



# Building A Reference Data Consolidation Strategy

Bob Schork, Team Leader, AVP  
Citi Architecture and Technology Engineering  
Warren, NJ



# Who am I?

- Over 25 yrs of IT Experience (including consulting)
- Over 13 years of Metadata Experience including Metadata Analysis
- Implemented Rochade and Platinum Repositories, including Maintenance and Reporting functions
- Was a Developer, DBA, Data Admin, Data Modeler, Data Architect, Metadata Architect
- Board member of DAMA NJ, Metadata Professional Organization (MPO)



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- Difficult Issues



# Reference Data Problems

- Silos - Redundancy
- Same Reference tables on different databases
- Similar Reference Data Tables
- Star vs. Snowflake issues
- Relational Data Model vs. Dimensional
- History vs. Current
- Master Data vs. Master Reference Data



# Master Data vs. Master Reference Data

According to Malcolm Chisholm:

- **Master Data** – The parties to the transactions of the enterprise. Data that represents the direct participants in a transaction, and which must be present before a transaction fires. E.G. Customer, Product, Order, Etc.
- **Master Reference Data** – Codes and descriptions. Tables containing this data usually have just a few rows and columns. Reference Data is any kind of data that is used solely to categorize other data found in a database, or solely for relating data in a database to information beyond the boundaries of the enterprise.



# Like Reference Data Tables

- Note the Surrogate Key
- Note the different code lengths

## DIM\_EDW\_UW\_UNIT

DIM\_EDW\_UW\_UNIT\_ID: NUMBER(15) NOT NULL

UW\_UNIT\_CD: CHAR(2) NOT NULL  
UW\_UNIT\_SHORT\_NM: CHAR(30) NOT NULL  
UW\_UNIT\_LONG\_NM: CHAR(60) NOT NULL  
RMX\_SBU\_ID: NUMBER(15) NOT NULL  
DW\_RMX\_REGION\_ID: NUMBER(15) NOT NULL

## ERD\_REF\_UW\_UNIT

UW\_UNIT\_CD: CHAR(2) NOT NULL

UW\_UNIT\_SHORT\_NM: CHAR(20) NULL  
UW\_UNIT\_LONG\_NM: CHAR(50) NULL

## CORP\_RMX\_UNDERWRITING\_UNIT

UW\_UNIT\_CODE: CHAR(6) NOT NULL

UW\_UNIT\_NAME: CHAR(40) NOT NULL



# Dimension Types

## TYPE1\_TABLE

TYPE1\_CD: CHAR(2) NOT NULL

TYPE1\_SHORT\_NM: CHAR(20) NOT NULL  
TYPE1\_LONG\_NM: CHAR(50) NOT NULL

## TYPE2\_TABLE

TYPE2\_CD: CHAR(2) NOT NULL  
TYPE2\_CURRENT\_ROW\_IN: CHAR(1) NOT NULL

TYPE2\_SHORT\_NM: CHAR(20) NOT NULL  
TYPE2\_LONG\_NM: CHAR(50) NOT NULL  
TYPE2\_ROW\_EFFEC\_DT: DATE NOT NULL  
TYPE2\_ROW\_EXPIR\_DT: DATE NOT NULL

## TYPE3\_TABLE

TYPE3\_CD: CHAR(2) NOT NULL  
TYPE3\_GROUP\_CD: CHAR(3) NOT NULL

TYPE3\_SHORT\_NM: CHAR(20) NOT NULL  
TYPE3\_LONG\_NM: CHAR(50) NOT NULL  
TYPE3\_ROW\_EFFEC\_DT: DATE NOT NULL  
TYPE3\_ROW\_EXPIR\_DT: DATE NOT NULL  
PREV\_TYPE3\_CD: CHAR(2) NOT NULL  
PREV\_TYPE3\_GROUP\_CD: CHAR(3) NOT NULL  
PREV\_TYPE3\_SHORT\_NM: CHAR(20) NOT NULL  
PREV\_TYPE3\_LONG\_NM: CHAR(50) NOT NULL

## TYPE4\_TABLE

TYPE4\_CD: CHAR(2) NOT NULL  
TYPE4\_GROUP\_CD: CHAR(3) NOT NULL  
TYPE4\_ROW\_OCCUR\_NO: NUMBER(10) NOT NULL  
TYPE4\_ROW\_VER\_NO: NUMBER(10) NOT NULL

TYPE4\_SHORT\_NM: CHAR(20) NOT NULL  
TYPE4\_LONG\_NM: CHAR(50) NOT NULL  
TYPE4\_ROW\_EFFEC\_DT: DATE NOT NULL  
TYPE4\_ROW\_EXPIR\_DT: DATE NOT NULL

## TYPE6\_TABLE

TYPE6\_SUROG\_ID: NUMBER(15) NOT NULL

TYPE6\_CD: CHAR(2) NOT NULL  
TYPE6\_GROUP\_CD: CHAR(3) NOT NULL  
TYPE6\_CURRENT\_ROW\_IN: CHAR(1) NOT NULL  
TYPE6\_SHORT\_NM: CHAR(20) NOT NULL  
TYPE6\_LONG\_NM: CHAR(50) NOT NULL  
TYPE6\_ROW\_EFFEC\_DT: DATE NOT NULL  
TYPE6\_ROW\_EXPIR\_DT: DATE NOT NULL  
CREATE\_TS: DATE NOT NULL  
CREATE\_USER\_ID: VARCHAR2(15) NOT NULL  
LAST\_UPDATE\_TS: DATE NULL  
LAST\_UPDATE\_USER\_ID: VARCHAR2(15) NULL  
ETL\_AUDIT\_TRAIL\_ID: NUMBER(10) NOT NULL



# History vs. Current

- **TYPE DESCRIPTIONS:**
- TYPE 1 - Overwrite code, lose history. Usually static.
- TYPE 2 - Preserve History. Write new row with dates.
- TYPE 3 - Preserve Historical version and grouping values in table.
- TYPE 4 - Two Sequence numbers, occurrence and version.
- TYPE 6 - Add surrogate key, Effective and Expiration dates. May also have audit information.

## TYPE1\_TABLE

```
TYPE1_CD: CHAR(2) NOT NULL
TYPE1_SHORT_NM: CHAR(20) NOT NULL
TYPE1_LONG_NM: CHAR(50) NOT NULL
```

## TYPE6\_TABLE

```
TYPE6_SUROG_ID: NUMBER(15) NOT NULL
TYPE6_CD: CHAR(2) NOT NULL
TYPE6_GROUP_CD: CHAR(3) NOT NULL
TYPE6_CURRENT_ROW_IN: CHAR(1) NOT NULL
TYPE6_SHORT_NM: CHAR(20) NOT NULL
TYPE6_LONG_NM: CHAR(50) NOT NULL
TYPE6_ROW_EFFEC_DT: DATE NOT NULL
TYPE6_ROW_EXPIR_DT: DATE NOT NULL
CREATE_TS: DATE NOT NULL
CREATE_USER_ID: VARCHAR2(15) NOT NULL
LAST_UPDATE_TS: DATE NULL
LAST_UPDATE_USER_ID: VARCHAR2(15) NULL
ETL_AUDIT_TRAIL_ID: NUMBER(10) NOT NULL
```



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- Difficult Issues



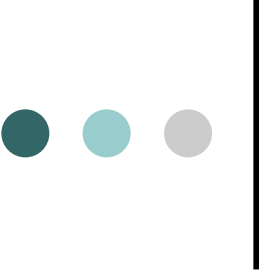
# Scoping

- Start small
- Resolve silos first, then expand to Business Units.
- Set up an Enterprise Governance committee to resolve conflicts among silos and Business Units.
- Don't be afraid to grandfather existing Reference Tables.
- Temper expectations. Remember, it will take a long time.



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- Difficult Issues



# True Upper Management Support – or lack thereof

- A CIO has a small window to prove they can turn around the corporations data and metadata needs.
- Lack of workers for the task
- Over 40% of Analysis and Developer time is spent on Maintenance
- A CIO has been burned before by the latest fad or vendor product.



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- Difficult Issues



# Vendor products

- Not all of them are good for your company.
- Why Vendor Implementations fail:
  - They brush over the analysis of the Reference Tables, no true integration.
  - Scoping problem - Try to resolve too much. Enterprise driven instead of Business Sector/Unit driven.
  - Some have standard “Golden Source” table.
  - They ignore the metadata analysis and modeling aspect of the projects.
  - They take 20 year old data structures and accept that as the Golden Source



# Vendor products

- How they can help:
  - Great for uncovering bad data
  - One tool holds all reference data in that Silo or Business Sector/Unit. Makes it easier to consolidate
  - Some have Artificial Intelligence
  - Some have cross platform tracking
  - Lots of Bells and Whistles



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- Difficult Issues



# Analysis

- Data Analysis and Metadata Analysis are needed
  - Do not confuse the two.
  - Must be done at the same time
  - Include naming standards, abbreviations, definitions, and integration issues
- It is a joint IT and Business collaboration
- Determine what Type of Table/Dimension is needed
- Determine what Database(s) you will use
- Loading, Presentation, and Maintenance issues must be resolved before coding begins.
- Avoid the **BIG** roll-out syndrome
- Don't be afraid of looking toward the future (what-ifs)



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- Difficult Issues



# How To ...

## Gather the information

1. Analyze what the data means, valid codes, etc and put that description in the Parking Lot Data Model.
2. Gather all like reference tables from a Business unit and load those structures into a Parking Lot Data Model.
  - Position all like tables (entities) on the same page of your data model.
  - Do not forget other reference data (groups, supporting codes) that feed these tables.
3. Create a Repository to hold your additional information
4. Validate with SME Analysis/Verification
5. Next Steps

# How To ...

## 1. Analyze

- Do these mean the same thing?
- Add any information to the description into the data model.
- Why is RISK\_TYPE\_CD in here?
- What is the Key for the new table?
- Put those elements in a parking lot (below)

### DIM\_EDW\_AGMT\_TYPE

DIM\_EDW\_AGMT\_TYPE\_ID: NUMBER(15) NOT NULL

AGMT\_TYPE\_CD: CHAR(2) NOT NULL  
AGMT\_TYPE\_SHORT\_NM: CHAR(20) NOT NULL  
AGMT\_TYPE\_LONG\_NM: CHAR(50) NOT NULL

### DIM\_ERD\_AGMT\_TYPE

AGMT\_TYPE\_CD: CHAR(2) NOT NULL

AGMT\_TYPE\_SHORT\_NM: CHAR(20) NULL  
AGMT\_TYPE\_LONG\_NM: CHAR(50) NULL

### CORP\_RMX\_AGREEMENT\_TYPE

TTY\_FAC\_DIR\_TYPE: CHAR(3) NOT NULL  
RISK\_TYPE\_CD: CHAR(2) NOT NULL

AGRMT\_TYPE\_NAME: CHAR(40) NOT NULL



# How To ...

## 2. Build a Parking Lot Data Model

- Assemble all Reference data for a Master Reference Data element.
- Include descriptions, usage (if known) and possible valid values.
- Use your target Database in the Data Model to create the new integrated table.
- Create a Usage Repository (See next page)



# How To ...

3. Build a Repository – If you have time
  - Use MS Access DB to track usage, values and descriptions
  - Do not forget the Project and Business Sector/Unit Information
  - Forerunner for your Metadata Strategy
  - Expect Business users to come to you about Business Term capturing



# How To ...

## 4. SME Analysis

- First on the Silo Level
- Both Business and IT reps need to be involved.
- May be more than 1 view of the Reference Data (mislabeling)
- Ask Business users for Business Term definitions



# How To ...

## 5. Next steps

- Expect the Business Sector consolidation effort to get legs and become the “Golden Source”
- Begin to formulate the Enterprise Data Council at this point
- Be aware of Bullies and Wimps. Have the right view.
- Be prepared to let some business users walk away.
- Publish and Promote (Intranet, SOA)



# AGENDA

- Reference Data Problems
- Scoping
- True Upper Management Support
- Vendor Products
- Analysis
- How to...
- **Difficult Issues**

# Difficult Issues - Politics

Never try to teach a pig to talk.  
It annoys the pig and frustrates you.





# Difficult Issues – 80/20 Rule

- Do not waste time running after ghosts. You can revisit it later.
- Business Sector Infighting
  - Assign a Data Steward, Owner
  - There may be more than 1 view of Reference table.
- Avoid the “Tyranny of the Urgent”.



# Difficult Issues – Enterprise

- Concentrate on the Business Sectors first, but keep a higher perspective.
- Must establish a Data Management Council to govern this.
  - Must consist of all business units and a Corporate wide focused member.
  - If you can not establish this, do not attempt to do it at the Enterprise level.
- Create a Roadmap.

# Difficult Issues – Grouping

You may need to group several Reference Tables together. An Assignment table can help.

## EDW\_RMX\_HRCHY

```
EDW_RMX_HRCHY_ID: NUMBER(10) NOT NULL
HRCHY_NM: VARCHAR2(50) NOT NULL
COMPL_HRCHY_IN: CHAR(1) NOT NULL
HRCHY_DEPTH_LVL_NO: NUMBER(1) NOT NULL
HRCHY_EFFEC_DT: DATE NOT NULL
HRCHY_EXPIR_DT: DATE NOT NULL
LVL1_CD: CHAR(3) NOT NULL
LVL1_SHORT_NM: VARCHAR2(30) NOT NULL
LVL1_LONG_NM: VARCHAR2(50) NOT NULL
LVL1_LVL_NM: VARCHAR2(50) NOT NULL
LVL1_EFFEC_DT: DATE NOT NULL
LVL1_EXPIR_DT: DATE NOT NULL
LVL2_CD: CHAR(3) NOT NULL
LVL2_SHORT_NM: VARCHAR2(30) NOT NULL
LVL2_LONG_NM: VARCHAR2(50) NOT NULL
LVL2_LVL_NM: VARCHAR2(50) NOT NULL
LVL2_EFFEC_DT: DATE NOT NULL
LVL2_EXPIR_DT: DATE NOT NULL
LVL3_CD: CHAR(3) NOT NULL
LVL3_SHORT_NM: VARCHAR2(30) NOT NULL
LVL3_LONG_NM: VARCHAR2(50) NOT NULL
LVL3_LVL_NM: VARCHAR2(50) NOT NULL
LVL3_EXPIR_DT: DATE NOT NULL
LVL3_EFFEC_DT: DATE NOT NULL
LVL4_CD: CHAR(3) NOT NULL
LVL4_SHORT_NM: VARCHAR2(30) NOT NULL
LVL4_LONG_NM: VARCHAR2(50) NOT NULL
LVL4_LVL_NM: VARCHAR2(50) NOT NULL
LVL4_EFFEC_DT: DATE NOT NULL
LVL4_EXPIR_DT: DATE NOT NULL
CREATE_TS: DATE NOT NULL
CREATE_USER_ID: VARCHAR2(15) NOT NULL
LAST_UPDATE_TS: DATE NOT NULL
LAST_UPDATE_USER_ID: VARCHAR2(15) NOT NULL
ESTG_CD_ETL_ID: NUMBER(10) NOT NULL
```

## EDW\_RMX\_ASSIGN

```
EDW_RMX_ASSIGN_ID: NUMBER(10) NOT NULL
EDW_BUS_XREF_ID: NUMBER(10) NOT NULL
RMX_RULE_YR: NUMBER(4) NOT NULL
CORP_LOB_CD: CHAR(3) NOT NULL
MRM_LOB_CD: CHAR(3) NOT NULL
RMX_ORG_ID_EDW_RMX_HRCHY_ID: NUMBER(10) NOT NULL (FK)
RMX_MRM_ISSNG_COM_RMX_HRCHY_ID: NUMBER(10) NOT NULL (FK)
RMX_AGRMT_ID_EDW_RMX_HRCHY_ID: NUMBER(10) NOT NULL (FK)
RMX_COVGE_ID_EDW_RMX_HRCHY_ID: NUMBER(10) NOT NULL (FK)
RMX_RISK_ATT_TYPE_ID_HRCHY_ID: NUMBER(10) NOT NULL (FK)
RMX_XL_LYR_POSITN_ID_HRCY_ID: NUMBER(10) NOT NULL (FK)
RMX_CORP_LINE_ID_RMX_HRCHY_ID: NUMBER(10) NOT NULL (FK)
RMX_MRM_LOB_ID_RMX_HRCHY_ID: NUMBER(10) NOT NULL (FK)
CREATE_TS: DATE NOT NULL
CREATE_USER_ID: VARCHAR2(15) NOT NULL
LAST_UPDATE_TS: DATE NOT NULL
LAST_UPDATE_USER_ID: VARCHAR2(15) NOT NULL
ESTG_CD_ETL_ID: NUMBER(10) NOT NULL
```

# Final Thoughts

Avoid **Werewolfing**  
(no silver bullets)



- Must have the good Data Modeling and Metadata Analysis skills handy.
- Don't get hung up on obscure reference tables that one group uses.
- Have weekly/bi-weekly status meetings to confirm/correct the direction.



# Questions ?

[Robert.Schork@citi.com](mailto:Robert.Schork@citi.com)

[bobschork@hotmail.com](mailto:bobschork@hotmail.com)

Team Leader, AVP

Citi Architecture and Technology  
Engineering (CATE),

Development Engineering,

Citi, Warren, NJ