Subject	Type	Standard
Data Dictionary	Identifier	S–3351
	Effective Date	July 2002
	Revision No.	1

Approval _____

CISSCO Program Director

A. PURPOSE

This standard specifies the content and format requirements for a Data Dictionary.

B. APPLICABILITY

This standard applies to any Data Dictionary produced by a project subject to the SDLCM Methodology. A project produces a Data Dictionary as part of designing and implementing the database(s) used by the system under development.

This standard is used by Development team members responsible for data modeling and database design to produce the Data Dictionary. It is also used by the Technical Project Manager and Quality Assurance personnel for reviewing the project's Data Dictionary.

The Data Dictionary is a living work product of detailed data and software design and implementation activities and is maintained for the life of the project. The Data Dictionary is made available to all project personnel, preferably in electronic form.

C. REFERENCE PUBLICATIONS

The following publications contain related information:

- *SDLCM Methodology Handbook*, Component 3
- *SDLCM Methodology Handbook*, Component 4
- *Systems Development CASE Tool Guidelines*, Systems Development and Integration Branch (SDIB) Office of Information Resources Management (OIRM), September 12, 1995
- *Standards and Conventions*, SDIB OIRM, August 28, 1995
- SDLCM Methodology Standard S–3172, Physical Design Document
- SDLCM Methodology Standard S–3151, Data Models
- SDLCM Methodology Procedure P–3101, Data Modeling

D. STANDARD

Refer to Sections 4.3 and 4.4 of the *Systems Development CASE Tool Guidelines* and to SDLCM Methodology Procedure P–3101, Data Modeling, for descriptions the processes used to develop and evaluate the Data Dictionary.

Refer to Sections 3.1 and 4.1 of the *Standards and Conventions* document for the conventions used to name each type of data object.

D.1 Data Dictionary Definition

A data dictionary is a mechanism for defining the data elements identified in structured analysis and structured design products (DFDs, functional specifications, or the dictionary itself). It is a place for users, designers, programmers, and testers to determine what constitutes data flows, data stores, and structure chart data couples; to look up unfamiliar terms; and to review data requirements.

To provide current and accurate information during development, establish one data dictionary on the project that is accessible to all personnel. Establish a means of controlling changes such that all users are aware of them and that the same entry cannot be changed simultaneously by several people.

Because of the amount of work necessary to maintain a large data dictionary, the use of automated tools is strongly recommended.

D.2 Data Dictionary Contents

Define the following items in a data dictionary:

- Each unique data flow or unique data couple and component included in a data flow
- Each unique data store or unique file and component included in a data store
- Anything else in the data flow diagrams (DFDs) or software architecture diagrams that needs a definition (for example, terms, abbreviations, or acronyms that are not part of a typical user's vocabulary)

Each entry in a data dictionary contains the name of the item being defined, all aliases by which it is known, its definition, and any notes or comments to further explain the item.

D.2.1 ITEM NAME

Include the name of the item as it appears in the DFD, software architecture diagram, or elsewhere in the data dictionary.

D.2.2 ALIASES

Items have aliases when they are known by two or more names. Avoid the use of aliases in the DFDs and software architecture diagrams, but when they do occur, give each its own entry in the data dictionary. List all aliases by which the entry is known. Do not include a redundant definition for each alias. Each alias should point to a single entry that contains the full definition.

D.2.3 DEFINITION

Each entry in the data dictionary must have a definition, which may be textual or composite.

D.2.3.1 Data Elements

Data elements are those items that do not need to be or cannot be further decomposed. Define data elements in the dictionary by their values and their corresponding meanings. To avoid introducing redundancy with functional specifications, do not include information about processing in the definition associated with the data element.

D.2.3.2 Composite Data

Composite data is composed of logical combinations of data elements or smaller divisions of composite data. Each item within the definition must appear as an entry in the data dictionary.

Separate the name from the definition with an equal sign (=), which is the shorthand notation for “is composed of."

Define a composite data element in terms of the following:

- Sequence (concatenation) of data elements: Use a plus sign (+) as the symbol for a logical AND.
- Repeated data elements: Indicate repeated elements or element sets by enclosing them in braces { }. Indicate a set number of iterations by placing numbers before and after the braces 1{...}5. The numbers indicate the lower and upper limits of the iteration.
- Selection from a set of data elements: Indicate a logical OR by a vertical line (|), and an EITHER .. OR by enclosing the choices in square brackets and separating them with the logical OR symbol. Any number of choices may be enclosed in the brackets [...|...|...]. Optional elements or composites are enclosed in parentheses ( ).
- Combinations of the above: The above symbols may be combined in any order to describe the item in the dictionary. An example of a composite data flow might be flow = a + b + [c | d] + 1{ e }5

D.2.4 NOTES

When relevant, apply notes to the data dictionary entry. Items that might appear in the notes field are

- Assumptions
- Response times
- Concerns
- Organization of data stores
- Access requirements
- Conditions that generate the defined control signal

Indicate comments in a data dictionary by enclosing them in asterisks (*).