

# Chart of Accounts Transformation Master Class: Tips for Designing Effective Charts

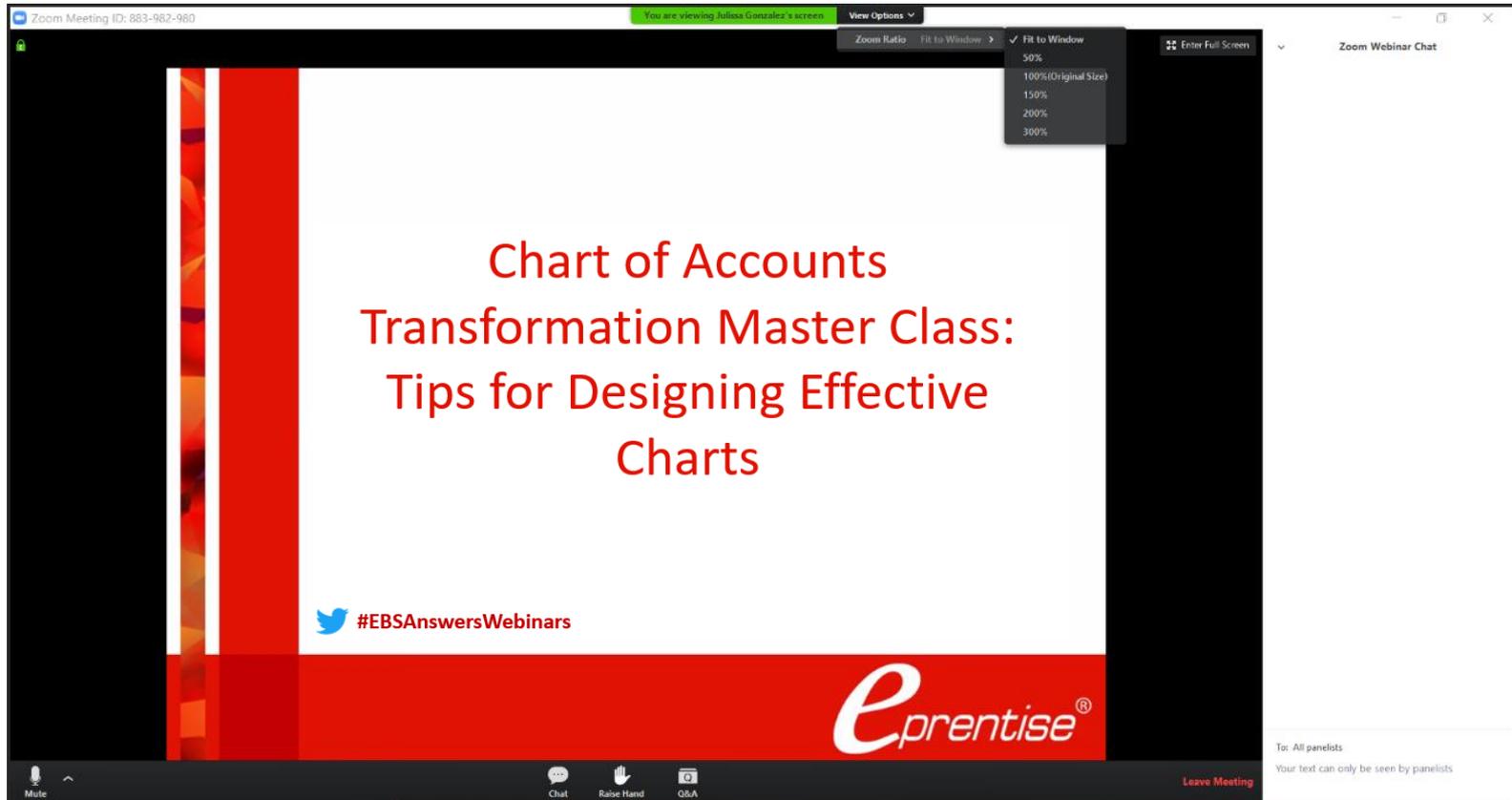


#EBSAnswersWebinars



# Webinar Mechanics

- Submit text questions.
- Q&A addressed at the end of the session and posted on LinkedIn.
- Everyone will receive an email with a link to view a recorded version of today's session.
- Polling questions will be presented during the session. If you want CPE credit for this webinar, you must answer all of the polling questions.



# About eprentise

## Getting the most out of your Oracle® E-Business Suite with Software that *Transforms* and *Optimizes* EBS



### ❑ **TRANSFORM** With Proven Out-of-the-Box Solutions That Align EBS With Your Business

- **Consolidation** merges and harmonizes one or more disparate, differently-configured, database instances into a single, fully-functional application
- **Divestiture** filters data when a company is carving out or selling off part of their business, to create a stand-alone fully functional environment for the divested entity with a limited data set
- **FlexField** changes the financial chart of accounts to support standardization and increase reporting reliability retaining all transactional history
- **Reorganization** changes or moves any configurations or set-ups and all related transactions. Reorganization is a broad category and includes software solutions for merging or separating organization units, ledgers, inventory organizations, or legal entities, calendar changes, currency changes, etc. Reorganization Software is used to comply with new regulatory or statutory changes, new organization structures, entry into new markets, and to support mergers, acquisitions, or divestitures.

### ❑ **OPTIMIZE** With Software That Gives You Visibility Into Your Operations

- **C Collection** analytics will identify potential problem areas with transactional data. The software increases the accuracy of the underlying data, provides data transparency, and allows EBS users to reduce costs, leverage opportunities across the enterprise, improve business processes and increase the confidence level of the users in their data, processes, and operations.
- **Automated Audit** is audit and assurance software for internal and external financial auditors and allows for drill-down from a balance sheet report into the transaction-level detail. The software covers hundreds of substantive procedures across all EBS modules for the entire enterprise domain and builds in consistent audit processes and workflows across the organization.

### ❑ **Does not violate Oracle Support Agreement**

# Agenda

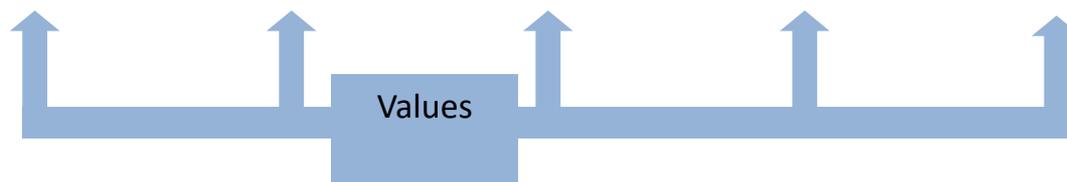
- ❑ Introduction
- ❑ Chart of accounts (COA) basics
- ❑ Designing a good COA
  - 10 Design considerations

# Chart of Accounts Basics: The Accounting Flexfield

## □ Example Structure

➤ Segments, Values, Code Combinations

Company	Business Unit	Cost Center	Region	Account
01	100	150	East	1111
02	100	210	West	2222



01.100.150.East.1111



Code Combination

# Chart of Accounts Basics: Segments

- ❑ The EBS accounting flexfield accommodates up to 30 segments for categorizing transactions
- ❑ Structurally analogous to columns in a table
- ❑ Have defined lengths for the values they contain
- ❑ **Common segment examples:**
  - Cost Center
  - Department
  - Fund
  - Location
  - Product Line

# Flexfields in the Form

Journals (Vision Operations (USA)) - 1124371 M1 Inventory 3680915: A 122438

Journal	WIP USD	Conversion	Currency	USD	Reverse	Date	
Description	2005/10/14 M1 "		Date	14-OCT-2005		Period	
Ledger	Vision Operations (U	Category	Type	User		Method	Switch Dr/Cr
Period	Oct-05	Effective Date	Rate	1		Status	Not Reversed
Balance Type	Actual	Budget				<input type="button" value="Reverse"/>	
Clearing Company		Tax					
Journal Type	Standard	Control Total					

Lines | Other Information

Line	Account	Debit (USD)	Credit (USD)	UOM	Qty	Description
1	01-000-1410-0000-000	0.00				2005/10/14 M1 "
2	01-000-1420-0000-000	0.00				2005/10/14 M1 "
3	01-000-1430-0000-000	0.00				2005/10/14 M1 "
4	01-000-1440-0000-000	0.00				2005/10/14 M1 "
5	01-000-1450-0000-000	0.00				2005/10/14 M1 "
6	01-520-5310-0000-000	0.00				2005/10/14 M1 "
7	01-520-5312-0000-000	0.00				2005/10/14 M1 "
8	01-520-5330-0000-000	0.00				2005/10/14 M1 "
		15,215.00	15,215.00			

Acct Desc: Operations-No Department-Inventory Material Value-No Sub Account-No Product

Buttons: Post, AutoCopy Batch..., Approve, Line Drilldown..., T Accounts..., Check Funds, Reserve Funds, View Results, Change Period..., Change Currency...

Annotations:

- Code Combinations (points to line 3)
- Accounting Key Flexfield (points to line 7)
- Value Descriptions (points to Acct Desc)

# Chart of Accounts Basics: Hierarchy

- ❑ The accounting flexfield incorporates parent-child relationships among values
  
- ❑ Roll-up Groups
  - A collection of parent values for a given segment
  - Used to create summary accounts
  - The most detail is at the lowest level
  
- ❑ Summary Accounts
  - Hierarchical rollup of children and/or parents
  - Faster reporting
  - Account balance inquiries

# 1<sup>st</sup> Design Consideration – Does the COA Pass The Mystery Accountant Test

- ❑ Your COA segments should be defined with enough clarity to pass the Mystery Accountant Test
- ❑ Do your segments answer the questions **Who**, **What**, **Where**, **Why**, and **How** giving your accounting flexfield the ability to classify each transaction
  - **Who** is responsible for the transaction?
  - **What** is the nature of the transaction?
  - **Where** is the activity occurring?
  - **How** are the funds being impacted?
  - **Why** was the transaction posted?

## 2<sup>nd</sup> Design Consideration: One Type of Information

- ❑ Information should not overlap across segments.
- ❑ If each segment contains one (and only one) type of information, you:
  - Reduce the maintenance of keeping information accurate in two places
  - Reduce the possibility of introducing errors into your accounting
- ❑ **Case:** If your cost center has the same type of information as a business unit segment, there is no need to implement both.
- ❑ **Case:** You shouldn't have a Department segment value such as **HR – Sacramento, CA** if there is also a Location segment in the chart.

## 3<sup>rd</sup> Design Consideration: Information Not Repeated

- ❑ The accounting flexfield should not repeat information that exists in other modules of EBS.
  - Reduced maintenance and errors (similar to Criteria 1)
- ❑ **Case:** If you are implementing Oracle Projects modules, there is no need to have a project segment in your accounting flexfield.
- ❑ **Case:** If you are implementing Receivables, then there is no need for a customer segment.

## 4<sup>th</sup> Design Consideration: Enough Room to Expand

- ❑ Define your segment lengths to be long enough to accommodate values added in the future.

**Note:** Although it likely won't be an issue, the maximum number of characters for a code combination string is 240, so there are upper limits on how long you can define your segments.

- ❑ When designing values, be sure to allow enough room for future growth within each rollup group:
  - Increment by at least 5 within each group
  - Increment by 10 if the group is likely to be a high growth area.
  - **Example:** If you have a location segment, allow enough room to add ten additional values between each of your lowest levels. Your Location segment hierarchy might look like this:

```
10000 US
    11000 Midwest
        11100 Detroit Metropolitan Area
            11110 Ann Arbor
            11120 Canton
            11130 Plymouth
20000 Canada
```

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

- ❑ Ranging your values logically promotes streamlined reporting, security, and maintenance.
  - Include a whole range of values in your rule (cross validation, security) and FSG report definitions
  - Exclude specific values if needed
  - Minimize the number of cross validation rules needed (under 50 using logical ranges compared to hundreds or thousands when not using logical ranges)



# 5<sup>th</sup> Design Consideration: Use Logical Ranges

- ❑ **Out of Range** — Cross-validation rules can get messy if your chart of accounts values are not organized in logical ranges.
  - Rule elements are much more complex due to the inability to rely on value ranges for exclusion
  - **Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):
    - 3001
    - 5057
    - 6124
    - 8537
    - 9905

INCLUDE	Low Value	High Value
Company	000	999
Department	0000	9999
Account	00000	99999
Product Line	000	ZZZ
Intercompany	000	999

## 5<sup>th</sup> Design Consideration: Use Logical Ranges

### ❑ Exclude Elements — 6 Steps Required

- Need Exclude elements for all the values above and below each of the department values

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

**Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):

3001 ■ 5057 ■ 6124 ■ 8537 ■ 9905

This statement prevents all departments below 3001 from being used:

EXCLUDE	Low Value	High Value
Company	000	999
Department	0000	3000
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

**Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):

3001 ■ 5057 ■ 6124 ■ 8537 ■ 9905

This statement prevents all departments greater than 3001 and less than 5057 from being used:

EXCLUDE	Low Value	High Value
Company	000	999
Department	3002	5056
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

**Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):

3001 ■ 5057 ■ 6124 ■ 8537 ■ 9905

This statement prevents all departments greater than 5057 and less than 6124 from being used:

EXCLUDE	Low Value	High Value
Company	000	999
Department	5058	6123
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

**Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):

3001 ■ 5057 ■ 6124 ■ 8537 ■ 9905

This statement prevents all departments greater than 6124 and less than 8537 from being used:

EXCLUDE	Low Value	High Value
Company	000	999
Department	6125	8536
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

**Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):

3001 ■ 5057 ■ 6124 ■ 8537 ■ 9905

This statement prevents all departments greater than 8537 and less than 9905 from being used:

EXCLUDE	Low Value	High Value
Company	000	999
Department	8538	9904
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

**Example Goal:** Prevent revenue account values between 30000 and 40000 from being used with any department values other than the following 5 values (we'll begin with the familiar global Include element):

3001 ■ 5057 ■ 6124 ■ 8537 ■ 9905



And finally, this statement prevents all departments greater than 9905 from being used:

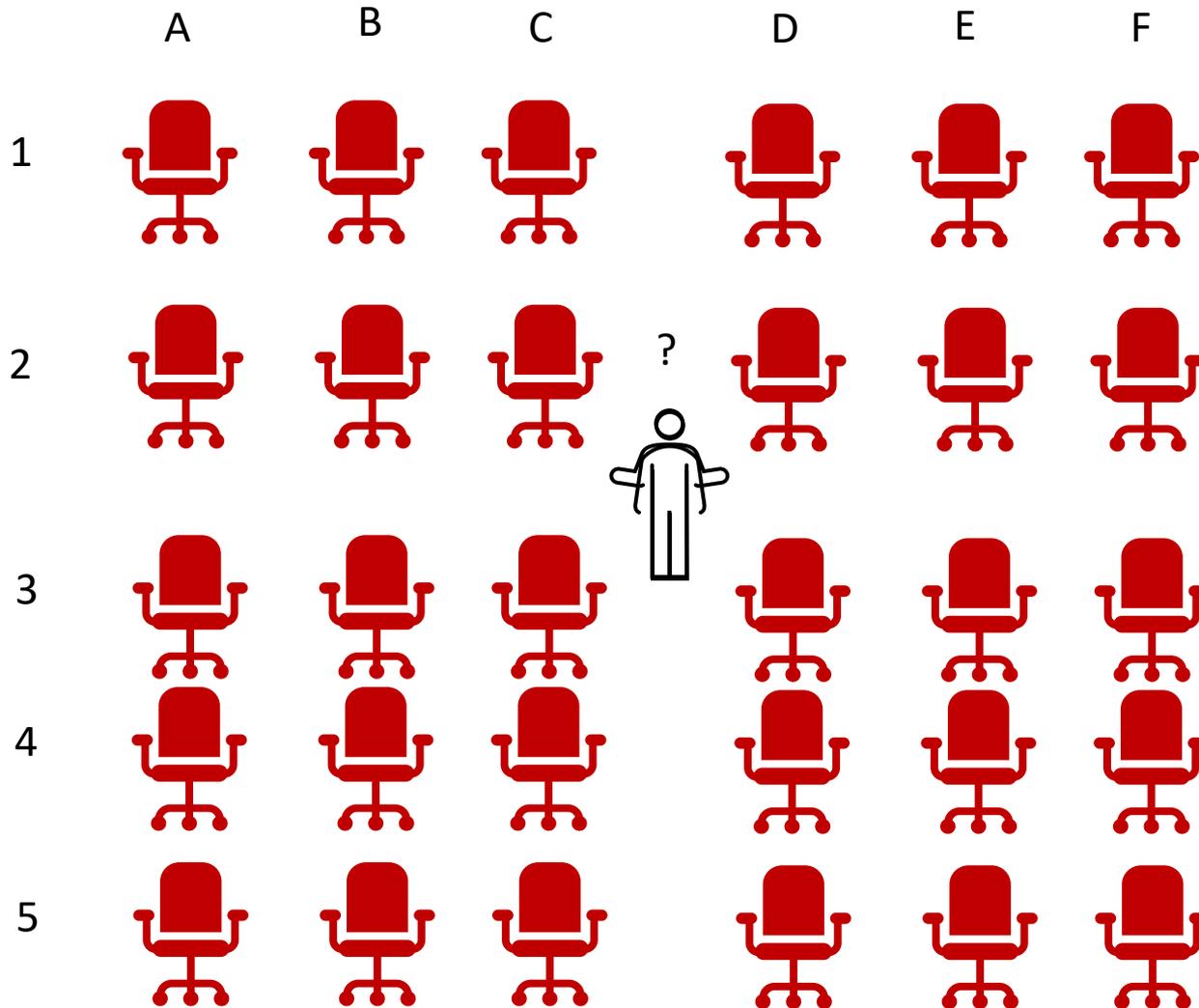
EXCLUDE	Low Value	High Value
Company	000	999
Department	9906	9999
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 5<sup>th</sup> Design Consideration: Use Logical Ranges

INCLUDE	Low Value	High Value
Company	000	999
Department	0000	9999
Account	00000	99999
Product Line	000	ZZZ
Intercompany	000	999

EXCLUDE	Low Value	High Value
Company	000	999
Department	9906	9999
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 6<sup>th</sup> Design Considerations: Don't Use a Dependent Segment



# 6<sup>th</sup> Design Considerations: Don't Use a Dependent Segment

- ❑ Using dependent segments generally results in decreased functionality and/or requires increased maintenance in:
  - Cross-Validation Rules
  - Security Rules
  - Mass Allocations
  - FSG's
  - Oracle Financial Analyzer
  - SLA Account Derivation Rules
  - Auto Accounting
  - Anything requiring account ranges

# 7<sup>th</sup> Design Consideration: No Reliance on Spreadsheets

- ❑ Be able to get the information you need from built-in EBS reports.
  - Real-time value
  - No errors introduced by use of spreadsheets
  - Streamlined workflow, lower resource requirements, and reduced maintenance
  - No need to integrate with 3<sup>rd</sup> party reporting applications
  
- ❑ Capitalize on a master row set for FSG reports.
  - Allows you to generate different reports without having to rewrite each report

## 8th Design Consideration: Avoid Alpha Characters Like The Plaque

- ❑ There is a perceived simplicity to using Alpha characters but they will introduce a maintenance nightmare.



# 8th Design Consideration: Avoid Alpha Characters Like The Plague

- ❑ **Goal** — Prevent revenue account values between 30000 and 40000 from being used with balance sheet department values between 1000 and 1999.
  - **Elements:** Global Include, single Exclude
  - **Error segment:** Department

INCLUDE	Low Value	High Value
Company	000	999
Department	0000	9999
Account	00000	99999
Product Line	000	ZZZ
Intercompany	000	999

EXCLUDE	Low Value	High Value
Company	000	999
Department	1000	1999
Account	30000	40000
Product Line	000	ZZZ
Intercompany	000	999

# 8th Design Consideration: Avoid Alpha Characters Like The Plague

- ❑ **Goal** — *Only* department 7640 is permitted for use with any product line that starts with S.
  - **Elements:** Global Include, three Excludes
  - **Error segment:** Department

INCLUDE	Low Value	High Value
Company	000	999
Department	0000	9999
Account	00000	99999
Product Line	000	ZZZ
Intercompany	000	999

EXCLUDE	Low Value	High Value
Company	000	999
Department	0000	7639
Account	00000	99999
Product Line	000	RZZ
Intercompany	000	999

# 8th Design Consideration: Avoid Alpha Characters Like The Plague

- ❑ **Goal** — *Only* department 7640 is permitted for use with any product line that starts with S.
  - **Elements:** Global Include, three Excludes
  - **Error segment:** Department

This exclude statement prevents the departments below 7640 from using any product line that starts with anything below S:

EXCLUDE	Low Value	High Value
Company	000	999
Department	0000	7639
Account	00000	99999
Product Line	000	RZZ
Intercompany	000	999

# 8th Design Consideration: Avoid Alpha Characters Like The Plague

- ❑ **Goal** — *Only* department 7640 is permitted for use with any product line that starts with S.
  - **Elements:** Global Include, three Excludes
  - **Error segment:** Department

This exclude statement prevents the departments above 7640 from using any product line that starts with anything below S:

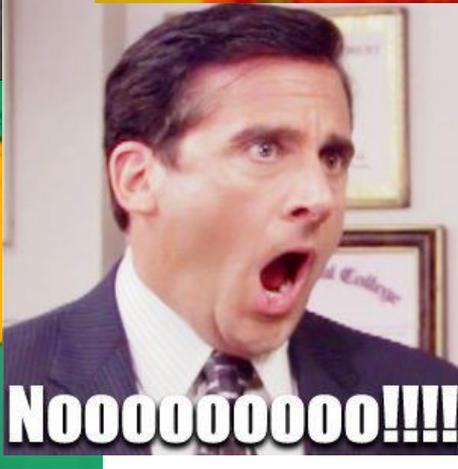
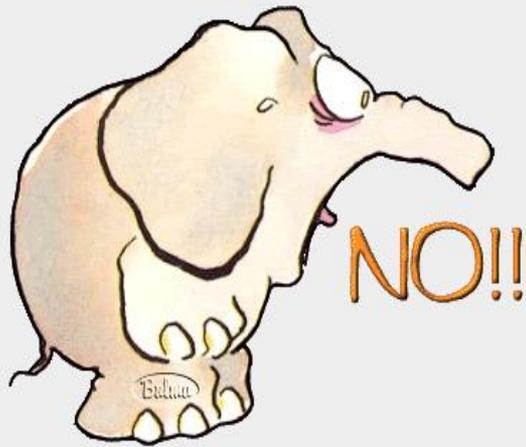
EXCLUDE	Low Value	High Value
Company	000	999
Department	7641	9999
Account	00000	99999
Product Line	000	RZZ
Intercompany	000	999

# 8th Design Consideration: Avoid Alpha Characters Like The Plague

- ❑ **Goal** — *Only* department 7640 is permitted for use with any product line that starts with S.
  - **Elements:** Global Include, three Excludes
  - **Error segment:** Department

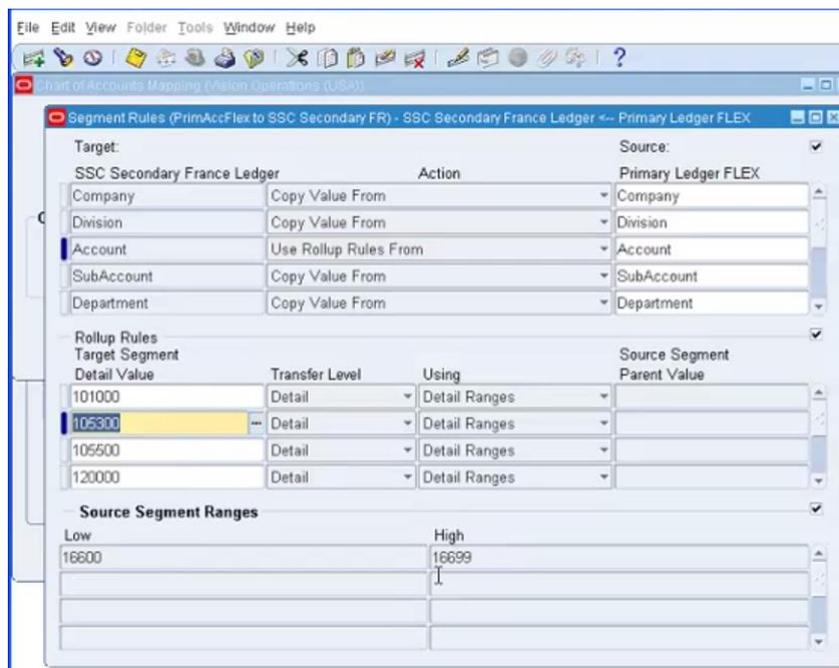
This exclude statement prevents the departments below 7640 from using any product line that starts with anything above S:

EXCLUDE	Low Value	High Value
Company	000	999
Department	0000	7639
Account	00000	99999
Product Line	T00	ZZZ
Intercompany	000	999



# 9th Design Consideration: Make Use of an Analysis Segment

Accounting Setups Manager > Chart of Accounts mapping



- Facilitates multiple ledger structure and greater ease of maintaining statutory compliance.

# Global Visibility – Local Compliance with R12

## Subledger Accounting, Accounting Methods Builder Replaces Global Accounting Engine

Compare Subledger Journal Entries		
	SSC France	SSC France SL (EUR)
Journal Entry Status	Final	Final
Balance Type	Actual	Actual
Journal Entry Type	Standard	Standard
Category	Purchase Invoices	Purchase Invoices
GL Date	17-Oct-2006	17-Oct-2006
Completion Date	17-Oct-2006 17:15:21	17-Oct-2006 17:15:21
Application Accounting Definition	Training AAD	RZ AP (No specific COA)
Description	SLA is the coolest!!! R2987653 is the invoice number.	Invoice Validated , Invoice Number : R2987653 , Date : 17-OCT-06 , Document Sequence Category : STD INV , Document Sequence Name : SSC FR AP , Invoice Voucher Number : 36 , Invoice Description :

Lines **Global COA** **Plan Comptable General**

SSC France					SSC France SL (EUR)				
Number	Accounting Class	Account	Accounted DR (EUR)	Accounted CR (EUR)	Number	Accounting Class	Account	Accounted DR (EUR)	Accounted CR (EUR)
1	Item Expense	101.100.63580.0000.730.000.000	6,000.00		1	Item Expense	101.100.616100.0000.730.000.000.000	6,000.00	
2	Item Expense	101.100.63580.0000.720.000.000	4,000.00		2	Item Expense	101.100.616100.0000.720.000.000.000	4,000.00	
3	Recoverable Tax	101.100.13200.3101.000.000.000	1,960.00		3	Recoverable Tax	101.100.445623.3101.000.000.000.000	1,960.00	
4	Liability	101.100.22100.0000.730.000.000		6,000.00	4	Liability	101.100.401009.0000.000.000.000.000		10,000.00
5	Liability	101.100.22100.0000.720.000.000		4,000.00	5	Liability	101.100.401009.0000.000.000.000.000		1,960.00
6	Liability	101.100.22100.0000.000.000.000		1,960.00					

# 10<sup>th</sup> Design Consideration: Move to A Global Chart of Accounts Where Possible

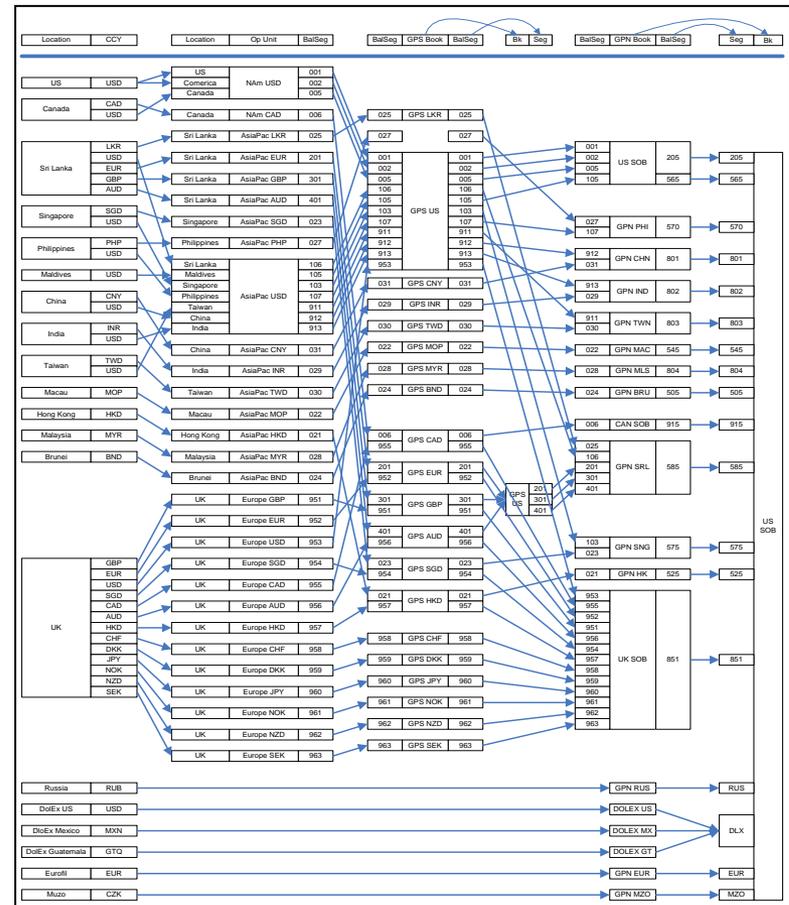
- ❑ Primary ledger is single source of truth for all accounting, reconciliation and analytical reporting
- ❑ Consistency but flexibility to accommodate different requirements
- ❑ External reporting without relying on a separate financial consolidation system
- ❑ Drill down to individual transactions in the subledgers without translation
- ❑ Transparency (3-5 years) to meet IFRS standards and international auditing requirements
- ❑ Common metrics and reporting structures with common interpretation

# A Global Chart of Accounts – Reduced Costs

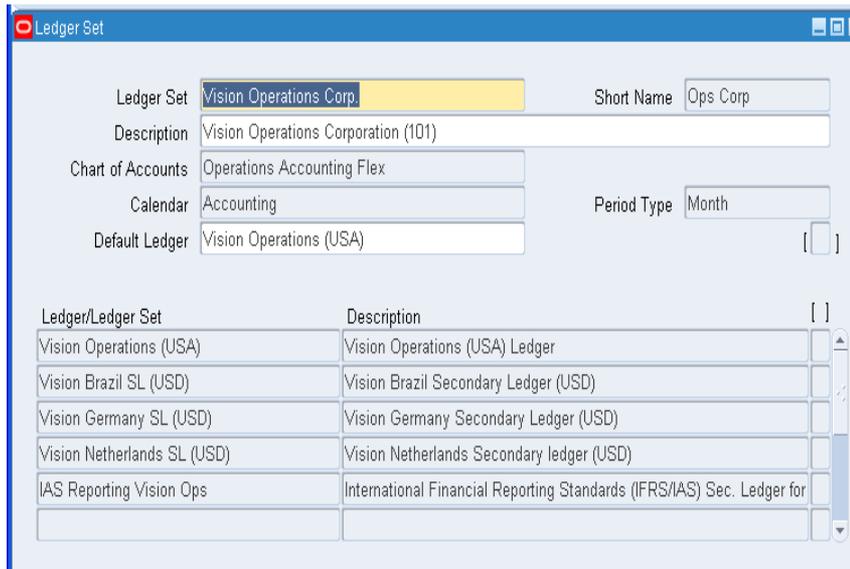
- ❑ Reduce complexity of configurations
  - Cross validation rules
  - Security rules
  - Reporting within ranges
- ❑ No conversions required for data warehouse queries, drill-down to subledgers, ad-hoc reporting
- ❑ Facilitate movement to shared service centers
  - Single COA to manage
  - Standardized training
- ❑ Enterprise governance and control of new combinations
  - Reduce redundancies

# A Global Chart of Accounts – Reduced Complexity

- Streamline the month- and year-end closing processes
- No more messy financial consolidations using spreadsheets
- R12 eliminates the need for using and maintaining multiple charts of accounts



# R12 Features - Ledger Sets



Ledger/Ledger Set	Description
Vision Operations (USA)	Vision Operations (USA) Ledger
Vision Brazil SL (USD)	Vision Brazil Secondary Ledger (USD)
Vision Germany SL (USD)	Vision Germany Secondary Ledger (USD)
Vision Netherlands SL (USD)	Vision Netherlands Secondary ledger (USD)
IAS Reporting Vision Ops	International Financial Reporting Standards (IFRS/IAS) Sec. Ledger for

## Perform the Following Across Ledgers

- Open/Close Periods
- Create Journals
- Allocations Across Ledgers
- Recurring Journals for All Ledgers
- Elimination Sets for All Ledgers
- Translate and Revalue Balances
- View Information without Changing Responsibilities
- View Journals and Account Balances Across Ledgers
- Submit Standard Reports
- Create Financial Statements that Include Data for Multiple Ledgers

# Global COA Design Recommendations

- ❑ Add an intercompany segment – take advantage of AGIS
- ❑ Add a segment to accommodate local requirements
  - Ranges, rollups defined for each country to use
    - Local bank accounts
    - Statutory reporting
  - Location segment (optional) but helps with security, cross validation
- ❑ Implement other modules for detailed tracking at a local level (through an OU)
  - Project Accounting
  - Collections
- ❑ Implement Multiple Reporting Currencies, secondary ledgers to report in different currencies

# 10 Design Considerations

1. Does the COA Pass The Mystery Accountant Test
2. One Type of Information
3. Information Not Repeated
4. Enough Room to Expand
5. Use Logical Ranges
6. Don't Use Dependent Segments
7. No Reliance on Spreadsheets
8. Avoid Alpha Characters  
*(Like the Plague)*
9. Make Use of an Analysis Segment
10. Move to a Global Chart of Accounts

# Questions?

## Contact

 #EBSAnswersWebinars

**Harrison Figura**  
*Product Manager*  
[hfigura@eprentise.com](mailto:hfigura@eprentise.com)



**- One World, One System, A Single Source of Truth -**

eprentise, LLC  
*Accelerating the time for change in  
Oracle® E-Business Suite*

[www.eprentise.com](http://www.eprentise.com)  
[www.AgilityByDesign.com](http://www.AgilityByDesign.com)  
[www.crystallizeanalytics.com](http://www.crystallizeanalytics.com)

# C Collection Analytics Software for Oracle EBS

# Thank You!

## Contact

 #EBSAnswersWebinars

**Harrison Figura**  
*Product Manager*  
[hfigura@eprentise.com](mailto:hfigura@eprentise.com)



**- One World, One System, A Single Source of Truth -**

eprentise, LLC  
*Accelerating the time for change in*  
*Oracle® E-Business Suite*

[www.eprentise.com](http://www.eprentise.com)  
[www.AgilityByDesign.com](http://www.AgilityByDesign.com)  
[www.crystallizeanalytics.com](http://www.crystallizeanalytics.com)