DATE	REVISION	BY
2/22/05	1	

Segment Data Architecture Development Methodology

CLIN 004.2; SOW 1.4.1.1

DATE: **March 22, 2005**

Submitted by the Cairo Corporation Team
Under Contract C-DEN-01916
HUD EDM Practice Support

ENTERPRISE DATA MANAGEMENT GROUP
202-708-1821, ea_team_support@hud.gov

Table of Contents

SECTION 1. INTRODUCTION

1.1 Objectives

The U.S. Department of Housing and Urban Development (HUD) has developed its first release of a target Enterprise Architecture (EA). In order to develop EA in greater detail and make the task more manageable, HUD has determined it must incrementally develop EA by the use of segment architectures. For a HUD segment to complete its architecture, a complete understanding and documentation of its data architecture is required.

The HUD data architecture is a framework for organizing data and its interrelationships to support HUD's business processes and align them with HUD's mission, functions and goals. It provides the underlying set of rules, descriptions of data and the relationships between data that govern how data supports the business processes defined in the HUD Business Reference Model (BRM).

The Segment Data Architecture Development Methodology (SDADM) is based on the findings of the "*Report of the Enterprise Data Management Group, Segment Data Architecture Best Practices Analysis, February 22, 2005*". The Enterprise Data Management Group's (EDMG) report identified eight best practices that were common among selected federal agencies as well as structured methods that contributed to a successful data architecture development. These are

- Top-down and bottom-up analysis
- Incremental development
- Alignment of data and business architectures
- Communicate, coordinate and collaborate
- Business Data Stewardship
- Established and enforced data architecture governance
- Architecture driven modernization efforts
- Centralized and effective access to EA artifacts.

A description of the eight best practices is found in Appendix A. The EDMG studied the data architecture and governance practices of three federal agencies, as well as EA methodology and frameworks, for best practices and artifacts. The Department of Defense Architecture Framework (DODAF) and Information Engineering Methodology (IEM) were chosen as best practices. The notations selected to document the data architecture artifacts within this methodology are primarily derived from these methods.

One goal of the Federal Enterprise Architecture (FEA) framework is to reduce software development cost and foster software or component reuse across the federal government. The FEA BRM provides a framework for federal agencies to identify common business functions across federal agencies. Agencies performing common functions should be able to share components to meet those business needs. The FEA Service Component Reference Model (SRM) provides a framework to record application components and the solution provided.

The Office of Management and Budget (OMB) requires agencies to review the SRM for components that satisfy a software need before submitting any request for funding new software development. [1] The OMB requires that any new development of software must be aligned with the

[1] http://www.whitehouse.gov/omb/circulars/a11/current_year/a_11_2004.pdf, pg, Section 53–4

agency's EA when there is not a component that meets an agency need. The federal solution development life cycle (SDLC) is evolving from one of a **design, code and test** process to one of an **architect, acquire and assemble** process[2].

The SDADM aligns with the three phases of the HUD Information Technology (IT) Lifecycle: architecture, investment and implementation. Please refer to Appendix D for details. The target segment logical data model (LDM) provides the standardized data requirements in order to **architect** a segment and enterprise-wide solution that supports the HUD business mission. The analysis of target and baseline architecture gaps and the identification of natural application components, along with authoritative data sources, provide input to **investment** decisions. The Target Area Data Model (TADM) provides the data specifications to **implement** or refine the authoritative source of data and interface requirements to exchange information.

The creation of components to support HUD's business activities is one goal of the SDADM. The HUD SDADM has the following additional goals:

- **Minimize the risk associated with the modernization of HUD's system portfolio.** The methodology uses an incremental approach that narrows the scope of analysis as the level of detail increases and results in specifications for components used to assemble target applications. This approach limits the risk of the overall modernization effort by providing a manageable scope for the analysis efforts and assembling of applications from proven components. Refer to Appendix C for additional details.

- **Reduce the cost of modernizing HUD's system portfolio.** The methodology aligns the target data architecture to the operational systems and data stores to identify a single, authoritative source of data for target data elements. A single authoritative source of data mitigates the maintenance costs and data integrity reconciliation associated with multiple data sources.

1.2 Scope

The scope of this report is to provide a detailed, technical description of the Segment Data Architecture Development Methodology (SDADM), provide a governance structure and describe alignment between the SDADM and the HUD System Development Methodology (SDM). This report is divided into the following sections:

- Section 2: Segment Data Architecture Development Methodology. This section details each phase of the methodology.
- Section 3: Segment Data Architecture Governance. This section provides a governance structure for segment data architecture development at HUD.
- Section 4: Software Development Methodology Alignment. This section aligns the HUD SDM to the SDADM. The SDADM produces artifacts that provide the information and documentation about an application's data needs that fulfill the requirements stated in the SDM.
- Appendix A: Segment Data Architecture Best Practices
- Appendix B: Parallel Decomposition

[2] Succeeding with Component-Based Architecture in e-Government, Concept Level White Paper, Industry Advisory Council (IAC) Enterprise Architecture SIG, March 2003

- Appendix C: Incremental Development Approach
- Appendix D: HUD Integrated IT Lifecycle Framework
- Appendix E: FIPS PUB 184 Data Modeling Method and Conventions
- Appendix F: Third Normal Form
- Appendix G: Data Quality Measures Documentation Template
- Appendix H: List of Acronyms

1.2.1 Segment Data Architecture Development Methodology

The SDADM is a phased, lifecycle approach that incrementally transitions HUD's conceptual DRM from a business view aligned with HUD's BRM to a physical view used for the implementation of the data specification and information exchanges. The methodology aligns the data architecture artifacts with the HUD BRM providing traceability or line of sight from HUD business functions to the supporting data structure.

The SDADM is organized in a phased workflow format (see Figure 1-1). This methodology provides the segment architecture development team with the inputs, outputs and requirements of each phase needed to develop a segment data architecture.

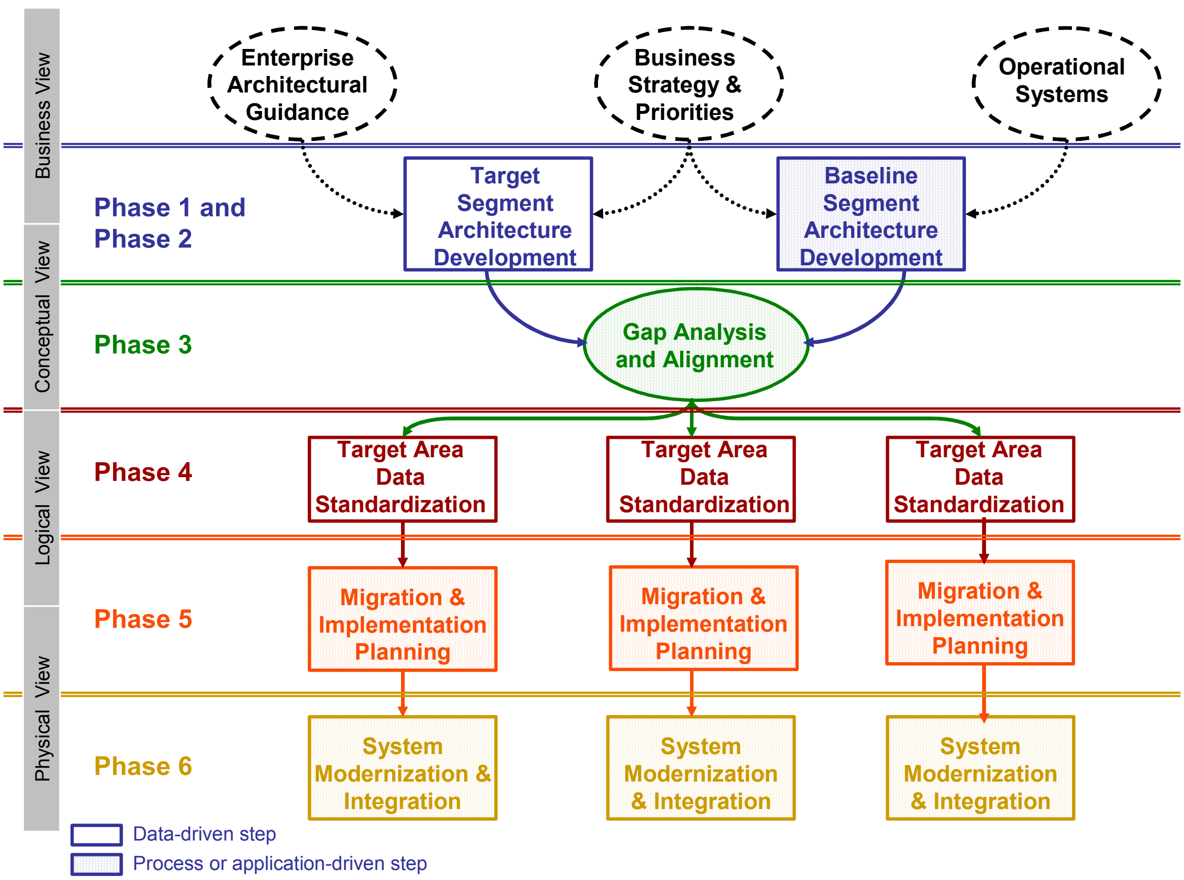

Figure 1-1. Segment Data Architecture Development Phased Approach

The methodology includes phases that are not data-driven but produce artifacts on which the segment data architecture is dependent to progress into the next level of development. These phases are included within the SDADM to

- Identify the dependencies between the development of business and data architecture
- Describe the content of the artifacts needed to validate the data architecture
- Provide an integrated perspective of the segment architecture development process.

The SDADM phases are:

- **Phase 1: Target Segment Architecture Development.** This phase describes the steps required to decompose and detail the conceptual HUD DRM to a segment logical data model (LDM). The segment LDM in conjunction with the segment Business Process Models (BPMs) identifies logical application components or targeted areas that are evaluated for further development.

- **Phase 2: Baseline Segment Architecture Development.** This phase describes the steps to document the baseline architecture based on HUD's operational systems and data sources. The Target Area Data Models (TADM) derived from the segment LDM are further detailed based on the harmonization and standardization of the baseline data elements.

- **Phase 3: Gap Analysis and Alignment.** This phase describes the steps to assess the 'fit' of the current operational systems compared to the HUD target architecture and determine the planned disposition of the operational systems. Operational systems are mapped to the candidate application components to provide input to the transition planning process.

- **Phase 4: Target Area Data Standardization.** This phase describes the steps required to harmonize data requirements and standardize data specifications based on segment target area business process requirements and baseline system data sources. The result is a TADM organized based on the taxonomy of the HUD DRM that leverages operational data elements to establish a Department-wide standard for the structure and definition of data.

- **Phase 5: Migration and Implementation Planning.** This phase describes the artifacts of the data architecture development process used to support the planning efforts to migrate and implement application components in the HUD target architecture.

- **Phase 6: System Modernization and Integration.** This phase describes the artifacts of the data architecture development process used to support the implementation of application components in the HUD target architecture.

1.2.2 HUD EA Governance Structure and HUD System Development Methodology

EA governance is the process and rules that ensure an organization's architecture is developed and used properly. HUD currently has a governance structure in place to control its EA activities, and the SDADM requires the creation of one role: a Segment Integrated Project Team (IPT) Data Architect.

The HUD SDM was created to guide the development and maintenance of HUD information systems to ensure that HUD's business needs are met. The SDADM artifacts that satisfy the SDM data documentation requirements are mapped to the relevant sections of the SDM.

SECTION 2. SEGMENT DATA ARCHITECTURE DEVELOPMENT METHODOLOGY

2.1 Segment Data Architecture Approach

The Segment Data Architecture Development Methodology (SDADM) is a phased, lifecycle approach to transition the HUD conceptual Data Reference Model (DRM) from a business view aligned with HUD's Business Reference Model (BRM) to a physical view used for the implementation of the data specification and information exchanges. The initial phases of the methodology define the high-level data requirements for the complete scope of a segment from both a business function and operational system perspective. The later phases of the methodology define detail requirements for targeted areas of a segment that have been selected for implementation (see Figure 2-1).

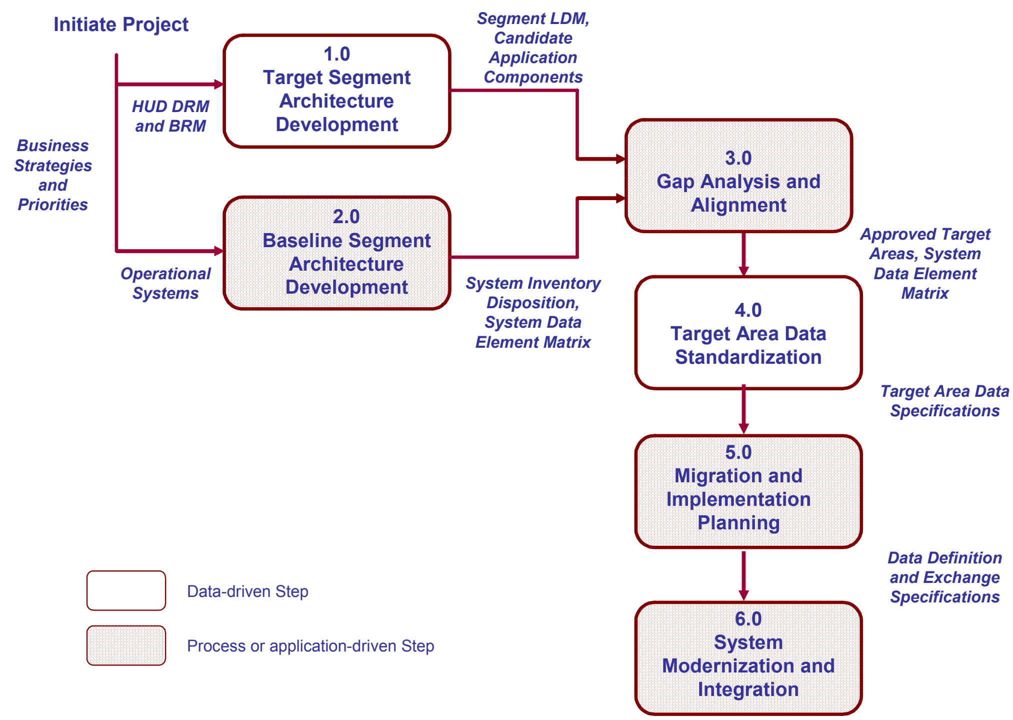

Figure 2-1. Segment Data Architecture Development Phases

The SDADM is comprised of both data-driven and process or application-driven phases. The methodology for data-driven phases identifies the inputs, outputs and steps to complete a task. For process or application-driven phases, the methodology identifies the artifacts produced by the phases that are required to develop the data architecture. A phase may contain both data-driven and process- or application-driven tasks. The distinction is noted within the workflow diagram for the phase.

The Target Segment Architecture Development phase uses the HUD BRM and DRM as the initial foundation for the segment target architecture. The segment target architecture is comprised of a logical data model and set of business models. The models are developed in parallel using a business-driven, top-down approach. The interactions between the models are mapped and provide the foundation to identify application components that are assembled to support the business function. The application components provide a manageable scope to incrementally detail and modernize ‘targeted areas’ of HUD’s baseline architecture.

The Baseline Segment Architecture Development phase supports the analysis and re-engineering of HUD’s operational systems. Data stores are the foundation for the segment baseline architecture. The results of this phase are an inventory of operational systems, system functions and data stores that are mapped to the target data and business models. It is recommended that Target and Baseline Segment Architecture Development phases be conducted in parallel to leverage the synergies between the phases. The target architecture is incrementally detailed and validated as it is developed using the baseline artifacts.

The Gap Analysis and Alignment phase creates and analyzes the mapping between the target and baseline architecture to identify gaps (i.e., missing business processes and data elements, as well as potential data integrity issues) between the architecture. The phase classifies operational systems based on the target architecture to identify redundant system functions and data stores, in addition to assisting HUD in determining the future disposition of the overlapping systems and data sources. The analysis results are used as input to develop a strategic roadmap with tactical recommendations to incrementally transition HUD’s systems from the baseline architecture to the target architecture within the scope of targeted areas.

The Target Area Data Standardization phase uses both the target and baseline data architectures to develop harmonized data requirements and standardize definition of data. The result is a data model organized based on the taxonomy of the target data architecture that provides a single, standardized data specification and structure. The target area data model is used to generate the specification to guide the implemented definition of data and the exchange of information between business processes.

The Migration and Implementation Planning phase uses the target area data specification and is mapped to the current system data elements to define data conversion and integration requirements, data replication requirements as well as to scope and sequence the implementation effort. It uses the exchange specifications to identify application interface requirements and select exchange protocols.

The System Modernization and Integration phase uses the data specifications to provide the standard for the implementation or modernization of the authoritative source of data. If the implemented definition of data is extended, it must map back to the specification to maintain the ‘line of sight’ from the HUD DRM to the implemented data source. The exchange specifications are used to provide the standard for how data is transmitted between the business processes.

2.2 1.0 Target Segment Architecture Development

This phase includes the steps required to decompose and detail the conceptual HUD DRM to a segment Logical Data Model (LDM). The segment LDM in conjunction with the segment Business Process Models (BPMs) is used to identify logical application components or targeted areas that need to be evaluated for further development (see Figure 2-2).

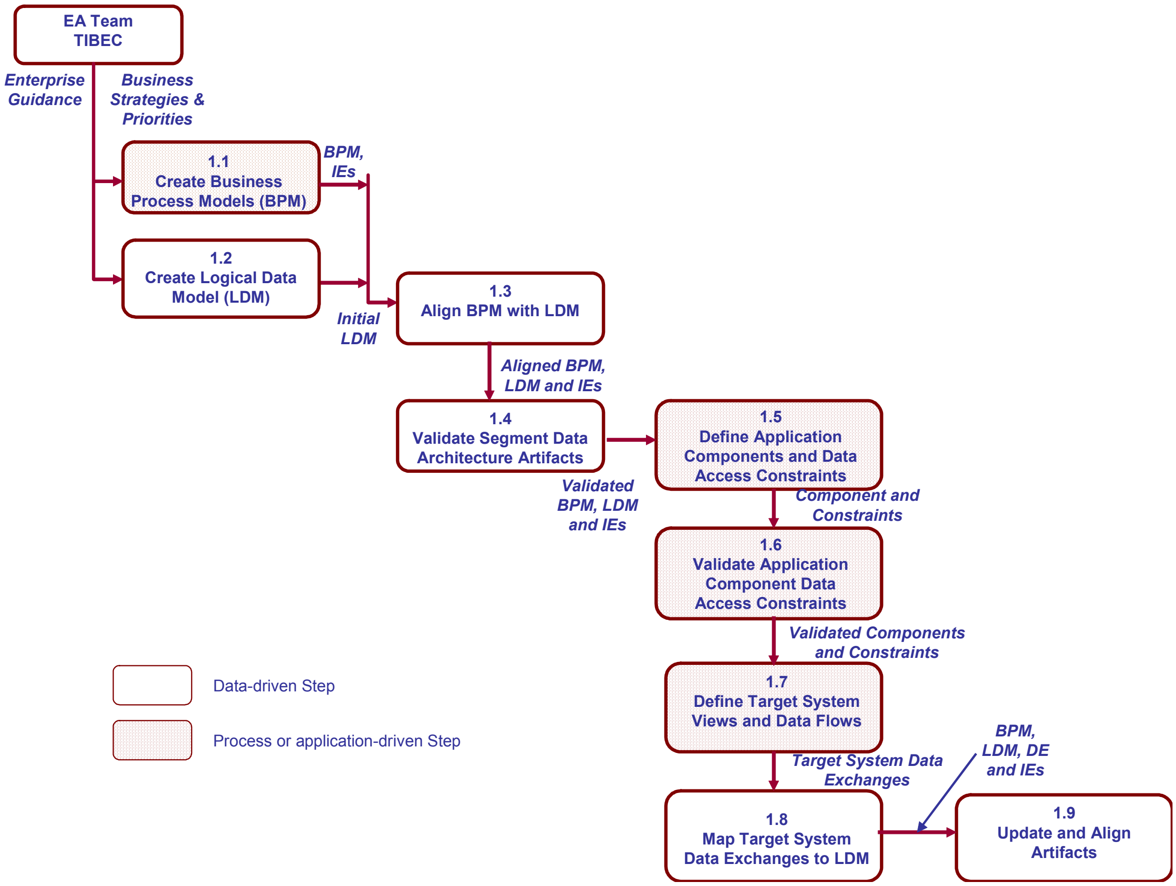

Figure 2-2. 1.0 Target Segment Data Architecture Development Steps

The segment LDM and BPMs are developed in parallel to ensure symmetry and alignment. The development of the LDM is dependent on BPM artifacts. The Create Business Process Models task is included within the SDADM to describe the artifacts used to guide the development and confirmation of the segment LDM.

The mapping of the segment BPM Information Exchanges (IEs) to the LDM validates the decomposition and completeness of both models. There is a direct correlation between the information exchanged and the business process with the data defined within the LDM. The mapping provides the foundation to identify natural clustering of process and data to define the scope of application components and the sequence of implementation. The recommended scope and sequencing of application components are used as input to the Gap Analysis and Alignment phase.

2.2.1 1.1 Create the Business Process Models

This task is process-driven. The artifacts required to develop the LDM are operational view diagrams for the segment's business function and sub-functions, as well as business process diagrams (BPDs) with information exchange (IE) requirements for the business process and activities (see Figure 2-3).

1.1 Create Business Process Models			
Input Artifact	Step	Output	Notations
HUD BRM Business Functions	1.1.1 Create Operational View of LOB or Business Function	Operational View of LOB/Business Function	OV-2 [1]
OV-2 for LOB/Business Function	1.1.2 Create Operational View of Sub-Function	Operational View of Business Sub-Functions	OV-2 [1]
OV-2 for Business Sub-Functions	1.1.3 Create BPDs and Information Exchanges between Business Activities	BPDs and Information Exchanges	BPMN [2] IDEF0 or IDEF3 [3]

Process or application-driven task

[1] *DoD Architecture Framework Version 1.0, Volume II: Product Descriptions,* http://www.defenselink.mil/nii/doc/DoDAF_v1_Volume_II.pdf, Section 4.2

[2] *Introduction to BPMN,* Steve A. White, IBM Corporation, http://www.bpmn.org/Documents/Introduction%20to%20BPMN.pdf

[3] *Integrated DEFinitions Methods,* Knowledge Based Systems, Inc., http://www.idef.com/IDEF3.html

Figure 2-3. 1.1 Business Process Artifacts

The operational view diagrams for the segment's business function and sub-functions provide the context of the business function and the stakeholders involved. Figure 2-4 is an illustrative example of a Department of Defense Architecture Framework (DoDAF) Operational Node Connectivity or OV-2 diagram for the business sub-function Process Grant Application.

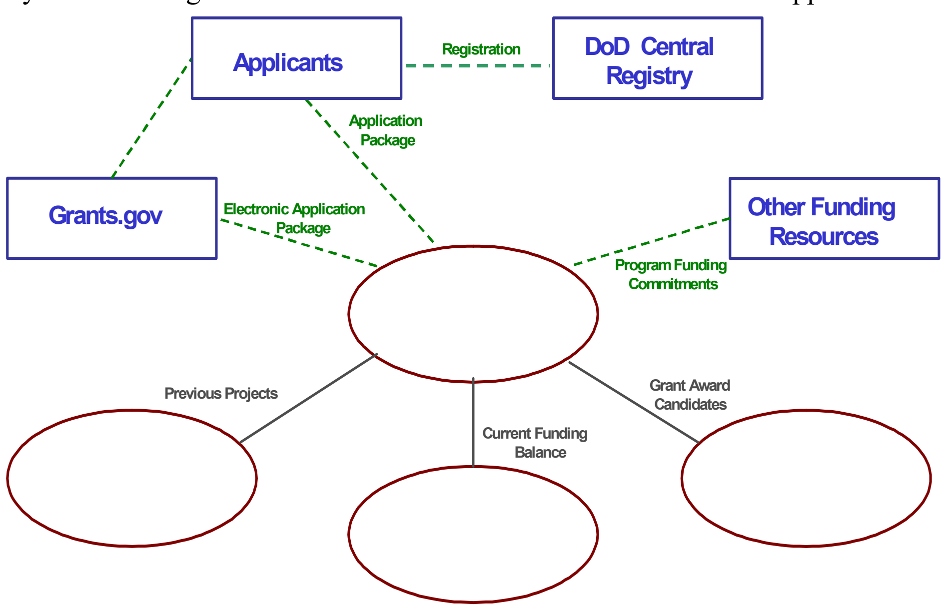

Figure 2-4. 1.1.2 Illustrative Example of Process Grant Application OV-2 Diagram

The documentation requirements for the business function and sub-function OV-2 diagrams and supporting documentation are

- External and internal stakeholders
- Type of information exchanged
- How and when the information is exchanged
- Standards used to exchange information
- Legislative and security requirements.

Figure 2-5 is an example of a BPD using Business Process Modeling Notation (BPMN) for the Establish Program process.

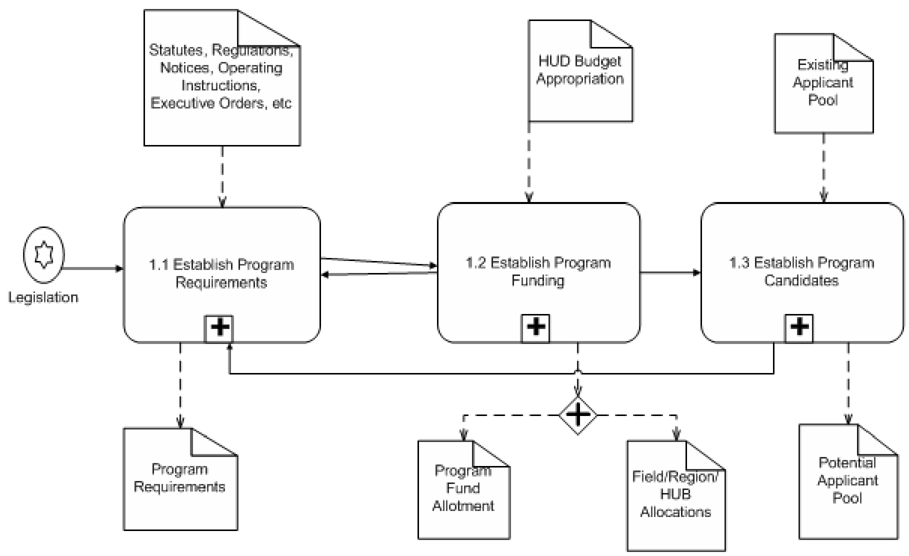

Figure 2-5. 1.1.3 BPMN Example of Establish Program Process

The documentation requirements for the BPDs and supporting documentation are:

- Business process activities
- Sequence and dependencies between the activities
- Data exchanged between the activities
- Security requirements to exchange data.

2.2.2 1.2 Create the Logical Data Model

The purpose of this task is to identify and validate the entity types and subject areas within the conceptual HUD DRM that are needed to develop an LDM for the segment's business function. Once the scope is determined, the LDM is created. Figure 2-6 identifies the steps and artifacts of this task.

1.2 Create the LDM			
Input Artifact	Step	Output	Notations
Conceptual HUD DRM	1.2.1 Identify and Validate the DRM Components	Scope of Segment LDM	
Conceptual HUD DRM, Scope of Segment LDM	1.2.2 Create the Segment LDM	Segment LDM	OV-7 [1] IDEF1x [2]

[1] *DoD Architecture Framework Version 1.0, Volume II: Product Descriptions,* http://www.defenselink.mil/nii/doc/DoDAF_v1_Volume_II.pdf, Section 4.7

[2] *Integrated DEFinitions Methods,* Knowledge Based Systems, Inc., http://www.idef.com/IDEF3.html

Figure 2-6. 1.2 Create the LDM Steps

2.2.2.1 1.2.1 Identify and Validate the DRM Components

The purpose of this step is to identify the conceptual HUD DRM entity types and subject areas that are used to create the segment's LDM. Each DRM subject area is assigned a principal data steward that aligns with a business function in the BRM. Select the entity types within the subject area(s) that are assigned to the segment's business function. Select entity types that provide context to the selected entity types.

Compare the selected entity types with the segment BPMs. The high-level IEs described within the BPMs should directly correlate to the entity types selected from the DRM. The scope of the segment's LDM and the correlation to the BPMs is validated with the EA Team and the segment's stakeholders. If revisions are required, the EA Team is responsible for updating the DRM and reestablishing the mapping to the target EA business and application layers.

2.2.2.2 1.2.2 Create the Segment LDM

The purpose of this step is to create the segment LDM using the selected HUD DRM central entity types and subject areas. Within the model, decompose and detail the central entity for which the segment is the principal data steward. Create new entity types that provide further detail about the central entity types and resolve complex relationships between central entity types. A central entity type is referred to as an independent entity type, and a descriptive entity type is referred to as a dependent entity type.

The Federal Information Processing Standard Publication (FIPS PUB) 184 (12/21/1993) issued by the National Institute of Standards and Technology (NIST) includes a prescribed approach to identifying and defining entity types, relationships, attributes and attribute properties. This approach is summarized in Appendix A, and the full document is accessed at the following link: http://www.idef.com/Downloads/pdf/Idef1x.pdf.

As entity types are created, subject classes are created to organize the entity types; relationships between the entity types are established; and attributes are defined to support the requirements of the business processes (see Figure 2-7).

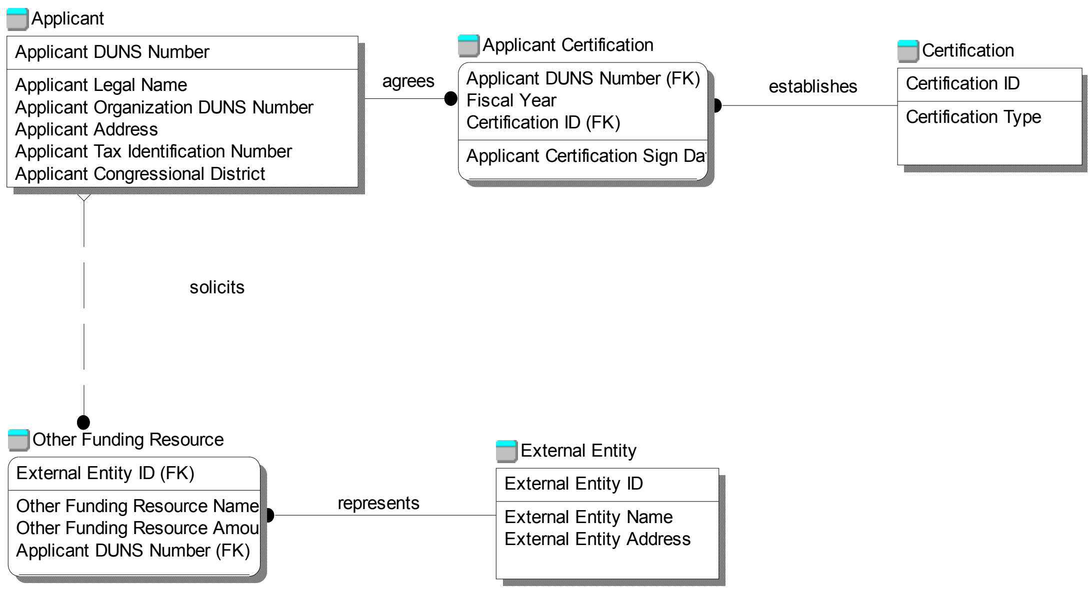

Figure 2-7 1.2.2 Example of a LDM for the Applicant Subject Class

Business requirements used to guide the decomposition and detail of the LDM are obtained from the segment's BPMs and information gathered from the function area (e.g., results of interviews, policies and procedures). The completion of this step is an LDM based on the segment's business processing requirements that includes

- Subject area
- Subject classes
- Central or dependent entity types
- Descriptive or independent entity types
- Relationships between the entity types
- Key attributes
- Natural identifiers of the entity types.

2.2.3 1.3 Align BPMs with the LDM

The purpose of this phase is to map the business process entity actions and IEs to the segment LDM. The mapping validates the decomposition and artifacts of both models and identifies disparities between the models that need to be resolved. Once the models are aligned and re-mapped, the business process entity actions are used to identify business stewards for the entity types. Figure 2-8 identifies the steps and artifacts of this task.

1.3 Align BPMs with LDM			
Input Artifact	**Step**	**Output**	**Notations**
BPMs, IEs, LDM	1.3.1 Map IEs to LDM	IE Matrix, Artifact Revisions	*Metadata Repository or Spreadsheet**
Business Process Entity Actions, LDM	1.3.2 Map Business Process Actions to LDM	CRUD Matrix, Artifact Revisions	*Metadata Repository or Spreadsheet**
Artifact Revisions	1.3.3 Revise BPMs, IEs, LDM, IE Matrix, CRUD Matrix	Revise BPMs, IEs, LDM, IE Matrix, CRUD Matrix	
CRUD Matrix	1.3.4 Identify Business Data Steward for each Entity Type	Business Data Steward Matrix	*Metadata Repository or Spreadsheet*

* = no formal notation

Figure 2-8. 1.3 Align BPMs with LDM Steps

2.2.3.1 1.3.1 Map IEs to LDM

The purpose of this step is to map the IEs to the LDM entity types and key attributes within a matrix or using a metadata repository. Analyze the mappings to identify

- New entity types and attributes to support the IEs
- Incomplete IE requirements.

Figure 2-9 provides an illustrative example of an IE matrix for the Process Application business process.

Process	Sub-Process	Information Exchanges	Data Objects										
			Applicant				Project Participant				Applicant Certifications		
			DUNS Number	Type Code	Legal Name	…	External Entity ID	Type Code	Project ID	…	Applicant DUNS Number	Certification RCEZEC	Certification Date
Process Application	**Validate Applicant Capacity**	Validate Applicant CCR	X	X	X								
		Estalish Applicant Organizational Resources	X	X			X	X	X				
		Verify Applicant Certifications	X	X							X	X	X
		Verify Applicant Funding Resources	…										
		Establish Applicant Project Participants	…										
		Identify Applicant Previous Projects	…										
		Determine Applicant Funding Allocation Status and History	…										
		Associate Applicant with Congressional Representation	…										

Figure 2-9. 1.3.1 Illustrative Example of Information Exchange Matrix for Process Applicant

In addition to the matrix, the following information is documented:

- Nature of Transaction (transaction type, triggering event)
- Performance Requirements (criticality, periodicity and timeliness)
- Information Assurance Requirements (availability, confidentiality and integrity)
- Security Requirements (protection, classification).

2.2.3.2 1.3.2 Map Business Process Actions to the LDM

The purpose of this step is to define and map the business process entity actions on the LDM entity types within a matrix or using a metadata repository. Analyze the mappings to identify

- New entity types and attributes to support the business process requirements
- Extraneous entity types and attributes not used by the business processes

- New business processes required to move the entity type through its life cycle from creation to disposition
- Extraneous business processes that do not act upon data within the segment, or which provide duplicative functionality.

Start with the business process activity level. Determine how each activity acts upon the data it consumes, processes internally, or exchanges at the entity type level. Record the action in the cell values using C for Create, D for Delete, U for Update and R for Read (CRUD). Figure 2-10 provides an illustrative example of a CRUD matrix for business process activities.

Process	Sub-Process	Activities	Data Objects			
			Applicant	Project Participant	Applicant Certification	Applicant Funding Resources
Process Applicant	**Validate Applicant Capacity**	Validate Applicant CCR	**C**			
		Establish Applicant Organizational Resources	**U**	**R**		
		Verify Applicant Certifications	**R**		**C**	
		Verify Applicant Funding Resources	**R**			**C**
		Establish Applicant Project Participants	**R**	**C**		
		Identify Applicant Previous Projects				
		Determine Applicant Funding Allocation Status and History				
		Associate Applicant with Congressional Representation				

Figure 2-10. 1.3.2 Illustrative Example of a CRUD Matrix for Process Applicant Activities

Roll up the CRUD action of each business process activity to the business process level. When more than one activity performs an action on the same entity, the highest level of action is recorded for the business process. The order of CRUD actions in highest to lowest order is: Create, Delete, Update and Read. Figure 2-11 provides an illustrative example of a CRUD matrix for a business process.

Process	Data Objects			
	Applicant	Project Participant	Applicant Certification	Applicant Funding Resources
Process Applicant	**C**	**C**	**C**	**C**

Figure 2-11. 1.3.2 Illustrative Example of a CRUD Matrix for Process Applicant Process

Compile all the business process CRUD matrixes into a single matrix to create a business function CRUD matrix. Figure 2-12 provides an illustrative example of a CRUD matrix for a business function.

Process	Entity Type													
	Program Office	Program	Funding Opportunity	Application	Applicant	Project Participant	Applicant Certification	Project	Project Activity	Project Activity Budget	Problem Need	Goal	Measurement Tool	Housing Site
Process Application	**R**	**R**	**R**	**C**	**R**									
Process Applicant					**C**	**C**	**C**							
Establish Program	**R**	**C**	**C**											
Establish Project			**R**	**R**	**R**			**C**			**C**	**C**	**C**	
Establish Project Activities			**R**	**R**	**R**			**U**	**C**	**C**				**C**

Figure 2-12. 1.3.2 Illustrative Example of a CRUD Matrix for Grants Management

2.2.3.3 1.3.3 Revise the BPMs, IEs, LDM and Matrixes

The purpose of this step is to modify the BPMs, IEs and LDM based on revisions identified from the analysis of IE and CRUD matrixes. Remap the IEs and business process entity actions to the LDM and update the IE Matrix, Business Process CRUD Matrix and Business Function CRUD Matrix.

2.2.3.4 1.3.4 Identify Business Data Steward for each Entity Type

The purpose of this step is to determine the natural Business Data Steward for all entity types using the business function CRUD matrix. The Business Data Steward is the organization responsible for the maintenance of the entity type. The steward is determined by the business function that creates the entity type and then the mapping of the business function to the business organization documented within the HUD BRM.

If more than one business function creates an entity type and those functions are mapped to different organizations, then the Business Data Steward needs to be determined by the Data Control Board (DCB). Each entity type is assigned only one Business Data Steward. Other organizations that perform actions on the entity types are identified as stakeholders. Figure 2-13 provides an illustrative example of a business data steward matrix.

Organization	Entity Type													
	Program Office	Program	Funding Opportunity	Application	Applicant	Project Participant	Applicant Certification	Project	Project Activity	Project Activity Budget	Problem Need	Goal	Measurement Tool	Housing Site
Program Office A	*	*	*	**S**	*									
Program Office B					**S**	**S**	**S**							
Program Office C	*	**S**	**S**											
Program Office D			*	*	*			**S**			**S**	**S**	**S**	
Program Office E			*	*	*			*	**S**	**S**				**S**

S = Business Data Steward
* = Stakeholder

Figure 2-13. 1.3.4 Illustrative Example of a Business Data Steward Matrix

2.2.4 1.4 Validate Segment Data Architecture Artifacts

The purpose this task is to validate the segment data architecture artifacts with the EA Team, Business Data Stewards and Data Control Board (DCB). Based on the feedback, update the impacted artifacts accordingly. Figure 2-14 identifies the steps and artifacts of this task.

1.4 Validate Segment Data Architecture Artifacts			
Input Artifact	Step	Output	Notations
LDM, BPMs, IEs, IE Matrix, CRUD Matrix, Business Data Steward Matrix	1.4.1 Validate Segment Data Architecture Artifacts with the EA Team	Artifact Revisions	
LDM, BPMs, IEs, IE Matrix, CRUD Matrix, Business Data Steward Matrix	1.4.2 Validate Segment Data Architecture Artifacts with the Business Data Steward and DCB	Artifact Revisions	
Artifact Revisions	1.4.3 Revise BPMs, LDM, IEs and Matrixes	Revised Artifacts	

Figure 2-14. 1.4 Validate BPM, LDM and IEs Steps

2.2.4.1 1.4.1 Validate Segment Data Architecture Artifacts with the EA Team

The purpose of this step is to validate with the EA Team the following segment data architecture artifacts:

- Segment BPMs
- Segment LDM
- IE Matrix
- CRUD Matrix
- Business Data Steward Matrix.

The EA Team is responsible for

- Ensuring data artifacts align with the HUD EA
- Verifying data artifacts' documentation and implementation consistently conform to the requirements
- Analyzing the impact of new and changed data architecture components
- Coordinating the dissemination of data architecture releases.

2.2.4.2 1.4.2 Validate Segment Data Architecture Artifacts with the Business Data Steward and DCB

The purpose of this step is to validate with the Business Data Steward and DCB the following segment data architecture artifacts:

- Segment BPMs
- Segment LDM
- IE Matrix
- CRUD Matrix
- Business Data Steward Matrix.

The Business Data Steward and DCB are responsible for

- Verifying data artifacts are accurate, complete and compliant with standards
- Ensuring data artifacts are accessible and reused appropriately across the Department
- Ensuring data artifacts are managed as prescribed by HUD EA policies and guidelines.

2.2.4.3 1.4.3 Revise BPMs, LDM, Information Exchanges and Matrixes

The purpose of this step is to update the impacted artifacts based on the recommended changes from the EA Team, Business Data Steward and DCB.

2.2.5 1.5 Define Application Components and Data Access Constraints

This task is process-driven and includes a description of the artifact required to define the data access constraint for an entity type. The artifacts required to define the data access constraints for an entity type are combined as a business process clustering matrix for the segment business function. The business process clustering matrix identifies application components and the entity actions those components perform against an authoritative source of data. The components and the permitted entity actions are documented within the LDM or supporting documentation as the data access constraints.

The clustering process uses the business function CRUD matrix to identify and sequence candidate application components. Further information about the cluster process and other usages of this matrix is described in an article that is accessed using the following link: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmaj/html/businesspatterns.asp.

Figure 2-15 provides an illustrative example of a cluster matrix.

Process	Entity Type													
	Program Office	Program	Funding Opportunity	Applicant	Project Participant	Applicant Certification	Application	Project	Problem Need	Goal	Measurement Tool	Project Activity	Project Activity Budget	Housing Site
Establish Program	R	C	C											
Process Applicant				C	C	C								
Process Application	R	R	R	U			C							
Establish Project			R	R				C	C	C	C			
Establish Project Activities			R	R				U				C	C	C

Figure 2-15. 1.5 Illustrative example of a Cluster Matrix for a Business Function

2.2.6 1.6 Validate Application Components and Data Access Constraints

This task is process-driven. Revisions recommended to the application components data access are updated in the entity type data access constraints documentation.

2.2.7 1.7 Define Target System Views and Data Flow Diagrams

This task is process-driven and includes a description of the artifacts required to map the target system data exchanges to the LDM. The artifacts required to map the target system data exchanges to the LDM are system view diagrams of the segment's business function and sub-functions, and system functionality or data flow diagrams of the target systems. Figure 2-16 identifies the step and artifacts of this task.

1.7 Define Target System Views and Data Flows			
Input Artifact	Step	Output	Notations
OV-2 for LOB/Business Function	1.7.1 Create System View of LOB or Business Function	System View of LOB/Business Function	SV-1 [1]
SV-1 for Business Function, OV-2 for Business Sub-Function	1.7.2 Create System Views of Sub-Functions	System Views of Business Sub-Functions	SV-1 [1]
SV-1 for Business Sub-Function, BPDs, Application Components	1.7.3 Create System Functionality/Data Flow Diagram for Target Systems	System Functionality/Data Flow Diagram for Target Systems	SV-4 [2]

Process or application-driven task

[1] *DoD Architecture Framework Version 1.0, Volume II: Product Descriptions,* http://www.defenselink.mil/nii/doc/DoDAF_v1_Volume_II.pdf, Section 5.1

[2] *DoD Architecture Framework Version 1.0, Volume II: Product Descriptions,* http://www.defenselink.mil/nii/doc/DoDAF_v1_Volume_II.pdf, Section 5.4

Figure 2-16. 1.7 Define Target System Views and Data Flows Steps

The System Views for a segment's business function and sub-functions identify target system nodes, systems and application components that support operational nodes and the interfaces between the nodes. The Department of Defense Architectural Framework (DoDAF) System Interface Description or SV-1 diagram links together the DoDAF Operational Node Connectivity Diagram or OV-2 diagram of the segment's business function and sub-functions with the target systems and application components assembled to support the business function. Figure 2.17 is an illustrative example of an SV-1 diagram for the Process Grant Application OV-2 diagram documented in Figure 2-4.

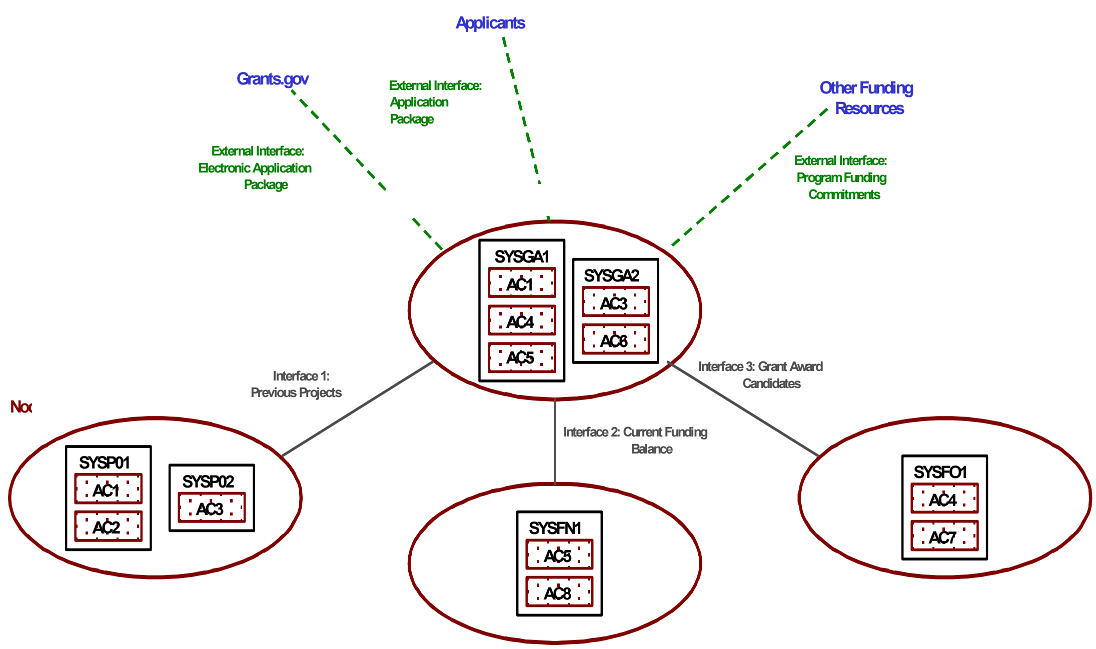

Figure 2-17. 1.7.1 Illustrative Example of Process Grant Application SV-1 Diagram

The documentation requirements for the segment's SV-1 diagrams and supporting documentation are

- Status of the data interface (e.g., existing, planned)
- Purpose of the data interface
- Event that triggers data exchange
- Means to exchange data
- Standards used to exchange data
- Type of data exchanged
- Legislative and security requirements applicable to the data exchange.

The System Functionality or Data Flow Diagram identifies the system functions or application components, internal system data flows, internal system data stores and external system data flows for a target system. Figure 2.18 is an illustrative example of a DoDAF Systems Functionality Description or SV-4 diagram for a target Grants Administration Node system (SYSGA1) documented in the illustrative example above.

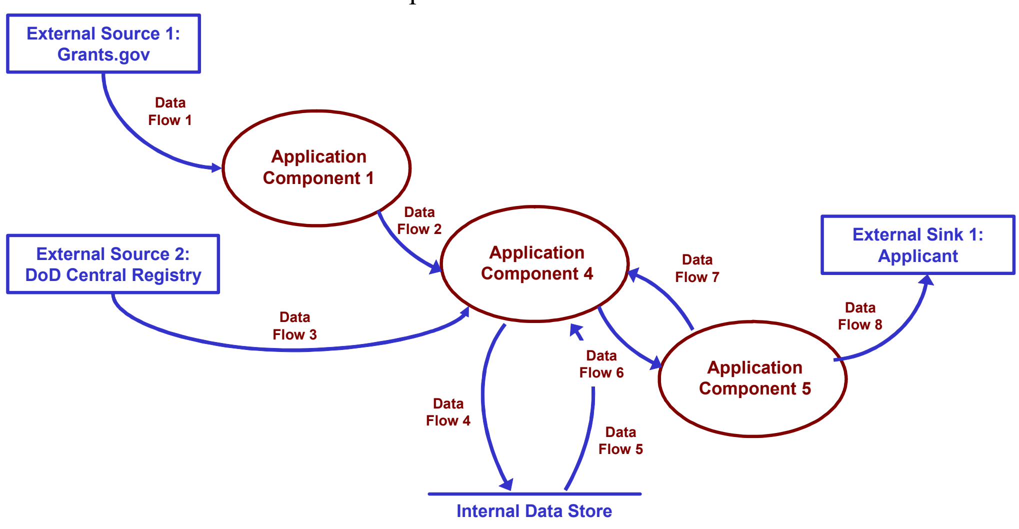

Figure 2-18. 1.7.3 Illustrative Example of Grants Administration SV-4 Diagram for SYSGA1 System

The documentation requirements for the segment's target system functionality or data flow diagrams and supporting documentation are

- Producing and consuming application component, internal system or external partner
- System data exchanged between producing and consuming components or systems including an indication of data exchange results (e.g., success or failure indicators)
- Dependencies between data exchanges
- Means to exchange data
- Standards and protocols used to exchange data
- Security requirements to exchange data.

2.2.8 1.8 Map Target System Data Exchanges to LDM

The purpose of this task is to map the data exchanges between the target system components to the segment LDM entity types and attributes. Figure 2-19 provides an illustrative example of a target system data exchange matrix.

Diagram #	Data Flow #	Data Objects										
		Applicant				Project Participant				Applicant Certifications		
		DUNS Number	Type Code	Legal Name	…	External Entity ID	Type Code	Project ID	…	Applicant DUNS Number	Certification RCEZEC	Certification Date
1	**Data Flow 1**	X	X	X		X	X	X		X		
	Data Flow 2	X	X	X								
	Data Flow 3	X										
	Data Flow 4	X	X	X		X	X	X		X		
	Data Flow 5											

Figure 2-19. 1.8 Illustrative Example of a Target System Data Exchange Matrix

In addition to the matrix, the following information is documented:

- Nature of Transaction (transaction type, triggering event)
- Performance Requirements (predicted volume, criticality, periodicity and timeliness)
- Information Assurance Requirements (predicted availability, confidentiality and integrity)
- Security Requirements (predicted protection, classification).

2.2.9 1.9 Update and Align Artifacts

The purpose of this task is to align the segment's LDM with the conceptual HUD DRM. This alignment is achieved by mapping all new entity types within the LDM to the central entity types within the conceptual HUD DRM. Alignment of the HUD BRM is achieved through the CRUD matrices.

Changes required to the HUD DRM or BRM to align the segment's data artifacts are facilitated by the Segment Target Area Data Model (IPT) Data Architect. The IPT Data Architect submits the recommended revisions to the EA Team to validate the justification for the revisions and analyze the impact to the overall HUD EA. The EA Team presents its recommended revisions and impact analysis results to the DCB for approval and prioritization. The approved revisions and release schedules are coordinated and disseminated by the EA Team.

2.3 2.0 Baseline Segment Architecture Development

This phase includes the steps to describe the baseline operational systems and data elements for the segment, then map the data elements to the segment logical data model (LDM). The Target Area Data Models (TADM) derived from the segment LDM are further detailed based on the harmonization and standardization of the baseline data elements (see Figure 2-20).

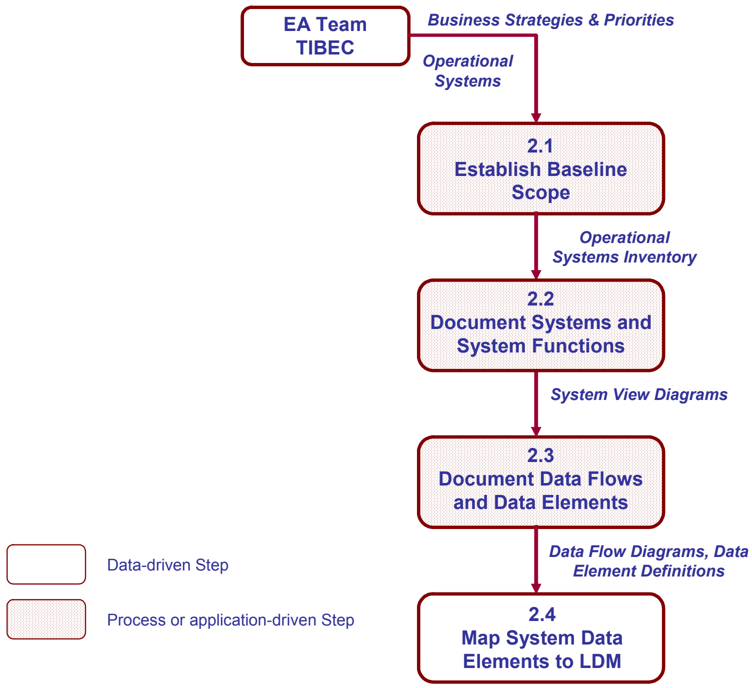

Figure 2-20. 2.0 Baseline Segment Data Architecture Development Steps

The majority of this phase includes process-driven tasks. The last task, Map System Data Elements to the LDM, is a collaborative effort between the operational system owners and the segment Integrated Project Team (IPT). The artifacts required to map the system data elements to the LDM are system view diagrams of the operational systems and system functions, data flows diagrams and data element descriptions. Figure 2-21 identifies the steps and artifacts of this phase.

2.0 Baseline Architecture Development			
Input Artifact	Step	Output	Notations
Business Strategies and Priorities, Operational Systems	2.1 Establish Baseline Scope	Operational System Inventory	*Metadata Repository or Spreadsheet**
Operational System Inventory	2.2 Document Systems and System Functions	System Views of Operational Systems and System Functions	SV-1 [1]
SV-1 for Operational Systems and System Functions	2.3 Document System Data Flows and Data Elements	System Functionality or Data Flow Diagrams, Data Element Definitions	SV-4 [2] *Metadata Repository or Spreadsheet**
Data Element Definitions, LDM	2.4 Map System Data Elements to LDM	System Data Element Matrix	*Metadata Repository or Spreadsheet**

* = no formal notation

Process or application-driven task

[1] *DoD Architecture Framework Version 1.0, Volume II: Product Descriptions,* http://www.defenselink.mil/nii/doc/DoDAF_v1_Volume_II.pdf, Section 5.1

[2] *DoD Architecture Framework Version 1.0, Volume II: Product Descriptions,* http://www.defenselink.mil/nii/doc/DoDAF_v1_Volume_II.pdf, Section 5.4

Figure 2-21. 2.0 Baseline Segment Data Architecture Development Steps

The system view diagrams of a segment's operational systems document the system nodes, systems, and system functions. The Department of Defense Architectural Framework (DoDAF) System Interface Description or SV-1 diagram is used to document the nodes, systems and interfaces at a various levels of detail. Figure 2-22 is an illustrative example of an SV-1 diagram for the Multi-family Rental Assistance Node.

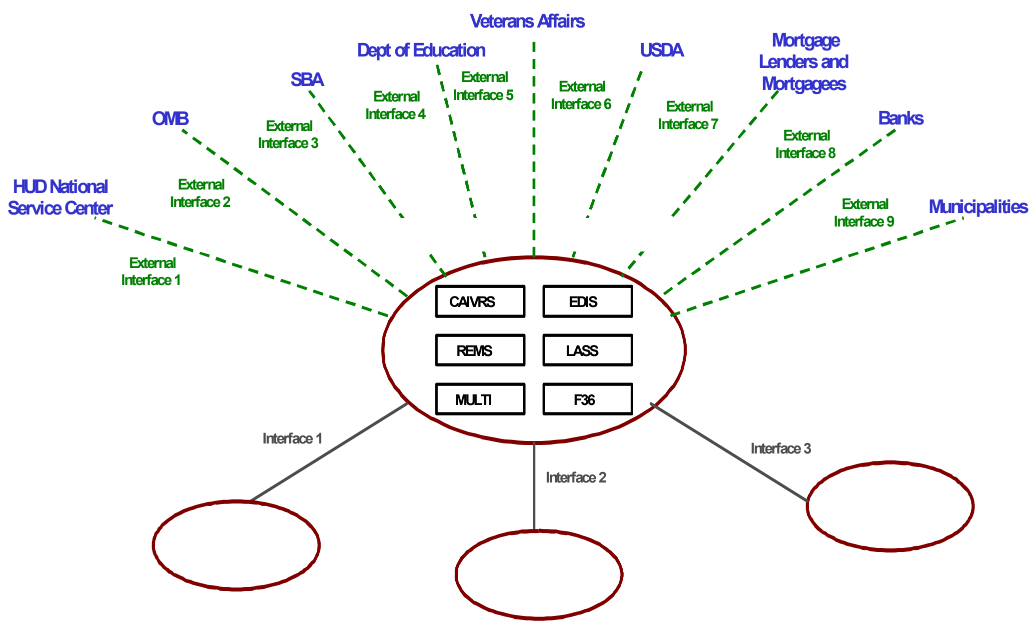

Figure 2-22. 2.2 Illustrative Example of Multi-family Rental Assistance Node

The documentation requirements for the segment's baseline SV-1 diagrams and supporting documentation are

- Purpose of the data interface

- Event that triggers data exchange
- Means to exchange data
- Standards used to exchange data
- Type of data exchanged
- Legislative and security requirements applicable to the data exchange.

The System Functionality or Data Flow Diagram identifies the systems and functions, internal system data flows, internal system data stores and external system data flows for a baseline system. Figure 2.23 is an illustrative example of a DoDAF Systems Functionality Description or SV-4 diagram for the operational EDIS system within the Multi-family Rental Assistance node.

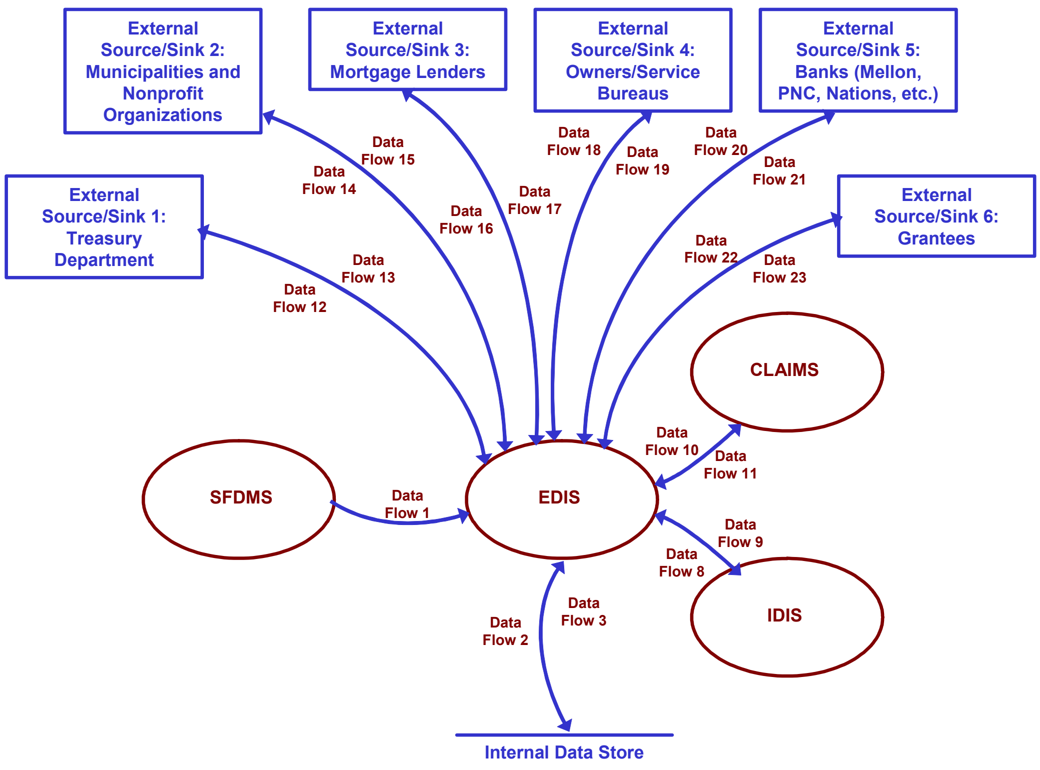

Figure 2-23. 2.3 Illustrative Example of EDIS System SV-4 Diagram

The documentation requirements for the segment's operational system functionality or data flow diagrams and supporting documentation are

- Producing and consuming internal system, system function or external partner
- Data elements exchanged between producing and consuming systems components including names, definition and properties (e.g. length, domain)
- Data elements acted upon (set, removed or changed) and saved within the internal data store during the system's execution
- Dependencies between data flows
- Means to exchange data
- Standards and protocols used to exchange data
- Security requirements to exchange data.

2.3.1 2.4 Map System Data Elements to LDM

The purpose of this task is to map the operational system data sources and elements to the segment LDM entity types and attributes. Figure 2-24 provides an illustrative example of a System Data Element Matrix.

Data Store	**Data Element**	**Entity Type and Attributes**																							
		Program				**Program Office**				**Funding Opportunity**					**Applicant**										
		ID	Office ID	Name	Description	ID	Name	Director Name	Description	ID	Program ID	Title	CFDA Title	Description	DUNS Number	Type Code	Legal Name	Organization DUNS Number	Organization Department Name	Address Street Name	Address City Name	Address State Name	Address Zip Code	Address Country Code
Data Store A	Field 1	X																							
	Field 2		X																						
	Field 3			X																					
	Field 4																								
Data Store B	Field 1	X																							
	Field 2				X																				
	Field 3					X																			
	Field 4								X																
	Field 5									X															
Data Store C	Field 1									X															
	Field 2														X										
	Field 3															X									
	Field 4																X								
	Field 5																	X							
	Field 6																		X						
	Field 7																			X					
	Field 8																			X	X	X	X	X	

Figure 2-24. Example of System Data Element Matrix

Changes required to the HUD DRM to capture central entity types and subject areas identified by mapping operational data sources to the segment LDM are facilitated by the Segment Target Area Data Model (IPT) Data Architect. The IPT Data Architect submits the recommended revisions to the EA Team to validate the justification for the revisions and analyze the impact to the overall HUD EA. The EA Team presents its recommended revisions and impact analysis results to the DCB for approval and prioritization. The approved revisions and release schedules are coordinated and disseminated by the EA Team.

2.4 3.0 Gap Analysis and Alignment

This phase includes the steps to assess the 'fit' of the operational systems compared to the HUD target architecture in determining the disposition. The operational systems are mapped to the candidate application components. Analysis of this mapping provides input to develop recommendations and a transition plan to incrementally migrate the HUD baseline architecture to its target architecture. Based on approval of the transitional plan by the Technology Investment Board Executive Committee (TIBEC), targeted area development projects are funded and resourced (see Figure 2-25).

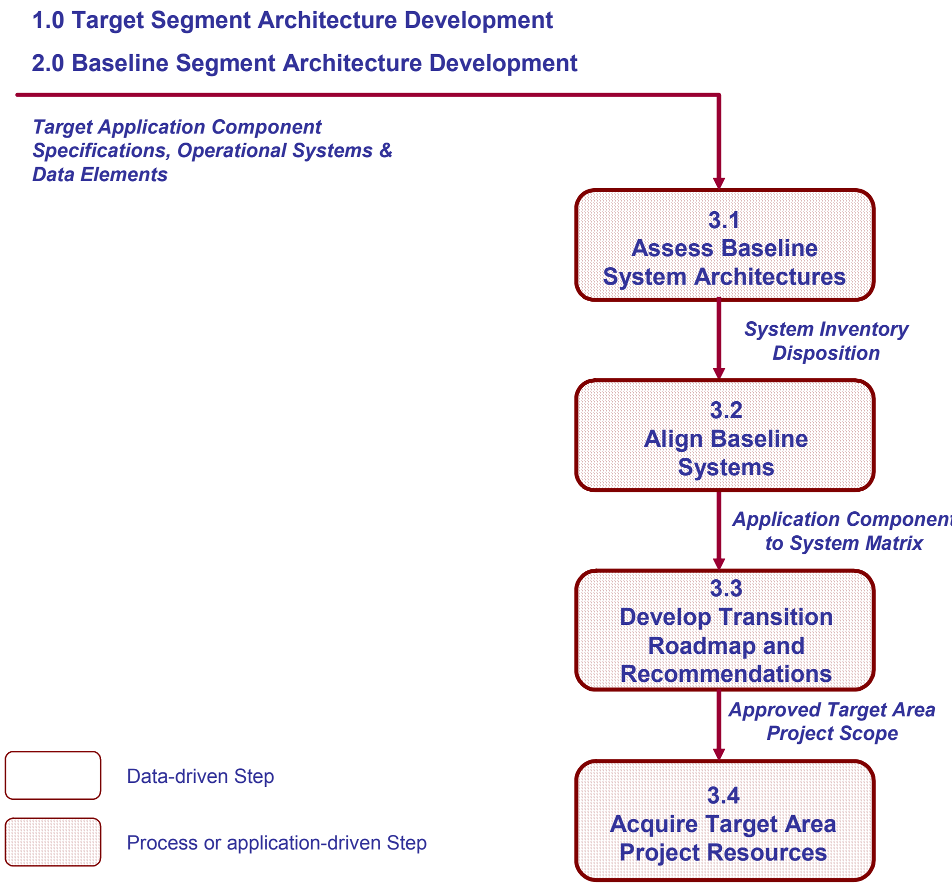

Figure 2-25. 3.0 Gap Analysis and Alignment Steps

This phase is an application and process-driven task. The Target Area Data Standardization phase is dependent on the approval of target area development projects and the scope of those projects. The scope of a target area development project is defined in a CRUD matrix.

2.5 4.0 Target Area Data Standardization

This phase includes the steps required to harmonize data requirements and standardize data specifications based on the target area business process requirements and baseline system data sources. The result is a Target Area Data Model (TADM) organized based on the taxonomy of the HUD DRM that leverages operational data elements to establish a Department-wide standard for the structure and definition of data (see Figure 2-26).

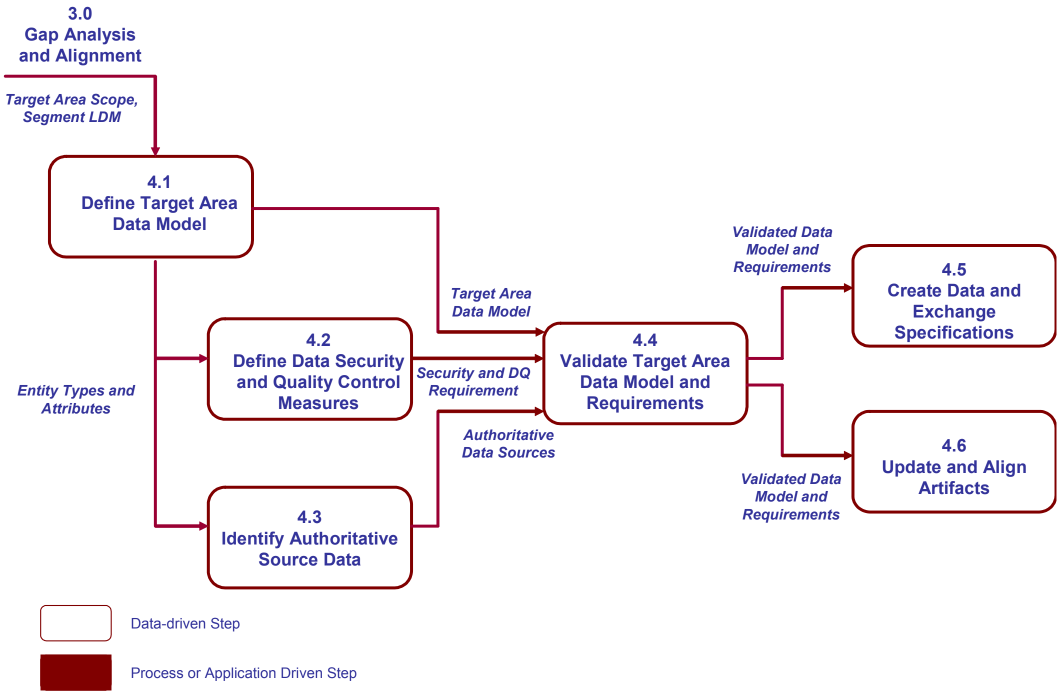

Figure 2-26. 4.0 Target Area Standardization Phase

2.5.1 4.1 Define Target Area Data Model

The purpose of this task is to create a TADM using the approved scope of the target area resulting from the Gap Analysis and Alignment phase to create a TADM from the segment LDM. The project TADM is then refined based on harmonization of the operational system data elements, normalization of the data structure and standardization of the data elements definitions and properties. Figure 2-27 identifies the steps and artifacts of this task.

4.1 Define TADM			
Input Artifact	Step	Output	Notations
Segment LDM, Target Area Scope	4.1.1 Create TADM	TADM	IDEF1x UML Class Diagram
TADM, Data Element Matrix	4.1.2 Harmonize Baseline and Target Data Elements	Harmonized TADM	IDEF1x UML Class Diagram
Harmonized TADM	4.1.3 Normalize TADM	Normalized TADM	IDEF1x UML Class Diagram
Normalized TADM	4.1.4 Standardize TADM	Standardized TADM	IDEF1x UML Class Diagram

Figure 2-27. 4.1 Define Target Area Data Model Steps

2.5.1.1 4.1.1 Create Target Area Data Model

The purpose of this step is to create a TADM from the segment's LDM based on the scope of the target area resulting from the Gap Analysis and Alignment phase. Collect all the supporting documentation (e.g., CRUD Matrix, BPMs, System Data Element Matrix, System Data Element Definitions) that aligns with the target area.

2.5.1.2 4.1.2 Harmonize Baseline and Target Data Elements

The purpose of this step is to analyze the operational system data sources and elements mapped to the entity types within the scope of the target area. Capture all the data requirements from the operational systems and define within the TADM. Figure 2-28 provides an example of this process.

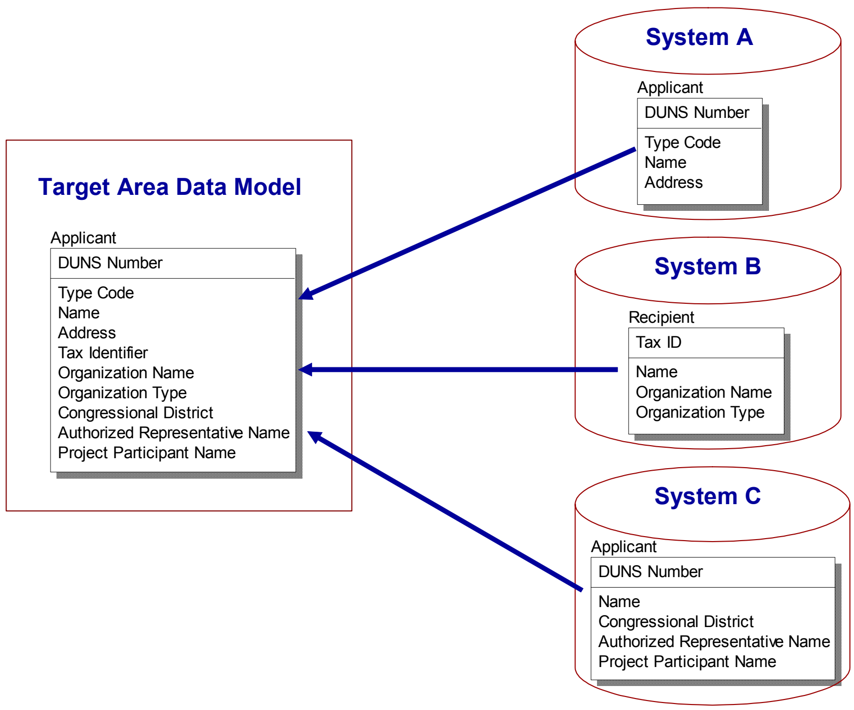

Figure 2-28. 4.2.1 Target Area Data Harmonization Example

2.5.1.3 4.1.3 Normalize Target Area Data Model

The purpose of this step is to normalize the TADM to the third normal form. Refer to Appendix F for an example of the normalization process. Figure 2-29 provides an example of putting the target Applicant in the example above in the third normal form at the entity type level.

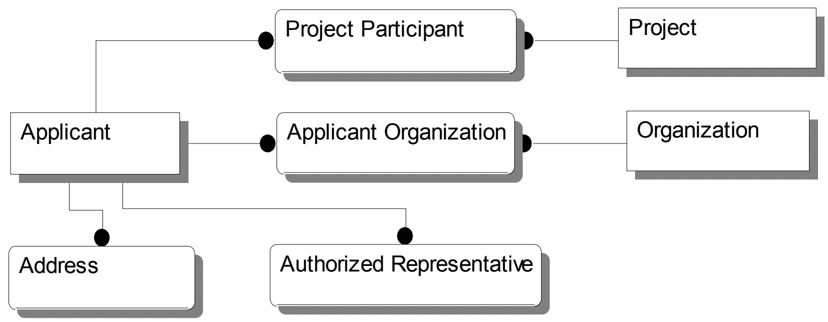

Figure 2-29. 4.1.3 Applicant Third Normal Form Example

4.1.4 Standardize Target Area Data Model

The purpose of this step is to ensure all entity types and attribute properties (e.g., name, definition, domain) conform to HUD and federal conventions (e.g., OMB, e-Gov). Figure 2-30 provides an example of attribute name and property conventions for conversion to Grants.gov.

Person Group				
Field Label	**Data Type**	**List Values**	**Min. # of Characters**	**Max # of Characters**
Prefix	AN	n/a	1	10
First Name	AN	n/a	1	35
Middle Name	AN	n/a	1	25
Last Name	AN	n/a	1	60
Suffix	AN	n/a	1	10

Figure 2-30. 4.1.4 Grants.gov Name Standardization Requirements

2.5.2 4.2 Define Data Security and Quality Control Measures

The purpose of the Define Data Security and Quality Requirements tasks is to identify and document the data security, privacy and quality requirements for the segment TADM entity types and attributes. Figure 2-3125 identifies the steps and artifacts of this task.

4.2 Define Data Security and Quality Control Measures			
Input Artifact	Step	Output	Notations
TADM Entity Types and Attributes, Business Requirements	4.2.1 Define and document Data Privacy and Security Measures	Data Privacy and Security Measures for Entity Types and Attributes	*Descriptions within Data Model or Spreadsheet **
TADM Entity Types and Attributes, Business Requirements	4.2.2 Define and document Data Quality Measures	Data Quality Measures for Entity Types and Attributes	*Descriptions within Data Model or Spreadsheet **
TADM Entity Types and Attributes, Business Requirements	4.2.3 Define and document Statistical Process Control Measures	Statistical Process Control Measures for Entity Types and Attributes	*Descriptions within Data Model or Spreadsheet **

** = No formal notation method*

Figure 2-31. 4.2 Define Data Security and Quality Control Measures Steps

2.5.2.1 4.2.1 Define and Document Data Security and Privacy Requirements

The purpose of this step is to identify and document the privacy and security measures within the data model itself or other supporting documentation for each entity type and attribute within the segment TADM. Private data are indicated using a flag (Y or N). Security measure documentation requirements for data include the requirements for (1) authentication, (2) access control and (3) confidentiality.

2.5.2.2 4.2.2 Define and Document Data Quality Measures

The purpose of this step is to identify and document the data quality measures within the data model itself or other supporting documentation for each entity type and attribute within the segment TADM. The minimum requirements are:

a. Data names, including business terms, abbreviated name, database or file name, standard screen name and standard report name
b. Definition
c. Valid value set (including value definition) or reasonable range of values
d. Business rules for data integrity
e. Physical specifications of data.[3]

An example from HUD's Total Information Quality Management Handbook of a template that can be used to document data quality measures is included in Appendix G.

2.5.2.3 4.2.3 Define and Document Statistical Process Control Measures

The purpose of this step is to identify and document

- Statistical performance benchmarks for the quality of data
- Monitoring requirements to alert users about variances in data quality.

Process control measures ensure that the information produced is correct, consistent and acceptable based on the variability of time. By monitoring the measures, the quality of information produced by a process can be validated against the agreed upon business requirements and standards. For example, a loans monitoring process has established benchmarks for a borrower's income versus the amount borrowed, and a periodic monitoring process measures the variances between the values and alert business users when the benchmark threshold is exceeded.

2.5.3 4.3 Identify Authoritative Source of Data

The purpose of this task is to identify a single data source that maintains the data within the HUD target architecture. Figure 2-32 identifies the steps and artifacts of this task.

---

[3] *Department of Housing and Urban Development Total Information Quality Management Handbook 3300.1,* Office of the Chief Information Officer (May 2003), p. 2-6.

4.3 Identity Authoritative Source of Data			
Input Artifact	Step	Output	Notations
Target Area Entity Types, Data Element Matrix, System Inventory Disposition	4.3.1 Identify and document Authoritative Source of Data for Target Area Entity Types	Operation Data Store Matrix with Authoritative Store Indicated	*Descriptions within Data Model or Spreadsheet **
Operation Data Store Matrix	4.3.2 Identify and document impacted Operation System Data Stores	Operation Data Store Matrix with Impacted Data Stores Indicated	*Descriptions within Data Model or Spreadsheet **

** = No formal notation method*

Figure 2-32. 4.3 Identify Authoritative Source of Data Steps

2.5.3.1 4.3.1 Identify and Document Authoritative Source of Data for Target Area Entity Types

The purpose of this step is to identify the 'best fit' operational system data store(s) to source the target entity type based on the operational system's disposition status using the Operational Data Store Matrix and the System Inventory Disposition. (Note: the disposition status is based on the operational system's fit to the target architecture. Possible statuses are 're-engineer', 'retire', 'integrate' or 'migrate').

Compare the selected data store elements to the target entity type attributes and analyze the gaps between the data specifications. Multiple operational data stores may source a single target entity type; conversely, a single operational data store may source multiple target entity types. If a 'fit' data source does not exist, then the best alternative is to create a new data source.

Based on this analysis, document the best method to source each target entity type within the Operational Data Store Matrix.

2.5.3.2 4.3.2 Identify and Document Impacted Operational Data Stores

The purpose of this step is to identify the impacted data sources using the Operational Data Store Matrix updated in the previous step. These are the data sources mapped to the target entity type but not selected as the authoritative source. During the transition and migration planning phase, it is determined if these data sources are populated using replication or retired.

Based on this analysis, document the impacted operational data sources in the Operational Data Store Matrix.

2.5.4 4.4 Validate Target Area Data Model and Requirements

The purpose of this task is to validate the TADM, data security and quality measures, as well as authoritative source data selections with the EA Team, Business Data Steward and DCB. Based on the feedback, update the impacted artifacts accordingly. Figure 2-33 identifies the steps and artifacts of this task.

4.4 Validate TADM and Requirements			
Input Artifact	**Step**	**Output**	**Notations**
Target Area Entity Types, Data Security and Quality Measures, Operational Data Store Matrix	4.4.1 Validate Target Area Data Model and Requirements with the EA Team	Artifact Revisions	
Target Area Entity Types, Data Security and Quality Measures, Operational Data Store Matrix	4.4.2 Validate Target Area Data Model and Requirements with the Business Data Stewards and DCB	Artifact Revisions	
Recommended Artifact Revisions	4.4.3 Revise TADM and Impacted Artifacts	Revised Artifacts	

Figure 2-33. 4.4 Validate TADM and Requirements Steps

2.5.4.1 4.4.1 Validate TADM and Requirements with the EA Team

The purpose of this step is to validate with the EA Team the following target area artifacts:

- TADM
- Entity type and attribute security requirements
- Entity type and attribute quality and statistical control measures
- Entity type authoritative data source selection and impacted data sources.

The EA Team is responsible for

- Ensuring data artifacts align with the HUD EA
- Verifying data artifacts' documentation and implementation consistently conform to the requirements
- Analyzing the impact of new and changed data architecture components
- Coordinating the dissemination of data architecture releases.

2.5.4.2 4.4.2 Validate TADM and Requirements with the Business Data Stewards and DCB

The purpose of this step is to validate with the Business Data Stewards and DCB the following target area artifacts:

- TADM
- Entity type and attribute security requirements
- Entity type and attribute quality and statistical control measures
- Entity type authoritative data source selection and impacted data sources.

The Business Data Stewards and DCB are responsible for:

- Verifying data artifacts are accurate, complete and compliant with standards
- Ensuring data artifactsare accessible and reused appropriately across the Department
- Ensuring data artifacts are managed as prescribed by HUD EA policies and guidelines.

2.5.4.3 4.4.3 Revise Target Area Data Model and Impacted Artifacts

The purpose of this step is to update the impacted artifacts based on the recommended changes from the EA Team, Business Data Steward and DCB.

2.5.5 4.5 Create Data and Exchange Specifications

The purpose of this phase is to create the physical data specifications to guide the implementation or transformation of the authoritative source of data and create the interface schema for the application component(s). Once the specifications and schema have been created, publish them in a central repository or file share. Figure 2-34 identifies the steps and artifacts of this task.

4.5 Create Data and Exchange Specifications			
Input Artifact	Step	Output	Notations
TADM	4.5.1 Create the Physical Data Specification	Physical Data Specification for the Authoritative Source of Data	DDL
TADM or Component Data Model	4.5.2 Create the Interface Schema for Application Component(s)	Interface Schema for Application Component(s)	XML
Recommended Artifact Revisions	4.5.3 Publish the Data Specification and Interface Schema	Data Specification and Interface Schema published in centralized repository or file share	

Figure 2-34. 4.5 Create Data and Exchange Specifications Steps

2.5.5.1 4.5.1 Create the Physical Data Specification

The purpose of this step is to create the physical data specification. The specification can be generated using automated transformation of the TADM into Data Definition Language (DDL) or manually using the physical properties documented within the TADM. The physical data specification provides the standard for the implementation of the authoritative and replicated sources of that data. The standard definition can be extended for a performance or data base management system (DBMS), but the implemented definition must map to the physical definition.

2.5.5.2 4.5.2 Create the Interface Schema

A TADM typically is scoped based on a single application component, but there may be instances when the planners decided to incorporate the data requirements for multiple application components into a single target area project. If the software tool used to document the model provides the functionality to generate the model's metadata schema into Extensible Markup Language (XML) format, it is recommended to create temporary component data models to generate the XML schema for the component interface. Otherwise, the XML schema for the application components interfaces need to be created manually.

2.5.5.3 4.5.3 Publish the Data Specification and Interface Schema

The purpose of this step is to publish the data specifications and component interface schema in a centralized repository or file share. Organize the data specifications and the XML schemas by business process IE.

2.5.6 4.6 Update and Align Artifacts

The purpose of this task is to integrate the target area data model into the segment's LDM, update the IE matrix, and validate the alignment of the segment LDM with the DRM. Figure 2-35 identifies the steps and artifacts of this task.

4.6 Update and Align Artifacts			
Input Artifact	Step	Output	Notations
TADM	4.6.1 Integrate Target Data Model into Segment LDM	Updated LDM	IDEF1x UML Class Diagram
TADM	4.6.2 Update the IE Matrix	Updated IE Matrix	*Metadata Repository or Spreadsheet **
Updated Segment LDM	4.6.3 Validate Alignment with DRM	Revised Segment LDM Mappings to DRM	*Metadata Repository or Spreadsheet **

** = No formal notation method*

Figure 2-35. 4.6 Update and Align Artifacts Steps

2.5.6.1 4.6.1 Integrate Target Area Data Model into Segment Data Model

The purpose of this step is to integrate all new and changed target area data model components back into the segment LDM in an automated or manual fashion. The TADM is a temporary model to support the development effort and eventually is to be deleted.

2.5.6.2 4.6.2 Update the IE Matrix

The purpose of this step is to update the IE matrix with the entity types and attribute changes completed in the TADM. Determine if entity types and attributes created in the TADM are to be included in the IEs or only support internal processing requirements.

2.5.6.3 4.6.3 Validate Alignment with the DRM

The purpose of this step is to validate and update the mapping of the segment LDM entity type to the conceptual HUD DRM central entity types and subject areas. Changes required to the DRM are facilitated by the Segment Target Area Data Model (IPT) Data Architect. The IPT Data Architect submits the recommended revisions to the EA Team to validate the justification for the revisions and analyze the impact to the overall HUD EA. The EA Team presents its recommended revisions and impact analysis results to the DCB for approval and prioritization. The approved revisions and release schedules are coordinated and disseminated by the EA Team.

2.6 5.0 Migration and Implementation Planning

The Migration and Implementation Planning phase is an application and process-driven task. The Target Area Data Standardization phase provides artifacts that guide this phase. The artifacts are:

- TADM
- IE Matrix
- Physical Data Specifications for the Authoritative Source of Data
- Interface Schemas for Application Components
- Operational Data Source Matrix.

From a data perspective, the planning phase addresses

- Baseline data source integration and interface requirements
- Data exchange protocols
- Data source implementation platforms
- Data conversion and quality cleansing
- Data replication (push versus pull)
- Data recovery and archive
- Alignment to the HUD DRM
- Data marts and warehouses to support reporting requirements.

Changes required to the Target Area Data Model (TADM) to support migration and implementation planning requirements are facilitated by the Segment Target Area Data Model (IPT) Data Architect. The IPT Data Architect submits the recommended revisions to the EA Team to validate the justification for the revisions and analyze the impact to the overall HUD EA. The EA Team presents its recommended revisions and impact analysis results to the DCB for approval and prioritization. The approved revisions and release schedules are coordinated and disseminated by the EA Team.

2.7 6.0 System Modernization and Integration

The System Modernization and Integration phase is an application and process-driven task. The Target Area Data Standardization phase provides artifacts that guide this phase. The artifacts are:

- Target Area Data Model (TADM)
- IE Matrix
- Physical Data Specifications for the Authoritative Source of Data
- Interface Schemas for Application Components
- Operational Data Source Matrix.

From a data perspective, the modernization phase needs to align to the

- Interface Schema for Application Components
- Physical Data Specifications or DDL of the Authoritative Source of Data and replications of the source.

The DDL is tweaked for implementation platform or performance optimization. If the implemented definition is extended, the resulting implemented DDL must map to the TADM definition. The standard interface schema must be complied by the internal system owner and the external business partner.

Changes required to the TADM to support modernization and system integration requirements are facilitated by the Segment IPT Data Architect. The IPT Data Architect submits the recommended revisions to the EA Team to validate the justification for the revisions and analyze the impact to the overall HUD EA. The EA Team presents its recommended revisions and impact analysis results to the DCB for approval and prioritization. The approved revisions and release schedules are coordinated and disseminated by the EA Team.

SECTION 3. SEGMENT DATA ARCHITECTURE GOVERNANCE

3.1 Segment Data Architecture Structure

Governance is best defined as the process and rules that ensure an organization's business is conducted properly. EA governance is the process and rules that ensure an organization's architecture is developed and used properly. The HUD Segment Data Architecture Development Methodology (SDADM) governance structure requires the new role of Segment Integrated Project Team (IPT) Data Architect (see Figure 3-1).

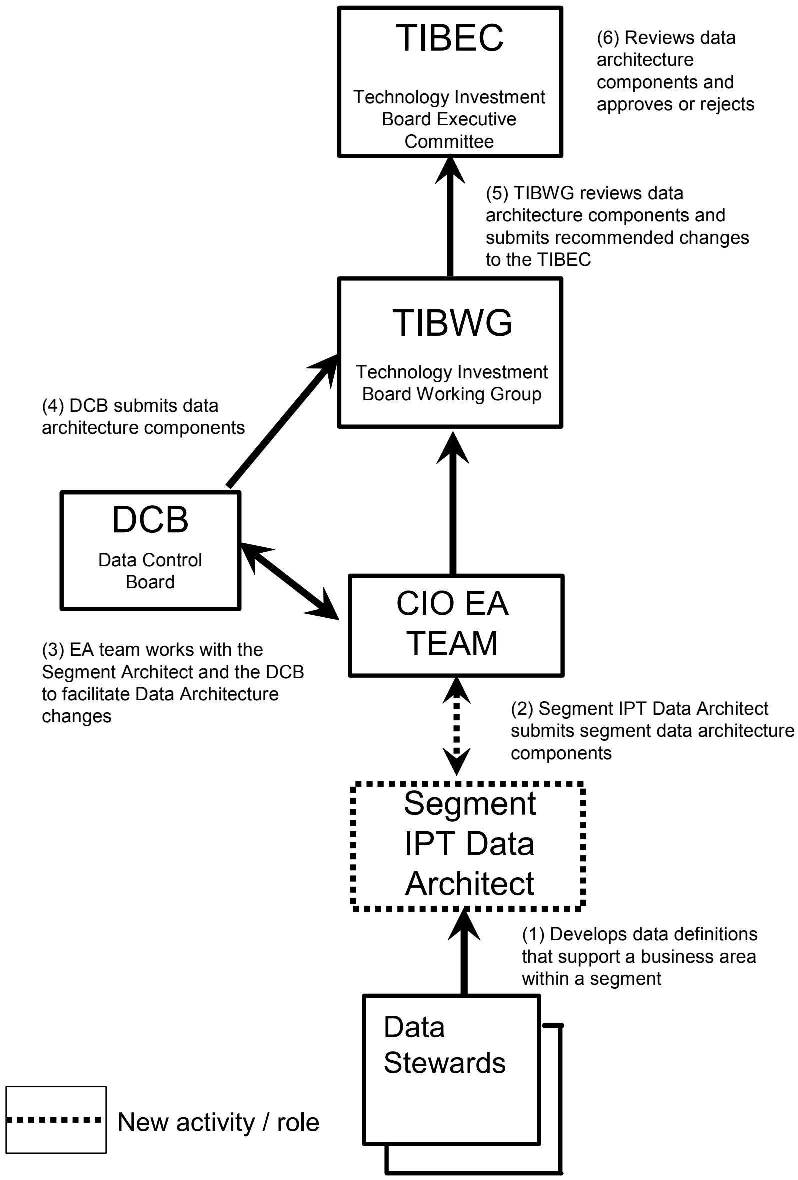

Figure 3-1. HUD EA- SDADM Governance Structure

3.1.1 Office of the Chief Information Officer (OCIO) EA Team

The Chief Architect and EA team of HUD's OCIO are responsible for developing and maintaining HUD's EA. These responsibilities include

- Managing the EA Practice and associated processes
- Maintaining and updating the EA
- Assisting in the creation of segment architectures, serving on the segment IPT and aligning the segment architecture with the HUD EA
- Coordinating and disseminating releases of baseline, target and interim target architectures
- Reporting changes to the baseline and interim target architectures
- Maintaining the EA migration plan
- Reviewing the EA for redundancies and obsolescence
- Analyzing and reporting on the crosscutting impact of major EA changes
- Providing EA assistance to the business and program areas, as needed
- Maintaining the Enterprise Architecture Management System (EAMS).

3.1.2 Technology Investment Board Working Group (TIBWG)

The purpose of the TIBWG, as quoted from its charter is[4]:

> The Technology Investment Board Working Group (TIBWG) is a permanent working group established to support the TIBEC in managing HUD's IT investment portfolio and EA.

The Business and Program Offices represented on the Technology Investment Board Executive Committee (TIBEC) provide representatives for the TIBWG. The EA team submits EA work products for review by the TIBWG (i.e., HUD's Target EA and EA policies). The TIBWG reviews the proposed products and works with the EA team on recommended changes before forwarding for approval by the TIBEC.

3.1.3 Technology Investment Board Executive Committee

The TIBEC responsibilities in regard to EA activities consist of

- Establishing EA as a Department-wide priority
- Approving target and interim target architectures
- Approving the IT portfolio, ensuring its alignment with the target architecture
- Approving EA and technology standards.

The Secretary and Deputy Secretary chair the TIBEC and the members are comprised of Assistant Secretaries and Principles of HUD. The detail functions of the TIBWG and the TIBEC are described in charters (refer to Appendix A and B).

Business and program offices have the following responsibilities in HUD's EA development:[5]

---

[4] U. S. Housing and Urban Development, Technology Investment Board Working Group Charter, July 7, 2002

- Provide architecture information for sponsored initiatives
- Update baseline architecture information of sponsored initiatives
- Ensure sponsored initiatives conform to the EA and technology standards
- Participate on the TIBEC and TIBWG with respect to EA concerns
- Conduct business opportunity analyses
- Establish internal office procedures for keeping architecture information accurate and current.

3.1.4 Data Control Board (DCB)

The mission of the DCB is to improve the quality of HUD's data. It functions as the steering body for data architecture components of the HUD EA. The responsibilities of the DCB are[6]:

- Establish an enterprise data management practice and promote the management of HUD's data as a strategic enterprise assets
- Review data architecture components and recommend components for submission to the TIBEC
- Review and recommend EDM policies, standards and procedures for submission to the TIBEC
- Promote data accuracy, completeness, consistency, validity and timeliness across HUD IT systems
- Review and approve data quality plans and monitor certification for all HUD mission-critical data systems
- Prioritize data management requirements and recommend related IT projects for submission to the TIBEC.

The DCB is chaired by a HUD OCIO representative. Membership consists of staff from primary offices within HUD.

3.1.5 Data Steward

A segment architecture has one or more Data Stewards based on the complexity of the segment's data requirements. The Data Steward role is filled by staff from lines of business within a segment. The responsibilities before the development of the SDADM were to[7]

- Collaborate with the EDMG and the DCB to manage data in conformance with the principles and guidelines established by HUD's enterprise data management practice to meet the requirements expressed by end-user organizations
- Participate on working groups and communities of practice to define data management standards
- Implement data management standards to promote data access and reuse across the Department
- Serve as the principal consultant to end users on the content, quality and appropriate use of data sets.

[5] U. S. Housing and Urban Development Enterprise Architecture Framework, July 2002, Pg 1-4.
[6] U. S. Housing and Urban Development Enterprise Data Control Board Charter, Pg 3.
[7] U. S. Housing and Urban Development Enterprise Data Management (EDM) Policy, Pg 1-5.

New responsibilities necessary to support the SDADM include

- Collaborating with the segment IPT data architect to validate data requirements and define data architecture components
- Being familiar with the business line's processes and having the ability to identify Information Exchanges (IE) within or externally to the business line
- Working with the business line's database administrators to describe the data's schema to the Segment IPT Data Architect for standardization activities.

3.1.6 Segment IPT Data Architect

The Segment IPT Data Architect manages the structure and standards of the data for a HUD segment. This role is the point of contact for the segment data architecture, the interface to the DCB, and the EA team for the segment's data components. This role requires a broad understanding of data management and data architecture practices, in addition to understanding the segment's business functions and goals. The responsibilities of this role follow:

- Collaborating with the business data stewards within the segment to integrate and document all data requirements according to HUD EA standards
- Understand the segment's business functions and the associated data requirements
- Direct all data related activities on the IPT and be responsible for all outputs from those activities
- Assist in the development of application component data requirements and in any modifications to the data architecture that the component requires
- Coordinate with the EA team and the DCB on the submissions of the segment's data architecture components into HUD's EA
- Coordinate with the DCB data standards submitted by other HUD segment data architects for potential impact

3.2 Segment Data Architecture Governance Process

The steps in the Segment Data Architecture governance process are depicted in Figure 3-1. They are

(1) Business data stewards develop data specifications during the development of a segment data architecture.
(2) Segment data architecture artifacts that impact HUD's EA are submitted to the EA team for review.
(3) If the artifacts adhere to HUD's EA data architecture standards, the EA team forwards the artifacts to the Data Control Board (DCB) for approval to be included in HUD's EA.
(4, 5, 6) These artifacts then follow HUD's existing governance procedures and are reviewed by the Technology Investment Board Working Group (TIBWG) after being submitted by the DCB. After submission to the Technology Investment Board Executive Committee (TIBEC), the artifacts are subject to approval for inclusion into HUD's EA.

SECTION 4. SOFTWARE DEVELOPMENT METHODOLOGY ALIGMENT

4.1 Purpose

The Segment Data Architecture Development Methodology (SDADM) develops artifacts that provide information necessary to build or acquire a software solution for a business need as defined in the segment's target architecture. Integrating the SDADM artifacts with the System Development Methodology (SDM) ensures consistency with HUD's segment data architectures.

4.2 HUD SDM and SDADM Alignment

As stated in the SDADM Governance Section, the Segment Integrated Project Team (IPT) Data Architect is responsible for providing the continuity from the development of the segment data architecture to the building of the application components that use the segment's data artifacts. If during any phase of the SDM the development team identifies missing data elements from the segment data architecture, the development team requests modification to the segment's data architecture through the Segment IPT Data Architect and the segment data architecture governance process.

4.2.1 1.0 Initiate Project Phase

The purpose of this phase is to identify an information management need and determine whether to commit resources to the solution. The segment data architecture identifies the segment's information needs in its target architecture and provides the alignment to HUD's business functions, mission and business needs.

4.2.2 2.0 Define System Phase

This phase contains three activities that produce artifacts that are also produced for the SDADM. The activities are primarily performed within the Determine Data Requirements step of this phase. A description of the three SDM activities and the SDADM artifacts that fulfill the requirements of this activity follows.

- **SDM Activity:** Develop a logical data model (LDM) showing how the data within the system are related in order to minimize duplication of data, provide flexibility and allow the data to be mapped to a wider variety of database designs. Define data attributes in the model. Document definitions in the Data Requirements Document.[8]

  SDADM Artifact: The segment LDM documents the results of this activity. The LDM includes entity types, data attributes, relationships and the properties associated with these data elements. Supporting matrices identify the business function actions on the data elements and the business data steward for each data element.

[8] Ibid, pg 2-3

- **SDM Activity:** Cross-check functional and data requirements to ensure consistency among requirements.[9]

  SDADM Artifact: The Business CRUD Matrix documents the results of this activity. The information exchanged between business processes and external stakeholders is validated against the LDM.

- **SDM Activity:** Use the requirements matrix to trace functional and data requirements to system objectives and needed functions approved by the Technology Investment Board (TIB).[10]

  SDADM Artifact: The segment architecture provides the mapping between data and business functions that align with HUD's mission, business needs and strategic plan.

4.2.3 3.0 Design System Phase

This phase contains two steps affected by the SDADM, the Design Database Specification and Design Program Specification. The SDM database specification requirement is fulfilled by the physical data specification generated from the Segment Architecture LDM. The physical data specification provides the structure and standards (e.g., naming conventions, domains, lengths) for the implementation of the database. The data specifications implemented within the database may be altered for performance and the target platform; the implemented data specifications must align to the physical data specification and the standards from the segment data architecture.

The SDADM provides the business rules for the segment's data. These rules ensure the security, privacy and quality of the authoritative source of the data. The Design Program Specification step needs to ensure the business rules documented within the artifacts of the SDADM are encapsulated within the specifications for the application component. The Information Exchange (IE) schema provides the specification for how data is transmitted between application components. The program specification needs to ensure application components adhere to this standard.

[9] Ibid, pg 2-3
[10] Ibid, pg 2-4

APPENDIX A. Segment Data Architecture Best Practices

In the report, "*Report of the Enterprise Data Management Group, Segment Data Architecture Best Practices Analysis, February 22, 2005*", the EDMG studied the data architecture and governance practices of three Federal Agencies and identified eight best practices that were common among these agencies. The three agencies studied were the Department of the Interior (DOI), the Department of the Navy (DON), and U. S. Customs and Border Protection (CBP).

The eight best practices that contributed to a successful data architecture development are the following.

- **Top-down and bottom-up analysis.** All the agencies established initiatives to build enterprise data architectures using both a top-down and bottom-up approach. The top-down perspective produced the taxonomy to organize the agencies' enterprise-wide data architecture based on the business functions in the BRM. The bottom-up efforts served to decompose, detail and validate the data architecture with the as-is artifacts.

- **Incremental development.** In order to make the decomposition and definition of its enterprise data architectures more manageable, each agency developed strategic plans based on business priorities to incrementally develop data architectures for targeted business areas.

- **Alignment of data and business architectures.** Processes were implemented to ensure that decomposition of the data and process models retained alignment so that the artifacts of the models could be mapped. This mapping served to (1) validate the synchronization of the data and BPMs, (2) validate the IEs, (3) identify missing data elements and business processes and (4) identify opportunities for data reuse and/or integration.

- **Communicate, coordinate and collaborate.** The "3 Cs" concept is common; all the agencies had established at least one working group as a means to foster the concept across the enterprise. These working groups contribute the structure and definition of the enterprise data architecture, coordinate the alignment and harmonization of the functional data architectures with the enterprise data architecture, and foster governance within the respective organizations.

- **Business Data Stewardship.** All the agencies interviewed had a Business Data Stewardship program in place.

- **Established and enforced data architecture governance.** Each agency stated that developing enterprise-level data architecture would not have been successful without established policy, procedures and standards for architecture development and data management. Furthermore, the governance had to be flexible enough to allow for program or functional-area extensions. Established governance was not enough; it also had to be supported by senior -level management to ensure that data artifacts produced by the various areas adhered to the agency's policies and standards.

- **Architecture driven modernization efforts.** The agencies use enterprise data architecture models to provide common semantics and structure for the physical data and information exchange specifications.

- **Centralized and effective access to EA artifacts.** The Department of the Interior (DOI) and the Department of the Navy (DON) have an extensive centralized repository or application that enables enterprise-wide access to the artifacts of the data architecture and other EA models.

APPENDIX B. Parallel Decomposition

The Segment Architectures will incrementally decompose both the business and data architecture in parallel from the conceptual to the logical level. The parallel decomposition of the architectures will provide

- The 'line of sight' from the implemented data specification to the HUD BRM
- Aligned data and process infrastructure to identify application components to implement HUD business functions
- Opportunities for information reuse and system integration
- Specification to exchange information with external stakeholders.

The Information Engineering Methodology founded by James Martin is based on the concept that the foundation for an overall architecture is dependent on the analysis of the data, activities and interactions between the data and activities (see Figure B-1).

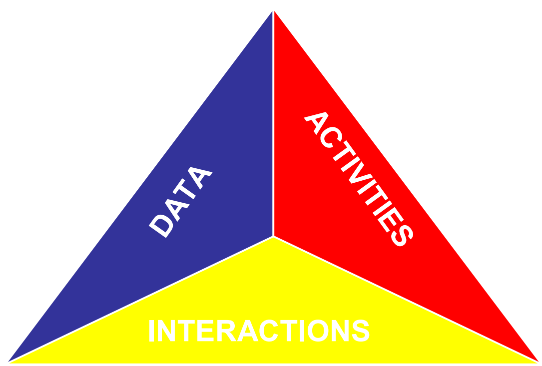

Figure B-4-1. Information Engineering View of Business Analysis

The diagram above indicates that analysis of a business area must incorporate the data view, the activity view and the interactions between these views. Applying this concept to the HUD EA Framework, the result is both segment data and business models that are aligned so that the interactions between the models (IEs and process actions) can be used to validate the context and content of each 'view'.

In the early days of Information Engineering the data and process models were often developed independently of each other. The resulting models were not cohesive, had excessive coupling and required extensive reconciliation efforts to define the interactions between models. The technique of parallel decomposition was employed to ensure symmetry between business process and data models at each layer of the model decomposition (see Figure B-2).

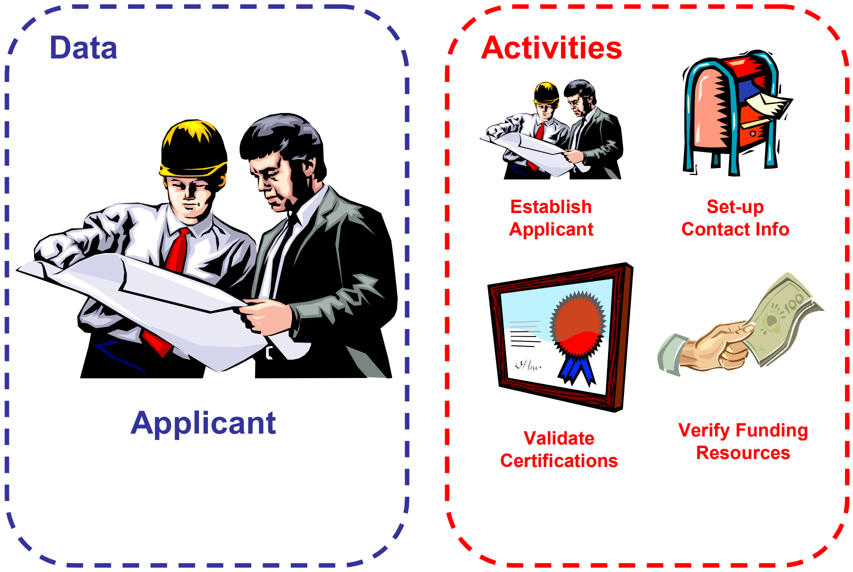

Figure B-4-2. Example of Parallel Decomposition

Decomposition of data and business process models (BPMs) in parallel enable the models to be balanced, synchronized and aligned to strategic business functions. The results of the decomposition of one of the models serve to confirm and guide the development of the other model. Construction of one model identifies and confirms objects in the other model.

Parallel decomposition is based on the premise that all data requirements identified should have corresponding activity requirements at the same level of detail, and vice versa. Figure B-3 demonstrates the results of parallel decomposition as it pertains to an illustrative example such as the Accept Applicant business process within the Grants Management business function.

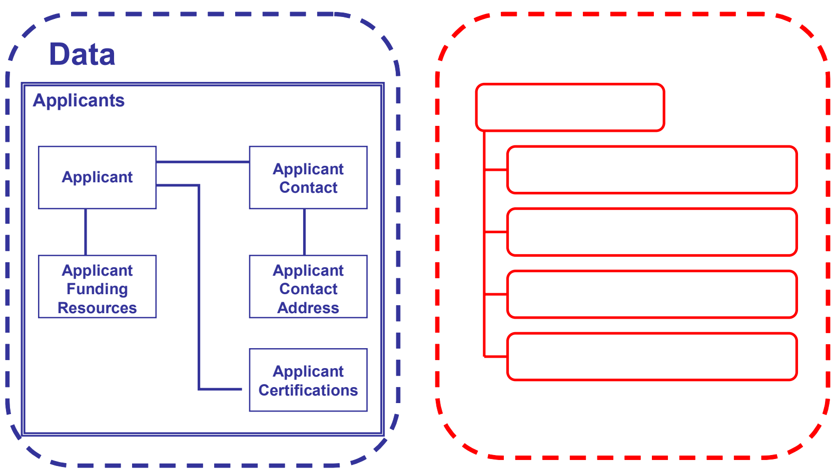

Figure B-4-3. Example of Accept Grant Applicant Parallel Decomposition

Parallel decomposition not only serves as a technique to confirm and guide the development of the process and data models within a segment, but it also sets the foundation to identify application components that fully encapsulate the data and processes to deliver the full functional capability of the services it provides. For example, the application component, Accept Applicant, might provide the following services:

- Establish Applicant
- Set-up Applicant Contact Information
- Validate Applicant Certifications
- Verify Applicant Funding Resources.

The component could be used for all or one of these services required by a consuming component. Furthermore, the component would be the authoritative source for creating the basic Applicant, Applicant Contact, Applicant Funding Resource and Applicant Certification data.

The interrelationship between the HUD BRM and DRM is maintained within the Segment Architectures. Specifically, the BRM provides the alignment to the HUD mission and strategic goals by identifying and defining business functions that define "what HUD does" without regard to organizational boundaries. The DRM provides the structure and high-level data requirements to support those business functions in terms of subject areas and entity types. The decomposition of those business functions and their data requirements into interface specifications and application components is achieved incrementally within the Segment Architectures (see Figure B-4).

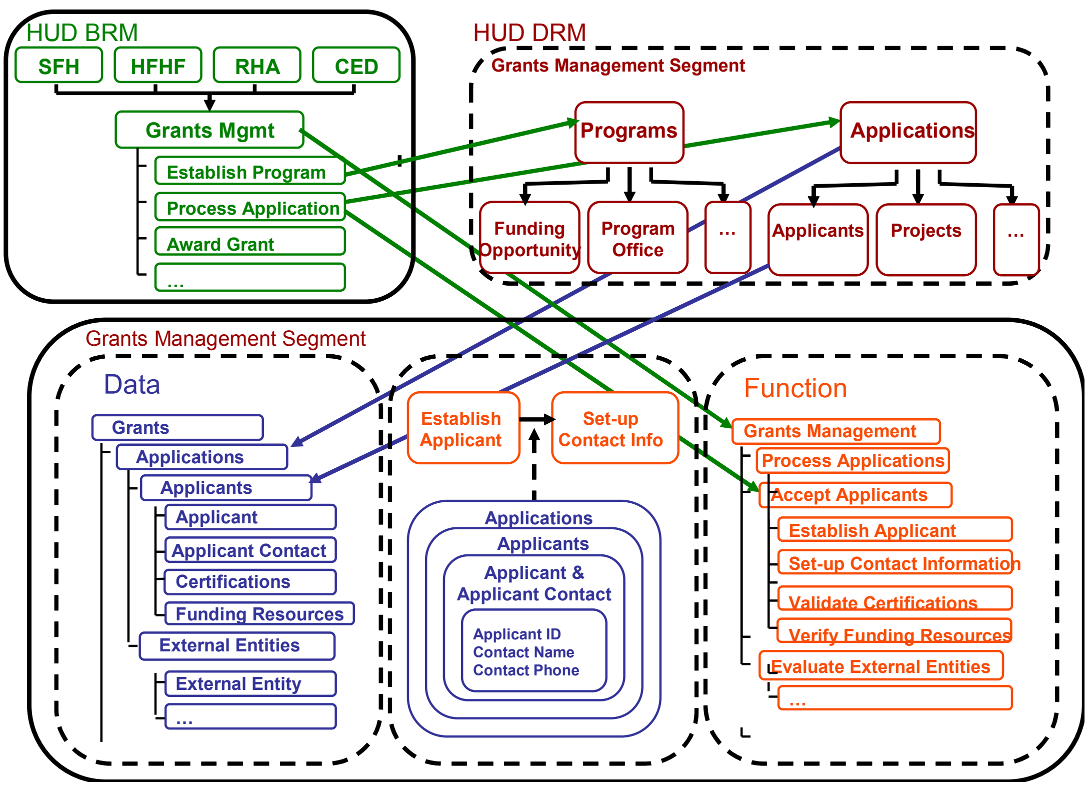

Figure B-4-4. Interrelationship between BRM, DRM and Segment Architectures

The Segment Architecture is seeded with the business function and sub-functions from the BRM and the subject areas, subject classes and central entity types from the DRM. The segment architecture teams decompose both the data and BPMs in parallel to (1) provide the standard for the internal and external interface exchanges, (2) identify internal application components, and (3) identify the interrelationships between the application components.

APPENDIX C. Incremental Development Approach

HUD uses Segment Architectures to incrementally detail EA efforts into attainable pieces. A Segment Architecture is an IT architecture for an individual Line of Business (LOB) (e.g., Multi-family Housing Finance) or a cross-cutting service (e.g., Tracking and Workflow, Grants Management).[11] These segments define the scope of the Segment Data Architectures development effort and are aligned with business functions within the HUD BRM. As such, the initial conceptual HUD DRM is decomposed into a logical view of HUD's information assets incrementally as the segment data architectures are developed (see Figure C-1).

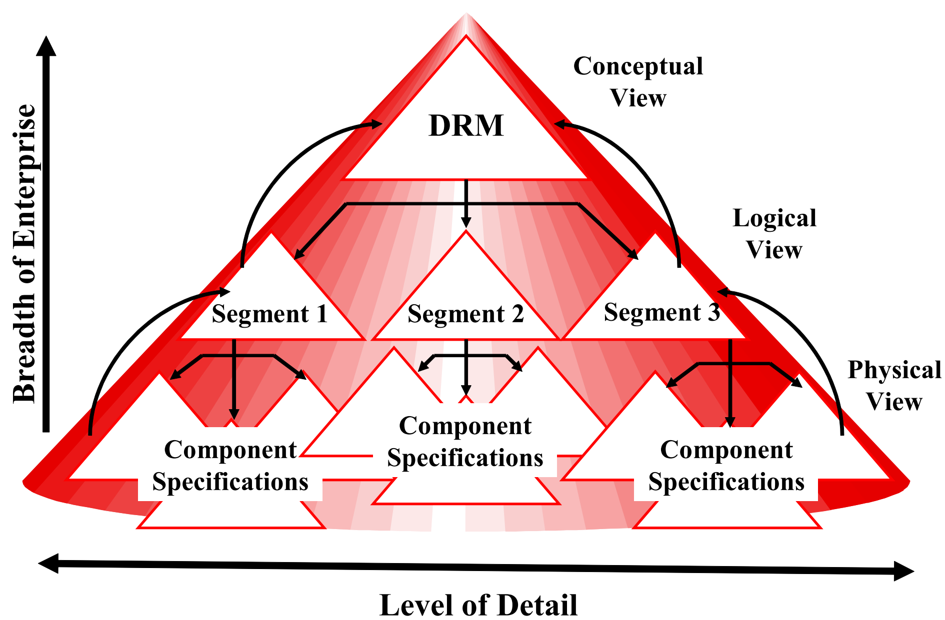

Figure C-1. Incremental Approach to Developing HUD DRM

The Segment Data Architecture Models are decomposed and detailed into standardized definitions of the segment's information assets within the taxonomy of the HUD DRM. The resulting logical view of the segment data will be comprised of subject areas, independent and dependent entity types, relationships, attributes, properties, and business rules. The segment's data model will serve to validate the taxonomy of the DRM, provide the logical data perspective for the DRM, and facilitate the identification of data-centric services or components for detailed design and implementation.

The Component Data Models further detail and refine targeted areas of a Segment Data Architecture Model to create the data specification and exchange schemas for component acquisition or development. It will serve to validate the standardization of the DRM data specification, provide the physical data perspective for the DRM and create the standard data specification and exchange schema for the implementation team.

[11] Housing and Urban Development, Target Enterprise Architecture Framework, July 2002

The incremental approach to narrow the scope of the data architecture being analyzed as the level of detail increases mitigates the overall risk to the modernization of HUD's information environment. The approach incrementally and systematically assembles components to implement targeted improvements to HUD's system inventory that align with HUD's overall strategic plan. Building an enterprise-wide application from scratch presents significant risk, but using proven components within a well-defined EA framework confines the risk to the component area being implemented.

APPENDIX D. HUD Integrated IT Lifecycle Framework

The SDADM follows the HUD Integrated IT Lifecycle Framework. This framework integrates HUD's EA practice with its Information Technology Investment Management (ITIM) and enables HUD to meet the current OMB requirements to justify its IT expenditures. The three phases of the framework are: (1) architecture, (2) investment and (3) implementation (see Figure D-1).

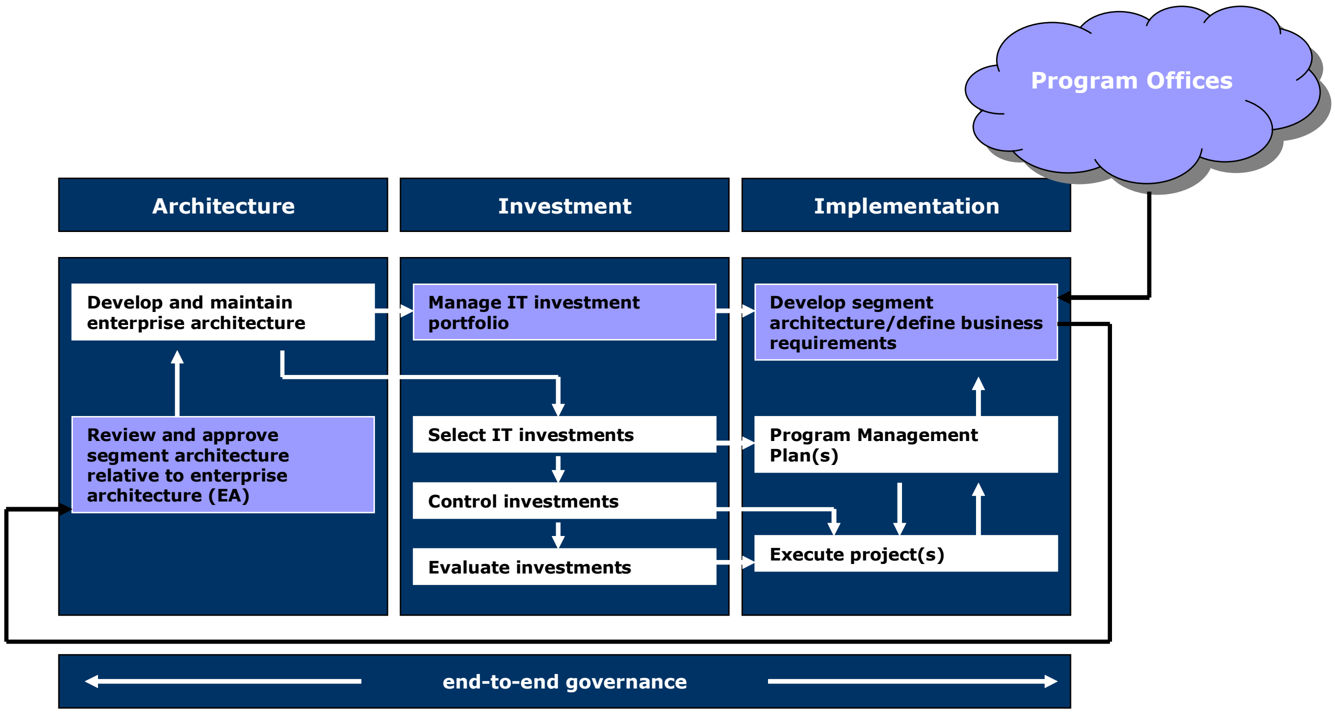

Figure D-1. HUD Integrated IT Lifecycle Framework[12]

The initial phase of the SDADM identifies the artifacts from the HUD EA business and data layer to create the Segment Data Architecture Model. The model is decomposed to additional layers of detail to validate the structure and context of the segment and establish the linkages between the segment data and business architectures. The resulting segment data architecture and its alignment to the overall HUD EA are reviewed and approved by stakeholders.

The linkages between the segment business and data architectures identify highly cohesive sets of functionality, or application components, that can be used to incrementally transition HUD's system inventory from its baseline state to the desired target state. The alignment of the baseline system inventory to the target application components identifies the gaps in functionality needed to support HUD's desired target state, as well as identify the redundancies across systems that provide the common functionality but with a "stovepipe" view. The target application components, along with an assessment of the baseline systems that currently support that functionality, provide input to the Manage the IT Investment Portfolio phase.

[12] *HUD's IT Lifecycle Framework Presentation,* OCIO EA Team, March 2005

Application components that are selected and receive funding to implement targeted improvements into HUD's baseline system inventory proceed onto the next phase of development. Data and business models of the selected components provide the high-level requirements. The models are further developed to provide functional specification based on detailed business requirements. The functional specifications can be used to make the decision whether to acquire the functionality from another agency or vendor, build the application function from scratch, or leverage legacy system functions.

APPENDIX E. FIPS PUB 184 Data Modeling Method and Conventions

The FIPS PUB 184 (12/21/1993) issued by National Institute of Standards and Technology (NIST) announced the adoption of Integration Definition for Information Modeling (IDEF1x) as a Federal Information Processing Standard. The FIPS PUB also prescribes an approach to identifying and defining data entities, the interrelationships between the entities (referred to as relationships) and the data element properties and representations (attributes and attribute properties).

The prescribed method within the FIPS PUB is summarized within this appendix. Further details about IDEF1x conventions and methods to detail data model components can be accessed using the following link: http://www.idef.com/Downloads/pdf/Idef1x.pdf.

- **Entity Definition** that entails identifying and defining the entities (or data elements in this case) that falls within the domain of the segment being analyzed. The method recommends defining the most obvious entities and requires the least amount of research. This includes
    - **Identify Entities** that represent a set of "things" of importance to the business and have data associated with them. These "things" can typically be identified from the source material as object noun phases. They are typically the subjects that the business processes act upon.
    - **Define Entities** by providing a name, definition and aliases (other names the entity might be known as) for each entity.
- **Relationship Definition** that entails identifying and defining the interrelationships between the entities. This includes
    - **Identify Related Entities** (or the pairing of the entities) that have a meaningful association or connection between entity instances. The methodology recommends not depicting all possible relationships but defining the interconnections between "parent" and "child" or independent and dependent entities.
    - **Define Relationships** by specifying a name, the dependency between the entities and a definition.
    - **Construct Entity-Level Diagrams** to graphically depict the entities and relationships. The "entity-level" diagram is typically not attributed and relationship properties of the relationship (e.g., optionality, cardinality) are not defined.
- **Key Definition** that entails refining the relationships, identifying key attributes and validating the model. This includes:
    - **Resolve Non-Specific Relationships,** or those relationships between entities that do not rely on each other for existence and are joined by many-to-many (M:N) relationships. Typically this process identifies new entities referred to as "associative" or "intersection" entities.

- **Depict Function Views** (referred to in this methodology as a Class) to manage the volume and complexity of the data model as more components are being identified and defined. These views or perspectives organize the data model that are directly related to functional aspects of the segment being analyzed and are instrumental in the validation of the model and the scope of application components moving forward. The standard recommends two methods to organize the model into Classes. One is to use a selected sample source material as the "topic" of the Class (e.g. Application); and the other is to relate the class to specific process.

- **Identify Key Attributes** to identify attribute (or property) values that uniquely identify each instance of an entity. A "key" can be comprised of a single attribute or a combination of attributes. Furthermore, there can be more than one set of attributes that uniquely identify an instance of an entity the specification of primary and alternative keys may be necessary.

- **Migrate the Primary Keys** to the related entities, usually replicated from the independent or "source" entity to the dependent or "receiving" entity, to create the foreign keys that represent all of the attributes of the primary key of the "source" entity. If the relationship participates in the identification of the "receiving" entity, then the foreign keys are part of the "receiving" entities' primary key. Note that this process typically occurs "automatically" during the definition of the relationship or the transforming of the physical model in most data modeling tools.

- **Attribute Definition** that entails identifying "facts about the business" or attributes (in our case, data element properties), determining which data element the attributes store facts about, and refining and validating the data model. This includes

  - **Identify Nonkey Attributes** that are "facts" about the entity that the business is interested in storing. These "facts" can typically be identified from the source material as descriptive noun phrases (noun phrases that are used to describe objects).

  - **Establish Attribute Ownership** and assign each non-key attribute to one owner entity. The entity to which the attributes are assigned must define a specific "fact" about that entity and only that entity.

  - **Define the Attribute** that includes the attribute name, definition and aliases/synonym(s).

  - **Refine the Model** to ensure the attributes do not "repeat" for each occurrence of the entity and are identified by the full primary key of the entity on which it resides. This analysis and resolving of the anomalies may identify additional entity types. For example, the attribute Grade may occur multiple times for a Student and is fully identified by the Student, the Course and for those who did not do too well the first time, and the time period the Grade was achieved. Therefore, the attribute Grade should reside on an entity identified by those three factors. Conversely, the attribute Student Name should not reside on the entity Student Course Grade because it is only identified by one of the attributes within the primary key -- Student ID and not Course ID or the Semester the course was taken.

APPENDIX F. Third Normal Form

The following article provides a good step by step example of taking an non-normalized data structure and putting it in 3rd normal form. The following article from the Database Journal can be accessed using the following link:
http://www.databasejournal.com/sqletc/article.php/26861_1428511_4

Title Database Normalization
By: **Ian Gilfillan**
Date: March 22, 2000

Let's run again through the example we've just done, this time without the data tables to guide us. After all, when you're designing a system, you usually won't have test data available at this stage. The tables were there to show you the consequences of storing data in unnormalized tables, but without them we can focus on dependency issues, which is the key to database normalization.

In the beginning, the data structure we had was as follows:

Project number
Project name
1-n Employee numbers (1-n indicates that there are many occurrences of this field - it is a repeating group)
1-n Employee names
1-n Rate categories
1-n Hourly rates

So, to begin the normalization process, we start by moving from zero normal form to 1st normal form.

The definition of 1st normal form
there are no repeating groups
all the key attributes are defined
all attributes are dependent on the primary key

So far, we have no keys, and there are repeating groups. So we remove the repeating groups, and define the primary key, and are left with the following:

Employee project table

Project number - primary key
Project name
Employee number - primary key
Employee name
Rate category
Hourly rate

This table is in 1st normal form.

A table is in 2nd normal form if
it's in 1st normal form
it includes no partial dependencies (where an attribute is dependent on only a part of a primary key).

So, we go through all the fields. Project name is only dependent on Project number. Employee name, Rate category and Hourly rate are dependent only on Employee number. So we remove them, and place these fields in a separate table, with the key being that part of the original key they are dependent on. So, we are left with the following 3 tables:

Employee project table

Project number - primary key
Employee number - primary key

Employee table

Employee number - primary key
Employee name
Rate category
Hourly rate

Project table

Project number - primary key
Project name

The table is now in 2nd normal form. Is it in 3rd normal form?

The definition of 3rd normal form
It's in 2nd normal form
It contains no transitive dependencies (where a non-key attribute is dependent on another non-key attribute).

We can narrow our search down to the Employee table, which is the only one with more than one non-key attribute. Employee name is not dependent on either Rate category or Hourly rate, the same applies to Rate category, but Hourly rate is dependent on Rate category. So, as before, we remove it, placing it in it's own table, with the attribute it was dependent on as key, as follows:

Employee project table

Project number - primary key
Employee number - primary key

Employee table

Employee number - primary key
Employee name
Rate Category

Rate table

Rate category - primary key
Hourly rate

Project table

Project number - primary key
Project name

These tables are all now in 3rd normal form, and ready to be implemented. There are other normal forms - Boyce-Codd normal form, and 4th normal form, but these are very rarely used for business applications. In most cases, tables in 3rd normal form are already in these normal forms anyway.

APPENDIX G. Data Quality Measures Documentation Template

This appendix includes a template from HUD's Total Information Quality Management (TIQM) Handbook to document data quality measures for entity types and attributes.

Business Concept	Name of the business object to be defined.
Status	[Status] as of x/x/02 [the date of the last revision]
Definition	A succinct, complete business description using common terms or previously defined terms – if new terms are needed, add them to the Business Concept Control Chart and reference in the next section.
Related Business Concepts	• A short-cut or hot-link to a previously defined term. • A description or annotation of an undefined term.
Data Integrity Rules	• A bulleted list of data integrity rules (things that "can" or "must" be present for accuracy, completeness).
Unique Identifier	Free form text, bullets, or whatever can be used to provide insights on the appropriate way to uniquely identify each instance of the data entities associated with the business concept.
Life Cycle	Free form text, bullets, or whatever can be used to provide insights on the states and transitions associated with the business concept.
Classification	Free form text, bullets, or whatever can be used to provide insights on the intrinsic classification necessary to understand the business concept.
Domain	• A bulleted list of all applicable domain values. It must be all-inclusive, however if it is not known at the time, a sample can be used while the definition is in process; indicate which approach is used. • Can be a diagram.
Special Usage	Free form text, bullets, or whatever can be used to highlight real-life cases of the use of the business concept.
Examples	Free form text, bullets, or whatever can be used to provide real-life or mock-up illustrations of the business concept
Issues & Concerns	Proposed definition concerns: • A bulleted list of issues or concerns associated with the definition proposed in this document. Existing definition concerns: • A bulleted list of issues or concerns associated with the existing definition, or definitions.
Background Documentation	Free form text and bullets as needed. For specific references to existing document, use the name and date of the document.

APPENDIX H. List of Acronyms

Acronym	Definition
BPD	Business Process Diagram
BPM	Business Process Model
BPMN	Business Process Modeling Notation
BRM	Business Reference Model
CRUD	Create Read Update Delete
DCB	Data Control Board
DDL	Data Definition Language
DoDAF	Department of Defense Architecture Framework
DRM	Data Reference Model
EA	Enterprise Architecture
EAMS	Enterprise Architecture Management System
EDMG	Enterprise Data Management Group
FEA	Federal Enterprise Architecture
FIPS PUB	Federal Information Processing Standard Publication
HUD	U.S. Department of Housing and Urban Development
IDEF	Integration Definition (for Information Modeling)
IE	Information Exchange
IPT	Integrated Project Team
IT	Information Technology
ITIM	Information Technology Investment Management
LDM	Logical Data Model
LOB	Line of Business
NIST	National Institute of Standards and Technology
OCIO	Office of the Chief Information Officer
OMB	Office of Management and Budget
OV-2	Operational Node Connectivity Diagram
SDADM	Segment Data Architecture Development Methodology
SDM	Software Development Methodology
SRM	Service Component Reference Model
TADM	Target Area Data Model
TIBEC	Technology Investment Board Executive Committee
TIBWG	Technology Investment Board Working Group
XML	Extensible Markup Language