

Hyperion Focus 17

Our Cost Allocation Journey

Simon Watts

M&G Ltd



Hyperion Focus 17

- **Overview and Introduction**
 - Our journey so far
- Bring on the SQL Allocations
- Summary/Business Benefits
 - Questions

Hyperion Focus 17

- Overview and Introduction
 - **Our journey so far**
- Bring on the SQL Allocations
- Summary/Business Benefits
 - Questions

Hyperion Focus 17

In 2009 we embarked on our Metify replacement project



Implemented Prior to 2002



2009 to 2011

Hyperion Focus 17

HPCM was chosen to do the cost allocations



MAS1



MAS2R



MAS2NR



1MCSTAL



2MCSTAL

11.1.1.3 to 11.1.1.4 (2010-2013)

Upgraded to 11.1.2.2 (2014)

Hyperion Focus 17

The introduction of Rolling Forecasting

	FY14												FY15												FY16												FY17												
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	
Jan Forecast	█																																																
Feb Forecast		█																																															
Mar Forecast			█																																														
Apr Forecast				█																																													
Budget (Jul-Sep)						█	█	█					█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	

	FY14												FY15												FY16												FY17												
	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	
Jan RF	█																																																
Feb RF		█																																															
Mar RF			█																																														
Apr RF				█																																													
Jul RF						█																																											
Aug RF							█																																										
Sep RF								█																																									

Hyperion Focus 17

HPCM pushed to its limits

```
'#####  
'# submit POV Genealogy without (the potential clashing) ASO clear (on retry onl  
'#####  
If bPOVCleared Then 'only submit if the clearing of POV was successful (it can  
'loading operations)  
| strHPMRun = strHPCMBatchClient & " processGenealogyPathsWithOutASOCubeClear "
```

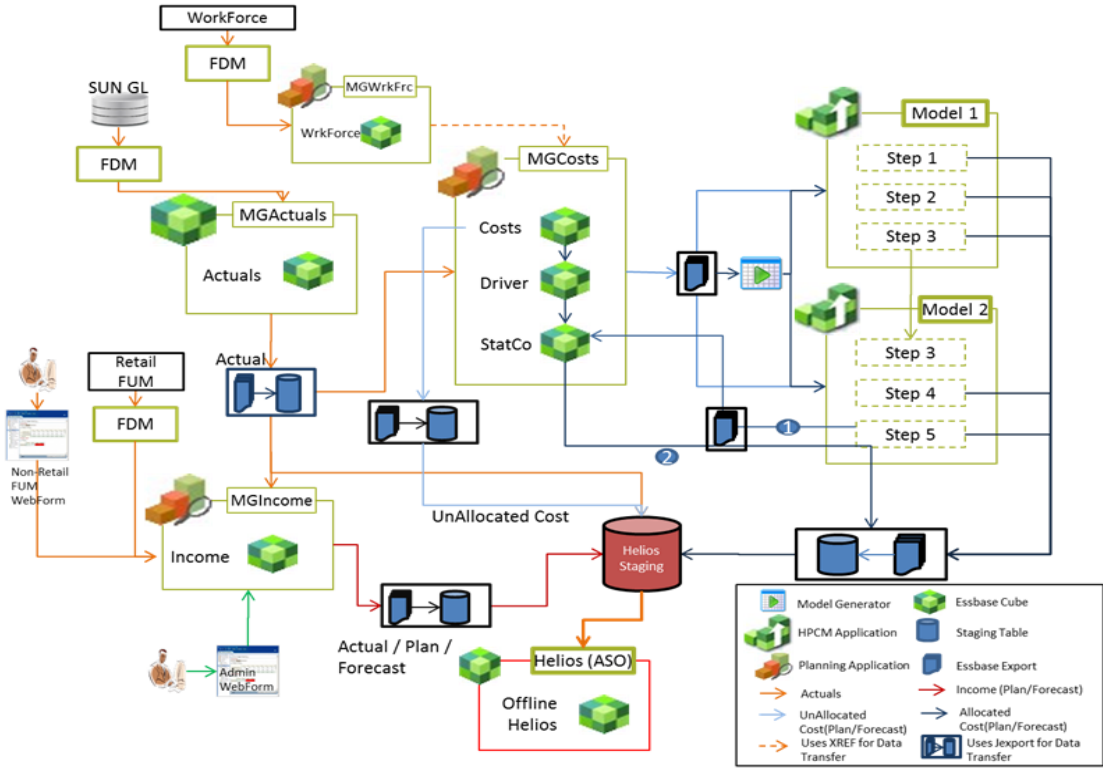
```
-----  
Got HPCM_CLIENT.done.COST.FY18.May : STATUS=SUCCESS  
##### Waiting 1min to check for HPCM_CLIENT.done.* (23 Completed, 1 Running)  
##### Waiting 1min to check for HPCM_CLIENT.done.* (23 Completed, 1 Running)  
##### Waiting 1min to check for HPCM_CLIENT.done.* (23 Completed, 1 Running)  
Got HPCM_CLIENT.done.COST.FY18.Nov : STATUS=SUCCESS  
##### Waiting 1min to check for HPCM_CLIENT.done.* (24 Completed, 0 Running)  
CS_AllocationCalcASO finished OK
```

Hyperion Focus 17

- Overview and Introduction
 - Our journey so far
- **Bring on the SQL Allocations**
- Summary/Business Benefits
 - Questions

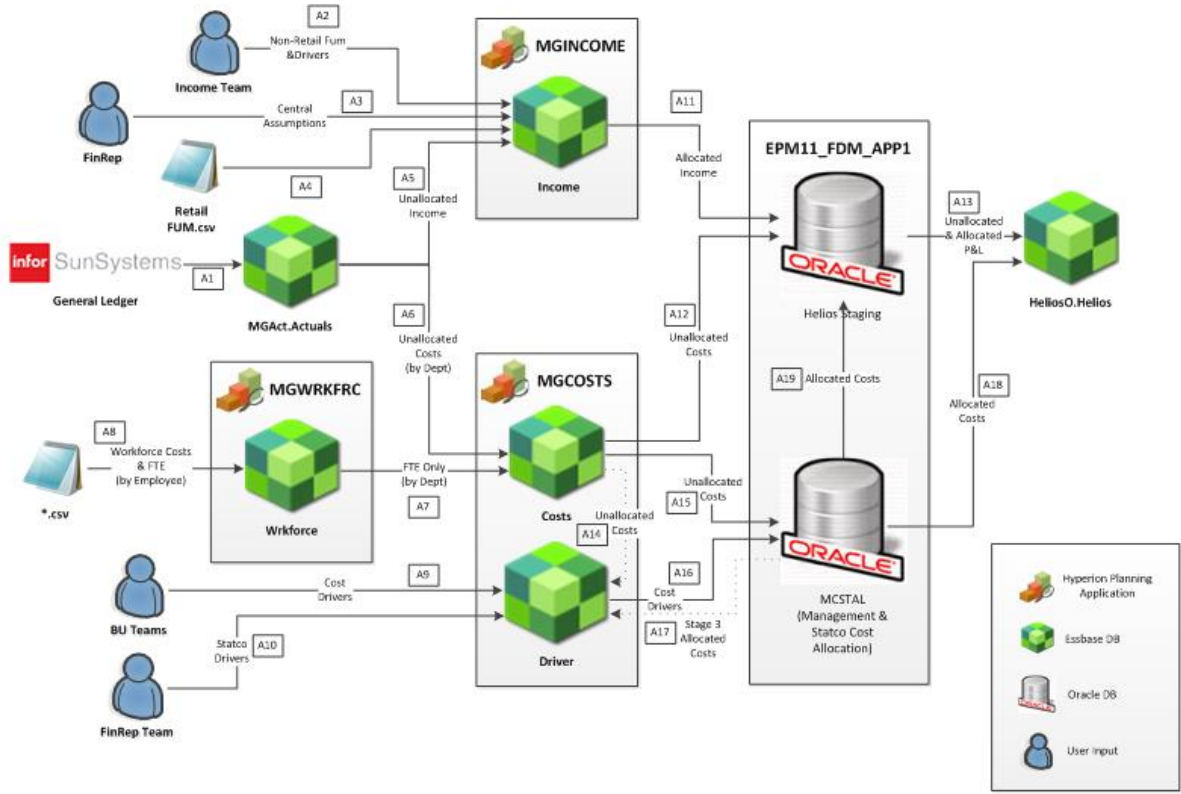
Hyperion Focus 17

System architecture before



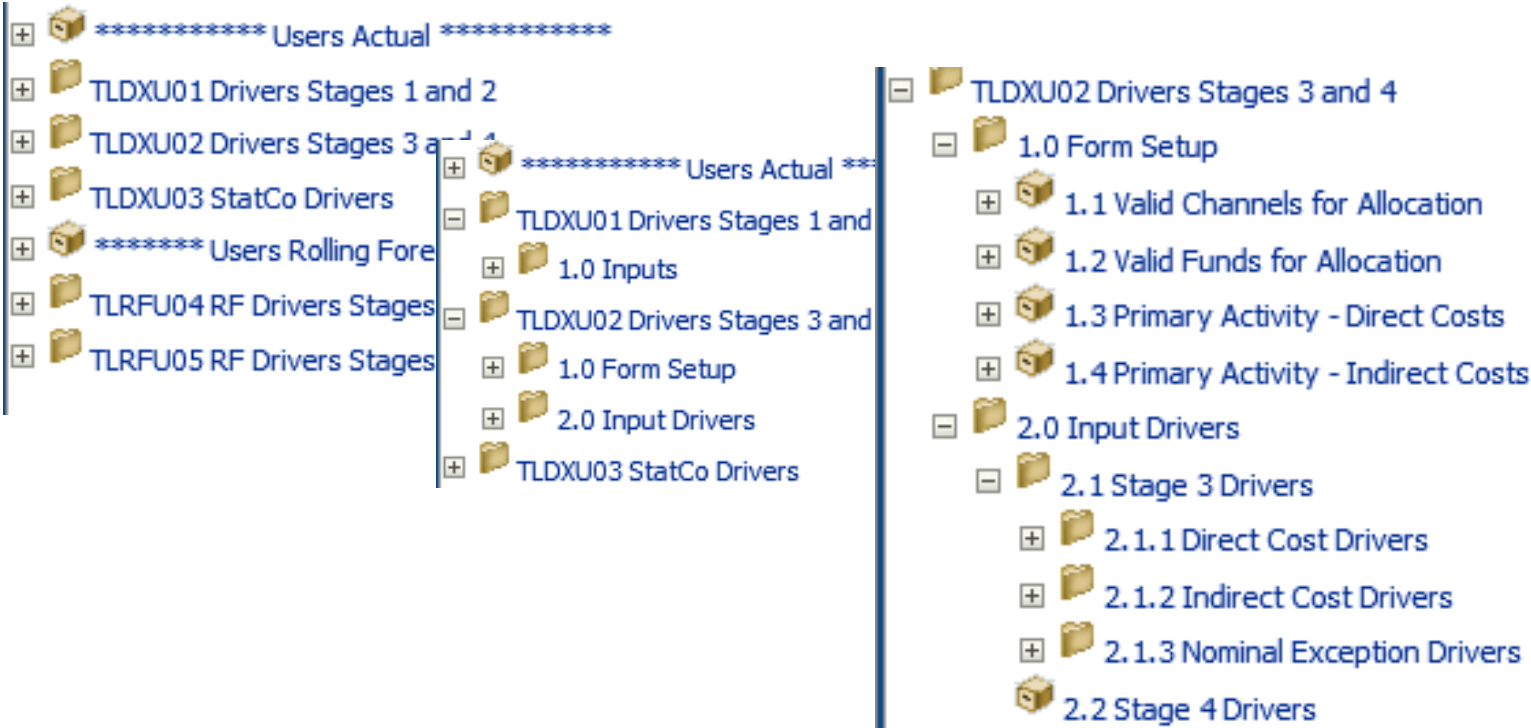
Hyperion Focus 17

System architecture after



Hyperion Focus 17

The art of driver capturing



Hyperion Focus 17

Essbase calculations creating drivers

	Direct Channel	Direct Fund	Direct Validation
Retail Base	Primary ▼	None ▼	OK
M & G RE Product Development	None ▼	Primary ▼	OK
M & G RE Client Services & Reporting	None ▼	Primary ▼	OK
M & G RE Marketing & Communications	None ▼	None ▼	OK
M & G RE Business Development	None ▼	None ▼	OK
M & G RE Placement Agent Fees	None ▼	None ▼	OK

```

#-----CHANNEL FIRST-----
Generate Cost Centre Drivers for InvOps and CntMgmServ based on Default and Exception drivers*/
FIX ( BegBalance, &FDM_RA_YEAR, NoView, NoStatCo, DataValue, NoFund, NoAccount, C_NoDepartment)
  FIX ( @Children( BusUnit_Base_Departments))

```

```

  FIX ( "S3_Exception", &CurrVer)
    AGG ( Channel);
  ENDFIX

```

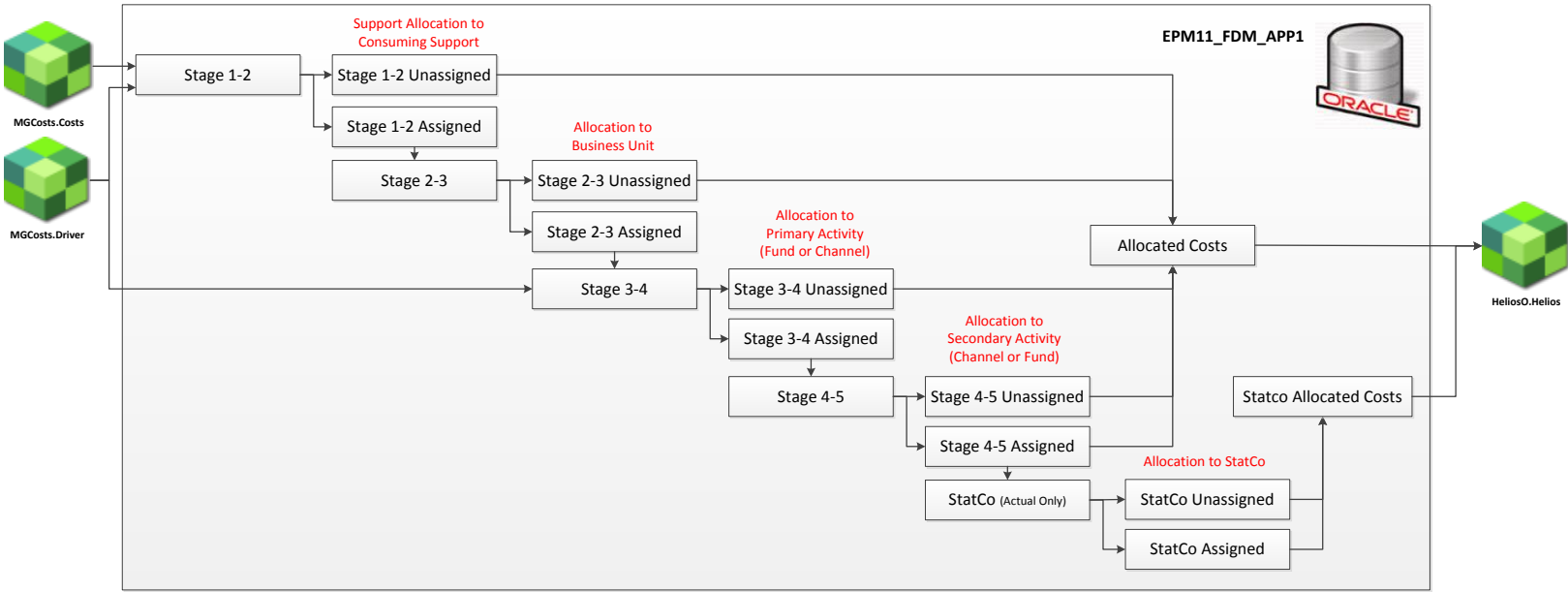
```

  FIX ( @Children( "S_InvOps_Departments"), @Relative( AllChannels, 0))
    &CurrVer (
      IF ( "S3_Exception" -> AllChannels != #MISSING AND "S3_Exception" -> AllChannels != 0)
        "S3_SourceSupp_Exception" = "S3_Exception";
      ELSE
        "S3_SourceSupp_Exception" = "S3_Default" -> "S_InvOps_BASE";
      ENDF
    )
  ENDFIX

```

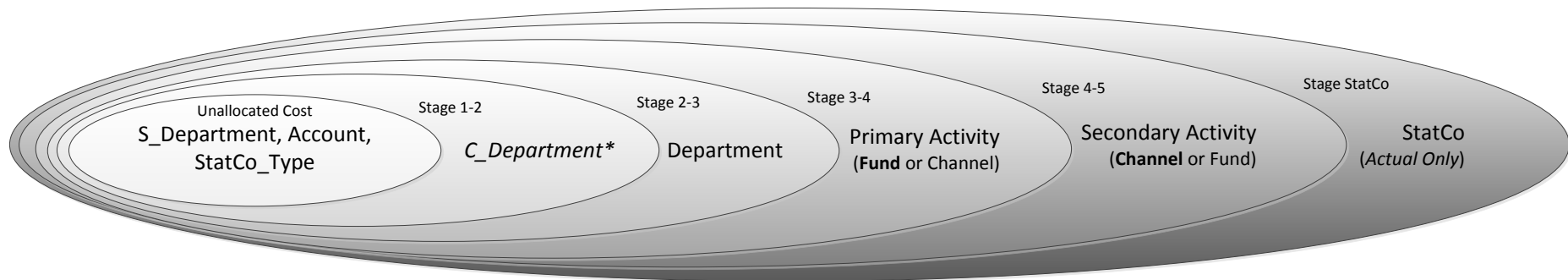
Hyperion Focus 17

Data flow through the views



Hyperion Focus 17

Data flow through the views



Hyperion Focus 17

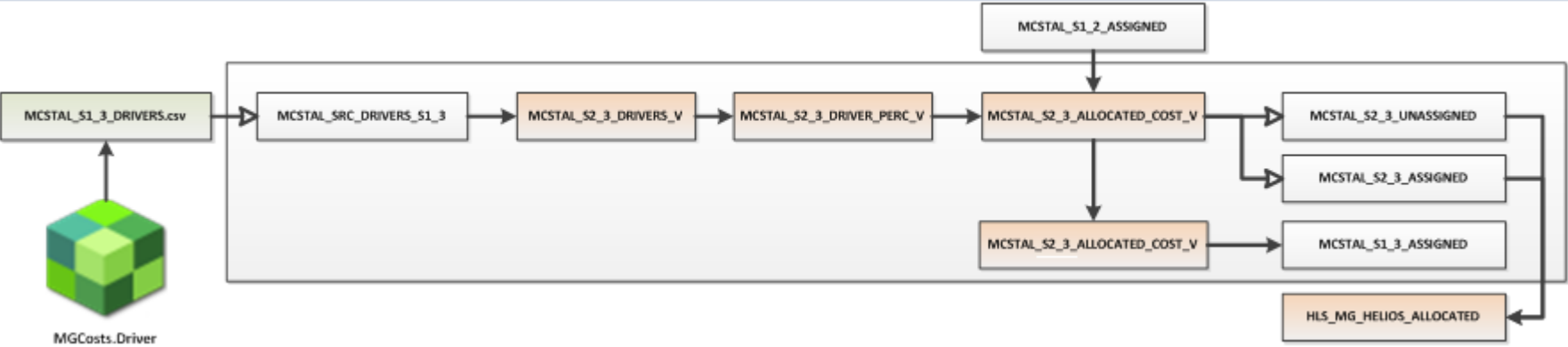
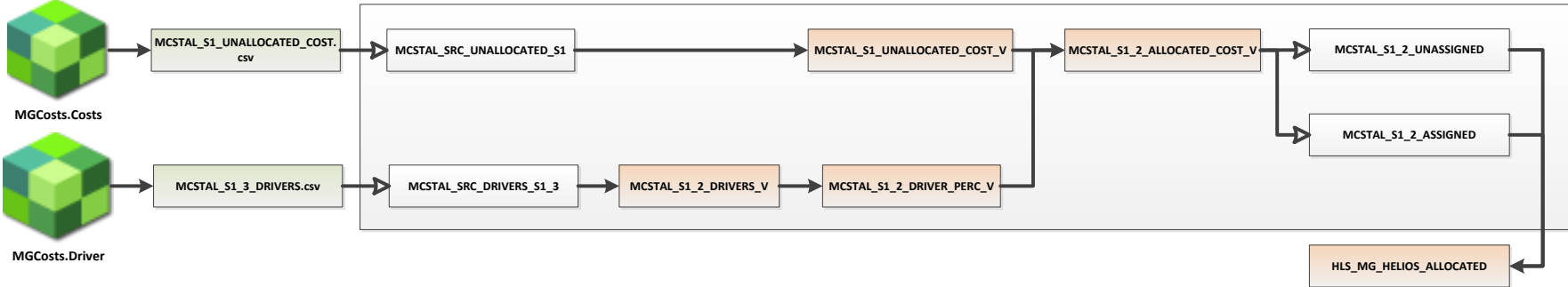
Custom subroutines to run the process

```
'#####  
'##### SQL Allocation #####  
Call RUNSQLAllocationProcess("SQL-MCSTAL", "fdjxDI")  
'#####
```

```
'#####  
'##### SQL Statco #####  
'Run StatCo Allocation only for Actuals  
Call RunSQLStatCo("SQL-STATCO")
```

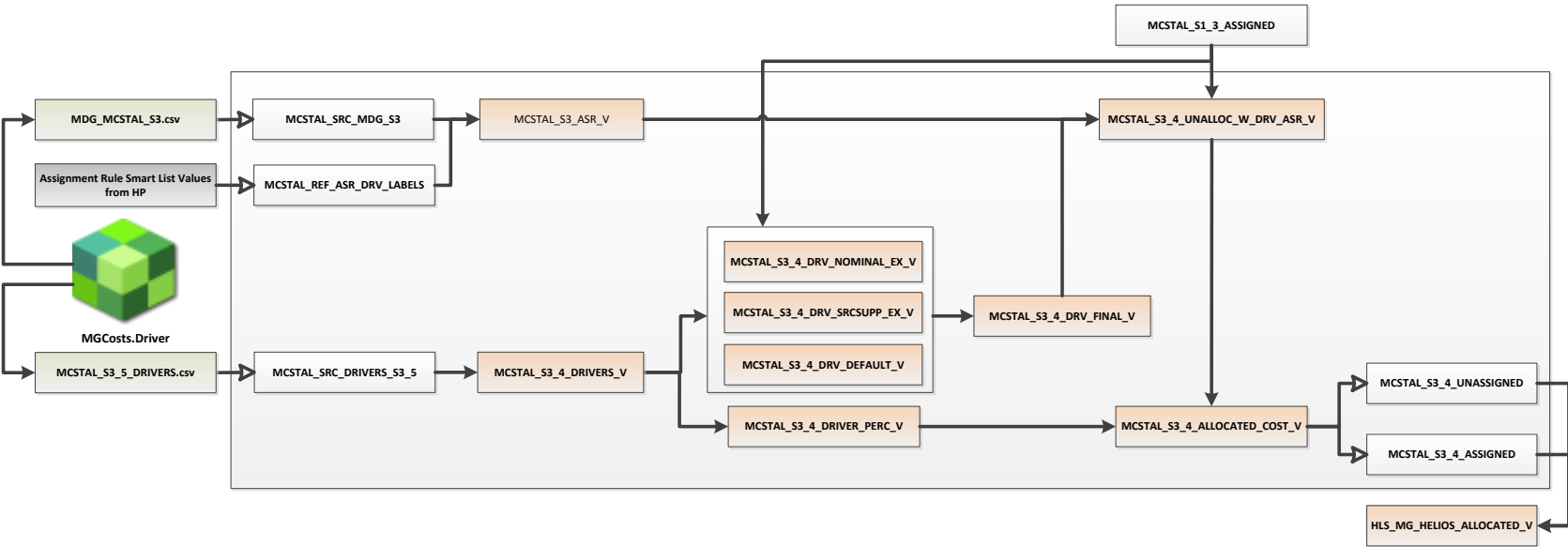
Hyperion Focus 17

SQL views to join cost to drivers



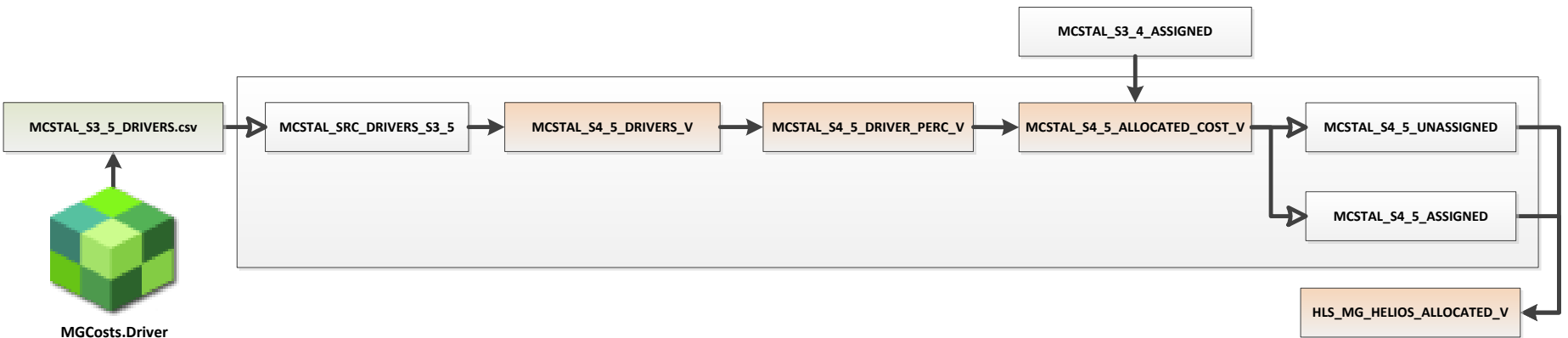
Hyperion Focus 17

SQL views to join cost to drivers



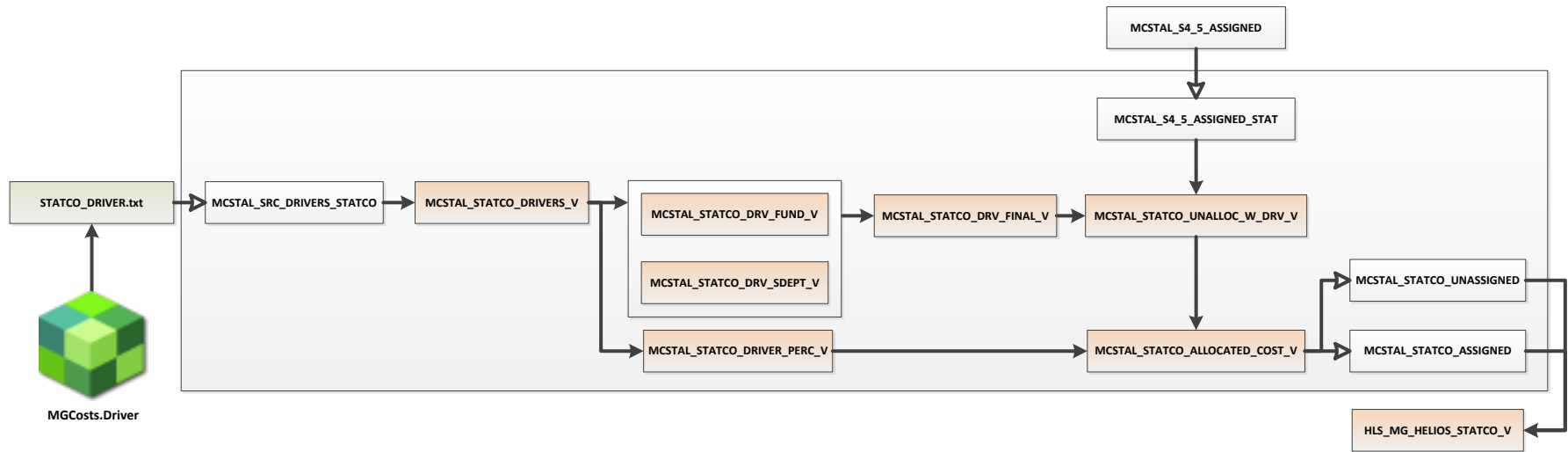
Hyperion Focus 17

SQL views to join cost to drivers



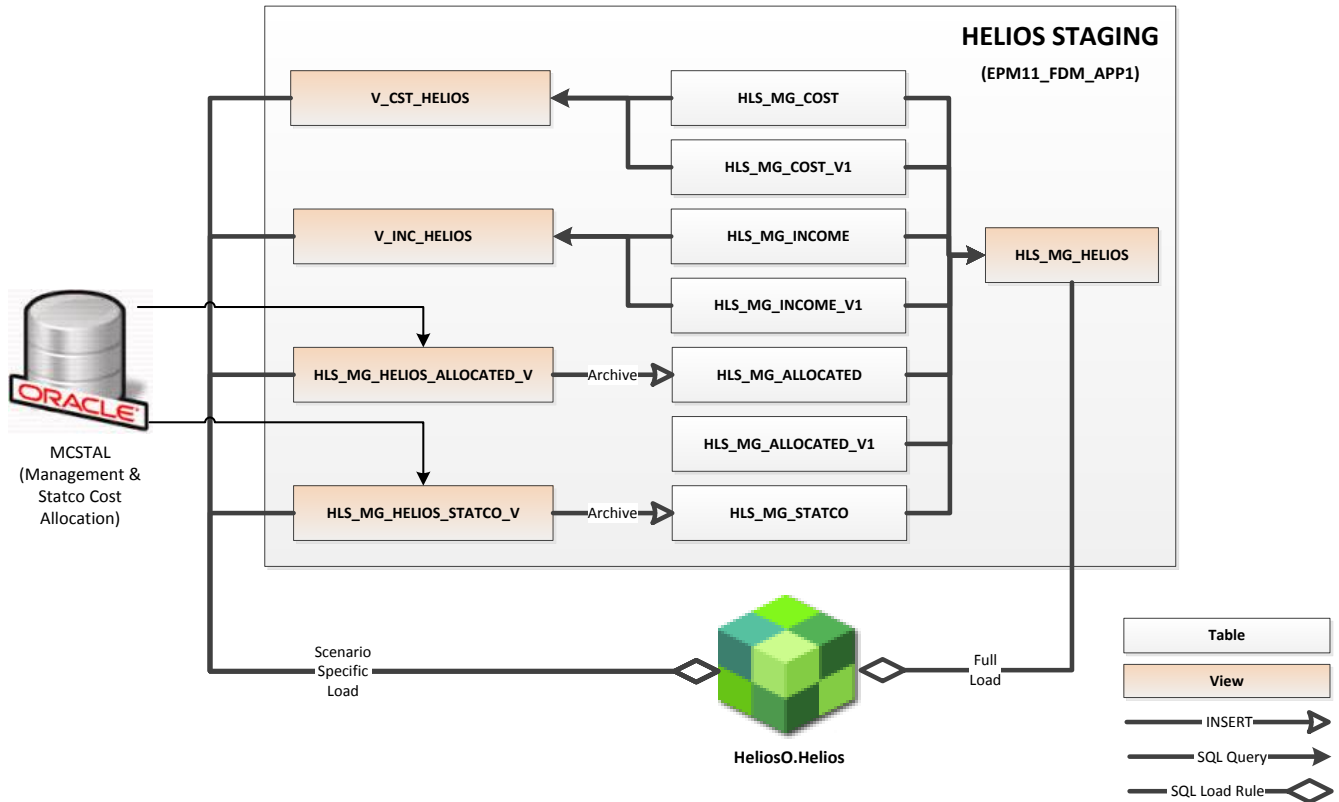
Hyperion Focus 17

SQL views to join cost to drivers



Hyperion Focus 17

SQL staging area



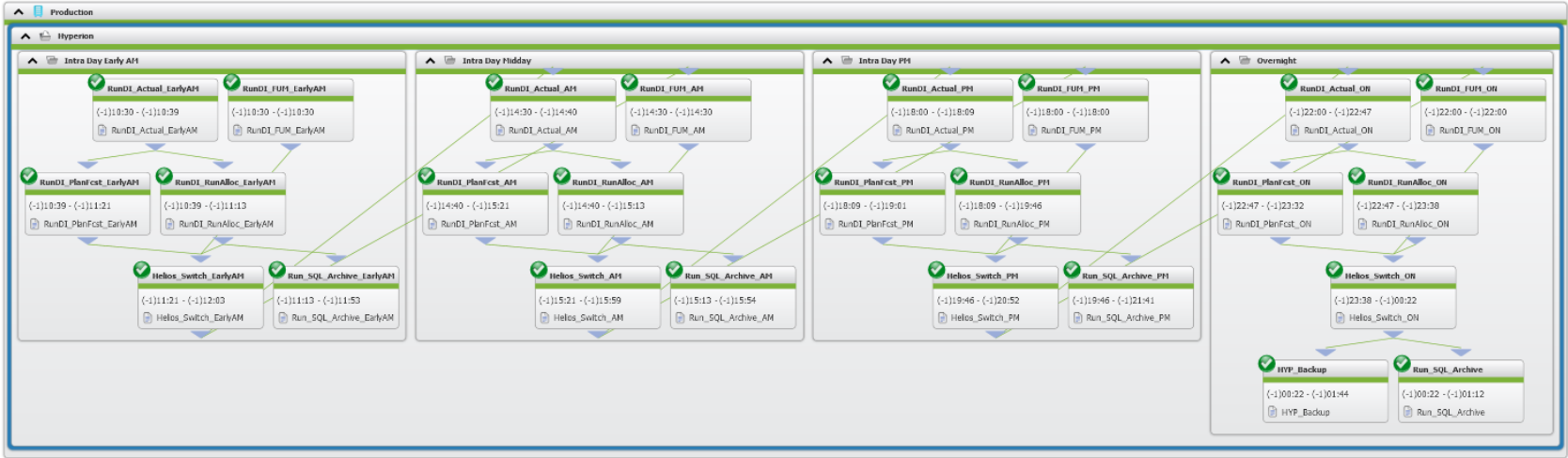
Hyperion Focus 17

Parallel loads into helios

```
import database $1.$2 data connect as $key $ESS_SQL_USER identified by $key $ESS_SQL_PASS
using multiple rules_file 'ldHLSACY','ldHLSAY1','ldHLSAY2','ldHLSAY3','ldHLSstc','ldHLSinc',
'ldHLSstc' to load_buffer_block starting with buffer_id 100 on error write to "$_MAXL_ERR_FILE";
```

Hyperion Focus 17

Ctrlm for automation



Hyperion Focus 17

Single planning form to control which scenario

	Refresh to Helios (Day)	Run Allocation (Day)	Refresh to Helios (Overnight)	Run Allocation (Overnight)
Actuals	Yes ▼	Yes ▼	Yes ▼	Yes ▼
Jan Rolling Forecast - Loaded	▼	▼	▼	▼
Feb Rolling Forecast - Loaded	▼	▼	▼	▼
Mar Rolling Forecast - Loaded	▼	▼	▼	▼
Apr Rolling Forecast - Loaded	▼	▼	▼	▼
May Rolling Forecast - Loaded	▼	▼	▼	▼
Jun Rolling Forecast - Loaded	▼	▼	▼	▼
Jul Rolling Forecast - Loaded	▼	▼	▼	▼
Aug Rolling Forecast - Loaded	▼	▼	▼	▼
Sep Rolling Forecast - Loaded	▼	▼	▼	▼
Oct Rolling Forecast - Loaded	▼	▼	▼	▼
Nov Rolling Forecast - Loaded	▼	▼	▼	▼
Dec Rolling Forecast - Loaded	▼	▼	▼	▼

Hyperion Focus 17

SQL tables and views

- MCSTAL_S1_2_ASSIGNED
- MCSTAL_S1_2_UNASSIGNED
- MCSTAL_S1_3_ASSIGNED
- MCSTAL_S2_3_ASSIGNED
- MCSTAL_S2_3_UNASSIGNED
- MCSTAL_S3_4_ASSIGNED
- MCSTAL_S3_4_UNASSIGNED
- MCSTAL_S4_5_ASSIGNED
- MCSTAL_S4_5_ASSIGNED_STAT
- MCSTAL_S4_5_UNASSIGNED
- MCSTAL_SRC_DRIVERS_S1_3
- MCSTAL_SRC_DRIVERS_S3_5
- MCSTAL_SRC_DRIVERS_STATCO

- MCSTAL_S1_2_ALLOCATED_COST_V
- MCSTAL_S1_2_DRIVER_PERC_V
- MCSTAL_S1_2_DRIVERS_V
- MCSTAL_S1_3_ALLOCATED_COST_V
- MCSTAL_S1_UNALLOCATED_COST_V
- MCSTAL_S2_3_ALLOCATED_COST_V
- MCSTAL_S2_3_DRIVER_PERC_V
- MCSTAL_S2_3_DRIVERS_V
- MCSTAL_S3_4_ALLOCATED_COST_V
- MCSTAL_S3_4_DRIVER_PERC_V
- MCSTAL_S3_4_DRIVERS_V
- MCSTAL_S3_4_DRV_DEFAULT_V
- MCSTAL_S3_4_DRV_FINAL_V
- MCSTAL_S3_4_DRV_NOMINAL_EX_V
- MCSTAL_S3_4_DRV_SRCSUPP_EX_V
- MCSTAL_S3_4_UNALLOC_W_DRV_AR_V

- MCSTAL_S3_ASR_V
- MCSTAL_S4_5_ALLOCATED_COST_V
- MCSTAL_S4_5_DRIVER_PERC_V
- MCSTAL_S4_5_DRIVERS_V
- MCSTAL_STATCO_ALLOCATED_COST_V
- MCSTAL_STATCO_DRIVER_PERC_V
- MCSTAL_STATCO_DRIVERS_V
- MCSTAL_STATCO_DRV_FINAL_V
- MCSTAL_STATCO_DRV_FUND_EX_V
- MCSTAL_STATCO_DRV_SDEPT_V
- MCSTAL_STATCO_UNALLOC_W_DRV_V

Hyperion Focus 17

Example SQL view

```
CREATE OR REPLACE FORCE VIEW "EPM11_FDM_APP1"."MCSTAL_S1_2_ALLOCATED_COST_V" ("YEARS", "SCENARIO", "VERSION", "STATCO_TYPE", "S_DEPARTMENT")
SELECT s1.YEARS, s1.SCENARIO, s1.VERSION, s1.STATCO_TYPE, CONCAT('S_',s1.DEPARTMENT) AS S_DEPARTMENT, drv.C_DEPARTMENT, s1.ACCOUNT, decode
decode(drv.PERCENTAGE,null,s1.JAN,s1.JAN * drv.PERCENTAGE) AS JAN,
decode(drv.PERCENTAGE,null,s1.FEB,s1.FEB * drv.PERCENTAGE) AS FEB,
decode(drv.PERCENTAGE,null,s1.MAR,s1.MAR * drv.PERCENTAGE) AS MAR,
decode(drv.PERCENTAGE,null,s1.APR,s1.APR * drv.PERCENTAGE) AS APR,
decode(drv.PERCENTAGE,null,s1.MAY,s1.MAY * drv.PERCENTAGE) AS MAY,
decode(drv.PERCENTAGE,null,s1.JUN,s1.JUN * drv.PERCENTAGE) AS JUN,
decode(drv.PERCENTAGE,null,s1.JUL,s1.JUL * drv.PERCENTAGE) AS JUL,
decode(drv.PERCENTAGE,null,s1.AUG,s1.AUG * drv.PERCENTAGE) AS AUG,
decode(drv.PERCENTAGE,null,s1.SEP,s1.SEP * drv.PERCENTAGE) AS SEP,
decode(drv.PERCENTAGE,null,s1.OCT,s1.OCT * drv.PERCENTAGE) AS OCT,
decode(drv.PERCENTAGE,null,s1.NOV,s1.NOV * drv.PERCENTAGE) AS NOV,
decode(drv.PERCENTAGE,null,s1.DEC,s1.DEC * drv.PERCENTAGE) AS DEC
FROM MCSTAL_S1_UNALLOCATED_COST_V s1 LEFT OUTER JOIN MCSTAL_S1_2_DRIVER_PERC_V drv
ON CONCAT('S_',s1.DEPARTMENT) = drv.S_DEPARTMENT AND s1.YEARS = drv.YEARS AND s1.SCENARIO = drv.SCENARIO AND s1.VERSION = drv.VERSION;
```

Hyperion Focus 17

Some stats on helios reporting cube

Helios Currently Contains

Approx 190-195m rows of data.

We reload 19-23m rows 4 x per day during RF

We reload 35-40m rows 4 x per day during Actual month end

Reloads take about 40-45 mins fully aggregated

On Average every row of cost ends up creating 600 odd rows of allocated cost

Hyperion Focus 17

- Overview and Introduction
 - Our journey so far
- Bring on the SQL Allocations
- **Summary/Business Benefits**
 - Questions

Hyperion Focus 17

Summary

- As an engine SQL is far superior for running allocations
- No limitations on genealogy
- Can include all detail from start to finish
- No issues deploying cubes with Essbase limitation on member combos
- Credit to Vince Wong for turning my vision into reality

Hyperion Focus 17

Business benefits

- RF gives us an earlier long range directional view of the business
- More flexibility with our model, one offs, large changes, frequency
- Workload is now smoothed out over year
- Less onerous planning process
- Scenario modelling with ease

Hyperion Focus 17

- Overview and Introduction
 - Our journey so far
- Bring on the SQL Allocations
- Summary/Business Benefits
- **Questions**

Hyperion Focus 17

Thank you

