‘Local Mapping Entry’ cube. Contains the same information but instead of Local code being entered as data, they would feature as elements in a dimension. This allows processes to perform a lookup against the local code to fetch the group code, using a much easier method than against a data entry cube

Validation of data prior to loading it into the Financials cube - prevents loading of data under certain criteria

When any data is uploaded into the Financials cube, there are many conditions under which a business wouldn't want the data imported - to give the opportunity to clean/fix the data first or prevent importing data accidentally. This allows data integrity and shortens the process to complete the accounts. Typical reasons to prevent data being imported:

- **Trial balance doesn't sum to zero:** A Trial Balance typically adds up to zero. A condition could be put in place to prevent loading data into the Financials cube if it didn't sum to zero or fell outside a certain tolerance.
- **Period is locked:** Once a financial period is closed, often businesses want to lock the figures to prevent them from changing. A condition could be put in place to prevent uploading the data if the period being uploaded related to a locked period.
- **Missing Cost Centres:** Many profit and loss accounts are split by cost centre, but sometimes the source data can be blank. In these cases, a condition could be put in place to prevent loading the data if the cost centre is blank. This forces the user to correct the source data before its loaded.