

Converting Legacy Data

Rob Yetter

Doug Bowser

Coeus User Group Conference 2008

Converting Legacy Data

- Overview of legacy data conversion into Coeus
- Case study: University of Maryland – Baltimore
- Questions

Do you Need to convert legacy data?

If you are going to use Coeus to feed a central data source you could just enter active information and merge the legacy data with the new Coeus data within the data mart /data warehouse

So you really need to convert legacy data...

- Do not reinvent your legacy system
 - You switched to Coeus for a reason, so try to exploit its strengths
 - Coeus was built with the input of numerous institutions, it should do all the things you need it to do
- Analyze how Coeus works
- Compare your business process to Coeus and compromise

Resources

- Converting legacy data involves technical and functional/ business resources
- Tools
 - What technologies do you have access to?
 - What skill sets do your people have?
 - What hardware do you have available for the conversion?

Assemble the Project team

Converting data requires cooperation among different groups

Research Office

- Subject matter experts
- Opportunity to develop an in house Coeus expert
- Should be individuals with a good functional understanding of sponsored research and some technical knowledge

Technical Office

- Provide the application environments
- Provide the resources to do the coding
- Will eventually support the software
- May not be able to dedicate same resource down the road

Project Manager

Methods of Loading Data

- Scripts from a flat file
- Database links and stored procedures
- Intermediary DB
- Code that reads- translates- writes to database

Characteristics of a good conversion process

- Method needs to also allow for multiple iterations of conversion
- Method should be incremental
- Conversion process will need QC along the way
- Need to be able clean out and replace data

Data structuring questions

- What modules will you be using?
 - Institute Proposal, Award, Protocol
- Map legacy data to Coeus data
- How do you want to structure your data?
 - Award Hierarchy
 - Contract data/ Grant data
 - How will you store larger funding structures – Task orders, Program Project Grants, Center, Core grants, etc.

Options of last resort

- Custom data fields
 - Advantage: Does not include modifying code
 - Disadvantage: Makes reporting more difficult
- Shadow systems to supplement Coeus
 - Advantage: Enhanced data access
 - Disadvantage: Another system to maintain
- Customizing the application
 - Advantage: Can do whatever you want
 - Disadvantage: upgrades

Supporting data areas

- Code tables

Review code tables and edit as permitted by the system

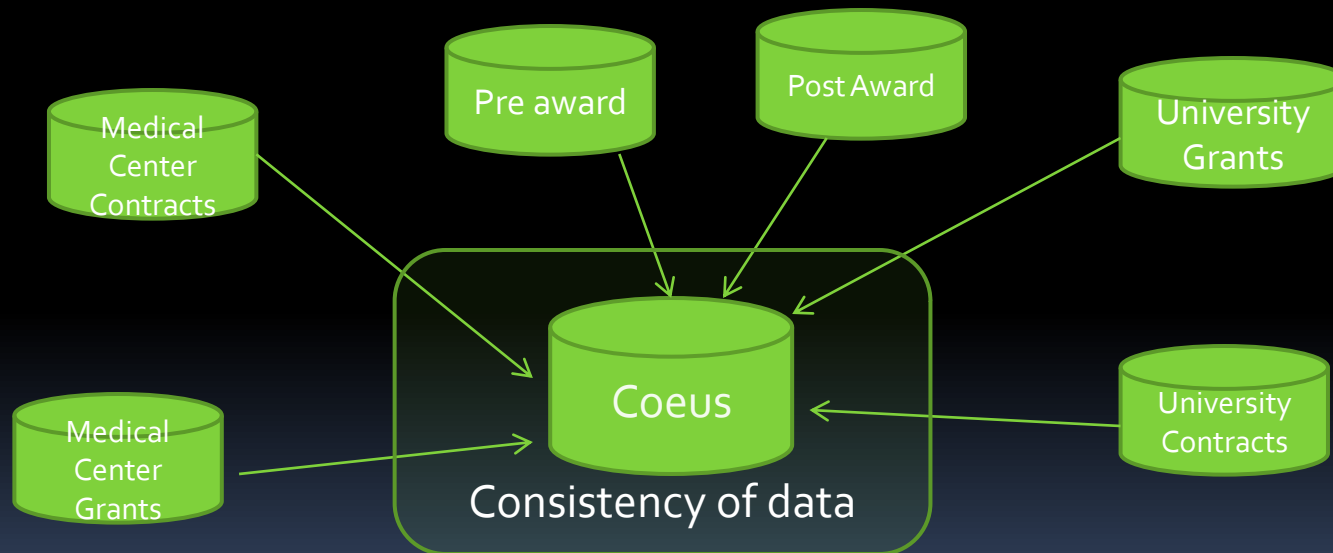
- Base tables

- Person
- Sponsor
- Rolodex
- Unit tables
- Organization

Combining data sources

University and Medical Center
Pre and Post award offices

Combining a grants and a
contracts database



Preparing the data

- Define the data before loading into Coeus
 - Load all historical data?
 - Load all types of data in the legacy system?

Preparing the legacy data

- Common Problems with legacy data
 - Data quality/ completeness is poor
 - Inconsistent design between old system and Coeus
 - Lack of knowledge/ documentation about the legacy system

Typical Legacy Data Problems

Problem	Potential Database Refactoring
A single column is used for several purposes	Split Column (to Notes)
The purpose of a column is determined by the value of one or more other columns	Remove Unused Column (to remove Date Type) Split Column (to Person Date)
Inconsistent data values	Introduce Trigger(s) for Calculated Column (between Birthdates and AgeInYears) Remove Redundant Column (to AgeInYears)
Inconsistent/incorrect data formatting	Introduce Common Format
Missing data	N/A
Missing columns	N/A
Additional columns	Introduce Default Value to a Column Remove Redundant Column
Multiple sources for the same data	N/A
Important entities, attributes, and relationships are hidden and floating in text fields	Replace Blob With Table Split Column
Data values that stray from their field descriptions and business rules	Split Column
Various key strategies for the same type of entity	Consolidate Key Strategy For Entity
Unrealized relationships between data records	Introduce Explicit Relationship
One attribute is stored in several fields	Combine Columns Representing a Single Concept
Inconsistent use of special characters	Introduce Common Format
Different data types for similar columns	Apply Standard Types to Similar Data
Different levels of detail	Introduce Calculated Column Replace Column
Different modes of operation	Separate Read-Only Data
Varying timeliness of data	Separate Data Based on Timeliness
Varying default values	Introduce Default Value to a Column
Various representations	Apply Standard Codes Apply Standard Types to Similar Data

Develop QC checks

- Test the data prior to loading
- While it is loading
- And after
- Use QC reports
 - Either SQL queries
 - Or the reporting tool you will use to report out of Coeus

Coeus objects

- Coeus creates sequences for proposals and awards
- Coeus uses sequences to build IP number and award number
- Referential Integrity constraints force the data to be loaded correctly
- You do not need to load every table

Load most data manually enter some

- There maybe some data that would be easier to manually enter rather than coding the conversion
- What will you need to fix after loading

Run systems in parallel

- Enter production data into Coeus and your legacy system to ensure consistency
- Validate reporting from both systems – Are you getting the same result from Coeus as you are from the legacy system?

Helpful links

- Coeus.org
 - ER Diagram
 - Structuring Awards
 - Code tables
 - Person
 - Unit Hierarchy
- [Scott W. Ambler Essay on Legacy data](#)

University of Maryland – Baltimore

Data conversion experience