

Specification for

Information systems products using SSADM (Structured Systems Analysis and Design Method) —

**Part 1: Implementation of SSADM
version 4**

Committees responsible for this British Standard

The preparation of this British Standard was entrusted by the Information Systems Technology Standards Policy Committee (IST/-) to Technical Committee IST/15, upon which the following bodies were represented:

- British Computer Society
- CCTA (the Government Centre for Information Systems)
- Department of Trade and Industry (IT Standards Unit ITD6A)
- Institution of Electrical Engineers
- International Computers Limited
- Lloyd’s Register of Shipping
- Ministry of Defence
- National Computing Centre Ltd.
- Oracle Corporation (UK) Ltd.

The following bodies were also represented in the drafting of the standard, through BSI Panel IST/15/-/200:

- Leeds Metropolitan University
- The International SSADM Users Group
- Information Systems Examination Board
- SSADM Design Authority Board
- Computer Services Association

This British Standard, having been prepared under the direction of the Information Systems Technology Standards Policy Committee, was published under the authority of the Standards Board and comes into effect on 15 June 1994

© BSI 04-2000

The following BSI references relate to the work on this standard:
Committee reference IST/15
Draft for comment 91/64274 DC

ISBN 0 580 22849 5

Amendments issued since publication

Amd. No.	Date	Comments
8939	December 1995	Indicated by a sideline in the margin

Contents

	Page
Committees responsible	Inside front cover
Foreword	iv
Introduction	1
Section 1. General	
1.1 Scope	4
1.2 References	4
1.3 Definitions	4
1.4 Abbreviations	23
Section 2. Conformity	
2.1 Individual products	24
2.2 Marking	24
Section 3. Detailed product specifications	
3.1 Business System Options products	25
3.2 Data Flow Modelling products	30
3.3 Dialogue Design products	46
3.4 Entity-Event Modelling products	52
3.5 Function Definition products	62
3.6 Logical Data Modelling products	68
3.7 Logical Database Process Design products	78
3.8 Processing Specification products	81
3.9 Relational Data Analysis products	84
3.10 Requirements Definition products	85
3.11 Specification Prototyping products	90
3.12 Technical System Options products	93
3.13 Physical Data Design products	96
3.14 Physical Process Specification products	99
3.15 Physical Environment Description products	102
3.16 Miscellaneous products	104
Annex A (normative) Products specified	109
Annex B (informative) Product groupings	110
Annex C (informative) Conventions	126
Annex D (normative) Conformity charts	128
Figure 1 — SSADM Modules and Stages	3
Figure 2 — Presentation conventions for the Context Diagram	31
Figure 3 — Presentation conventions for Data Stores	33
Figure 4 — Presentation conventions for External Entities	34
Figure 5 — Presentation conventions for Processes	35
Figure 6 — Presentation conventions for Data Flows	36
Figure 7 — Presentation conventions for Process Decomposition	37
Figure 8 — Legal sources and destinations of data flows	38
Figure 9 — Presentation conventions for Dialogue Structure	49
Figure 10 — Presentation conventions for Menu Structures	50
Figure 11 — Presentation conventions for Effect Correspondence Diagrams	53
Figure 12 — Presentation conventions for Effect Correspondence Diagrams: Entity Roles and Effect Qualifiers	55
Figure 13 — Presentation conventions for Entity Life Histories	56
Figure 14 — Presentation conventions for Entity Life Histories: Parallel lives	57

	Page
Figure 15 — Presentation conventions for Entity Life Histories: State Indicators	58
Figure 16 — Presentation conventions for Entity Life Histories: Entity Roles	59
Figure 17 — Presentation conventions for Entity Life Histories: Effect Qualifiers	60
Figure 18 — Presentation conventions for Entity Life Histories: Quit and Resume-1	61
Figure 19 — Presentation conventions for Entity Life Histories: Quit and Resume-2	62
Figure 20 — Presentation conventions for Enquiry Access Paths	63
Figure 21 — Presentation conventions for I/O Structure Diagram	67
Figure 22 — Presentation conventions for Logical Data Structures	74
Figure 23 — Presentation conventions for Logical Data Structures: exclusive arcs	75
Figure 24 — Presentation conventions for Logical Data Structures: Recursive relationships	76
Figure 25 — Presentation conventions for Logical Data Structures: partitions	77
Figure 26 — Presentation conventions for Enquiry Process Models	79
Figure 27 — Presentation conventions for Update Process Models	82
Figure 28 — Presentation conventions for the Prototype Pathway	92
Figure 29 — Presentation conventions for the SSADM Structure Diagram	107
Figure 30 — Presentation conventions for the SSADM Structure Diagram: parallel structure	108
Figure B.1 — SSADM Version 4, Feasibility Study Module, Stage 0: Product Grouping	111
Figure B.2 — SSADM Version 4, Requirements Analysis Module, Stage 1: Product Grouping	112
Figure B.3 — SSADM Version 4, Requirements Analysis Module, Stage 2: Product Grouping	113
Figure B.4 — SSADM Version 4, Requirements Specification Module, Stage 3: Product Grouping	114
Figure B.5 — SSADM Version 4, Logical System Specification Module, Stage 4: Product Grouping	115
Figure B.6 — SSADM Version 4, Logical System Specification Module, Stage 5: Product Grouping	116
Figure B.7 — SSADM Version 4, Physical Design Module, Stage 6: Product Grouping	117
Figure B.8 — Application Development Standards	118
Figure B.9 — Data Flow Model	118
Figure B.10 — Data Catalogue	119
Figure B.11 — Logical Data Model	119
Figure B.12 — Function Definitions	120
Figure B.13 — Processing Specification	121
Figure B.14 — Dialogues	122
Figure B.15 — Logical Process Model	123
Figure B.16 — Physical Process Specification	123
Figure B.17 — Physical Environment Classification	124
Figure B.18 — Requirements Specification	124
Figure B.19 — Physical Design	125

	Page
Figure B.20 — I/O Structure (for all functions)	125
Figure B.21 — Logical System Specification	126
Figure B.22 — Physical System Specification	126
Table A.1 — Products specified	109
Table D.1 — Conformity chart for Feasibility Module, Stage 0: Feasibility	129
Table D.2 — Conformity chart for Requirements Analysis Module, Stage 1: Investigation of Current Environment	129
Table D.3 — Conformity chart for Requirements Analysis Module, Stage 2: Business Systems Options	130
Table D.4 — Conformity chart for Requirements Specification Module, Stage 3: Definition of Requirements	130
Table D.5 — Conformity chart for Logical System Specification Module, Stage 4: Technical System Options	131
Table D.6 — Conformity chart for Logical System Specification Module, Stage 5: Logical Design	131
Table D.7 — Conformity chart for Physical Design Module, Stage 6: Physical Design	131
Table D.8 — Conformity chart for Data Catalogue	132
Table D.9 — Conformity chart for Data Flow Model	132
Table D.10 — Conformity chart for Logical Data Model	132
Table D.11 — Conformity chart for Current Services Description	133
Table D.12 — Conformity chart for Requirements Specification	133
Table D.13 — Conformity chart for Processing Specification	133
Table D.14 — Conformity chart for Function Definitions (set)	134
Table D.15 — Conformity chart for Logical Design	134
Table D.16 — Conformity chart for Logical Process Model	134
Table D.17 — Conformity chart for Physical System Specification	134
Table D.18 — Conformity chart for Application Development Standards	135
Table D.19 — Conformity chart for Physical Design	135
Table D.20 — Conformity chart for Physical Process Specification	135
Table D.21 — Conformity chart for Physical Environment Classification	135
Table D.22 — Conformity chart for Dialogues	136
Table D.23 — Conformity chart for I/O Structures	136
List of references	Inside back cover

Foreword

This British Standard has been prepared under the direction of the Information Systems Technology Standards Policy Committee.

NOTE To comply with this specification, the user has to comply with all its relevant requirements. He may depart from recommendations, but this would be on his own responsibility and he would be expected to have good reasons for doing so.

The Structured Systems Analysis and Design Method (SSADM) is a systematic approach to the analysis and design of information systems applications.

SSADM Version 4 is a product-oriented specification of what practitioners and managers need to do to undertake the tasks of systems analysis and design. It describes an effective process of system development, which users can adapt to suit project needs.

Relationship between the standard and the method

Presently under development is:

BS 7738-2, *Implementation of SSADM version 4.2*.

Other parts may be added to BS 7738 as and when other versions of the SSADM manuals are published.

The details of the method itself are contained in the current version of the SSADM manuals (SSADM Version 4 NCC/Blackwell, 1990 [1]). This British Standard is designed to be self-contained. It will not depend upon the SSADM manuals in its role as a standard. It will not, however, be independent of the method (and of the manuals in a wider sense). The standard will be the responsibility of BSI; the method will be the responsibility of the Design Authority Board (DAB). The two bodies understand the importance of maintaining a clearly understood relationship with mutual recognition of purpose by each organization.

Control of the method

The method was developed by the CCTA¹⁾, it is now controlled by the Design Authority Board (DAB). The DAB provides a forum for discussion of the method, its application and evolution. The DAB includes representatives from all major groups with interests in the method: government and private sector, customers and suppliers of SSADM-related products and services and members of relevant professional bodies. The relationship between the standard and the method will be maintained by liaison between the DAB and the BSI Technical Committee; a representative of the Technical Committee being a member of the DAB. The DAB members are solely responsible for the method, its publication in the form of the manuals and supporting documentation, and any changes to the method.

BSI acknowledges that the copyright in SSADM is the property of the Crown. This British Standard is published with the consent of CCTA and the permission of the Controller of HMSO.

Change control

The evolution of the standard is the responsibility of the Technical Panel (IST/15/-/200) of BSI, reporting directly to Technical Committee IST/15 Software Engineering. The standard will be revised and updated to reflect changes in the method as agreed by DAB and as published in the manuals. In addition, the standard itself will be reviewed periodically, and in response to suggestions for revision received from users of the standard.

This edition of this standard makes specific reference to the 1990 version of the manuals for SSADM. Future releases of the standard will similarly make explicit reference to the prevailing edition. If the method changes, as reflected in the manuals, the Technical Panel will consider the changes and amend the standard accordingly. The minimum change would be an alteration to the normative reference.

¹⁾ Formerly the Central Computer and Telecommunications Agency of HM Treasury, now CCTA — The Government Centre for Information Systems.

Production certification. Users of this British Standard are advised to consider the desirability of third party certification of product conformity with this British Standard based on testing and continuing product surveillance which may be coupled with assessment of a supplier's quality systems against the appropriate Part of BS 5750.

Enquiries as to the availability of third party certification schemes are forwarded by BSI to the Association of Certification Bodies. If a third party certification scheme does not already exist, users should consider approaching an appropriate body from the list of Association members.

Copyright is waived by BSI in respect of Table D.1 to Table D.23, Figure 1 to Figure 30, and Figure B.1 to Figure B.22.

A British Standard does not purport to include all the necessary provisions of a contract. Users of British Standards are responsible for their correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, pages i to vi, pages 1 to 136, an inside back cover and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

Introduction

Purpose of the SSADM standard

The Structured Systems Analysis and Design Method (SSADM) is a method widely used in the United Kingdom for the development of information systems. In developing such systems, many organizations require the procurement of products and services from external suppliers. In this context there is a need for a clearly established and uniformly recognized statement of the specification for information systems products using SSADM. This standard provides a clearly identified basis for the interface between customers for, and suppliers of, SSADM application products for information systems development. Application products are those normally associated with the development of a computer system.

Aligning with the modular structure of SSADM, there is recognition of the necessity to be able to locate products within the identified modules and stages of the method, thereby enabling procurement of coherent groups of products.

The establishment of a standard in this fashion is intended to promote consensus and good practice in information systems development.

Specifically this British Standard will:

- a) contribute to an open market in SSADM products and services, by providing a recognized specification of products;
- b) provide a basis for negotiation between customers for and suppliers of SSADM products and services;
- c) facilitate the transfer of products across stages and modules of the method; supporting the flexibility of SSADM;
- d) facilitate the procurement of services, training, tool support, and other related support facilities for an SSADM environment.

NOTE Guidance on the management of product design can be found in BS 7000.

Readership

The audience for the SSADM manuals includes practitioners, user managers, trainers, service and product providers, project and technical managers and IT directorate staff, standards makers and co-ordinators, data administrators information managers and academics. This British Standard is aimed at a subset of this group: those involved directly in the clear specification of required SSADM products delivered by suppliers to customers in a contractual context. It is, however, specifically aimed at those non-specialists involved in such contexts, who would not expect to have to be familiar with the detailed knowledge of SSADM contained in the manuals. It will, therefore, not be necessary for users of this standard to have a detailed knowledge of the method itself.

A specification for products

This British Standard has been produced as a specification for products (see BS 0-3). The scope for such a standard is to specify complete or partial products in terms of certain key characteristics. This standard is not intended to be a specification for the method (SSADM) itself.

Structure of SSADM

SSADM is modular in structure. (See Figure 1 for the modules and stages involved.) It consists of five modules, each being composed of one or more stages. Completion of any stage or module is marked by acknowledged completion of a defined and agreed set of products. Initiation of any stage or module is marked by acceptance of inputs; these inputs may be products from preceding stages or modules, or they may be derived from non-SSADM origins.

SSADM activities

SSADM stages are further subdivided into steps, each step consisting of a number of activities or tasks. The details relating to the performance of these tasks are contained in the SSADM manuals. They are not of direct concern to this standard for the specification for products.

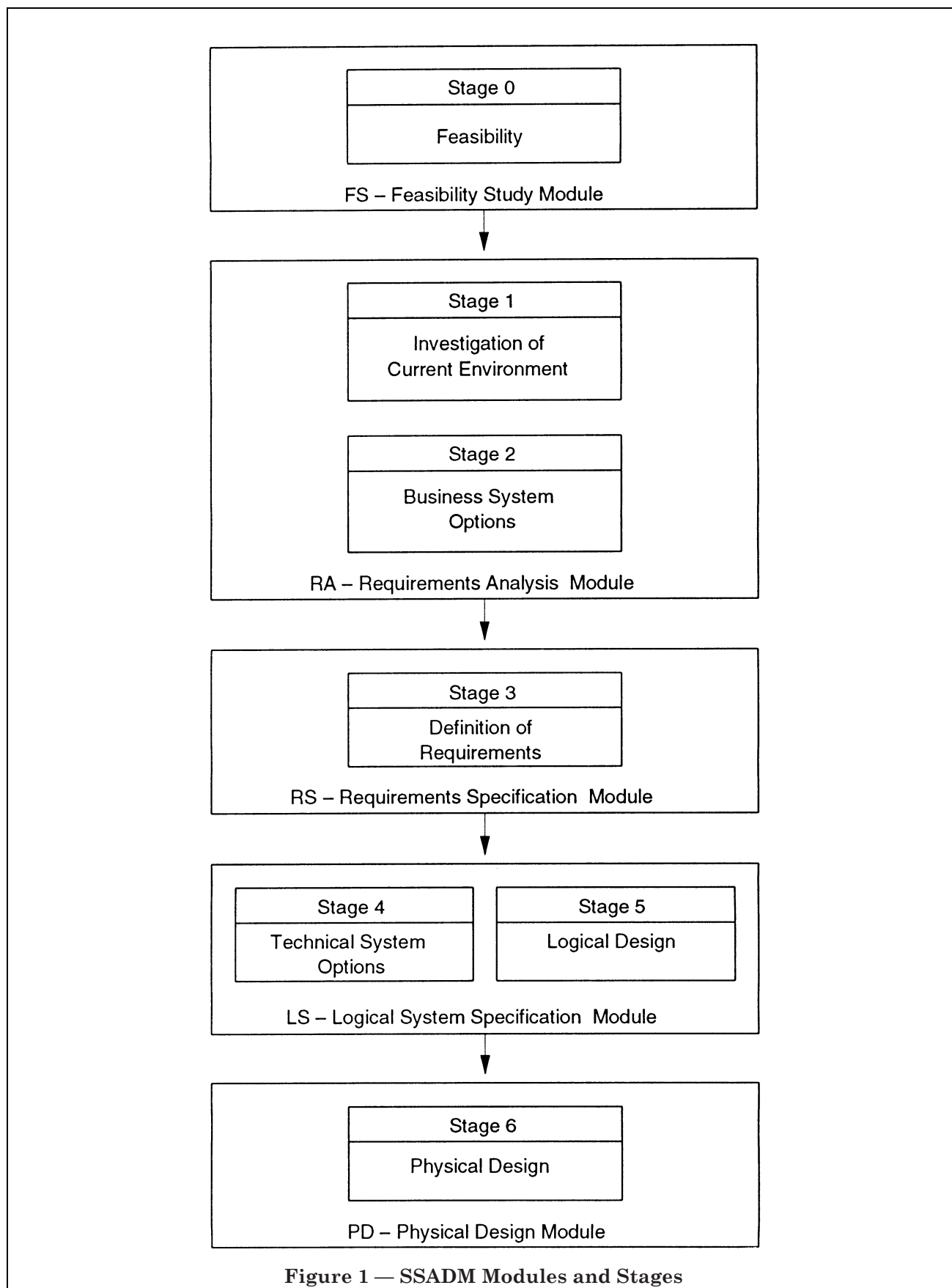
Conformity by presentation

The manual for SSADM version 4 suggests notations and conventions for aspects of products that can be represented graphically. In this standard where a product can be represented graphically the product entry in section 3 includes details of presentation. The aim of such detail is to present illustrations of components of the product. It is not the aim of this standard to offer completed examples of the products themselves, but to define the preferred notation for component parts.

This “preferred presentation” is a stronger view of presentation than that defined in the manuals. It permits the standard to offer a baseline for readability and comprehensibility across a range of contexts: between customers, suppliers and practitioners.

Conformity to the standard may, however, be accomplished in terms of individual products without precluding alternative presentation conventions, as long as the products delivered can be shown to fulfil the other criteria specified in the standard.

In most cases the preferred presentation has been that exemplified in the SSADM manuals. In other cases it has been necessary to select one amongst several options for the preferred presentation.



Section 1. General

1.1 Scope

This British Standard specifies the application products encompassed by the use of SSADM in the development of information systems. The complete list of products covered by this standard is given in Annex A.

This British Standard focuses on the interface between customer and supplier when supply of products and services relating to systems development, using SSADM, is the key concern. The focus of the standard is on products, and the means for assessing delivery of completed products.

The products produced during an SSADM development project can be classified in a number of ways:

- a) by Module or Stage;
- b) by concept and technique;
- c) alphabetically.

Annex B groups products according to the Module/Stage in which they are produced. Section 3 gives more detailed entries for products grouped by concept and technique; this precludes the necessity to repeat common details for products which occur in more than one Stage or Module of the method. Annex A lists products alphabetically, cross-referenced to the relevant section in this standard.

1.2 References

1.2.1 Normative references

This British Standard incorporates, by dated or undated reference, provisions from other publications. These normative references are made at the appropriate places in the text and the cited publications are listed on the inside back cover. For dated references, only the edition cited applies; any subsequent amendments to or revisions of the cited publication apply to this British Standard only when incorporated in the reference by amendment or revision. For undated references, the latest edition of the cited publication applies, together with any amendments.

1.2.2 Informative references

This British Standard refers to other publications that provide information or guidance. Editions of these publications current at the time of issue of this standard are listed on the inside back cover, but reference should be made to the latest editions.

1.3 Definitions

For the purposes of this British Standard, the following definitions apply.

NOTE See C.1 for an explanation of the conventions used in this clause.

1.3.1

acceptance (testing) criteria

Object

criteria setting out the details of how the final implemented system is required to perform specific tasks so that the users are satisfied that it caters for their needs

NOTE Initial details will be compiled based on SSADM documentation, but will require further development outside of SSADM.

1.3.2

access path (A)

Object

route to be taken through the Logical Data Model from an entry point to the entity, or entities, required for a particular piece of processing

NOTE The Logical Data Model is constructed with a view to implementing update processing, so update processing access paths are not documented separately. Enquiry processing access paths need to be documented explicitly.

1.3.3

activity (N)

General

that which, within SSADM steps, effects transformation in one or more products

1.3.4

activity descriptions (A)

Product

details of what the transformation (in one or more products) is, the inputs, the references, and the participants

1.3.5

activity network

Product

all of the activities placed into logical sequence, thus enabling timescales to be estimated and work to be scheduled

1.3.6

ad-hoc enquiry (A)

Object

enquiry which is not pre-defined but is created by the user as and when it is needed

NOTE Ad-hoc enquiries may be documented for each applicable user role as individual function definitions.

1.3.7 analysis of requirements

Product

module Product formed from the Requirements Analysis Module

NOTE It consists of the Current Services Description, Requirements Catalogue, User Catalogue and the Selected Business System Option.

1.3.8 application development standards

Product

standards defining the standards which apply to the physical design and development activities for this project/application

1.3.9 application naming standards

Product

standards defining the naming conventions for all aspects of the application under development, with particular emphasis being placed on constraints imposed by the (physical) implementation environment

1.3.10 application style guide

Product

set of standards, covering the user interface, to be followed within a particular application development

NOTE 1 This document is based on the Installation Style Guide and tailored to the specific needs of this project.

NOTE 2 This is also known as Style Guide.

1.3.11 attribute (A)

Object

characteristic property of an entity type, that is, any detail that serves to describe, qualify, identify, classify, quantify or express the state of an entity

NOTE 1 If possible the logical attributes for an entity will be translated directly into data items within the physical representation.

NOTE 2 An attribute may be optional for an entity type, meaning that it does not apply to all occurrences of the entity.

1.3.12 attribute/data item description

Product

each description within this set documenting all known details about an attribute (data item) in the logical system

NOTE Details are later translated into the physical data item(s) for the implementation of the system. Also known as Data Item/Attribute Description.

1.3.13 batch

Object

logical grouping of events (or functions) which always happen within the same time-frame

NOTE See also 1.3.102.

1.3.14 bottom-level process

Object

process on a Data Flow Diagram which is not further decomposed into another Data Flow Diagram

NOTE See also 1.3.61.

1.3.15 business system option (BSO)

Technique

the means by which users agree desired functionality of the new application with developers

NOTE BSOs are used to define the functionality needs and the boundary for the system, with reference to the business needs.

1.3.16 business system options

Structural Model Element. Stage 2

the Stage aiming to take the Requirements Catalogue, Current Services Description and User Catalogue and use this information as the basis on which to decide the most appropriate way for development to meet the business needs

NOTE This Stage has two Steps:

- a) Define Business System Options;
- b) Select Business System Options.

1.3.17 business system options

Product

set of Business System Options which is compiled so that a selection can be made

1.3.18 capacity planning

Technique

predicting the (hardware/software) configuration required to satisfy the constraints and requirements of the proposed system

NOTE It is also used to assist in the development of service level agreements (outside of SSADM).

1.3.19 capacity planning input

Product

processing and data information passed outside of SSADM activities to capacity planning techniques, explicitly during development of Technical System Options

1.3.20 command structure

Product

product showing the directions that control can take when a user decides to complete or terminate a particular dialogue

NOTE This allows navigation to be implemented with or without menus.

1.3.21 common process

Object

aspect of processing identified as common to several processes (or functions) within a SSADM system specification

NOTE Functionality such as a calculation of entitlement under a benefits scheme or discount arrangements may occur in more than one process or function. Details for each common process need to be documented in an Elementary Process Description with the relevant cross-references. These may be carried forward as part of the Function Definitions within the Requirements Specification. Common processes are vital components of the Physical Design, documented in the Function Component Implementation Map.

1.3.22 component

Object

object represented in the Function Component Implementation Map

NOTE It is usually classified as one of the following:

- super function;
- function;
- I/O process;
- database process;
- common process.

1.3.23 composite data flow

Object

a data flow on a Data Flow Diagram which is decomposed into simpler data flows on a lower-level Data Flow Diagram

1.3.24 context diagram

Product

diagram drawn to illustrate the initial scope of the proposed system

NOTE The diagram concentrates on the major inputs and outputs of the system and shows the external sources and recipients of system data.

1.3.25 core SSADM

General

central, fundamental part of SSADM which refers to the following Modules:

- Feasibility Study (FS);
- Requirements Analysis (RA);

- Requirements Specification (RS);
- Logical System Specification (LS);
- Physical Design (PD).

1.3.26 cost/benefit analysis

Product

objective way to judge the merit of one option against another

NOTE This forms the vital financial part of the specification of each option (business or technical).

1.3.27 current environment

General

current services (computer and manual) and the wish list for the future system

NOTE Where no current system exists this will simply reflect the wish list for the new system. This is a generic term.

1.3.28 current environment logical data model

Product

product providing a detailed description of the information used or produced by the current environment

NOTE It is developed during the Requirements Analysis Module and forms part of the Current Services Description. (See also 1.3.117 and 3.6.5.)

1.3.29 current physical data flow model

Product

model showing how the current services are organized and processing is undertaken

NOTE An overview of current services is provided by documenting only the level 1 Data Flow Diagram. (See also 1.3.35 and 3.2.4.)

1.3.30 current services

General

services encompassing all existing processing within the business area being addressed by the project, irrespective of whether those functions are manual or computerized

NOTE This is a generic term.

1.3.31 current services description

Product

product providing the details of the logicalized current system which, with the Requirements Catalogue and User Catalogue, is output from Stage 1: Investigation of Current Environment

1.3.32 data catalogue

Product

central repository for all the descriptive information about items of data

NOTE 1 This includes physical details which may be found during data flow modelling activities as well as physical design activities. Logical data modelling will provide information about attributes (the logical equivalent to data items).

NOTE 2 See also 1.3.12, 1.3.101, 3.6.1 and 3.6.4.

1.3.33 data flow

Object

object showing where data is being passed between different elements on a Data Flow Diagram

NOTE The name associated with the data flow should be meaningful to those reviewing the Data Flow Diagram. Data flows will pass into and out of the system and between processes (generally via data stores except on the Current Physical Data Flow Model where process-to-process flows may reflect inadequacies in the current system). When the system boundary is being defined there may even be data flows between external entities. At the lowest level of the Data Flow Diagram these are simple data flows, though they may be combined into composite data flows on higher level diagrams.

1.3.34 data flow diagram

Product

diagram showing how services are organized and processing is undertaken

NOTE It should be a simple diagram that is readily understood, so that it can act as an effective means of communication between analysts and users. (See also 1.3.35 and 3.2.4.)

1.3.35 data flow model

Product

set of Data Flow Diagrams and their associated documentation

NOTE The diagrams form a hierarchy with the Data Flow Diagram Level 1 showing the scope of the system and the lower level diagrams expanding the detail as appropriate. Additional documentation provides a description of the processes, input/output data flows and external entities.

1.3.36 data flow modelling (DFM)

Technique

technique used to help define the scope of the system and ensure that the analysts have a clear understanding of the users' problems and requirements

NOTE The technique is used to build a model of the information flows and not to define the detail of the processing performed by the system.

1.3.37 data item (A)

Object

element or field of a data store or component of inputs or outputs (which may be held within an entity type on the Logical Data Structure)

NOTE A data item is any detail that serves to describe, qualify, identify, classify, quantify or express the state of the data store. The logical attributes for an entity will, in general, translate directly into data items within the physical representation.

1.3.38 data store (A)

Object

collection of any type of data in any form as represented on a Data Flow Diagram

NOTE This may be a computer file or a box of documents or any other means of storing data. Data stores may be standing, long-term files such as sales and purchase ledgers, or short-term accumulations such as daily batches of documents. Data stores on logical DFDs cannot be manual.

1.3.39 database management system (DBMS)

General

mechanism for managing data held within a computerized system

NOTE Conceptually data is held within one file regardless of how the content is physically organized.

1.3.40 database process

Object

a physical process to be applied to the implemented database

NOTE The valid processes will vary depending on the DBMS used for implementation. They will include read and write operations to varying levels of sophistication.

1.3.41 DBMS data storage classification

Product

product for analysing and recording data storage and retrieval mechanisms of DBMS or file handler

1.3.42 DBMS performance classification

Product

product recording the factors which impact on the performance of a DBMS or file handler

1.3.43 definition of requirements

Structural Model Element. Stage 3

stage aiming to take the Analysis of Requirements and produce the Requirements Specification within the boundary identified in the Selected Business System Option

NOTE This Stage has eight Steps:

- Define Required System Processing;
- Develop Required Data Model;
- Derive System Functions;
- Enhance Required Data Model;
- Develop Specification Prototypes;
- Develop Processing Specification;
- Confirm System Objectives;
- Assemble Requirements Specification.

1.3.44

detail entity (A)

Object

the entity type at the several end where two entity types are connected by a 1 : m relationship, a single instance of one entity type is related to several instances of the other

NOTE See also 1.3.132.

1.3.45

determinant (A)

Object

values for some of the attributes within an entity occurrence directly dependent on the value of a group of specific attributes (possibly a group of one), which effectively identify this occurrence of the entity

NOTE The identification attributes are known as the determinant within relational data analysis.

1.3.46

dialogue

Object

on-line activity required by a particular user role so that they can action (perform) a particular function

1.3.47

dialogue control table (A)

Product

table used to identify and capture the navigation between the logical groupings of dialogue elements

1.3.48

dialogue design (DD)

Technique

technique used to define the on-line activity of the system

NOTE Dialogues are identified as part of the Requirements Specification and then logically designed explicitly as part of the Logical Design. Physical dialogue design activities are undertaken during the physical design activities to complete the design prior to system implementation.

1.3.49

dialogue element

Object

section of an input or output data flow which may consist of many data items

NOTE Each dialogue element is represented as a box on a Dialogue Structure.

1.3.50

dialogue element description

Product

product used to describe a dialogue element

NOTE A set of these provides the detailed documentation for a Dialogue Structure.

1.3.51

dialogue identification (A)

Technique

technique used to identify the complete set of the requirements for dialogues within the proposed system

NOTE Dialogues which are identified as critical in function definition may be subjected to specification prototyping.

1.3.52

dialogue level help

Product

product used to detail the level of help that the user (user role) requires to progress through this dialogue

1.3.53

dialogue structure (A)

Product

diagrammatic representation of a dialogue

1.3.54

dialogues

Product

product used to package together details of all identified dialogues within the system

1.3.55

document flow diagram

Product

diagram showing how documents pass around the system

NOTE This may be the initial diagram drawn within the data flow modelling technique to assist in defining/identifying the boundary of the system. This diagram will be produced if the current system is predominantly clerical and involves the passing of information using forms or other documents.

1.3.56

domain

Object

pool of values from which the actual values for an attribute can be drawn

1.3.57

effect (A)

Object

change caused to a single entity occurrence because of a single event

NOTE An effect can be one of four kinds: create (birth), modify (update), logical delete (death) or update of state indicators.

The effect has to be qualified where an occurrence of an entity is affected in one of severally mutually exclusive ways by an event.

1.3.58

effect correspondence diagram (ECD)

Product

diagram showing all the effects an event has on data within the system and how those effects impact upon each other

1.3.59

effect correspondence diagramming

Technique

technique used to analyse and document how effects are related to each other for each event

1.3.60

effect qualifier (N)

Object

object given after the event name to qualify the effect where an occurrence of an entity is affected in one of severally mutually exclusive ways by an event

NOTE The qualifier is bracketed after the event name and such brackets should differ in nature to those used to qualify an entity by its role.

1.3.61

elementary process

Object

lowest-level process on a Data Flow Diagram which relates directly to the user (business) environment in which the system (computerized or manual) is to be operated

NOTE 1 It will assist in the identification of functions.

NOTE 2 Whole, or partial, elementary processes may be regarded as common across a number of processes, in which case they will be separately described.

1.3.62

elementary process description (EPD)

Product

description of the business environment in which the process is trying to operate

NOTE A requirement for common processing may also be described within an Elementary Process Description and cross-referenced to the elementary processes or functions which use it.

1.3.63

end-module product (N)

Object

a product which is delivered at the end of a Module, having been created or updated within that Module

1.3.64

end-stage product (N)

Object

a product which is delivered at the end of a Stage, having been created or updated within that Stage

1.3.65

enquiry access path (EAD)

Product

route through the Logical Data Model from an entry point to the entity type(s), required for a particular enquiry

1.3.66

enquiry element

Object

element which requires information to be read from the database but involves no update processing

NOTE 1 Some update functions contain enquiries as well as updates (events); these elements are also called enquiry elements.

NOTE 2 This is also known as enquiry.

1.3.67

enquiry function (A)

Object

function which requires information to be read from the database but involves no associated update processing

1.3.68

enquiry process model (EPM) (A)

Product

model consisting of a structure diagram for an enquiry processing requirement and the associated Operations List

NOTE The structure is based on the Enquiry Access Path and the appropriate I/O Structure.

1.3.69

enquiry trigger

Object

data items that have to be input to the system to initiate an enquiry

1.3.70

entity (A)

Object

something, whether concrete or abstract, which is recognized as being of importance within the environment under investigation and which is capable of an independent existence

NOTE 1 Each occurrence of an entity (type) should be uniquely identifiable.

NOTE 2 Logical data modelling identifies types of entity, not individual occurrences, i.e. tenant and applicant not John Smith. (See 1.3.76 and 1.3.78.)

1.3.71

entity description

Product

product documenting all of the details concerned with an entity type on the Logical Data Structure, including details of state indicators which are applied during entity life history analysis

NOTE There will be associated Relationship Descriptions for each related entity type on the Logical Data Structure.

1.3.72 entity-event modelling (EEM)

Technique

combination of entity life history analysis and effect correspondence diagramming

1.3.73 entity life histories (ELH) (A)

Product

set of structures combining possible lives of possible occurrences of an entity type

1.3.74 entity life history (A)

Object

object that charts the events that may cause any one occurrence of an entity type to be changed in any way

NOTE It shows the valid structure of events (initially identified through use of data flow modelling and function definition techniques) affecting an entity type on the Logical Data Structure.

1.3.75 entity life history analysis

Technique

technique bringing together and validating the processing and data requirements of the system

NOTE This is done by investigating the Required System Logical Data Model. Events which affect the entity lives are documented showing the sequence of effects which take place.

1.3.76 entity occurrence (N)

Object

specific (uniquely identifiable) instance of an entity type

1.3.77 entity roles

Object

the entity deemed to be assuming different roles if a single event affects more than one occurrence of a particular entity type and the effects are different for each occurrence affected

NOTE Each possible role has to be separately identified on the ELH diagram for the relevant entity, as separate processing will have to be specified for each role. The qualifier for the role is placed in brackets after the event name to provide a complete identification of the entity role and brackets should differ in nature to those used to qualify an event by its effect.

1.3.78 entity type (N)

Object

something, whether concrete or abstract, which is recognized as being of importance within the context of the system being developed, and which is capable of an independent existence

NOTE Each occurrence of an entity type should be uniquely identifiable.

1.3.79 event (A)

Object

whatever triggers a process (on a Data Flow Diagram) to update the values or status of the system

NOTE An event may cause more than one entity occurrence or type to be changed. In the logical system, an event initiates an update process.

1.3.80 event/entity matrix

Product

grid used to identify which entity types are affected by a particular event

NOTE 1 It provides two checks: that each event affects at least one entity type, and that each entity type is affected by at least one event.

NOTE 2 Also known as Entity/Event Matrix.

1.3.81 evolving product (N)

Object

product which is delivered but expected to evolve or be the subject of further refinement in later Stages or Modules

NOTE A distinction is made between end-stage/end-module products which are delivered as complete at the end of the Stage or Module.

1.3.82 exclusive relationship group

Object

group where the participation of an entity occurrence in one relationship precludes its participation in one or more other relationships

1.3.83 expression

General

statement which can be evaluated

NOTE The statement may be forming a comparison between individual items. An expression may itself consist of several sub-expressions linked by “and” or “or”.

1.3.84**external entity***Object*

source or recipient (or both) of data which exists outside the boundary of the defined system but which communicates with the system

NOTE An external entity may be another system, an organization, an individual or a group of people. These are documented within the Data Flow Model.

1.3.85**external entity description (EED)***Product*

brief explanation of the relevance of an external entity in relation to the existing or proposed system

NOTE The detail will cover responsibilities or functions of the external entity and any constraints on the interface with the proposed system.

1.3.86**feasibility***Structural model element. Stage 0*

stage aiming to investigate the requirements laid down within the Project Initiation Document and suggest the way ahead

NOTE This Stage has four Steps:

- Prepare for the Feasibility Study;
- Define the problem;
- Select Feasibility Options;
- Assemble Feasibility Report.

1.3.87**feasibility***Technique*

technique used to investigate the possibilities for future work in a particular area of the business

NOTE The scope of the study will be laid down within the Project Initiation Document. Much of the analysis will be done using other techniques including logical data modelling, data flow modelling, requirements definition and dialogue design.

1.3.88**feasibility options***Product*

set of Feasibility Options which is compiled so that a selection can be made

NOTE Each option documents the functions to be incorporated and details implementation requirements. Each description is textual with some planning information.

1.3.89**feasibility report***Product*

module Product from the Feasibility Study Module, documenting the possible approaches to the system development and assessing the impact of each so that the most appropriate way ahead can be fully investigated

1.3.90**feasibility study module (FS)***Structural Model Element*

module whose objective is to produce the Feasibility Report which will suggest the way ahead for the project

NOTE 1 The activities form a short assessment of a proposed information system to determine whether the system would be feasible and appropriate to the business needs of the organization. Feasibility is assessed in terms of the managerial, business, financial, technological and cultural needs of the organization.

NOTE 2 This Module has one Stage: Stage 0: Feasibility.

1.3.91**fragment***Object*

defined processing element which has a specific identity, purpose, set of inputs, outputs and correspondences

NOTE It may be at any level. 1 : 1 with component or 1 : 1 with operation, e.g. Read Master. It may be in support of a process, or a data group, screen or error message.

1.3.92**function***Object*

set of system processing which the users wish to schedule together, to support their business activity

1.3.93**function component implementation map (FCIM)***Product*

classification and specification of all implementation fragments for all function components defined in the Function Definitions to meet the processing requirements

NOTE See also 1.3.22.

1.3.94**function definition***Product*

description of the function providing a cross-reference to other associated SSADM products

1.3.95**function definition (FD)***Technique*

identification and documentation functions which are the units of processing carried forward to physical design

1.3.96**function definitions***Product*

packaging of all details about functions to be included in the Requirements Specification

NOTE These details are further expanded during physical design activities. Basic information about a function consists of a Function Definition with one or more I/O Structures. Further information may be found in Enquiry Access Paths (for enquiry functions) and (common) Elementary Process Descriptions.

1.3.97 function type

General

category of the function

NOTE There are three ways of categorizing functions:

- processing requirements: update or enquiry;
- access details: on-line or off-line;
- method of initiation: user or system.

1.3.98 functional area (A)

General (in DFM)

view that the user has of the processing system based upon the tasks which they undertake

NOTE Each of these views will represent one or more Functional Areas depending upon how widely the individual user uses the system. Each Functional Area will be directly related to a specific task which the user undertakes.

1.3.99 functional requirement

Object

requirement for a particular facility or feature which describes what the system should do, e.g. include updates, enquiries and reports

1.3.100 grouped domain

Object

pool of values associated with validation rules and formats common to several attributes

1.3.101 grouped domain description

Product

products used to document validation rules and formats common to several attributes

1.3.102 group of events

Object

collection of event occurrences containing more than one event type

NOTE There may be more than one occurrence of each event type. The term is used to describe the batching or grouping of data for input to a function.

1.3.103 help

General

facility provided in the implemented system to guide the user through the options available to them at any point in time

NOTE Dialogue level help may be associated with Dialogue Structures so that the need to consider (and implement) these features can be passed forward into the physical design activities.

1.3.104 I/O descriptions

Product

product used to document data flows which cross the Data Flow Model system boundary

1.3.105 I/O process

Object

process, documented in the physical design, for producing detail to or accepting information from a specific piece of functional processing

1.3.106 I/O structure

Product

product documenting the input to and outputs from a function, or part of a function

NOTE See also 3.5.5 and 3.5.6.

1.3.107 I/O structures (for all functions)

Product

packaging of all I/O Structures for all identified functions

NOTE This product exists because some SSADM activities do not require the complete set of documentation for all functions.

1.3.108 impact analysis

Product

description of the effects of the option (business or technical) on the user environment covering issues concerned with organization, procedures, and implementation factors

NOTE This product is used to document the ramifications of pursuing a particular course of action.

1.3.109 installation development standards (A)

Product

standards documenting the criteria which should be used during the development of all Information Systems within the organization/installation

1.3.110 installation style guide

Product

set of standards about the nature, approach and style of the human factors aspects of computerized systems

NOTE 1 The standards should be followed by all projects undertaken within an organization.

NOTE 2 Also known as Style Guide.

1.3.111 integrity errors

Object

inconsistencies in a database

NOTE The data in a database have to be consistent for the system to be able to function correctly. Where inconsistency arises the database is not maintaining referential integrity of the database.

1.3.112 interim product (N)

Object

product created during a stage or module, but which is not specifically offered as a delivered product at the end of the stage or module itself

NOTE It is most often an in-progress product created as part of the continuing development process, and used in the derivation of other (end-stage, end-module) products.

1.3.113 investigation of current environment

Structural Model Element. Stage 1

stage aiming to look at the existing services and requirements and produce a logical view of the processing requirements

NOTE This Stage has six Steps:

- Establish Analysis Framework;
- Investigate and Define Requirements;
- Investigate Current Processing;
- Investigate Current Data;
- Derive Logical View of Current Services;
- Assemble Investigation Results.

1.3.114 key (A)

Object

attribute or a combination of attributes which can be used to identify (uniquely) each occurrence of an entity type

1.3.115 list

Object (in DBMS)

representation of relationship between master and detail entities by a list of keys, including the key of the master and the key of each detail entity, with pointers connecting each entry to the next entry

1.3.116 logical data flow model

Product

set of Data Flow Diagrams and their associated documentation produced during logicalization in Stage 1

NOTE See also 1.3.35.

1.3.117 logical data model

Product

product providing an accurate model of the information requirements of all or part of an organization

NOTE This serves as a basis for file and database design, but is independent of any specific implementation technique or product. The Logical Data Model consists of a Logical Data Structure, Entity Descriptions and Relationship Descriptions. Associated descriptions of attribute/data items and grouped domains are maintained in the Data Catalogue.

1.3.118 logical data modelling (LDM)

Technique

technique used to produce a conceptual model of the information requirements of all or part of an organization

1.3.119 logical data store/entity cross-reference (A)

Product

matrix showing the correspondence between logical data stores in the Data Flow Model and the entity types on the Logical (current or required) Data Model

NOTE 1 This is used to ensure that a main data store corresponds to an entity type or group of types. Also each entity type on the Logical Data Model has to be held completely within one and only one main data store. There may be two variants of this product: Logical, Requirements Specification.

NOTE 2 Also known as Entity/Logical Data Store Cross-reference.

1.3.120 logical data structure (LDS) (A)

Product

diagrammatic representation of the information needs of an organization in the form of entity types and the important relationships between them

1.3.121 logical database process design (LDPD)

Technique

technique used to translate the information gathered during the Requirements Specification Module into a logical specification which can then be translated into a physical design for the system in any implementation

NOTE This also provides a logical definition of the system which assists in maintenance of the implemented system. This technique covers the definition of the processing for data input to, and output from, the database.

1.3.122
logical design*Product*

stage Product from Stage 5: Logical Design, packaging the logical view of processing with the Required System Logical Data Model and the Requirements Catalogue

1.3.123
logical design*Structural Model Element. Stage 5*

stage aiming to take the Requirements Specification and further develop the processing requirements in such a way as to be implementation independent prior to physical design activities

NOTE This Stage has four Steps:

- Define User Dialogues;
- Define Update Processes;
- Define Enquiry Processes;
- Assemble Logical Design.

1.3.124
logical grouping of dialogue elements (LGDE)*Object*

dialogue elements within a particular Dialogue Structure grouped together where input/output requirements suggest that there are benefits to be gained by doing so, such as simpler processing

1.3.125
logicalization*Technique*

sub-technique of data flow modelling, used in Stage 1, which converts the Current Physical Data Flow Model into a logical model with no physical implications

1.3.126
logical key (A)*Object*

form of key meaningful to the users of the system and which is usually allocated by them

NOTE Each instance of such a key uniquely identifies one record. It does not tell the system where the data is physically stored. (See also 1.3.162.)

1.3.127
logical/physical data store cross-reference (A)*product*

matrix that is used within the data flow modelling technique (logicalization) to cross-reference the logical view of data organization from the current (physical) services

1.3.128
logical process model*Product*

product used to package all processing details within the Logical Design

1.3.129
logical system specification*Product*

module product from the Logical System Specification Module consisting of the Selected Technical System Option, the Technical Environment Description and the Logical Design

1.3.130
logical system specification module (LS)*Structural Model Element*

module with the objective to produce the Logical System Specification

NOTE The Selected Technical System Option and the Technical Environment Description define the scope of the physical implementation. This detail has to be consistent with the Logical Design. The Application Style Guide is also developed during this Module. This Module has two Stages:

- Stage 4. Technical System Options;
- Stage 5. Logical Design.

1.3.131
main data store*Object*

representing elements on Data Flow Diagrams; the data that is held centrally in a system so that it can be used by a number of different processes

NOTE Entities on the Logical Data Model have to reside in one and only one main data store. Each main data store has to have one or more entities associated with it.

1.3.132
master entity (A)*Object*

the entity at the single end where two entity types are connected by a 1 : m relationship and a single instance of one entity is related to several instances of the other

NOTE During Stage 3, all relationships on the LDM have to be represented as 1 : m relationships. This provides a clear hierarchical structure to provide the necessary input to the following techniques: entity life history analysis, logical database process design and physical design. (See also 1.3.44.)

1.3.133
menu*Object*

hierarchical structure used to provide a user (role) with access to available and applicable functions

1.3.134 menu structure

Product

product providing a diagrammatic representation of the menus to be used within the system

1.3.135 message pair

Object

exchange between the user and the system comprising an input message and an output message

1.3.136 module

Structural Model Element

element comprising a defined set of products and activities, a finite lifespan and an organizational structure

NOTE The SSADM framework requires a project to be subdivided into a number of Modules each of which consists of one or more Stages. Each Module forms a distinct unit for management purposes. The production of the defined end products, to agreed quality standards, signals the completion of the Module.

1.3.137 non-functional requirement

Object

requirement describing how, how well or to what level of quality a facility of the system should be provided

NOTE Examples include service level requirements, access restrictions, security, monitoring, audit and control, and constraints.

1.3.138 non-procedural specification

Object

specification of a process (or entire function), generally used with a program generator, which may contain no more than the specification of the inputs and outputs, and the correspondences between them

1.3.139 normal form

General

result of applying relational data analysis techniques

NOTE 1 There are several stages of normalization; relations are translated into:

- first normal form (1NF);
- second normal form (2NF);
- third normal form (3NF);
- fourth normal form (4NF);
- fifth normal form (5NF).

NOTE 2 SSADM recognizes the existence of fourth and fifth normal form but does not tend to put them to practical use.

1.3.140 normalization

Technique

relational data analysis, effectively using rules to analyse the way items of data depend upon one another for their meaning

NOTE In a normalized Logical Data Model all entity types, considered as relations, have to be in third (or higher) normal form.

1.3.141 normalized relation (A)

Object

result of normalization

NOTE It is a group of data (items) which are a convenient package for access and manipulation purposes. The information held within a normalized relation will be concerned with a particular concept.

1.3.142 object type (N)

Object

object having a defined range of behaviours

NOTE Typically, a function component (e.g. I/O process) or an element type supported in the physical environment. Similar to an entity type.

1.3.143 off-line function

Object

function where all the data are input and the whole of the database processing for the function is completed without further interaction with the user

1.3.144 on-line function

Object

function where the system and the user communicate through input and output messages, i.e. message pairs

NOTE The system responds in time to influence the next input message. On-line functions may include off-line elements such as printing an off-line report.

1.3.145 operation (A)

Object

discrete pieces of processing which may combine to constitute effects; initially identified during entity life history analysis and expanded during process modelling

1.3.146 operations list

Product

list of all the operations which are detailed on a processing structure which may be for an enquiry or an update

NOTE See also 3.7.1 and 3.7.3.

1.3.147
outline current environment description (A)*Product*

description of the current services and any existing problems; produced during the Feasibility Study

NOTE The analysis is similar to that carried out in Stage 1, but is not completed to the same level of detail.

1.3.148
outline development plan (A)*Product*

product providing management with information on the development strategy for the remainder of the project for the specific Technical System Option so that provisional timescales and resource requirements, and therefore development costs, can be estimated

1.3.149
outline required environment description (A)*Product*

description of the requirements to be included within the proposed system; produced during the Feasibility Study

NOTE The analysis reflects that carried out in Stage 1 but is not completed to the same level of detail.

1.3.150
overview logical data structure (A)*Product*

product describing the major entity types and their relationships within the area of the investigation

NOTE See also 1.3.117.

1.3.151
parallel structure (A)*Object*

structure appearing on Entity Life Histories, used to show where certain events may happen within the lifetime of an entity, but not in any prescribed order

1.3.152
physical data design*Product*

definition for the physical database which is to be implemented

NOTE The design is developed in two Steps; the first produces a first-cut design based on applying rules about the DBMS to the Required System Logical Data Model; the second is a design optimized for performance reasons.

1.3.153
physical data design (PDD)*Technique*

technique taking the Required System Logical Data Model and translating it into a product specific database design within the chosen technical environment with consideration having been given to performance and space constraints

1.3.154
physical design*Product*

module product from the Physical Design Module which defines the data and processing elements of the implementable system

1.3.155
physical design*Structural Model Element. Stage 6*

stage whose objective is to specify the physical data and physical processes using the language and features of the chosen implementation environment and incorporating installation standards

NOTE This Stage has seven Steps:

- Prepare for Physical Design;
- Create Physical Data Design;
- Create Function Component Implementation Map;
- Optimize Physical Data Design;
- Complete Function Specifications;
- Consolidate Process Data Interface;
- Assemble Physical Design.

1.3.156
physical design*Technique*

technique taking the Logical Design and developing it into a specification for the implementation which is wholly dependent on the technical environment as documented in the Physical Environment Specification

NOTE This technique has two aspects: physical data design, and physical process specification.

1.3.157
physical design module (PD)*Structural Model Element*

module whose objective is to produce the Physical Design for the system based on the Logical System Specification and the Physical Environment Specification, i.e. it will be implementation dependent

NOTE This Module has one Stage: Stage 6, Physical Design.

1.3.158**physical design strategy***Product*

product documenting all aspects relating to designing the physical implementation of the application including all planning documentation

1.3.159**physical environment***Object*

implementation environment including, where appropriate, the development environment and the migration path between them

1.3.160**physical environment classification***Product*

classification of the environment in which the application is to be implemented which also describes the development environment and migration path where necessary

NOTE See also 1.3.41, 1.3.42 and 1.3.172.

1.3.161**physical environment specification (PES)***Product*

specification of the hardware and software products and services to be supplied, commissioned and made available for implementation

NOTE Generally this will be provided by the vendor.

1.3.162**physical key (A)***Object*

object which tells the system the address on the magnetic disc where a specified record is to be found

NOTE This is often called a pointer.

1.3.163**physical process specification***Product*

packaging of all of the specifications for processing which are required in the proposed system

1.3.164**physical process specification***Technique*

technique taking the Logical Design and tailoring it to the physical environment to produce the specification for all necessary processing (programs)

1.3.165**physical system specification***Product*

specification comprising the Physical Design, Application Development Standards (both from SSADM) and the Physical Environment Specification (which is likely to be the response to a procurement exercise)

1.3.166**problem definition statement***Product*

statement of the user requirement for the system produced during the Feasibility Study

NOTE This may include charts and diagrams to supplement the detail.

1.3.167**procedural specification (A)***Object*

specification of a process, derived from the input and output data relevant to a procedure, including operations and conditions allocated to it

NOTE It may take the form of a diagrammatic structure, based on SSADM structure diagram.

1.3.168**process***Object*

object transforming or manipulating information (data) in a system

1.3.169**process data interface***Product*

product documenting how the Logical Data Model can be mapped onto the Physical Data Design, showing how it interfaces with the Physical Processing Specification

NOTE It allows the designer to implement the logical update and enquiry processes as physical programs, independently of the physical database structure.

1.3.170**process/entity matrix***Product*

cross check that each entity type is used as the basic information for at least one process and identifying logical groupings of bottom-level processes during logicalization of the Data Flow Model

NOTE Also known as Entity/Process Matrix.

1.3.171**processing specification (PS)***Product*

development of this product may identify errors in other products and force previous Steps to be revisited

NOTE Product of Stage 3 (Develop Processing Specification). This product is not shown explicitly on the SSADM structural model.

1.3.172 processing system classification

Product

classification of the details of the processing environment which is to be used for implementation

NOTE Where appropriate, it also defines the development environment.

1.3.173 product (A)

Object

item of software, hardware, or documentation which may itself be a collection of other products

NOTE Each product is predefined and its production and delivery is planned within the project.
(See also 1.3.63, 1.3.64, 1.3.81, 1.3.112.)

1.3.174 product specification (N)

General

description of the purpose, form, components and completion criteria of a product

1.3.175 program (A)

Object

stored set of computer instructions assembled to meet the requirements of one or more functions

1.3.176 project

General

undertaking having the following characteristics:

- a defined and unique set of technical products to meet the business needs;
- a corresponding set of activities to construct those products;
- a certain amount of resources;
- a finite lifespan;
- an organizational structure with defined responsibilities.

1.3.177 project initiation document (PID)

Product

definition of the terms of reference and objectives for the project, approved by the project board at project initiation

NOTE It is used to identify business requirements, as well as organizational and general information needs, security aspects and an initial Project Plan.

1.3.178 prototype

Object

animated view provided for the user of how the system being developed will work

NOTE It enhances user understanding, allowing better identification of discrepancies and deficiencies in the user requirement.

1.3.179 prototype demonstration objective document

Product

list of the points of discussion to be addressed between the user and the analyst during the prototype demonstration completed prior to any prototype demonstration for each Prototype Pathway

NOTE Assumptions and queries for each menu, screen and/or report are listed under their respective component number.

1.3.180 prototype pathway

Product

combination of screen and report components, identified within the prototyping activities, with the existing menus

NOTE The pathways are a script for the prototyping session and hence use a limited and simple serial diagrammatic representation to convey the structure.

1.3.181 prototype result log

Product

document used to record the results of the prototype demonstration

NOTE This document is used in a similar capacity to minutes of a meeting. Each request made by the user is documented on the log, with a change grade, and the log is updated later to show what changes are required.

1.3.182 prototyping

Technique

variety of techniques which may be used to produce prototypes

NOTE See also 1.3.210.

1.3.183 prototyping report

Product

report prepared by the technical manager of the prototyping exercise to report back to the project board setting out whether the objectives for the exercise were achieved, or alternatively, the reason why they were not

NOTE It includes estimates of the value of the work done and where necessary suggests whether more work would (or would not) be beneficial.

1.3.184 prototyping scope

Product

product used by the project board to define the boundaries and objectives for the prototyping activities

1.3.185 quality

General

the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs

NOTE This definition is identical with that given in BS 4778-1.

1.3.186 quit and resume (A)

Object

object appearing on an Entity Life History to show how the prescribed sequence of events for an entity may be altered in particular circumstances

1.3.187 random event

Object

event which may occur at any point either during the life of an entity or during a particular stage of an entity life

1.3.188 RDA working paper

Product

documentation of the progress through relational data analysis, taking relations which are un-normalized through to third normal form

1.3.189 relation

Object

group of data items (or attributes)

NOTE When the Logical Data Model is validated using relational data analysis, a relation equates to an entity.

1.3.190 relational data analysis (RDA)

Technique

method of deriving data structures which have the least redundant data and the most flexibility

NOTE The flexibility is achieved by breaking down the data groups into smaller groups without losing any of the original information. It is the objective of this technique to transform all relations into at least third normal form.

1.3.191 relationship (A)

Object

association between two entity types, or one entity type and itself (recursion/involution), to which all instances (occurrences) of the relationship conform

1.3.192 relationship degree (A)

Object

indication where two entity types are directly related, of the number of instances of each such entity type that may participate in one instance of the relationship

1.3.193 relationship description (A)

Product

documentation of the details of a relationship between two entity types on the Logical Data Structure

1.3.194 report format

Product

product showing the layout of a printed report as desired by the user

1.3.195 required system data flow model

Product

model showing how the proposed services are to be organized and what processing is to be undertaken

NOTE See also 1.3.35.

1.3.196 required system logical data model

Product

product providing the detail of the proposed system information requirements which is developed during the Requirements Specification and Logical System Specification Modules

NOTE See also 1.3.117.

1.3.197 requirement (A)

Object

description of required feature of the proposed system

NOTE Requirements may be functional (describing what the system should do) and non-functional (describing how a facility should be provided, or how well, or to what level of quality). (See also 1.3.99 and 1.3.137.)

1.3.198 requirements analysis module (RA)

Structural Model Element

element where objective is to produce the Analysis of Requirements, within which the Selected Business System Option will define the scope of further investigation

NOTE This Module has two Stages:

- Stage 1. Investigation of Current Environment;
- Stage 2. Business System Options.

1.3.199 requirements catalogue

Product

central repository for information covering all identified requirements, both functional and non-functional, in which each entry is textual and describes a required facility or feature of the proposed system

1.3.200 requirements definition

Technique

technique focusing on the future, required system, used to identify and describe requirements for the proposed system which meet the needs of users, and of the business as a whole

NOTE As the project progresses, requirements are resolved using more formal techniques such as function definition.

1.3.201 requirements specification

Product

module product from the Requirements Specification Module, packaging all of the details which are required in order to decide upon the technical direction of the project

1.3.202 requirements specification module (RS)

Structural Model Element

module whose objective is to produce the Requirements Specification

NOTE This Module has one Stage: Stage 3. Definition of Requirements.

1.3.203 resource flow diagram

Product

diagram documenting how resources move within an organization

NOTE This may be the initial diagram drawn within the data flow modelling technique to define the boundary of the system. This diagram will be produced if the current system is predominantly concerned with movement of physical objects, e.g. goods.

1.3.204 screen format

Product

product showing the layout that the user requires on the visual display unit screen

1.3.205 selected business system option

Product

description of a chosen system development direction documenting the system boundary, inputs, outputs and the transformation taking place within the boundary

NOTE Essentially the description is textual with supporting (annotated) elements from the Current Services Description.

1.3.206 selected technical system option (TSO) (A)

Product

description of the chosen technical system option

NOTE The technical details are placed in the Technical Environment Description.

1.3.207 service level requirement

Object

non-functional requirement which states the required quality of service the user expects from a functional aspect of the system

NOTE The service level requirement may be associated with Function Definitions.

1.3.208 space estimation

Product

assessment of the storage requirements of the data design using a particular implementation environment

NOTE The relevant information will vary widely depending upon the technical environment and has to be designed and compiled for each application.

1.3.209 specific function model

Object

model of the components of a particular SSADM function

1.3.210 specification prototyping (SP) (A)

Technique

technique used to identify and trap errors in the specification of the user requirement, and so to enhance requirements definitions prior to detailed logical design activities being undertaken

1.3.211 SSADM structure diagram

Product

diagrammatic representation of structures in terms of sequences, selections and iterations which may be extended to include parallel constructs and operations lists

1.3.212 stage

Structural Model Element

unit of activity with a single goal and rationale

NOTE The SSADM framework requires a project to be subdivided into a number of Modules each of which consists of one or more Stages.

1.3.213 state indicator

Object

attribute used as a re-expression of the structure of an Entity Life History, in a format which will be used during the Logical Design Stage to enforce the defined sequence of events

NOTE A state indicator should be thought of as an additional attribute within each entity.

1.3.214 step

Structural Model Element

subdivision of a Stage with a defined set of products and activities

NOTE The SSADM framework requires a project to be subdivided into a number of Modules each of which consists of one or more Stages, each of which consists of one or more Steps.

1.3.215 success unit

Object

set of processing which has to succeed or fail as a whole within the system

NOTE 1 When a failure occurs the system is restored to the state it was in immediately before the success unit began.

NOTE 2 Also known as logical success unit.

1.3.216 super function

Object

function comprising several lower-level functions

1.3.217 system

General

complete technical output of the project including all technical products (i.e. all hardware, software, documentation, etc.)

NOTE The system will live beyond the life of the project and, in the case of an enhancement project, will have existed before the project.

1.3.218 system description

Product

product showing how the Requirements Specification is met by the Technical Environment Description for a particular Technical System Option

NOTE In many cases the major decisions in this area will have already been taken in choosing a Business System Option.

1.3.219 table

Object (in DBMS)

tabular representation of the relationship between master and detail entities represented by the physical contiguity of the key of the master entity and the keys of each of its detail entities

1.3.220 take-on requirements description

Product

details of the data conversion requirements which have to be implemented before a fully working system can be available compiled during technical system option activities

1.3.221 technical environment description (TED)

Product

details of the specification of the technical environment which is produced once the Technical System Option has been selected

NOTE This detail is then passed on to physical design activities.

1.3.222 technical system option (TSO)

Technique

means by which users agree the new applications implementation strategy incorporating the desired functionality, as defined in the Requirements Specification

NOTE Several Technical System Options are developed and one is selected (or combined elements from several). This gives the technical direction for future development.

1.3.223 technical system options (TSO)

Product

set of Technical System Options which has been developed so that the system development direction can be chosen

NOTE Each option documents the functions to be incorporated and details implementation requirements. Each description is textual with some planning information. Functional elements are taken directly from the Requirements Specification.

1.3.224 technical system options (TSO)

Structural Model Element. Stage 4

stage aiming to take the Requirements Specification and decide on the most appropriate way for development to meet the technical needs

NOTE This Stage has two Steps:
— Define Technical System Options;
— Select Technical System Options.

1.3.225 testing outline

Product

product compiled during technical system option activities which documents the basic requirements for testing the implementable system so that it will meet user requirements

1.3.226 testing timing factors definition

Product

product used to identify the DBMS read and write operations that will be used to implement the types of logical reads and writes used in the logical update and enquiry processes

NOTE Disc and processor performance information is required for all relevant operations.

1.3.227 timing estimation

Product

product used to assess the timing requirements using the data design within a particular implementation environment

NOTE The relevant information will vary wildly depending upon the technical environment and has to be compiled for each application.

1.3.228 training requirements description

Product

product compiled during technical system option activities which documents the amount of training required for the staff who will be using/working on the new system, so that they will be fully effective at the appropriate time (i.e. when the system is available for use)

1.3.229 transient data store

Object

element where transient data is held for a short time before being used by a process and then deleted

NOTE Data held in transient data stores may not be structured in the same way as the data in a main data store. These are elements of a Data Flow Diagram but are not reflected in the Logical Data Model. They are likely to be used to overcome a constraint within the physical system implementation.

1.3.230 universal function model (UFM)

Object

standard model used to identify the general components of an SSADM function

1.3.231 update function

Object

function containing some update processing of the database

NOTE This may also include some enquiry functions.

1.3.232 update process model (UPM)

Product

structure diagram for the update (event) processing and the associated operations list

NOTE This is based on the Effect Correspondence Diagrams which provide a data-oriented view of the system, and the associated Entity Life Histories which provide an event-oriented or process-oriented view of the system.

1.3.233 user catalogue

Product

description of the on-line users of the proposed system, including details of job titles and the tasks undertaken by each of the identified users

1.3.234 user manual requirements description

Product

product compiled during technical system option activities, which documents the basic information which can be supplied from the SSADM products and other available sources

NOTE It relates to the system and its smooth running from the user point of view.

1.3.235 user role

Object

collection of job holders who share a large proportion of common tasks

1.3.236 user role/function matrix

Product

the mapping of correspondences between functions and user roles, thus identifying the dialogues as being the cross-reference between user roles and on-line functions (either enquiry or update)

NOTE 1 Reading down from the user roles axis also provides the (initial) Menu Structure for the system.

NOTE 2 Also known as Function/User Role Matrix.

1.3.237**user roles***Product*

documentation of the details for each user role identified as having a direct interest in the required system

1.3.238**variant (N)***General*

product developed to meet specific requirements which differ from other types of the same product, as well as meeting requirements generally applicable to all types of the same product

NOTE The specification for each variant has to have a unique name. A product may have several variants.

1.3.239**version (A)***General*

subdivision of a product, or variant, relating to the continuing development of that product

NOTE A product, or variant, may have a number of versions.

1.4 Abbreviations

For the purposes of this British Standard the following abbreviations apply.

1NF	first normal form
2NF	second normal form
3NF	third normal form
BSO	business system option (the technique) Business System Options (the product) Selected Business System Option (the product)
CBA	Cost/Benefit Analysis
CCTA	CCTA — The Government Centre for Information Systems
DBA	Database Administrator
DBMS	Database Management System
DD	dialogue design
DFD	Document Flow Diagram
DFM	data flow modelling
EAP	Enquiry Access Path
ECD	Effect Correspondence Diagram
EED	External Entity Description
EEM	entity event modelling
ELH	Entity Life Histories (the product) entity life history analysis (the technique)
EPD	Elementary Process Description
EPM	Enquiry Process Model
FCIM	Function Component Implementation Map

FD	function definition
FS	Feasibility Study
I/O	Input/Output
LDM	Logical Data Model (the product) logical data modelling (the technique)
LDPD	logical database process design
LDS	Logical Data Structure
LGDE	logical grouping of dialogue elements
LS	Logical System Specification
PD	physical design
PDD	physical data design (the technique)
PDI	Process Data Interface
PED	Physical Environment Description
PES	Physical Environment Specification
PID	Project Initiation Document
PPS	Physical Process Specification
PS	Processing Specification
RA	Requirements Analysis
RD	Relationship Degree
RDA	relational data analysis
RS	Requirements Specification
SP	specification prototyping
TED	Technical Environment Description
TSO	technical system option (the technique) Selected Technical System Option (the product) Technical System Options (the product)
UFM	universal function model
UPM	Update Process Model

Section 2. Conformity

2.1 Individual products

To claim compliance with this British Standard, each individual product (as specified in section 3) supplied shall conform to the relevant clauses only, i.e. those relating to the specific product. Upon request, the supplier shall supply a copy of the charts (given in Annex D) appropriately completed and annotated for the agreed range of products.

For any product, conformity shall be stated in one of the following ways.

a) *Conformity by product.* Conformity in respect of delivery of the product specified (at the overall level of the product itself) reference being made to the relevant purpose, composition, completion criteria, status and presentation conventions clauses.

b) *Conformity by specified component.* Conformity in respect of delivery of the product specified, but with specific reference to the composition clause of the product.

NOTE This is to permit agreed delivery of a product which does not incorporate all aspects of the stated composition, but which is deemed to serve its purpose in the specific development context.

c) *Conformity by presentation.* In addition, a product or component can conform in terms of the presentation conventions given in the specification for that particular product.

2.2 Marking

Each individual product supplied shall be labelled or supplied with documentation stating the following:

- a) the number and date of this British Standard, i.e. BS 7738-1:1995²⁾;
- b) means of identifying the supplier.

²⁾ Marking BS 7738-1:1995 on or in relation to a product represents a manufacturer's declaration of conformity, i.e. a claim by or on behalf of the manufacturer that the product meets the requirements of the standard. The accuracy of the claim is solely the claimant's responsibility. Such a declaration is not to be confused with third party certification of conformity, which may also be desirable.

Section 3. Detailed product specifications

NOTE See C.2 for an explanation of the conventions used in this section.

3.1 Business System Options products

COMMENTARY AND RECOMMENDATIONS ON 3.1. A *Business System Option (BSO)* is a description of a system, its boundary, inputs, outputs and the transformations taking place within it. It is not concerned with how the transformations are effected, except where the mechanisms for carrying them out impact the transformations themselves.

Following the establishment of the Requirements Catalogue, and preceding the Requirements Specification Module, BSO products are presented to users to enable them to decide upon the functionality of the required system.

Each specific option comprises sets of features designed to satisfy Requirements Catalogue entries, and to assist the evaluation of the impacts and benefits of the possible new system upon the organizational context.

3.1.1 Business System Options

3.1.1.1 Purpose

A set of Business System Options shall encompass a number of possible solutions, each of which shall satisfy the user requirements to a greater or lesser extent. Each option shall present a high-level system design, and be developed and evaluated from the business aspects.

Each BSO shall contain descriptions of the functional areas to be included in the system (i.e. encompassed by the system boundary), the requirements to be addressed and the potential impact on the organization.

NOTE 1 There may be only one BSO.

NOTE 2 Presentation and consideration of BSOs provides users with the opportunity to assess alternative scenarios for meeting selections among required functions. It also allows users to decide to abandon the project, having decided that requirements cannot be adequately satisfied within the pre-defined constraints.

3.1.1.2 Composition

Business System Options shall be textual documents, which may be supported by Data Flow Diagrams and a Logical Data Structure, as follows:

- a) heading:
 - Option name and/or identifier;
- b) detailed documentation describing:
 - functionality (system) boundary;
 - levels of functionality (of the entire application and its components);
 - other technical considerations such as operating constraints;
 - Cost/Benefit Analysis;

- Impact Analysis;
- training needs.

3.1.1.3 Completion criteria

The following shall be considered for any example:

- a) for each:
 - 1) Is the scope of the system described consistent with the constraints defined in the project initiation document, the Feasibility Report, or both?
 - 2) Are the minimum requirements met by this option?
 - 3) Is the proposed system (option) feasible?
 - 4) Will the proposed option meet the installation standards for data, processing, communications, etc.?
 - 5) Is the format of the option in keeping with installation standards?
- b) for the set:
 - 1) Are all options documented here?
 - 2) Does the set of options cover all business requirements?

3.1.1.4 Status

The Business System Option shall be an interim product (completed); an interim product produced during Stage 2 (RA), but options shall be presented to users, and the selected BSO will be an end-stage product (RA).

COMMENTARY AND RECOMMENDATIONS ON 3.1.1. For any proposed system a number, or set, of Business System Options can be compiled and presented to users in order that a specific option be selected. Alternatively it may be decided by the users that no option embodies a feasible basis for development and that the project be abandoned.

Derivation:

Requirements Analysis:

- Current Services Description;
- Feasibility Report (if it exists);
- project initiation document;
- Requirements Catalogue;
- User Catalogue.

Where used:

- Business System Options: RA.

Presentation conventions are not applicable.

3.1.2 Selected Business System Option

3.1.2.1 Purpose

The Selected Business System Option shall document the description of the system functionality of the chosen option, so supporting the further development of the Requirements Specification.

3.1.2.2 Composition

The Selected Business System Option shall be essentially a textual document, which may be supported by Data Flow Diagrams and a Logical Data Structure.

Detailed documentation shall describe the following:

- functionality (system) boundary;
- levels of functionality (of system and components);
- technical considerations (e.g. operating constraints);
- Cost/Benefit Analysis;
- Impact Analysis;
- selection reasoning (documenting reasons for choosing this option, and for not selecting other available options).

3.1.2.3 Completion criteria

The following shall be considered for any example.

- a) Is the scope of the system described consistent with the requirements defined in the Project Initiation Document?
- b) Has one and only one option been selected? (This may incorporate aspects from several of the original options.)
- c) Are the minimum requirements met by this option?
- d) Is the proposed system (option) feasible and possible?
- e) Is the proposed system (option) economically sound and within budget?
- f) Will the proposed option meet the installation standards?
- g) Is the format of the option in keeping with installation standards?
- h) Does the Selected Business Option look as if it will meet the project objectives?
- i) Will the Selected Business Option contribute towards enabling the organization's objectives?

3.1.2.4 Status

The Selected Business System Option shall be delivered complete at the end of Stage 2.

COMMENTARY AND RECOMMENDATIONS ON **3.1.2.** *The Selected Business System Option provides the basis for further development of the Requirements Specification.*

Derivation:

Requirements Specification:

- *Business System Options;*
- *Current Services Description;*
- *Feasibility Report;*
- *project initiation document;*
- *Requirements Catalogue;*
- *User Catalogue.*

Where used:

- *Business System Options: RA;*
- *Data Flow Modelling: RS;*
- *Dialogue Design: RS;*
- *Logical Data Modelling: RS;*
- *Requirements Definition: RS;*
- *Technical Systems Options: LS.*

Presentation conventions are not applicable.

3.1.3 Cost/Benefit Analysis

3.1.3.1 Purpose

The CBA shall present costs of a proposed option for completion of the project which are balanced against benefits, therefore assisting in the decision making process.

3.1.3.2 Composition

The composition shall be as follows:

- a) variant identification (one of):
 - Feasibility Option;
 - Business System Option;
 - Technical System Option;
- b) header information:
 - option name and/or identifier;
- c) details:
 - development costs;
 - operating costs;
 - replaced costs;
 - benefits;
 - tangible benefits (visible/quantifiable);
 - projected cost containment (how proposed system is cheaper than current);
 - intangible benefits (meeting critical requirements).

3.1.3.3 Completion criteria

The following shall be considered for any example.

- a) Is the option type identified as one of feasibility, business system, or technical system?
- b) Is the option name/identifier correct?
- c) Are the stated direct tangible benefits valued correctly? Do they indeed exist?
- d) Are the projected cost containment figures reasonable?
- e) Are the intangible benefits believable?
- f) Has the source for all estimates been clearly identified?
- g) Are the development, operating and replaced costs believable and reasonable?
- h) Are relevant benefits metrics available?

3.1.3.4 Status

NOTE CBAs may be produced for several options.

For a specific selected option the CBA shall be delivered as a complete product at the end of the relevant module as follows:

- Feasibility CBA: FM;
- BSO CBA: RA;
- TSO CBA: LS.

For non-selected options, the CBA shall be an interim product.

COMMENTARY AND RECOMMENDATIONS ON **3.1.3.** *Cost Benefit Analysis offers users a focus for discriminating between different systems options, by providing details of the potential benefits and costs of an option. Such an analysis may be produced for Feasibility Options (Feasibility Module), Business System Options (Requirements Analysis), and/or Technical System Options (Logical System Specification).*

Derivation:

Feasibility Module (Feasibility Options):

- *Feasibility Option;*
- *Impact Analysis.*

Requirements Analysis (Business System Options):

- *(Selected) Business System Option;*
- *Impact Analysis.*

Logical System Specification (Technical System Options):

- *Impact Analysis;*
- *Outline Development Plan;*
- *Technical Environment Description;*
- *(Selected) Technical System Option.*

Where used:

- *Feasibility CBA: FS;*
- *BSO CBA: RA;*
- *TSO CBA: LS.*

Presentation conventions are not applicable.

3.1.4 Feasibility Options

3.1.4.1 Purpose

The set of Feasibility Options shall encompass all the options to be considered, during a feasibility study, in selecting the way ahead. Each alternative shall present a high-level system design, and be evaluated from the following viewpoints:

- a) business (support for business requirements and objectives);
- b) organizational (impact on people and tasks);
- c) technical (IS requirements, viability of development and implementation route);
- d) financial (cost, benefits and risks).

3.1.4.2 Composition

Feasibility Options shall be essentially textual documents, which may be augmented by components of a Logical Data Model and a Data Flow Model, as follows:

- a) heading:
 - option name and/or identifier;
- b) detailed documentation including:
 - 1) textual description of the scope of the information system (clerical and IT aspects) and the requirements to be addressed, possibly supported by models of the major processes, entities and external interfaces of the proposed information system;
 - 2) an overview of the software and hardware configuration required to run the information system option and the technical resources required to develop it;
 - 3) Impact Analysis: an outline assessment of the impact of the option on the organization;
 - 4) an approximate investment/appraisal (Cost/Benefit Analysis) detailing a broad order of cost and financial, non-financial and indirect benefits;
 - 5) resource requirements;
 - 6) outline time scale/plan for each implementation;
 - 7) risks in terms of business, technical, financial and cultural factors;
 - 8) advantages and disadvantages of the options and a conclusion as to whether it is achievable and desirable.

3.1.4.3 Completion criteria

The following shall be considered for any example.

- a) For each: Is the option feasible from all of the following viewpoints:
 - business;
 - organizational;
 - technical;
 - financial?
- b) For the set: Is the set of options complete?

3.1.4.4 Status

The Feasibility Options shall be end-module products; delivered complete at the end of the Feasibility Module.

COMMENTARY AND RECOMMENDATIONS ON **3.1.4.** *The set of Feasibility Options is compiled so that a selection can be made. Each option documents the functions to be incorporated and details implementation requirements. Each description is textual with some planning information.*

Derivation:

Feasibility Module:

Outline Current Environment Description;
Outline Required Environment Description;
Problem Definition Statement;
Requirements Catalogue;
User Catalogue.

Where used:

Feasibility: FS;
Business System Options: FS;
Technical System Options: FS;
Logical Data Modelling: FS;
Data Flow Modelling: FS;
Requirements Definition: FS.

Presentation conventions are not applicable.

3.1.5 Feasibility Report

3.1.5.1 Purpose

The feasibility report shall serve several purposes as follows.

- a) It records management decisions about the options for further development, including whether a proposed information system should be cancelled, re-scoped, split or merged with another.
- b) It forms the basis for any decision to commit resources for a more detailed study.
- c) It provides information for a more detailed study as a record of decisions, assumptions, estimates, user requirements and outline options.

- d) It provides an outline Project Plan for managing a more detailed study.
- e) It records the findings of the study team according to the terms of reference agreed at the start of the study.

3.1.5.2 Composition

The composition shall be as follows:

- a) introduction:
 - reasons for study;
 - terms of reference;
 - study objectives;
 - scope of study;
 - constraints;
 - completion date;
 - consultation;
 - management of study;
- b) executive summary:
 - recommended solution;
 - options considered but rejected;
 - plans for full study;
 - preferred procurement path;
 - implementation plan;
- c) study approach:
 - details of costs and conduct of study;
- d) existing business and IS support to the organization, documenting the current situation in the area of study;
 - business objectives;
 - functions and processes currently undertaken;
 - organization of the business area, with various roles and responsibilities;
 - current and potential areas of strength and weakness;
 - any relationships with other business areas and organizations;
 - any existing IS support detailing functions supported or not, and weaknesses, technological opportunities and constraints;
- e) future IS support required:
 - description of the system's place in the IS strategy;
 - overview of scope of required system and functionality;
 - details expressed in measurable terms for requirements;
 - implications of any geographic distribution of IS support;

- details of service performance required of the proposed system;
- f) proposed system, describing how the requirements can be met:
 - narrative overview of the logical system, based on selected business option(s);
 - outline of the alternative technological options, with summary of the necessary technical framework;
 - advantages and disadvantages of the proposals;
- g) options considered but rejected:
 - described in fashion similar to that for the proposed system, but with less detail;
- h) financial assessment:
 - summary of costs comparing proposed system with options;
 - summary of costs against benefits;
- i) project plan:
 - resource requirements;
 - expected implementation timescale;
 - proposed management structure for development and implementation;
- j) conclusion and recommendations;
- k) annexes:
 - SSADM support documentation.

3.1.5.3 Completion criteria

The following shall be considered for any example.

- a) Does the report conform to installation standards?
- b) Is the proposed solution within the identified budget constraints?
- c) Is the project within the scope set? Can it be continued on this basis?
- d) Do the users agree that their requirements have been addressed?
- e) Is one and only one option selected as the way ahead?
- f) Are all the requirements mutually consistent? If not, have priorities been identified?
- g) Is it an accurate statement of requirements, constraints, and possible future enhancements to the system?

3.1.5.4 Status

The Feasibility Report shall be a completed product, delivered at the end of the Feasibility Module.

COMMENTARY AND RECOMMENDATIONS ON **3.1.5. The Feasibility Report** provides the basis upon which decisions can be made to continue or redefine a proposed IS project.

Derivation:

- FS.

Where used:

- Feasibility: FS;
- Requirements Definition: FS, RA;
- Data Flow Modelling: FS, RA;
- Logical Data Modelling: FS, RA;
- BSO: FS, RS;
- TSO: FM;
- Dialogue Design: RA.

Presentation conventions are not applicable.

3.1.6 Impact Analysis

3.1.6.1 Purpose

Impact Analysis shall explain the effects of the option under consideration on the user environment. This document shall be used to supplement Feasibility Options, Business System Options and Technical System Options.

3.1.6.2 Composition

The composition shall be as follows:

- a) version identification (one of):
 - Feasibility Option;
 - Business System Option;
 - Technical System Option (unselected);
 - Technical Environment Description (selected);
- b) heading information:
 - option name and/or identifier;
- c) details:
 - Take-On Requirements Description;
 - testing outline;
 - Training Requirements Description;
 - User Manual Requirements Description;
 - additional points to be covered including organization and staffing; changes in user operating procedures;
 - implementation considerations including: training requirements, conversion and staff learning curves;
 - savings, in terms of replaced equipment, maintenance, supplies, etc.;
 - comparative impact (advantages and disadvantages) with the other Technical System Options.

NOTE For Feasibility Options and Business System Options there will be less detail available and some of these items may have no associated section.

3.1.6.3 Completion criteria

The following shall be considered for any example.

- a) Is the option type identified correctly (feasibility, business system, technical system)?
- b) Is the option name/identifier correct?
- c) Are the following aspects covered:
 - improved turnover/work rates;
 - ease and speed of implementation;
 - scale of development costs;
 - reliability;
 - projected cost containment;
 - resourcing consequences;
 - staff and equipment;
 - performance and service;
 - limitations;
 - difficulties in implementation and/or increased time scale?

3.1.6.4 Status

Impact Analysis shall be delivered complete for a specified option and type of option; dependent on Stage/Module by option type.

COMMENTARY AND RECOMMENDATIONS ON 3.1.6. Impact Analysis describes the effects of a specified option (feasibility, business or technical) on the user environment and will cover issues concerned with organization, procedures, and implementation factors. This product is used to document the ramifications of selecting a particular option.

Derivation:

- a) Feasibility Module (Feasibility Options):
 - Feasibility Option;
 - Outline Current Environment Description;
 - Outline Required Environment Description;
- b) Requirements Analysis (Business System Options):
 - (Selected) Business System Option;
 - Current Services Description;
 - Requirements Catalogue;
 - User Catalogue.
- c) Logical System Specification (Technical System Options):
 - Requirements Specification;
 - (Selected) Technical System Option;
 - Technical Environment Description.

Where used:

- Business System Options: RA;
- Technical System Options: LS;
- Feasibility: FS;

— Physical Design: PD.

Presentation conventions are not applicable.

3.2 Data Flow Modelling products

COMMENTARY AND RECOMMENDATIONS ON 3.2. Data Flow Modelling is used to develop a hierarchical decomposition of a specific system into identified data sources and receivers, functions, data stores and data flows between them. The techniques employed have the following objectives:

- a) define the system context in terms of its boundary, scope and environment;
- b) identify the external entities that supply data to the system and/or receive data output by the system;
- c) identify data flows across the system boundary;
- d) identify data stores within the system boundary;
- e) establish the main functions, or groups of functions, to be performed by the system.

The technique plays a crucial part in the development of the Requirements Definition of the system under development. The Data Flow Model, being hierarchical, moves from a top-level user-oriented view to a more detailed view at the lower levels. These lower levels, consisting of descriptions of elementary processes, provide the bases for products completed in the Logical Systems Specification Module.

Derivation:

For many of these products there might be three variants: Current Physical, Logical and Required System. These will be produced in the order listed. Consequently the derivation of later variants will depend on the production of earlier ones if produced.

3.2.1 Context Diagram

3.2.1.1 Purpose

The Context Diagram shall provide an overall view of the external data flows to and from the system and also the external entities that supply data to and/or receive data from the system within the boundary. The diagram shall be drawn by the analyst using data flow modelling notation to represent the context in which the system operates.

3.2.1.2 Composition

The composition shall be as follows:

- the system (a single process);
- multiple external entities;
- links (data flows) between external entities and processes.

3.2.1.3 Completion criteria

The following shall be considered for any example.

- Is the diagram syntactically correct and semantically adequate?
- Is the scope of the system clearly shown?
- Are all known external entities represented?
- Is there one and only one “process” box, representing the system?

3.2.1.4 Status

The Context Diagram shall be an evolving product if produced during Feasibility. It shall be a completed, end-stage product delivered at the end of Stage 1 (RA).

3.2.1.5 Presentation conventions

The components of Context Diagrams shall be in accordance with Figure 2.

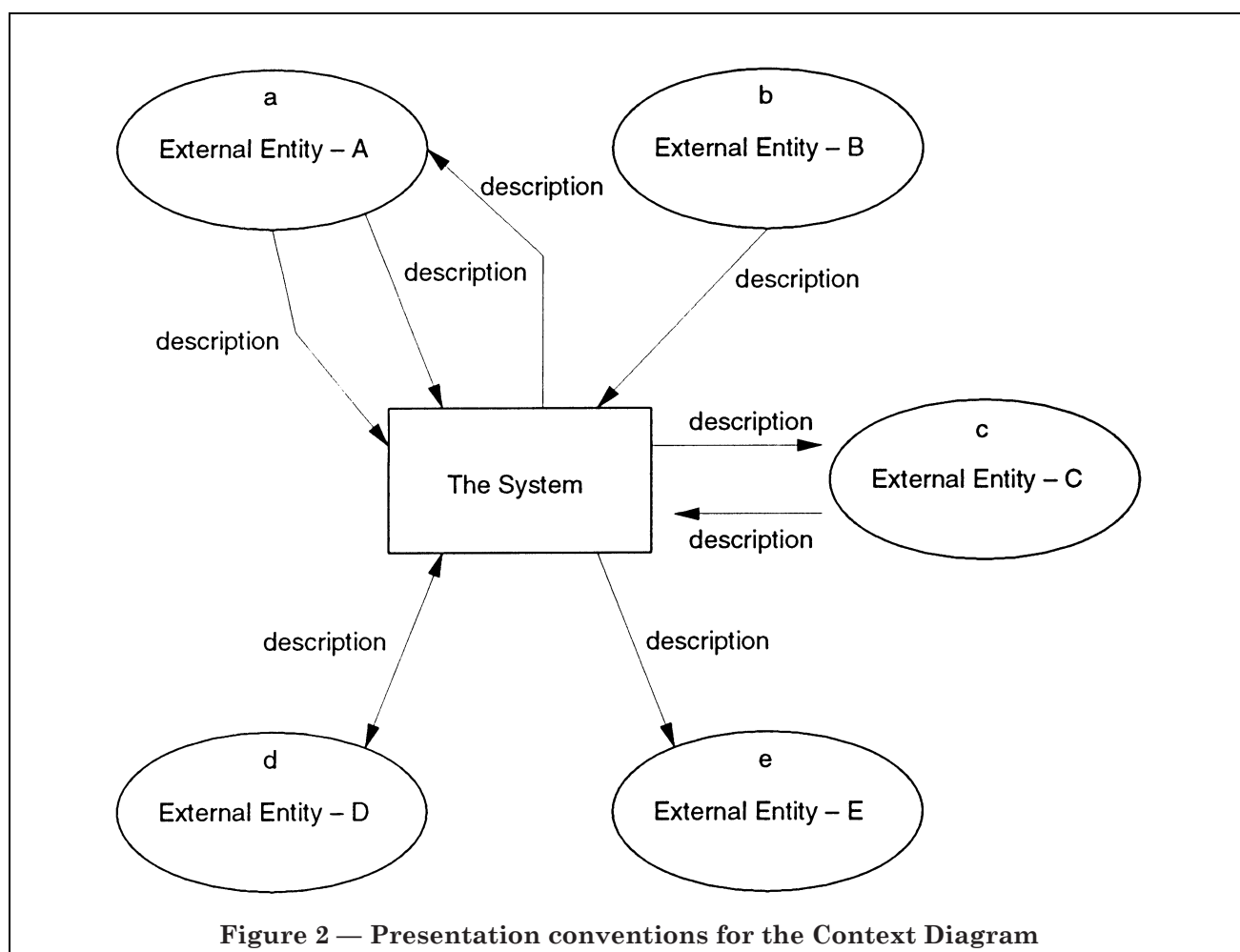
COMMENTARY AND RECOMMENDATIONS ON 3.2.1. *The Context Diagram is drawn to illustrate the initial scope of the proposed system. The diagram concentrates on the major inputs and outputs of the system and shows the external sources and recipients of system data.*

Derivation:

- Feasibility:*
 - discussion with users;
 - existing system documentation;
- Requirements Analysis:*
 - discussion with users;
 - existing system documentation;
 - Context Diagram (if produced during Feasibility Study).

Where used

- Feasibility;
- Data Flow Modelling: RA.



3.2.2 Data Flow Diagram: Level 1

3.2.2.1 Purpose

The Level 1 Data Flow Diagram shall serve to contain the overall information flow model of the system.

NOTE It will be revised during the lifetime of the system to reflect current, logical or required functional areas.

There shall be three possible variants of the Data Flow Diagram: Level 1: Current System, Logical and Required System.

The Level 1 Data Flow Diagram of the Current System shall be drawn by the analyst to describe the scope of the system under investigation.

The Level 1 Diagram of the Logical Data Flow Model shall be built from the bottom up, by grouping bottom-level processes into more abstract units and drawing the data flows that represent information flowing at the lower level.

The Level 1 Diagram of the Required System Data Flow Model shall be reached when processes map on to a user's view of the functions that the system is intended to support.

3.2.2.2 Composition

The composition shall be as follows:

- a) variant identifier (one of):
 - Current Physical;
 - Logical;
 - Required System;
- b) diagram elements representing:
 - processes;
 - data flows;
 - data stores;
 - external entities;
 - resource flows;
 - resource stores.

3.2.2.3 Completion criteria

The following shall be considered for all variants.

- a) Is the variant identifier properly completed?
- b) Are agreed notational conventions correctly applied?
- c) Is the boundary of the system clear?
- d) Are all user-perceived functional areas represented?
- e) Are meaningful names used for processes and data stores?
- f) Are all identifiers unique?
- g) Do external entity names accurately reflect the environment outside the system?
- h) Does the diagram accurately reflect the physical, logical or required system in terms of external entities, data stores and data flows?

- i) Does the diagram avoid giving an inappropriate level of detail, such as sequencing or detailed processing logic?

The following shall be considered for the Logical Data Flow Diagram.

- 1) Are all physical aspects of the current system removed, unless they are constraints on the requirement?
- 2) Are any enquiries remaining after logicalization major ones?
- 3) Are all external entities identified as potential users of the system?

It shall be considered for the Required System Data Flow Diagram if all the facilities defined by the Selected Business System Option, and nothing else, are modelled in the Required System Data Flow Diagrams.

3.2.2.4 Status

There shall be three possible variants of this product: Current Physical, Logical and Required System.

Where the Current Physical variant is started during Feasibility, it shall then be delivered as an evolving end-module product. It shall then be updated in Stage 1 and delivered complete as an interim product during Stage 1.

The Logical variant shall be produced during Stage 1, and shall be delivered as an end-stage product on completion of Stage 1.

The Required System variant shall be developed during Stage 3 as an interim product.

3.2.2.5 Presentation conventions

The components of the Level 1 Data Flow Diagram shall be in accordance with Figure 3 to Figure 8.

COMMENTARY AND RECOMMENDATIONS ON 3.2.2. *The Level 1 diagram illustrates the main functional areas contained within the scope of the system. It should be a simple diagram readily understood by users, so that it can act as an effective means of communication between analysts and users. (See also 3.2.4.)*

Derivation:

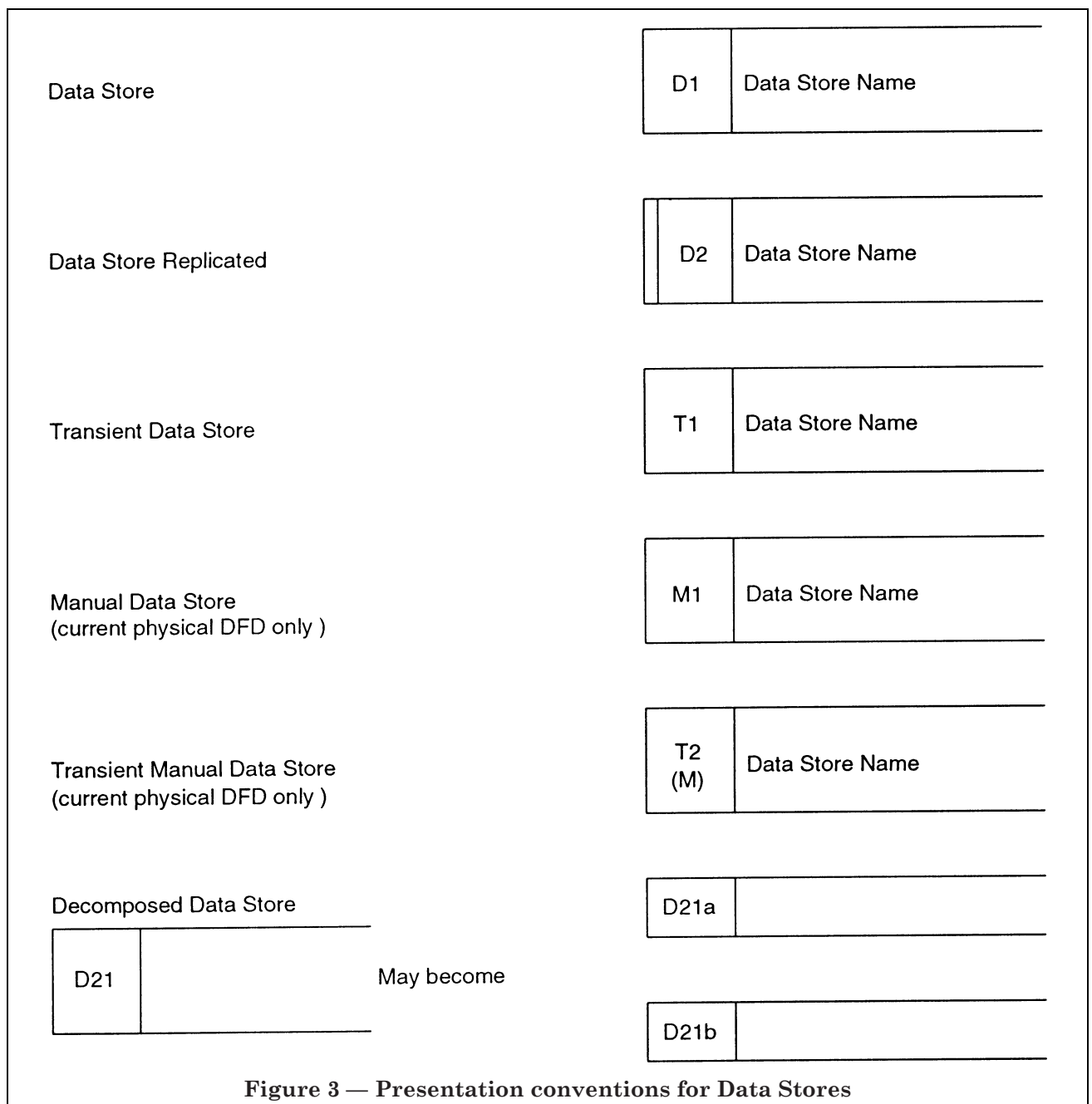
- a) *Feasibility (for Current Physical Data Flow Diagram Level 1):*
 - existing system documentation;
 - project initiation document;
- b) *Requirements Analysis (for Current Physical Data Flow Diagram Level 1):*
 - Context Diagram;
 - Current Physical Data Flow Diagram Level 1;
 - Document Flow Diagram;

- *existing system documentation;*
- *Feasibility Report (if it exists);*
- *Resource Flow Diagram.*
- c) *Requirements Analysis (for Logical Data Flow Diagram Level 1):*
 - *Current Physical Data Flow Diagram Level 1.*
- d) *Requirements Specification (for Required System Data Flow Diagram Level 1):*
 - *Selected Business System Option;*

— *Logical Data Flow Diagram Level 1.*

Where used:

- *Data Flow Modelling (all three variants);*
- *Feasibility: Current Physical;*
- *Logical Data Modelling (all three variants);*
- *Business System Options: Logical, Required variants;*
- *Function Definition: Required;*
- *Entity-event Modelling: Required.*



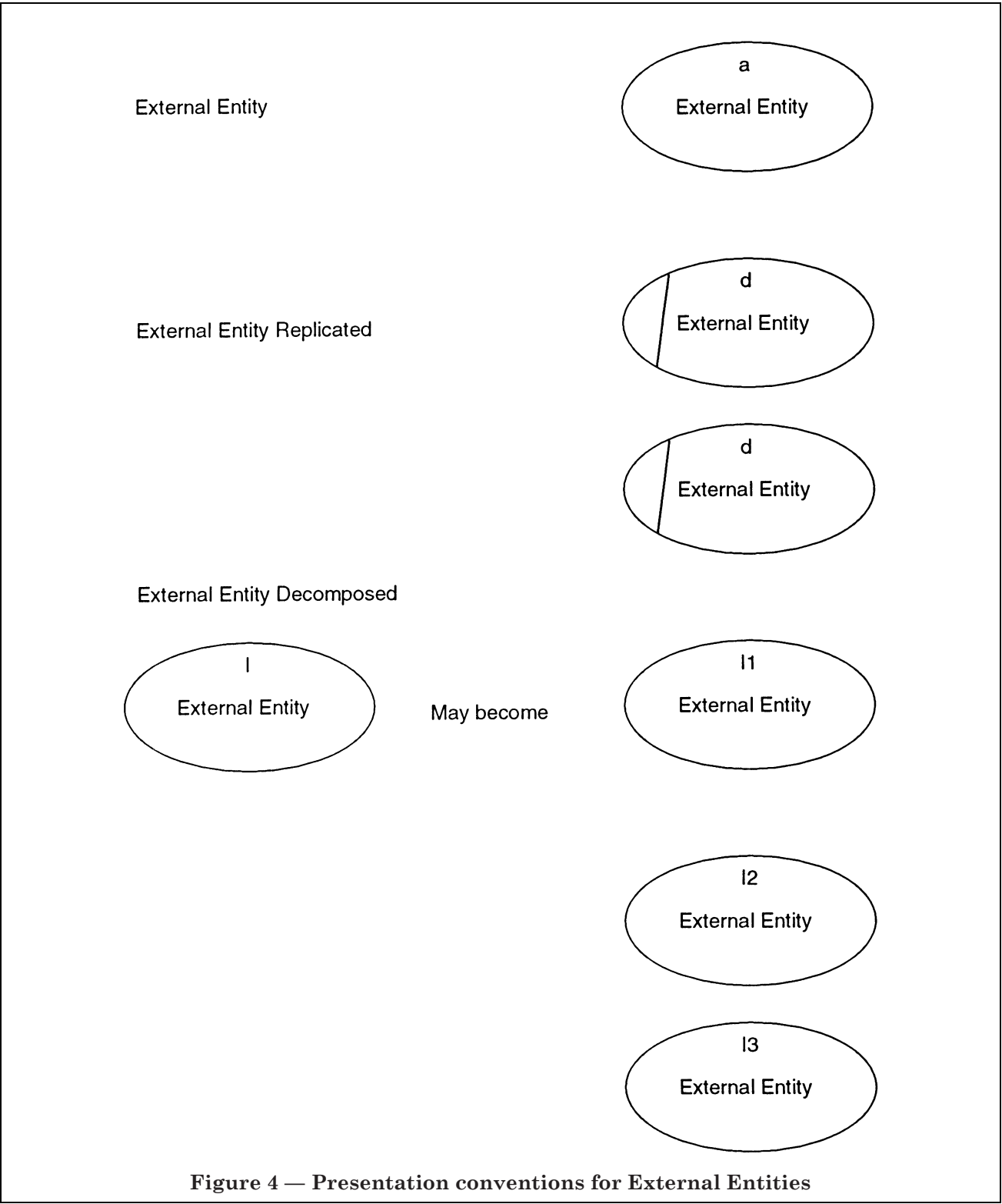


Figure 4 — Presentation conventions for External Entities

