

1. Dimensional Data Design - Data Mart Life Cycle

1.1. Introduction

A *data mart* is a persistent physical store of operational and aggregated data statistically processed data that supports businesspeople in making decisions based primarily on analyses of past activities and results. A data mart contains a predefined subset of enterprise data organized for rapid analysis and reporting.

This section provides the life-cycle steps for developing and enhancing data marts.

Additional standards that may be relevant to data mart projects include the following:

- Data Administration (DA) standards, which are available on the CMS Internet;
- Data Base Administration (DBA) Standards, which are available on the on the CMS Internet.
- Systems Development Standards. CMS's system development standards are outlined in the *CMS Integrated IT Investment & System Life Cycle Framework*, which is available on the CMS Intranet.

The [Dimensional Data Design- Data Mart Life Cycle diagram](#) depicts the milestones, control points, and deliverables as they occur during the following steps:

Dimensional Data Design Processes:

[Initiate Data Administration Services](#)

[Data Mart Development](#)

[Data Mart Enhancement](#)

Supplemental Information:

[Appendix: Dimensional Data Modeling Standards](#)

[Appendix: Related DA Standards, Forms, Tools](#)

Key Deliverable(s):

- *Dimensional Logical Data Model*
- *Dimensional Physical Data Model*
- *Data Definition Language*
- *Production Data Mart*

1.2. Initiate Data Administration Services

Introduction:

It is important to request DA support early in the Requirements phase of a data mart project. Early involvement enables the Project Data Analyst to gain a thorough understanding of project needs and to plan the data mart appropriately. Use the following procedure to initiate a data services request.

- [Submit a Data Services Request](#)

Deliverable(s):

- *A Data Services Request*

1.2.1 Submit a Data Services Request

- | | |
|------------------------------|--|
| <i>Project Manager</i> | A. Early in the Requirements phase of the project, the Project Manager/GTL makes a request for data support and services. To provide background for this request, the Project Manager completes the required sections of the <i>Data Services Request</i> and submits the form according to its instructions. The Data Service Request procedure and form is available from the main Data Administration web page. |
| <i>Data Services Manager</i> | B. Evaluate the data services request. If it is appropriate and feasible to provide DA support, assign a qualified Project Data Analyst to the project, appropriately to the level of anticipated complexity. |
| <i>Project Data Analyst</i> | C. Meet with the project team. Team members introduce themselves and explain their roles. The Project Data Analyst provides an overview of DA standards and the data mart development life cycle. |

From this point on, the Project Data Analyst is a member of the project team and is included in team meetings and communications.

1.3. Data Mart Development

Introduction:

The objective of data mart design is easy access to relevant information. The *star schema* is the design method of choice at CMS, as it enables a relational database to emulate the analytical functionality of a multidimensional database.

The following processes describe the prescribed activities and standards for development of data marts at CMS, using the *star schema* approach.

- [Define Data Mart Requirements](#)
- [Create Logical View of Dimensional Model](#)
- [Approve Logical View of Dimensional Model](#)
- [Develop Physical View of Model](#)
- [Approve Physical View of Model](#)
- [Create and Test Development Data Mart](#)
- [Perform Acceptance Testing](#)
- [Implement in Production](#)

Deliverable(s):

- *Data Mart Requirements*
- *Approved Dimensional Logical Data Model*
- *Approved Dimensional Physical Data Model*
- *Data Definition Language*
- *Production Data Mart*

1.3.1 Define Data Mart Requirements

*Business Analyst
Project Data Analyst*

A. Identify the business requirements for the new data mart. The requirements specification should include, but is not limited to, the following information:

- Categories of users and the likely business questions to be asked by each group;
- Geographic location(s) of users;
- Data required to answer user questions;
- Source(s) for the data;
- Likely volumes and access patterns;
- Architectural concept views of the data mart as it may relate to enterprise business, data, applications and technology architectures;
- Security requirements.

Project Data Analyst

B. Work with the Business Analyst to identify and provide models and metadata for the data source(s) to be used by the data mart. An Enterprise Data Model (EDM) extract including the source entities and attributes may be provided, as well as individual logical and physical models of source databases.

C. (Future) The enterprise Conformed Dimensions and Conformed Facts will be examined for suitability, and any required changes or additions will be submitted to Central DA and the Enterprise Architecture team.

*Business Analyst
Project Data Analyst*

D. Conduct a review of documented data requirements. The requirements review provides an opportunity for the Business Analyst to confirm that the requirements specification is complete and accurate. If a review is held, members of the project team may attend.

*Project Manager,
Business Owner/Sponsor*

E. Sign-off on documented data requirements.

1.3.2 Create Logical View of Dimensional Model

Data Administrator
Project Data Analyst

- A. Provide source models and metadata for enterprise Conformed Dimensions and Conformed Facts. ("Conformed" dimension attributes and facts are those designed for standardized use by data marts across the agency.)

Data Administration provides source models and associated metadata for these data structures.

Project Data Analyst

- B. Work with the Business Analyst and subject matter experts to develop the logical view of the dimensional model, based on the approved documented data requirements. See the [Dimensional Model Standards](#) for a complete description of this deliverable.

The model must also be compliant in the naming of [Entities](#) and [Attributes](#), including use of standard business terms. The Standard and Abbreviations Terms List is available from the Standards Terms page (accessible from the Data Administration home web page). If a needed term is not on the list, follow the procedure outlined on the Standard Terms page.

All CMS data models must be documented in the CMS standard data modeling tool. For more information about use of the standard modeling tool, see the *DA Standard Tools* topic, which is accessible from the Data Administration home web page.

1.3.3 Approve Logical View of Dimensional Model

Project Manager

- A. Every dimensional data mart project must conduct a review of the logical view of the data model. This review should be completed prior to the development of physical names.

If the model has been previously approved or enhancements are being made to an existing approved dimensional model, only the changes to that model are reviewed. If the existing model was never reviewed, or if a new model was created, the entire model must be reviewed.

Review Purpose

The purpose of the model review is to: verify that the dimensional model meets user requirements; confirm that

preferred data sources have been identified; validate the technical quality of the model; and ensure that existing data structures and metadata are being reused appropriately.

Review Timing

It is advisable for the Project Data Analyst to hold interim reviews during the development of the dimensional model. Interim reviews provide the opportunity to familiarize the Business Analyst and development team with the model and obtain feedback on business and technical modeling issues. If an interim review is held, a representative from Data Administration should participate.

The final review of the dimensional model takes place when the Project Data Analyst believes that the model meets all known business requirements, meets DA standards, and includes all required metadata.

Review Package Contents

Provide the necessary information to guide participants through the review process. The review package must include the following:

- A soft copy of the of the data model. Distribute this via email.
- A separate printed diagram for each fact table and its associated dimension tables. (Creating a separate subject area in the model for each fact table and its associated dimension tables will enable printing of separate diagrams.)
- An entity/definition report, including all entities (tables).
- An entity/attribute/attribute-definition report, including all attributes (columns).
- A report including valid values and value descriptions for any code sets documented for this data mart. See the [Dimensional Model Standards](#) for information on code sets (domains) that must be documented.
- Business requirements documentation. Include an overview of the business requirements. Include user views (reports or screen designs) if available. Reviewers may request additional documentation e.g., detailed business requirements and, notes from interviews with users.

- Project Data Analyst* B. Schedule reviews of the dimensional model to include three components: the review kick-off meeting, the syntax or technical review, and the review against business requirements. At the end of the review, reviewers are expected to approve the model by email message.
- Project Manager, Project Data Analyst* C. Arrange and conduct review kick-off meeting; prepare and distribute review package.
- Project Manager, Project Data Analyst, Project Database Administrator, Data Administrator* D. Participate in kick-off meeting.
- Project Data Analyst* E. Conduct the review session(s) and performs the tasks below.
- Distribute the review package.
 - Provide information on project status and critical dates.
 - Walk through model at a high level (informational), pointing out the major fact and dimension tables.
 - Establish responsibilities, events, and dates for the remainder of the review.
- Business Analyst* F. Resolve issues requiring business needs.
- Project Data Analyst* G. Review Model against business requirements. Structure and conduct walkthrough sessions. Explain the way in which the model addresses the business requirements. Document and problems with solutions or alternatives.
- Business Analyst, Project Data Analyst* H. Confirm that the model meets business requirements and communicate the outcome – whether it is concurrence or disagreement. Point out any inconsistencies between model and business requirements. Raise other issues as applicable.
- Project Database Administrator* I. Provide guidance for the anticipated data mart DBMS deployment.

- Project Data Analyst* J. Conduct the actual review session and perform the tasks below.
- Walk through the model in an orderly manner, pointing out the major fact and dimension tables.
 - Identify data sources and explains the major processes used to derive data from these sources.
 - Explain expected usage and navigation patterns and the way in which the model will support these.
 - Explain any unusual features of the model. These must be documented in a cover sheet, as stated in the [Dimensional Model Standards](#).

Project Manager, Business Owner / Sponsor, Data Administrator K. Approve model after all required changes have been made

Project Data Analyst L. Coordinate the transmission of approvals by email message. When all reviewers are ready to sign off, Data Administration sends them an email message asking for confirmation of their approval by return email. The Project Data Analyst forwards these approvals to Data Administration; the project team and all reviewers are to receive a copy of the approval messages.

Project Data Analyst M. Package the dimensional model with approval messages. Pass the approved model package to Data Administration, where it is saved in the DA model library.

1.3.4 Develop Physical View of Dimensional Model

Project Data Analyst A. This phase includes development of the first-cut physical view and final physical view of the model.

Generate a first-cut physical view in the standard modeling tool. Participate in end-of-phase review.

Data Administrator, Database Administration B. Assist in generating first-cut physical view as needed. Review/approve physical names. Participate in end-of-phase review.

Project Data Analyst C. With the assistance of Data Administration, generate the first-cut physical view in the standard modeling tool. The first-cut physical view includes physical names for the logical elements. Detailed standards and procedures for generating physical names may be found in [Assign a Column or Element Name](#).

Note: All elements in the physical model, including physical-only elements, must have meaningful definitions. If a single file with logical and physical views is used for the logical and physical models, all definitions will be available in both models automatically. If separate files are used, the modeler must ensure that the definitions are brought over into the physical model.

1.3.5 Approve Physical View of Dimensional Model

*Project Data Analyst,
Data Administrator* A. Review the physical names with Data Administration. Data Administration approves the names or recommends changes and, when appropriate, a follow-up review.

Data Administration has no role in the development of the physical view after this point, except to assist in developing physical names for any additional attributes. For information on updates to logical and physical names, see [Assign a Column or Element Name](#).

Data Administration participates in the end-of-phase review to ensure that any new elements comply with naming standards, and that meaningful definitions have been provided for all elements.

Data Administrator B. When the physical names have been approved, pass a copy of the model and any associated documentation to the Project Database Administrator. See the Database Administration web page for more information on DBA standards and procedures for the target DBMS environment.

*Project Data Analyst,
Project Manager* C. Obtain the dimensional model with sign-offs indicating approval of the logical view by the Business Owner / Sponsor, Data Administrator, and Project Manager. Database Administration approval may also be required.

Forward the approved model and approval messages to Data Administration.

Data Administrator D. After the end-of-phase review, the approved dimensional model is saved in the DA model library.

1.3.6 Create and Test Development Data Mart

- Database Administrator* A. Use the approved physical data model and generate DBMS Data Definition Language (DDL) for the data mart.
- Database Administrator* B. Deploy the Dimensional Data Mart to the development DBMS environment.
Notify the Project Manager of the test environment availability.

1.3.7 Perform Acceptance Testing

- Project Manager* A. Notify the Business Owner / Sponsor and business users (or designated testers) of the available data mart test environment.
- Business Users,
Business Analyst* B. Test the data mart in the Validation/Acceptance Test environment and verify that it meets business needs.
- Project Database
Administrator* C. Support the data mart test environment as needed.
- Project Data Analyst,
Project Database
Administrator* D. Participate in end-of-phase review. If there are changes to the data mart requirements, enter modifications in the appropriate dimensional model -- logical or physical data model. Changes to models must be resubmitted to the approval processes at the highest level of change. For example, if the logical model is changed, restart the process at [Approve Logical View of Dimensional Model](#); if the physical model is changed, restart the process at [Approve Physical View of Dimensional Model](#). To expedite the process, limit the review and approvals to the changed areas.

1.3.8 Implement in Production

Database Administrator

A. Prepare production operational facilities and components and deploy the data mart to the production DBMS environment.

Database Administrator

B. Monitor and support the data mart in the production DBMS environment.

1.4. Data Mart Enhancement

Introduction

All enhancements to a dimensional data mart, whether major or minor, must be incorporated in the dimensional model for the data mart and comply with the [Dimensional Modeling Standard](#). All changes must be reported to Data Administration (DA) and Database Administration (DBA). In addition, major enhancements -- such as those addressing new high-level business requirements, supporting new business processes, and/or requiring additional tables--must follow the life cycle steps described in this document.

Enhancements to data marts begin with a baseline of the existing operating data mart. The processes to meet new data mart requirements are:

- [Enhance Data Mart Requirements](#)
- [Update Existing Logical View of Dimensional Model](#)

Note: the remaining processes consists of steps that are the same as those performed for development of a Data Mart.

- [Approve Logical View of Dimensional Model](#)
- [Develop Physical View of Model](#)
- [Approve Physical View of Model](#)
- [Create and Test Development Data Mart](#)
- [Perform Acceptance Testing](#)
- [Implement in Production](#)

Deliverable(s):

- *New Data Mart Requirements*
- *Approved Dimensional Logical Data Model*
- *Approved Dimensional Physical Data Model*
- *Data Definition Language*
- *Production Data Mart*

1.4.1 Enhance Data Mart Requirements

- Data Administrator* A. Provide extracts from the Enterprise Data Model (EDM) and/or provides models and associated metadata from the source databases that will feed the data mart.

- Business Analyst* B. Identify the new business requirements for the data mart.
Project Data Analyst The requirements specification should include, but is not limited to, the following information:
- Any new categories of users and the likely business questions to be asked by each
 - Geographic location(s) of new users
 - Additional business needs of existing users
 - Data required to address new business needs
 - Source(s) for the data
 - Any changes to volumes and access patterns
 - Any changes to security requirements
- Identify new business requirements, data to support requirements, and sources for data.
- Project Data Analyst* C. Work with the Business Analyst to identify and provide models and metadata for the data source(s) to be used by the data mart. An Enterprise Data Model (EDM) extract including the source entities and attributes may be provided, as well as individual logical and physical models of source databases.
- Business Analyst* D. Conduct a review of documented data requirements. The
Project Data Analyst requirements review provides an opportunity to confirm that the requirements specification is complete and accurate. If a review is held, any member of the project team may attend.

1.4.2 Update Existing Logical View of Dimensional Model

- Data Administrator* A. If the approved dimensional model for the existing data mart is stored in the DA model library, provide a copy for the project team.
- Provide source models and metadata for enterprise Conformed Dimensions and Conformed Facts. ("Conformed" dimension attributes and facts are those designed for standardized use by data marts across the agency.)

Project Data Analyst B. The dimensional model documents both the business metadata (business names and definitions) and technical metadata (the physical *star schema* structure) for the data mart. The approved model for the existing data mart should be available from Data Administration. This model must be reviewed and if necessary updated to ensure that it accurately represents the existing data mart.

Develop the logical view of the dimensional model. See the [Dimensional Data Modeling Standard](#) for a complete description of this deliverable.

Work with the Business Analyst develop and develop the logical view of the dimensional model in the CMS standard modeling tool.

Business Analyst C. Provide additional information on business requirements as needed.

Project Data Analyst D. Review the existing dimensional model and if necessary update it. Ensure that the model accurately reflects the existing data mart.

- Enhance the logical view of the dimensional model in CMS's standard modeling tool.
- If any new "conformed" dimension tables are included in the model, Data Administration provides source models and associated metadata for these tables, if available. ("Conformed" dimension tables are those designed for use by multiple data marts.)

Project Data Analyst E. If the approved dimensional model for the existing data mart is stored in the DA model library, Data Administration provides a copy for the project team. Note: If the dimensional model for the existing data mart is not in the DA model library, the project team is responsible for obtaining the existing model or creating a new model. All models must comply with the [Dimensional Data Modeling Standard](#).

Update the dimensional model as needed to ensure that it is synchronized with the existing operating data mart. This up-to-date baseline dimensional model then becomes the starting point for modeling the enhancements.

Project Data Analyst F. Follow the remaining approval and testing processes necessary for data mart implementation. See [Data Mart Enhancement](#).

1.5. Dimensional Data Modeling Standards

[Dimensional Model Definition](#)

[Diagramming and Documentation Conventions](#)

[Naming Conventions](#)

[Relationships](#)

[Required Logical Metadata](#)

[Best Practices](#)

1.5.1 Dimensional Model Definition

A dimensional model is a graphical representation of information structures that allows business information to be viewed from many perspectives. Dimensional data design is applicable in an analytical processing or decision support environment. Dimensional modeling captures metadata similar to that in an Entity-Relationship (E-R) model but uses a different structure--i.e., the star schema. The primary components of the star schema are fact tables and dimension tables--the fact table is at the center of the star, and the dimension tables form the points. At CMS, dimensional modeling is used for data mart design.

Dimensional models must comply with standards in the following areas:

- Diagramming conventions
- Naming conventions
- Relationships
- Required logical metadata
- Best practices (these are not standards, but compliance is expected unless the model developer can show that specific circumstances justify an exception).

Note: The logical view of the dimensional model may not be in third normal form, but should be similar in structure to the physical view.

1.5.2 Diagramming and Documentation Conventions

- Generate the final diagram using the standard modeling tool.
- In the Model Properties dialogue, select the Dimensional feature within the Enable Modeling Features area of the dialogue.
- If capturing transformation logic in the standard data modeling tool, in the Model Properties dialogue, select the Dimensional and Data Movement feature within the Enable Modeling Features area of the dialogue..
- Select Dimensional Modeling as the Physical Notation, and Select IDEF1X as the Logical Notation.
- Include the name of the model (or project), the name of the project contact, and the last change date in a label within the graphic.

- Include all dimension and fact tables in a single data model, but create a separate subject area for each fact table and its associated dimension tables. This will enable the printing of a separate graphic for each subject area when the model is reviewed.
- Visual enhancements to the model (color, special fonts, etc.) are optional. If visual enhancements are used, provide a legend within the graphic explaining the significance of each visual feature.
- Separate physical and logical models (as distinct from physical and logical views within a single model) are not required or recommended for dimensional models.
- In addition to the documentation produced from the data model, a cover document is required. This document should describe the general features of the model and any unusual characteristics. The following should be included:

- Grain statements for all fact tables and dimension tables.

The grain statement for a fact table defines the lowest level of detail needed to support the business analytical requirements. E.g., "The grain of the Annual Hospital Bills by State Fact Table is the year in which the medical service was rendered, the State of the patient's residence, and the hospital identifier."

The grain statement for a dimension table identifies the natural key of the table (the natural key may be single-part or concatenated). E.g., "The grain of the Provider dimension table is provider number"; "The grain of the Beneficiary Demographics dimension table is age, ethnicity, and gender."

- Method of handling any changing dimensions.
- Handling of semi-additive or non-additive facts (facts which usually cannot be summarized--e.g., patient counts across multiple health care facilities cannot be summarized to produce a count of distinct Medicare beneficiaries, and temperature readings taken at various locations in a building would typically be averaged rather than summarized).
- Explanations of any dimensions with multiple hierarchies (e.g., a time dimension with fiscal year and calendar year hierarchies).
- Explanations of any degenerate dimensions. (Degenerate dimensions usually occur in line-item-oriented fact table designs. For example, if an invoice number is captured in the fact table, but all the other fact table items are at the level of the invoice line item, the invoice number might be considered the "key" of a "degenerate"--i.e., missing--invoice dimension.)
- Explanations of any "factless fact tables." (A "factless fact table" may be used to count occurrences of an event where no information is needed except the count itself. It may also be used to record events that did not take place--e.g., no sales occurred for a given product.)

1.5.3 Naming Conventions

Business Names: General Rules

- If a data object in the dimensional model corresponds exactly to one in the source system or in another dimensional model approved by DA, use the name from the source system or other model. **Exception:** If the object exists in the EDM, and the EDM name is different from the source or other model name, the EDM name is preferred. Consult DA if there is an overriding reason for not using the EDM name in this situation.
- Spell out each word in any name that is added; do not use abbreviations. Use of acronyms in names is limited to those acronyms on the Standard Terms and Abbreviations list.
- Make nouns singular, except in cases where the plural is the only form commonly used--e.g., "savings," referring to an amount saved.
- Use only the alphabetic characters A-Z and the space character. Do not use punctuation marks or special characters, including the slash (/) or the hyphen (-). Numbers are not allowed.
- Do not use possessive nouns--e.g., use "Beneficiary Health Assessment" rather than "Beneficiary's Health Assessment."
- Use a maximum of 120 characters, including spaces.

Business Names: Fact and Dimension Tables (Entities)

Note: Although most discussions of dimensional modeling refer to fact and dimension "tables" **and make no distinction between physical and logical terminology, ERwin uses the term "entity"** in the logical view.

- The name must be a noun (e.g., "Beneficiary") or noun phrase (e.g., "Beneficiary Screening").
- If DA has provided "conformed" dimension tables, use the existing business or logical names for these tables.
- Use "Fact" as the last word in a fact table name (but do not use "Dimension" as the last word in a dimension table name).
- Create fact table names that identify the business process or entity on which information is being collected--e.g., "Medical Insurance Enrollment Fact," identifying a process, or "Hospice Usage Fact," identifying an entity (in this case a concept).
- Include a modifier in each fact table name that indicates the grain of the table--e.g., "Claim Summary Month Fact."

Business Names: Elements (Attributes)

- Each element or attribute name must include a representation term (class word) and an object class term (prime word), optionally, a property and qualifier term (modifier). The representation term (class word) must be the last word in the name. For more information on naming elements, see the [Attribute](#) standard.
- Make each element or attribute name unique within the data model. This is a general requirement, but there may be exception situations. If a waiver is justified, see the DA web page for the Standards Waiver Request Form and instructions for its submission.
- If a "conformed" fact is being reused, and the existing level of granularity is matched, use the existing name.
- If aggregate or summary data from a source system is used, the data name must indicate a total or summary--e.g., "Travel Total Amount."
- For an element or attribute that captures a quantity, include the unit of measure in the element name--e.g., Claim Blood Deductible Pint Quantity.
- If the element or attribute captures a count of unique or non-duplicated values (e.g., a count of unique beneficiaries in a set of Medicare claims), the element name must include the word "unique" (e.g., "Beneficiary Unique Count").
- **Note:** Consult DA's list of standard business names, acronyms, and abbreviations while developing business names. The *Standard Name and Abbreviation List* will list approved terms, abbreviations, and acronyms. The Standard Terms and Abbreviations List is available from the Standards Terms page (accessible from the Data Administration home web page).

If a needed term is not on the approved list, follow the procedure to request on the Standard Terms page to request a new term, which is accessible from the Data Administration web page.

Physical Names

Business or logical names must be converted to physical names using a standard method and standard abbreviations, and physical names must be approved by Data Administration. Detailed standards and procedures for the generation of physical names may be found in topic: [Assign a Column or Element Name](#).

1.5.4 Relationships

If a fact table has multiple relationships to a single dimension table, then each relationship must be assigned a unique rolename. Other relationships need not be named.

1.5.5 Required Logical Metadata

Table and Element Descriptions

- The description must be a noun phrase. If necessary, add one or more sentences to complete the description. E.g., the element "Provider Payment Scheduled Date" might be defined as "The

scheduled date for payment to the provider" (noun phrase). "This date is considered to be the date of payment, since no information on the actual payment date is available" (additional sentence).

- The description must be clear, precise, and unambiguous. It must identify the data object and distinguish it from all other objects. Add examples or exclusions as needed to improve clarity.
- If the element is in a time dimension, specify how the unit of time is being measured. E.g., does "year" refer to calendar year or fiscal year; is "month" equivalent to calendar month, etc.
- Describe each data object in terms of business rules, not in terms of information processing rules or physical considerations.
- If a dimension table or element in the model corresponds exactly to one in the source system or in another dimensional model approved by DA, use the description from the source system or other model. **Exception:** If the item exists in the EDM, and the EDM description is different from the description on the source system or other model, the EDM description is preferred. Consult DA if there is an overriding reason for using the source system description in this situation.
- If a description from the EDM or a source system model is being reused, it is not necessary to copy the source description into the project model. Instead, reference the source using the following format:

REFER TO: xxx, where xxx is the model name, exactly as in the source model.

Note: Cross referencing is permitted only if the name of the data object is exactly the same as the name in the source model.

Elements: Additional Metadata Requirements

- **Domain.** Specify the valid values or range of allowable choices.
 - If the domain is a set of code values, list the valid values and provide a description for each value. This information can be entered in the standard modeling tool (in the physical view of the project model), documented in a spreadsheet, or provided in another appropriate format. **Exception:** If the number of codes is large and is documented elsewhere, provide a reference to the source document location. Enter this information at the end of the element description. Begin a new paragraph and use the following format:

VALID VALUE SOURCE: xxx, where xxx is free-form text.
 - If the domain is a range of values (e.g., numeric, 1-33), document the range in the physical view of the project model. Only a single range can be entered in the modeling tool. If there are multiple value ranges, document them in an appropriate format.
- **Format.** Identify the element data type, such as numeric, character, date, etc. If non-integer values are needed, state the precision (the number of decimal places to be accommodated). If very large numbers are expected, over a billion, state the largest value expected. This will help the DBA select the correct data type.

- **Security Classification.** If this element captures sensitive data, specify the degree of sensitivity. Enter this information at the end of the element definition. Begin a new paragraph and use the following format:

SECURITY CLASSIFICATION: xxx, where xxx is free-form text.

- **Alternate Business Names.** If alternate names are in use for the same element within the business community, enter the alternate names in Erwin's User Defined Properties. Start a new paragraph and use the following format:

ALTERNATE BUSINESS NAME(S): xxx, where xxx is free-form text.

- **Source System.** If the source for the element data is a CMS system or database, enter the acronym for the source system at the end of the element definition. Start a new paragraph and use the following format:

SOURCE SYSTEM: xxx, where xxx is the system acronym.

1.5.6 Best Practices

Fact Tables

- For every business process there should be at least one fact table.
- Each item in a fact table should have a default aggregation (or derivation) rule--e.g., sum, min, max, semi-additive, not additive. Any complexities in the aggregation method must be documented.

Enter the aggregation/derivation information at the end of the definition. Begin a new paragraph and use the following format:

DERIVATION RULES: xxx, where xxx is free-form text.

- The grain, or granularity, of the fact table should be at the lowest level for which a need has been identified and a requirement approved. Performance and storage constraints must also be considered.
- The grain of all items in the fact table should be the same. If there is a need for aggregation at more than one level, a separate fact table for each level of aggregation may be needed.

Note: Aggregation tables are either (a) transparent to the user, such that all SQL is written to go against the lowest level of granularity, or (b) explicit--i.e., seen and queried by the user. Since the dimensional model is, among other things, a tool for communication with the user, it is preferable to include only those tables that the user will see.

Dimension Tables

- Each dimension table has one and only one lowest level element, called the dimension grain.

- Dimension tables that are referenced or are likely to be referenced by multiple fact tables are "conformed dimensions." If conformed dimensions already exist for any of the dimensions in the model, their reuse is expected. If new dimensions with potential for usage across the agency are being developed, the design must support anticipated cross-agency needs.
- Each non-key element should appear in only one dimension table.
- Most models should have at least one period or time dimension. There may be more than one period dimension. Date and time may be split into two separate dimensions, especially if time is being captured at the hour or minute level.
- If a dimension table includes a code, in most cases the code description should be included. For example, if branch locations are identified by a branch code, and each code represents a branch name, both the code and the name should be included. An alternative is to include the description and omit the code--e.g., State = California, Status = Active.
- Generally, there should be no more than twenty dimension tables per fact table; the designer should provide justifications if more than twenty dimension tables are required.

Keys

- The primary key of a dimension table should be a surrogate key. A source system production key should not be used as a primary key.
- The primary keys of the dimension tables should be included in the fact table as foreign keys. Together these (and only these) foreign keys make up the fact table primary key (in the logical view).

1.6. Appendix: Summary of Related DA Standards, Forms, and Tools

Standards	Forms	Tools
<ul style="list-style-type: none"> ▪ Dimensional Model Standards ▪ Entities ▪ Attributes ▪ Assign a Column or Element Name. 	<ul style="list-style-type: none"> ▪ Data Services Request Form ▪ Data Standard Waiver Request Form <p>Access forms from the main Data Administration web page.</p>	<ul style="list-style-type: none"> ▪ DA Standard Tools <p>Access information about DA tools from the main Data Administration web page.</p>