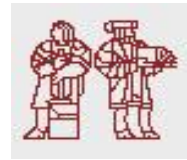




The MIT Information Quality Industry Symposium, 2007



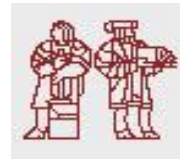
# ***Enterprise Data Validation Architecture (EDVA):***

***Fixing Data Quality for Enterprise Interoperability***

Cambridge, Massachusetts, USA

July 18-19, 2007

Bill McMullen




# Outline

- Purpose
- Nature of the Problem
- Elements of a Solution
- Business Case Considerations
- Q&A

**Business Enterprise Architecture (BEA) & Enterprise Transition Plan (ETP)**  
September 2006

An Objective – “Transform the Department’s supply chain information environment by: 1) improving data integrity and visibility by defining, managing, and utilizing item, customer, and vendor master data; and 2) reducing complexity and minimizing variability on the supply chain business transactions by adopting standardized transaction and business rules.”


Department of Defense  
Enterprise Architecture Federation Strategy



Draft Version 1.0  
18 September 2006

Prepared by the DoD CIO

DoD 8320.02-G

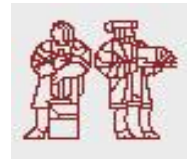


**Guidance for Implementing Net-Centric Data Sharing**

April 12, 2006  
Assistant Secretary of Defense for Networks and Information Integration/  
Department of Defense Chief Information Officer

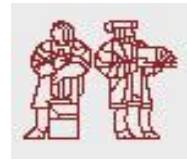
CHAPTER 4 - DATA SHARING IMPLEMENTATION

- C4.1. CHAPTER OVERVIEW
- C4.2. MAKING DATA VISIBLE
- C4.3. MAKING DATA ACCESSIBLE
- C4.4. MAKING DATA UNDERSTANDABLE
- C4.5. PROMOTING TRUST



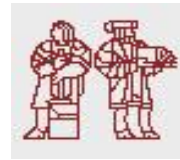
## Purpose

- Purpose of Briefing
  - Provides an overview of an Enterprise Data Validation Engine  
**(This may be an important management option for certain business cases.)**
- Purpose of an Enterprise Data Validation Engine
  - Aligns data across previously independent legacy system sources
  - Eliminates routine manual reconciliation efforts previously needed to coordinate legacy data sources
  - Improves decision making at the domain and/or enterprise levels through enhanced legacy data quality
  - Can be applied in many ways across the enterprise



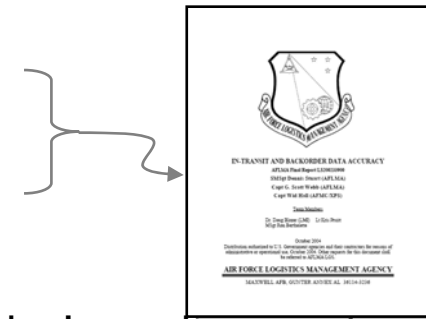
## Decision Making Capability for the Next Decade

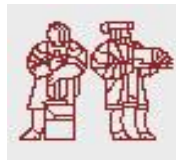
- ERP promises highly accurate data and superior decision making ability, but a wait is involved
  - ECSS is expected to be deployed about 2011+
- It is not necessary to wait to increase data and decision integrity
  - Selected “legacy” systems, with minimal effort, can provide more integrity in the meantime
  - Global Logistics Support Capability (GLSC) requires reengineering both the business processes and their supporting systems
- It is possible to make improvements without distracting from ECSS-ERP
  - The proposed approach will actually facilitate the migration of valid data to ECSS



## Nature of the Problem Regarding Enterprise Data Validation

- The Business Problem
  - Data from separate legacy systems frequently can not reliably be combined at domain or enterprise levels
  - When combined, may not yield trustworthy results for quality enterprise decision making
  - Examples
    - **Case #1: 29% mismatch among two requisition systems (re GLSC)**
      - “Almost 30 percent of Air Force backorder data is inaccurate.”
      - “Forty-two percent of in-transit records ... were invalid. This equates to 4,627 transactions, valued at \$325M.”
    - **Case #2: 98% failure to integrate databases across domain**
    - **Case #3: 25% mismatch (est.) among several vehicle systems**
  - Manual reconciliation to compensate is not satisfactory in today's environment (AFSO21)
    - Airman resources no longer will be available to “fix” data (re GLSC)
- The Technical Issues
  - Synchronization of multiple source system data
  - Association of multiple source data



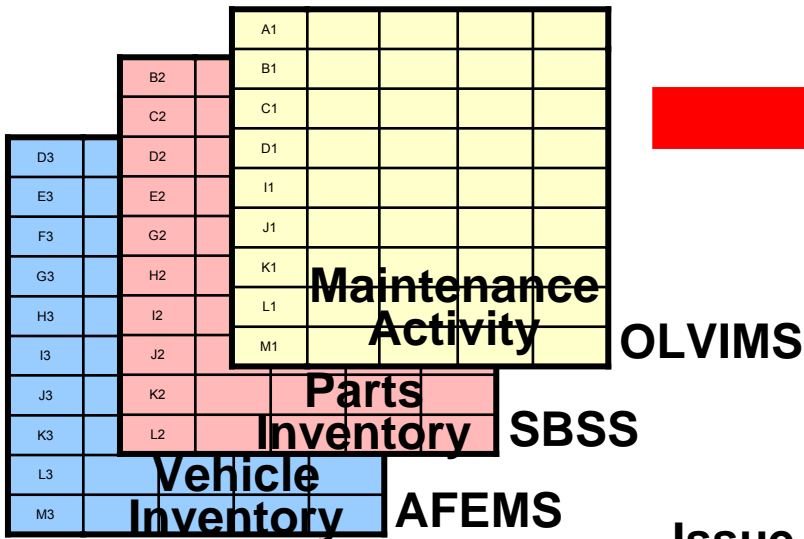


# The Systems Gap: Misaligned Data Leads to Questions

Source systems often describe different aspects of the same asset.

Integration blends the data from the multiple source systems into a global “mismatched” view.

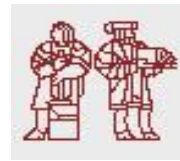
## Data for Vehicles Within 3 Systems



A				VOID?						VOID?	
B											VOID?
C											
D											
E											
F	VOID?			VOID?							
G											
H											
I											
J	Maintenance Activity					Parts Inventory				Vehicle Inventory	
K											
L											
M								VOID?			

Note: The Synchronization Issue shows up in a database when data from multiple source systems does not correspond across the sources in a logical manner, or as expected, given the business rules.

**Issue – Time and Instance Integrity Across Systems**  
**Problem – Reporting Produces Inconsistencies**  
**(Q: How much can we trust this?)**



# Enforce Business Rules & Link Records

## Are the Voids in Gray Valid?

**Equipment Information (SBSS & AFEMS)**

A1				
B1				
C1				
D1				
I1				
J1				
K1				
L1				
M1				

B2								
C2								
D2								
E2								
G2								
H2								
I2								
J2								
K2								
L2								

**Organizational Information (MPES)**

D3		
E3		
F3		
G3		
H3		
I3		
J3		
K3		
L3		
M3		

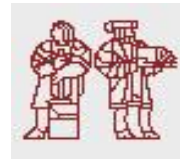
**Aircraft Information (ForceTab)**

A1		
B1	B2	
C1	C2	
D1	D2	D3
	E2	E3
		F3
	G2	G3
	H2	H3
I1	I2	I3
J1	J2	J3
K1	K2	K3
L1	L2	L3
M1		M3

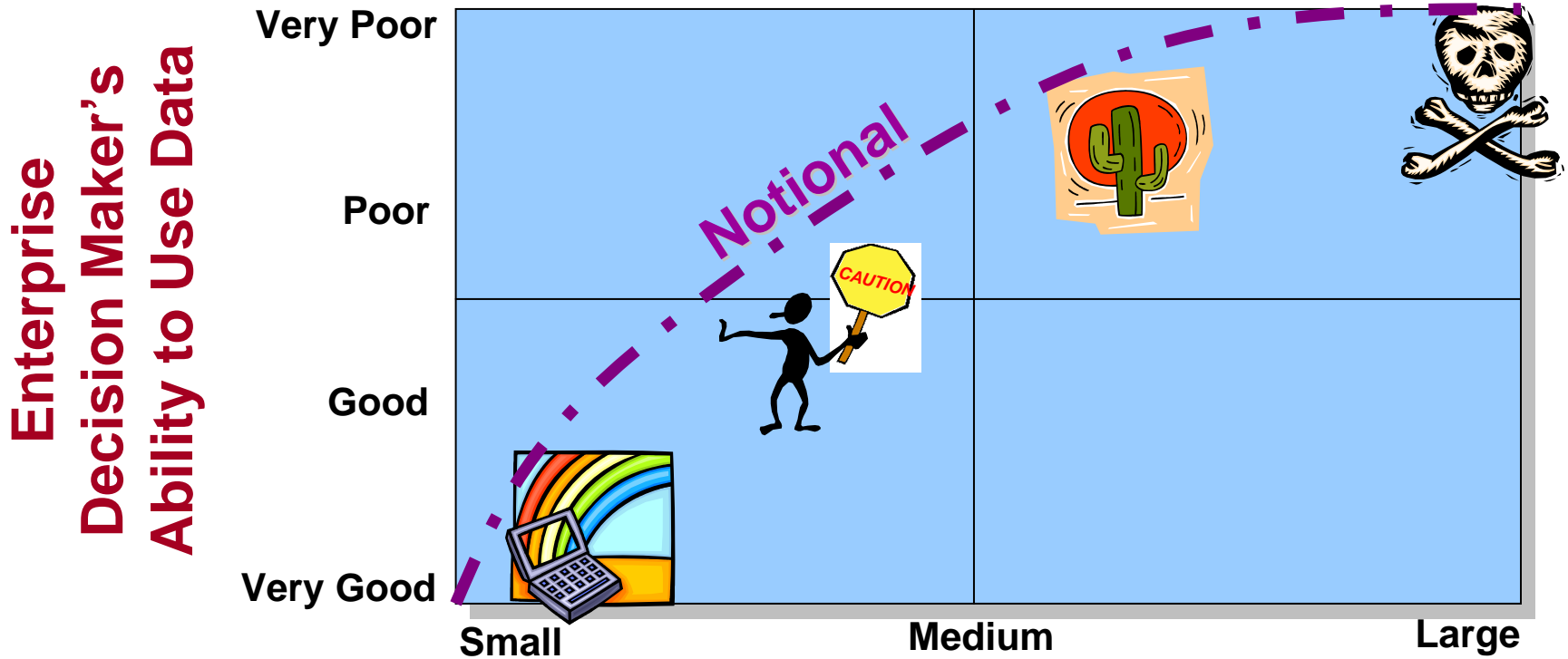
**Cross-Reference Information (GFM-DI)**

- Check against enterprise business rules
- Make corrections at this point or they usually won't (can't) get done later

**Note:** The Association Issue shows up in a federated database when invalid associations among object is erroneously permitted. Validation against business rules must be performed prior to a source system allowing such a transaction to update its own database.



# Problems Compound as More Systems Are Considered

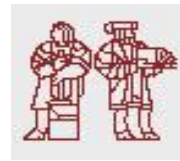


\* Only the topic of Object Association is addressed hereafter, because the solution to this single problem facilitates the solution of Attribute Synchronization (even though its solution can be attempted separately).

## Compounding of Missing, Mismatched, or Unsynchronized Data

(Due to improperly associated and serialized enterprise-wide data and enforcement of enterprise-wide business rules)



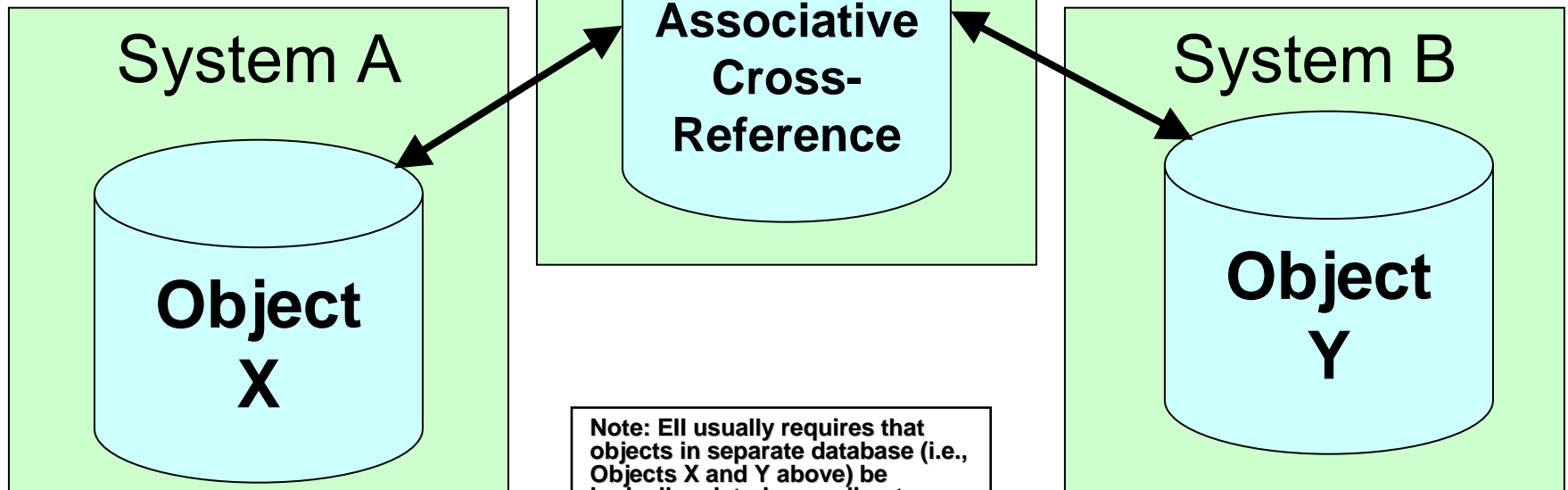


# Some Elements of a Solution for Using an Enterprise Data Validation Engine

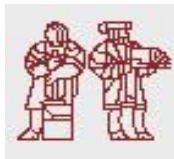
## Associations Across Systems

Determine how different objects of two systems are associated.

New business rules must be enforced by Systems A and B.



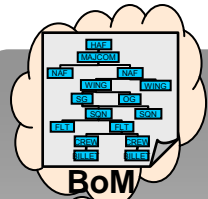
Note: EII usually requires that objects in separate database (i.e., Objects X and Y above) be logically related according to certain enterprise business rules.



# Leveraging the Air Force Enterprise Data Validation Architecture

Unclassified / NIPRNet

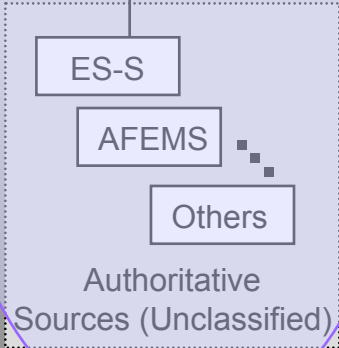
Classified / SIPRNet



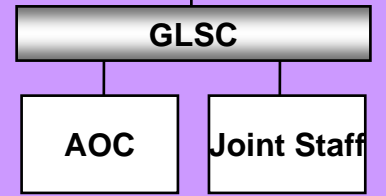
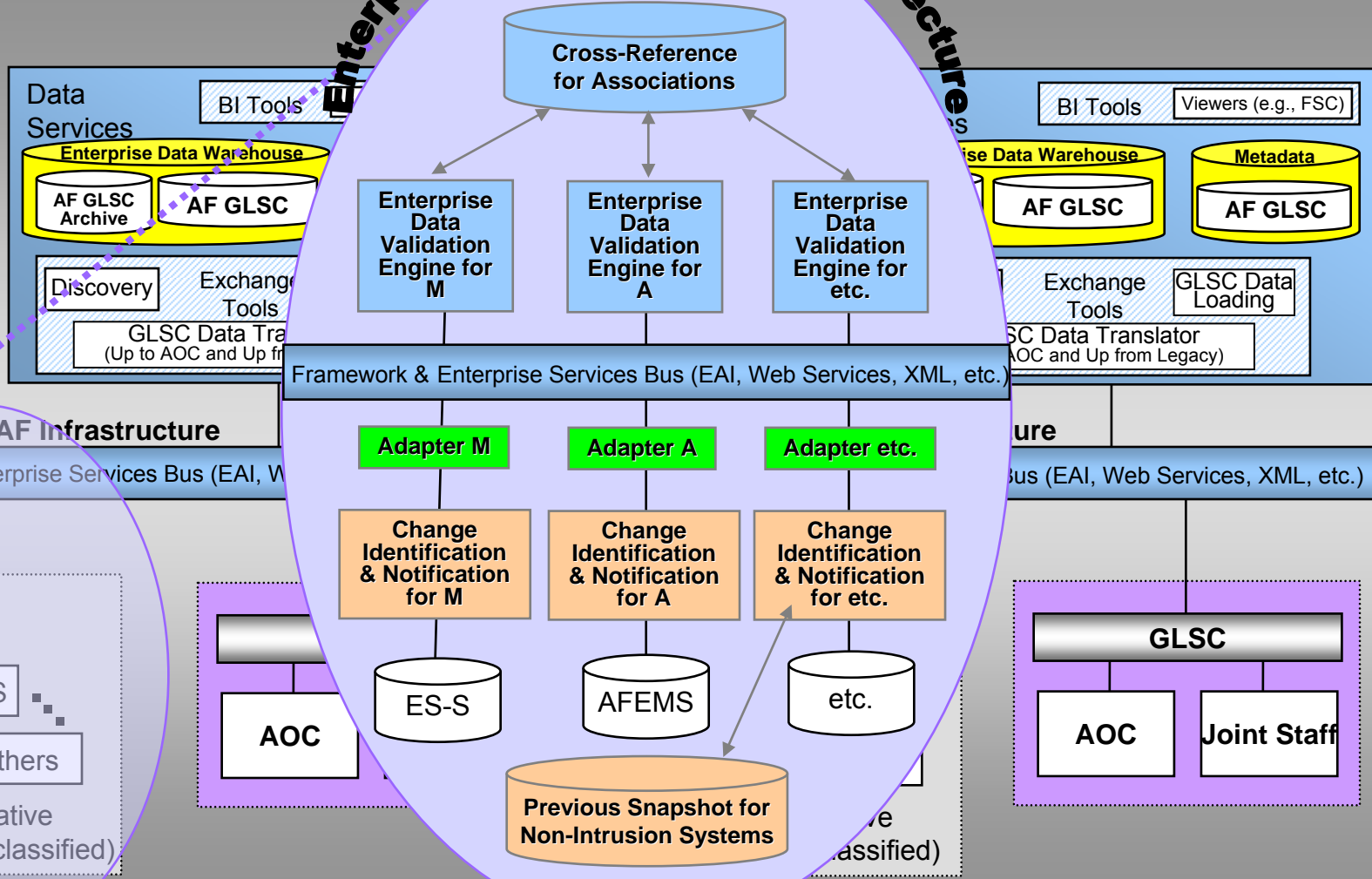
AF Portal

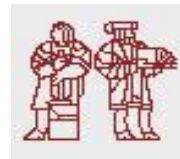
GCSS-AF Infrastructure

Framework & Enterprise Services Bus (EAI, Web Services, XML, etc.)



## Enterprise Data Validation Architecture

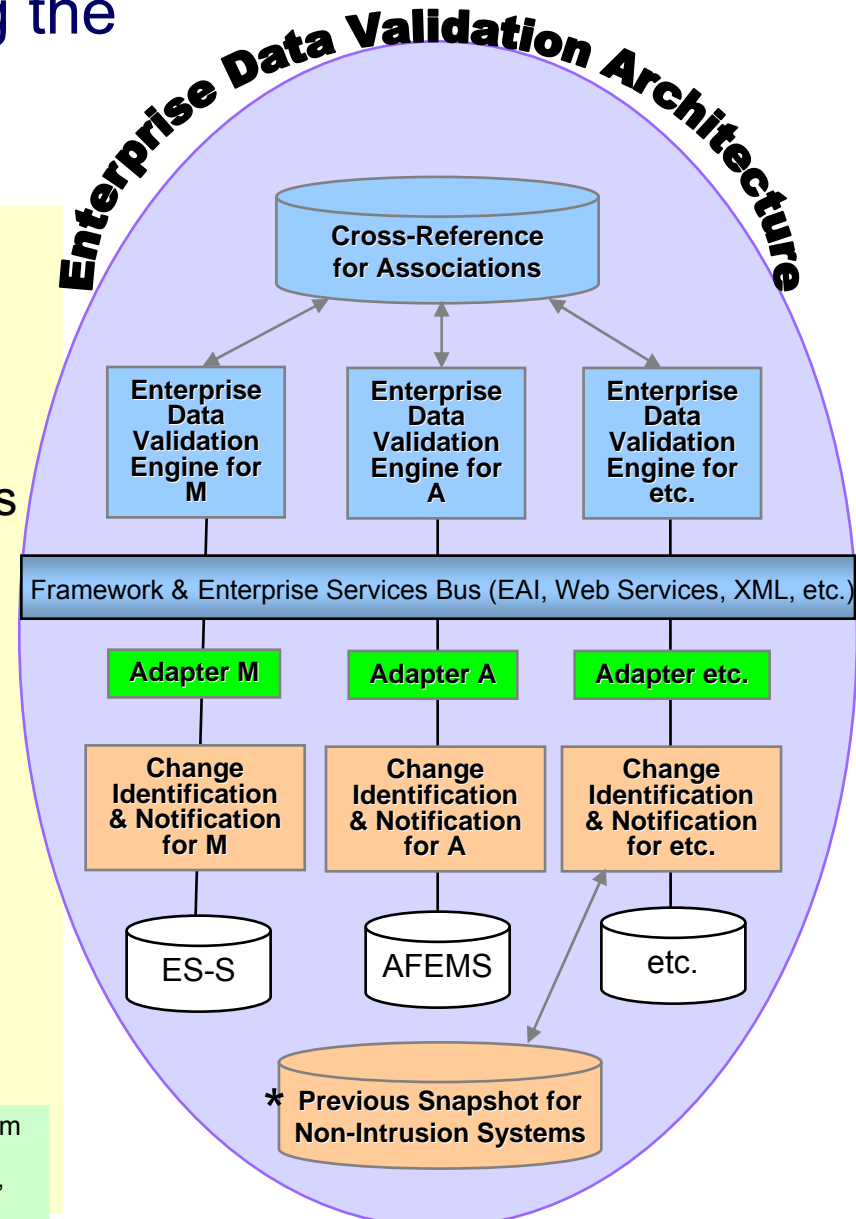




# Source System Architecture Using the Enterprise Data Validation Engine

- Leads to greatly enhanced data agreement across systems that currently lack common standards or interoperability
- Enterprise Data Validation Engine ensures data integrity as it trickles in from the various systems
- Enterprise Data Validation Architecture
  - Components include
    - Cross-Reference for Associations
    - Enterprise Data Validation Engine
    - Change Identifier and Notifying
    - Interoperability Protocols (7 possible choices)

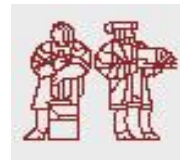
[Database Snapshot for Non-Intrusion\*]



\* Can be non-intrusive to source system, but with critical tradeoffs


- Performance
- Data Quality

This architecture is an adaptation from "Protocols for Integrity Constraint Checking in Federated Databases", P. Grefen and J. Widom, 1997



# Data Entry Scenario Where Legacy Modifications Allowed

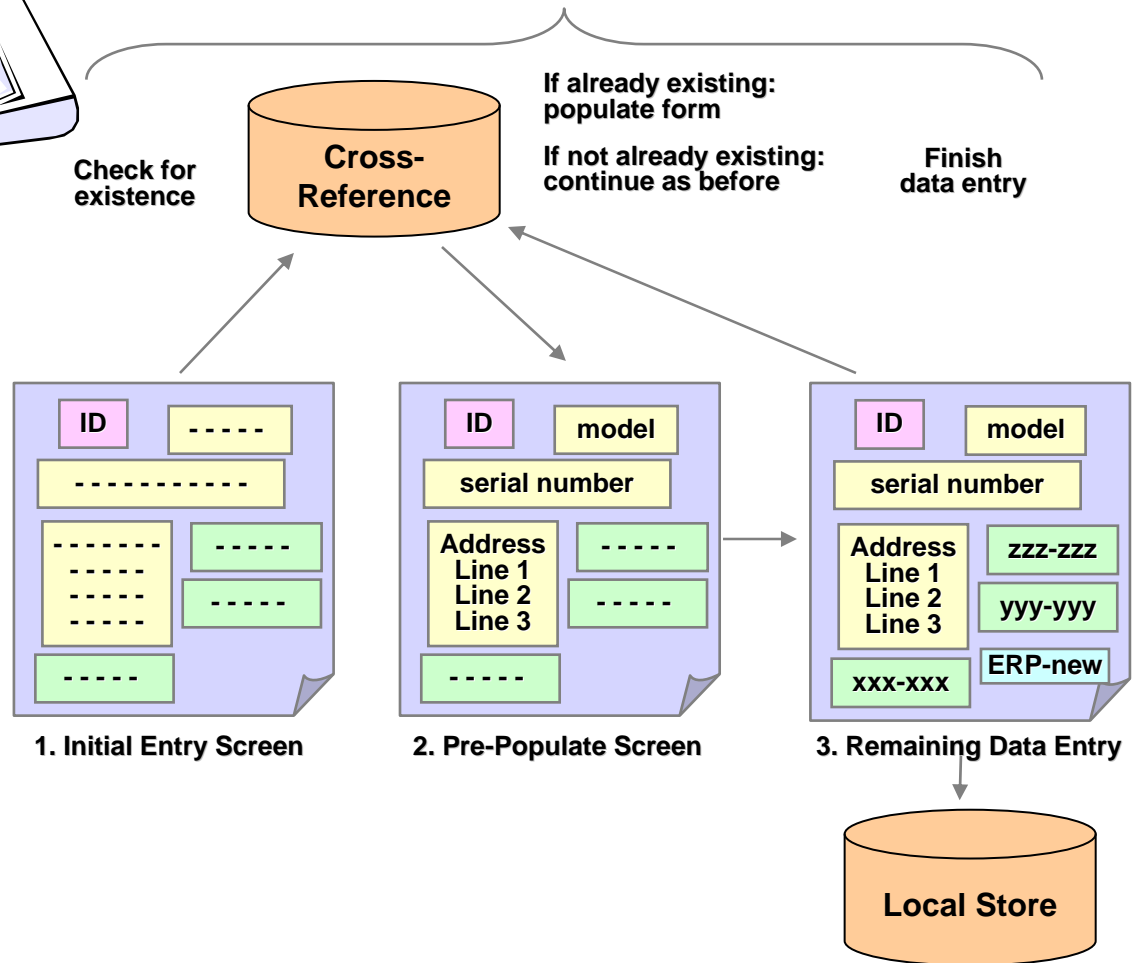
## Current Legacy Data Entry Process

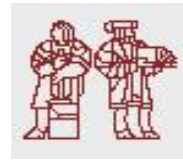


ID	model
serial number	
Address Line 1	zzz-zzz
Address Line 2	yyy-yyy
Address Line 3	
xxx-xxx	



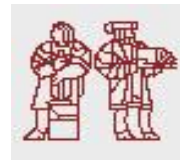
## Improved Legacy Data Entry Process





## Business Case Considerations

- **Program-Centric Aspects**
  - **Development Effort and Time [Investment]**
    - **Added Effort**
      - **Systems Aspects (enhanced data integrity capabilities vs. system revisions)**
      - **Operational Aspects (elimination of reconciliation activities vs. data entry revisions)**
    - **Coordination Among Other Stakeholders (via a COI?)**
  - **Lifecycle Benefits (may accrue over time) [Return]**
    - **Improved Data Integrity**
    - **Improved Data Timeliness**
    - **Eliminate Redundant Data Entry**
    - **Eliminate Reconciliation Effort**
- **Enterprise-Centric Aspects (in addition to above)**
  - **Generic Solution Template**
  - **Prerequisite to ERPs (DEAMS, ECSS, DIMHRS) Migration**
  - **Improved Integrity of Enterprise-Level Decision Making**



# Considerations for Use of Enterprise Data Validation

*Where would the Enterprise Data Validation Engine be considered for use?*

*After conducting Enterprise-Level Cross-System Data Quality assessments for given products, then perform the following Data Fixing Strategy as applicable.*

Note: The term "data" herein includes any data aggregation (element, record, file, cross-system composite, etc.).

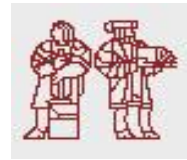
Rule \ Situation	1	2	3	4	5	6	7	8
Can Data Be "Abandoned"?	Yes	No	No	No	No	No	No	
Is Current Quality OK?	-	Yes	No	No	No	No	No	
Feasibly to Re-Entered?	-	-	Yes	No	No	No	No	
Feasible to Reconcile?	-	-	-	Manually	Auto-matically	Auto-matically	Auto-matically	
Can Apply Fix at Migration Time?	-	-	-	-	Yes	No	No	
Fixing Sources Cost Acceptable?	-	-	-	-	-	Yes*	No	
Data Fixing Strategy	Do Nothing	Do Nothing	Manually Re-Entry	Manually Reconcile	Do Data Cleansing	Enforce Enterprise Data Validation	???	

This issue is similar to the Y2K problem in that you can't wait until the last minute to think about or act upon. It has the additional characteristic that the sooner it is addressed the better the data quality gets over time.

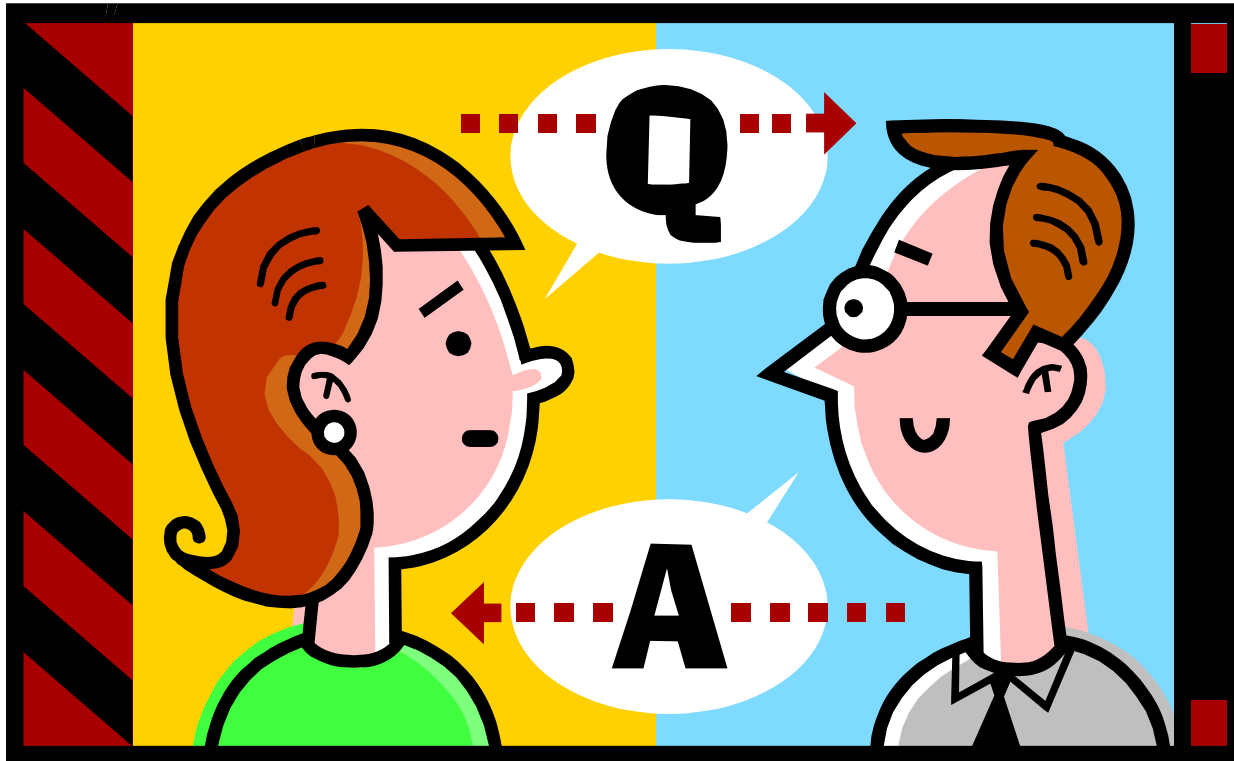
\* If interfaces are being changed anyway, then these costs may be negligible to use EDVA template.

Data Quality Improvements

Indicates Inherent Government Responsibilities

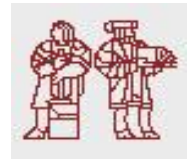


## Questions and Answers



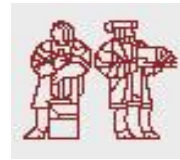


The MIT Information Quality Industry Symposium, 2007



## Backup Slides

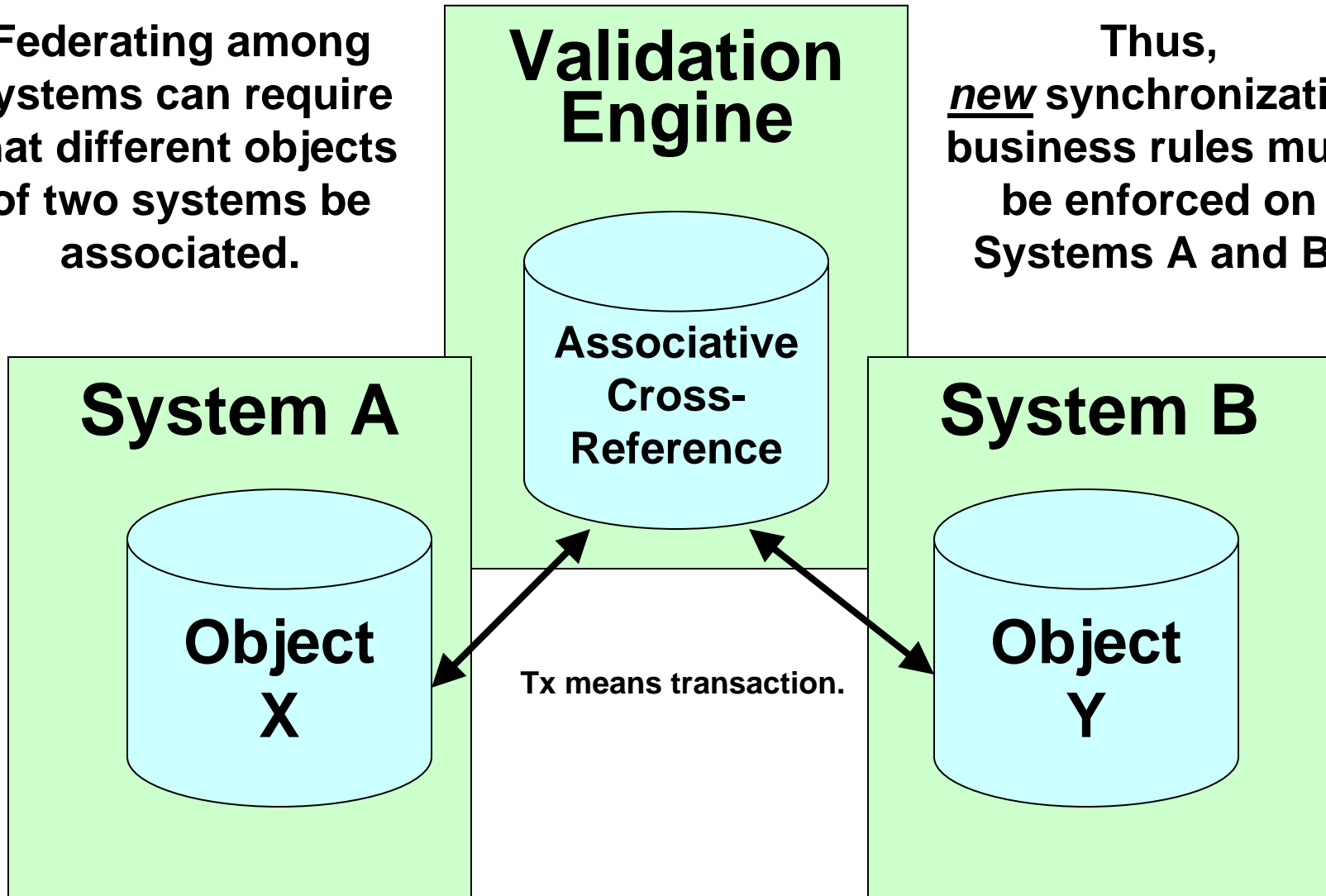


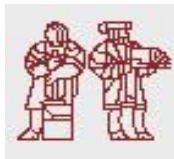


## Associations Between Objects

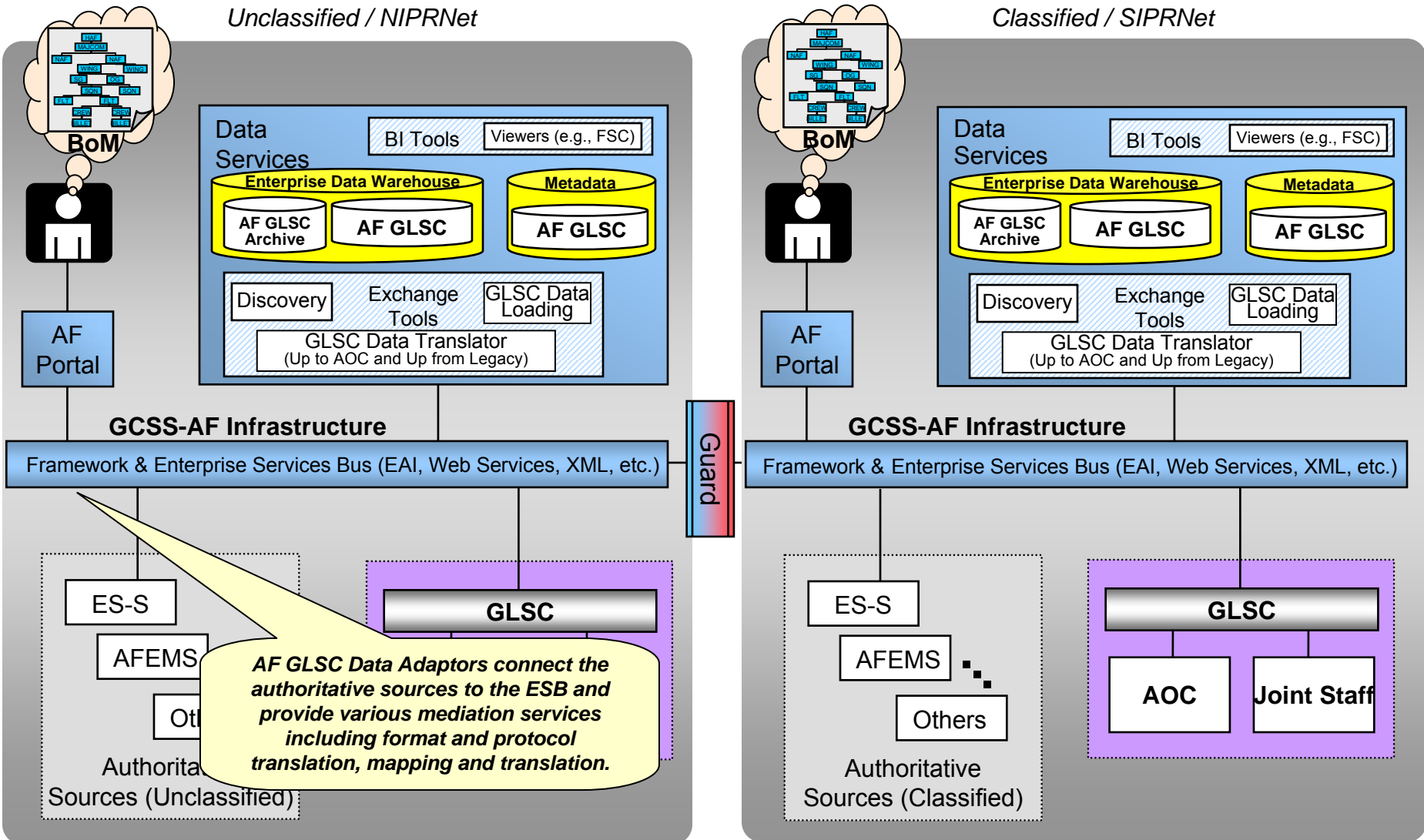
Federating among systems can require that different objects of two systems be associated.

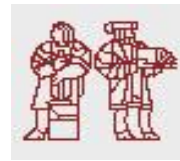
Thus, new synchronization business rules must be enforced on Systems A and B.



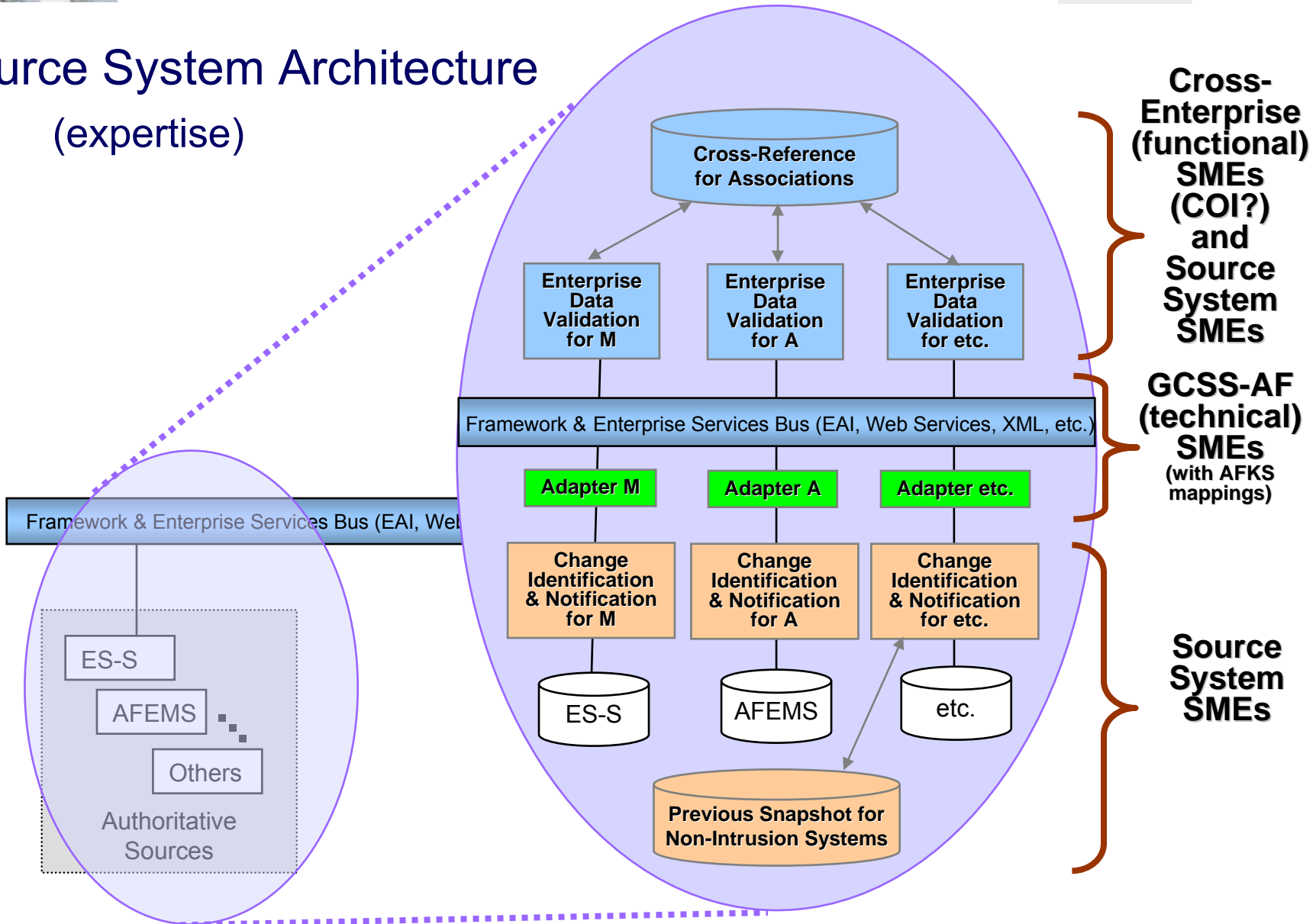


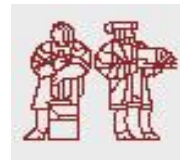
# Vision Architecture (using GLSC example)



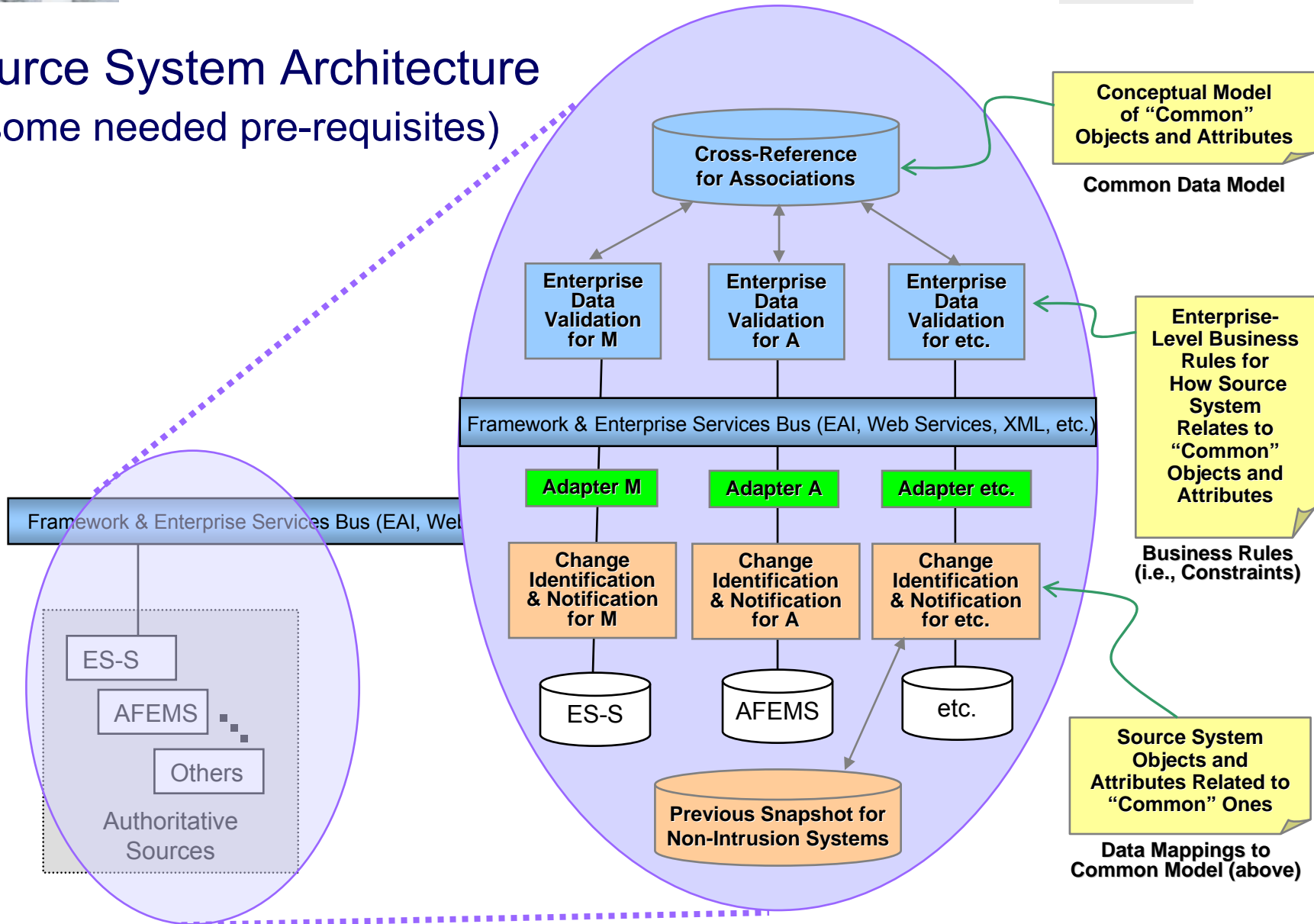


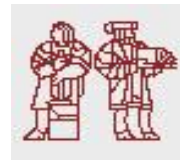
# Source System Architecture (expertise)



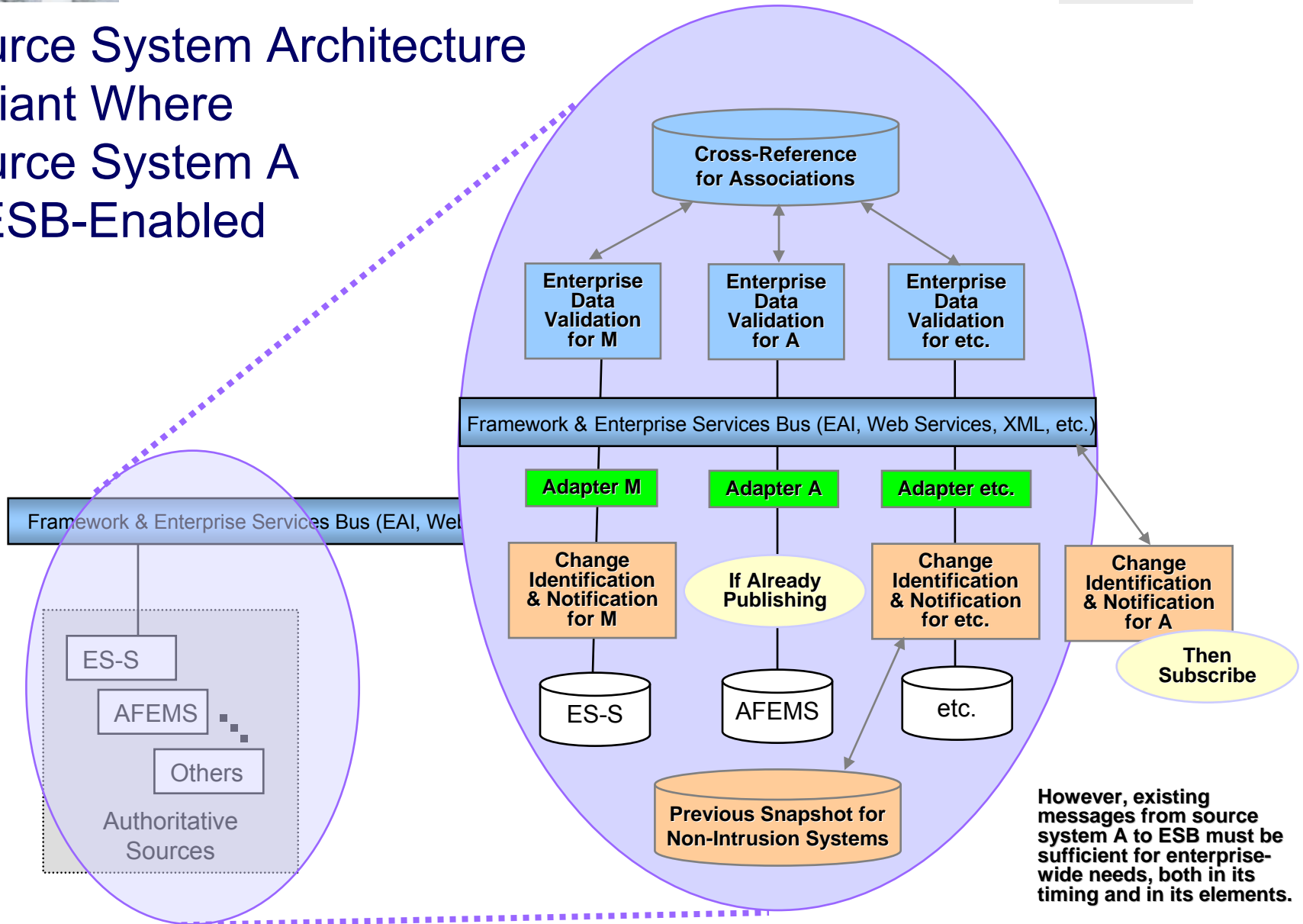


# Source System Architecture (some needed pre-requisites)

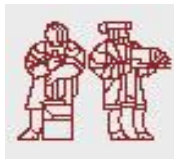




# Source System Architecture Variant Where Source System A Is ESB-Enabled

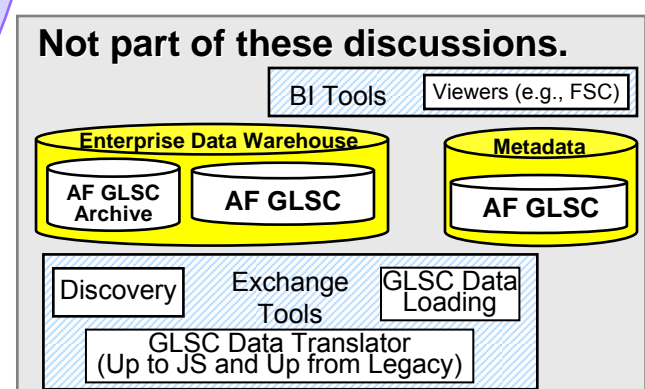
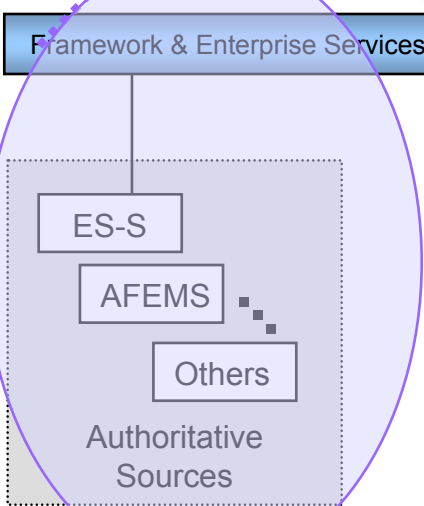
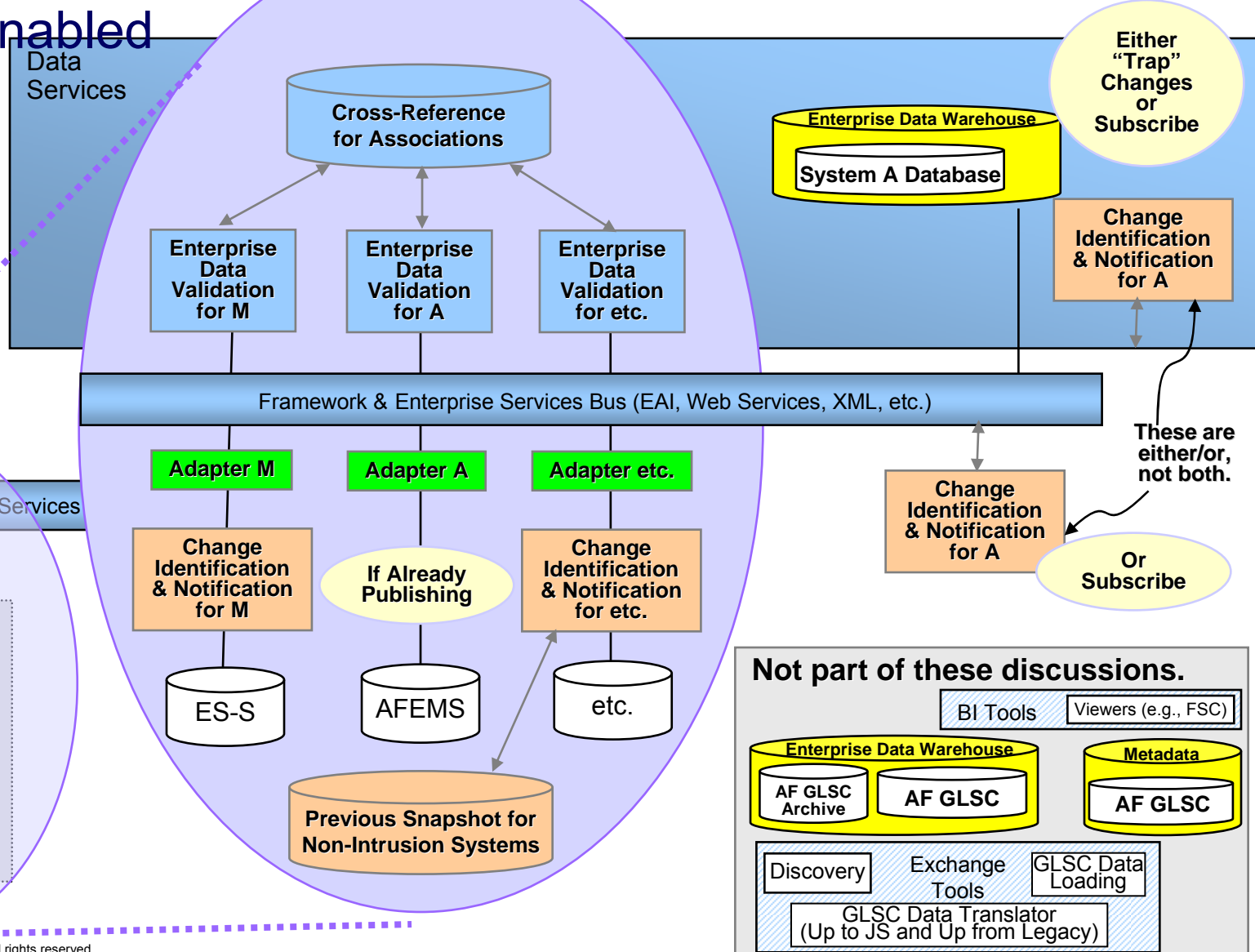


However, existing messages from source system A to ESB must be sufficient for enterprise-wide needs, both in its timing and in its elements.



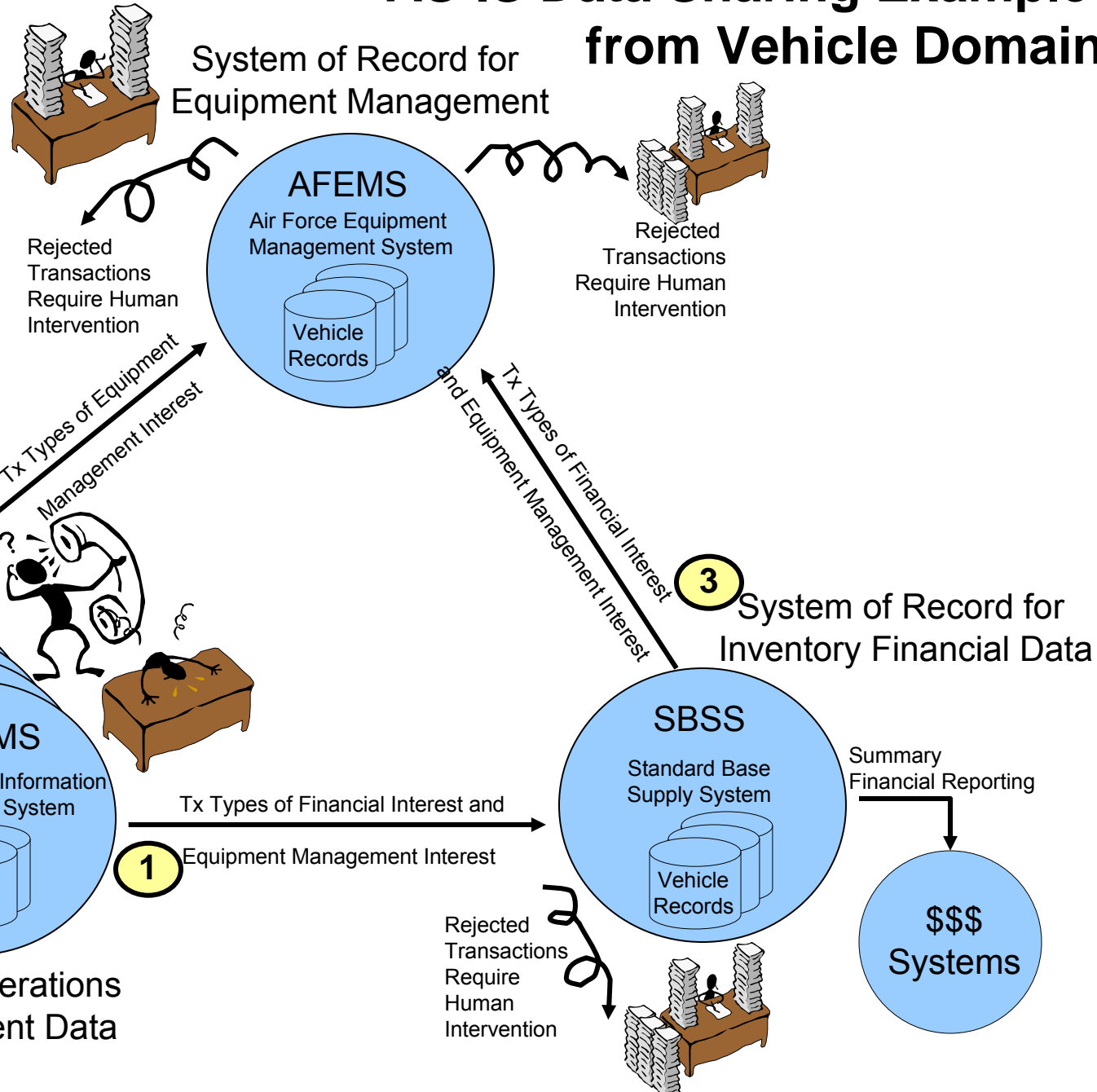
# Source System Architecture Variant where Source System A Is AFKS-Enabled

However, existing messages from source system A to ESB must be sufficient for enterprise-wide needs, both in its timing and in its elements.

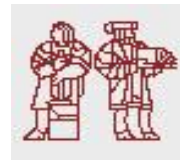


# AS-IS Data Sharing Example from Vehicle Domain

	Core vehicle data copies with additional domain data
	Interfaces with limited automation; mostly abandoned in favor of dual data entry
	Rejected tx have limited notification processes and must each be cleared by workers
<b>1</b>	Tx are edited for syntax by SBSS
<b>2</b>	Tx are edited for syntax and semantics by AFEMS
<b>3</b>	Some tx are simply forwarded after syntax check (no other processing done by SBSS)



- About 200 Sites
- Distributed Data
- Stand Alone Operations



# Vehicle Life Cycle



❖ Need: One Pickup

❖ Acquire

❖ Wait for

**Priority Buy List**  
\$ APPROVED \$

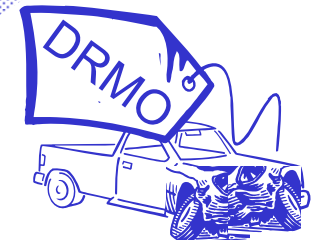
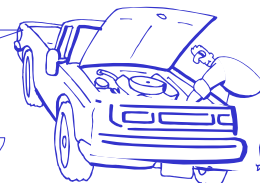
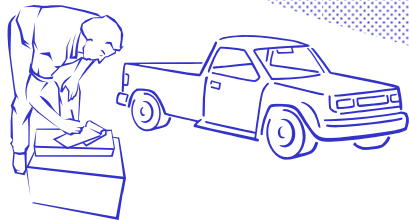
❖ Receive

❖ Operate and Maintain

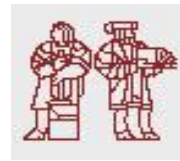
- Mission Capable
- Mission in Progress
- Needs Preventive Maintenance
- Work Order in Progress

❖ Turn - In

Change Highlights	Generating Parties	Interested Parties	Participating Systems
Wait for (SPR)	Acquisition	Vehicle Management, Equipment Management, Inventory Management	OLVIMS, AFEMS, SBSS
Receive (REC)	Vehicle Management	Vehicle Management, Equipment Management, Inventory Management	OLVIMS, AFEMS, SBSS
Operate (mission capable)	Vehicle Operations	Vehicle Management, Equipment Management	OLVIMS, AFEMS
Maintain (non-mission capable)	Vehicle Maintenance	Vehicle Management, Equipment Management	OLVIMS, AFEMS
Maintain (awaiting parts)	Vehicle Maintenance	Vehicle Management	OLVIMS
Turn-In (salvage)	Vehicle Management	Vehicle Management, Equipment Management	OLVIMS, AFEMS, SBSS

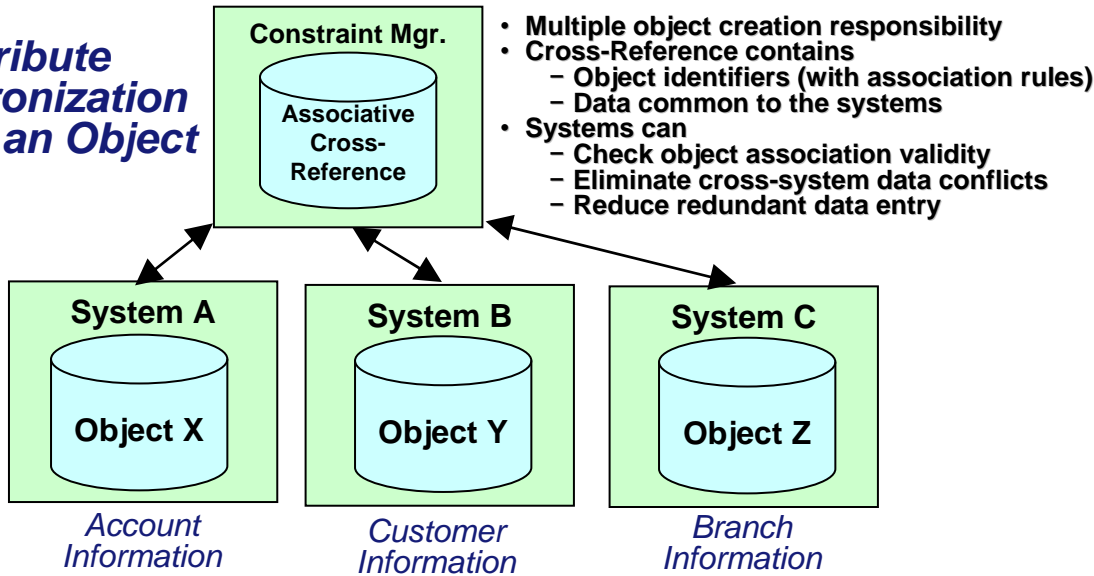






# Characteristics of The Situations

## Attribute Synchronization Across an Object



## Associations Among Objects

