==
MESSAGE IDENTIFIER: ISO/IEC JTC1/SC7/Secretariat/1996-61
==

ISO/IEC JTC1/SC7 N1498
1996-04-14

TITLE: Conceptual Model and Taxonomy for Information Systems Engineering

SOURCE: Commission of the European Communities

WORK ITEM:

STATUS: Final Report

REFERENCES: Resolution 427

ACTION: For discussion at the next SC7 Plenary in Prague.
Formal comments required no later than 1996-10-15.

Francois Coallier
Jean-Normand Drouin
ISO/IEC JTC1/SC7 Secretariat

Attached is an uuencoded pkzipped MS Word for Windows 6.0 (.doc) file in letter forma
t.
To convert to A4 in Word, go in the "File" menu, "Page Setup" sub-menu and select the "Paper Size" tab. Page numbers may change. The TOC and LOF can be updated by placin
g
the cursor in any of the two and pressing F9.

Size:	uuencoded	-	165 298 bytes
	zipped	-	117 951 bytes
	native	-	427 520 bytes

number of pages: 103

Conceptual Model and Taxonomy for Information Systems Engineering

Final Report of Project Team
CEN/TC 311 PT01

April 1995

"The first countries to enter the information society will reap the greatest rewards. They will set the agenda for those who must follow. By contrast, countries which temporise, or favour half-hearted solutions, could, in less than a decade, face disastrous declines in investment and squeeze on jobs."

'Europe and the Global Information Society'
The Bangemann Report

Foreword

Recognising there was a problem Europe needed to address, the Commission of the European Communities mandated CEN[1] to set up a project team to investigate Europe's requirements for standards in the field of Information Systems Engineering (ISE). In its approved final report (May 1991), the project team identified urgent reasons for ISE standardisation in Europe relating to the process of creating a single market in countries with diverse languages, cultures, business practices and system engineering methods. Although components of ISE standardisation were being addressed by various national and international standards bodies, nobody was looking at ISE as a whole.

This report led to the creation of CEN/BT WG63 to define the way forward. In its final report, WG63 proposed that a CEN Technical Committee be set up to undertake the ISE standardisation programme. As a result, CEN/TC 311 was established in June 1993.

CEN/TC 311's scope is standardisation in the field of ISE. Its role is to support Europe's business, economic, political, cultural and legislative needs by identifying the role of and need for existing or new agreed standards in the field of ISE and encouraging and enabling their preparation and application. The use of good ISE standards in Europe will:

- contribute to the removal of barriers to trade and overcome language and cultural barriers, enabling organisations to compete on equal terms throughout Europe;
- support the establishment of information systems needed to implement the single European market;
- provide long term economic benefits as European companies influence the development of products world-wide;
- reduce the risks associated with ISE products, with benefits to both the acquirers and the providers of Europe's information systems;
- contribute to the efficiency of ISE in Europe in order to increase Europe's competitiveness in the global market.

This document addresses the terms of the CEC mandate SOGITS N695.2 SOGT 93/45.2:

> *"to produce a conceptual model and corresponding taxonomy to ensure the coherence and completeness of standards work in the ISE area, taking into account the user requirements."*

[1] Ref. BC-IT-014SI

The document was produced by project team CEN/TC 311 PT01 which had the following membership:

Team Leader:

Jenny Thornton	The Bedwyn Consultancy Ltd UK

Editor:

Marty Sanders	Catalyst Software Ltd Ireland

Core Team:

Annie Combelles	Objectif Technologie France
Stefano Nocentini	IBM South Europe, Middle East and Africa Italy
Friðrik Sigurðsson	TölvuMyndir hf. Iceland

Support Team:

Mogens Brammer	Herlev Hospital Denmark
Hugh Davis	ICL UK
Jacques Hagelstein	Sema Group Benelux Belgium
Veikko Hyytiäinen	TIEKE Finland
Jens Kaasbøll	University of Oslo Norway
Antoni Michalski	Ministry of Finance Poland
Brian Millis	Independent Consultant UK
Freddie Vogelius	Dansk Standard Denmark
Brian Wichmann	National Physical Laboratory UK

1. Table of Contents

2. Figures

1. Scope and field of application

1.1 Scope

The scope of this document is information systems engineering, approached from the perspective of a description of the concepts of ISE and the classification of those concepts in a taxonomy.

Information systems engineering concerns the provision of information systems to organisations, including both the provision of development services and the provision of operational services.

1.2 Audience

The audience for the document includes:

- IS procurers and users;
- ISE practitioners;
- ISE researchers and educators;
- CEN/TC 311;
- CEN/BT S7;
- SOGITS and its associated committees;
- CEN member bodies;
- Other interested standards makers;
- Euromethod developers and users.

1.3 Rationale and purpose

The wide scope of ISE standardisation and the number of standards organisations working in the field make the job of co-ordinating ISE standards development a challenging one. In these circumstances it is important to have agreed reference documents, including carefully defined terminology. Such reference documents are variously called conceptual models, reference models, frameworks and architectures. They are intended primarily to be used by standards-makers so that they can communicate their ideas, test the overlap and cohesion of their work and establish consensus across the industry.

The purpose of the *'Conceptual Model and Taxonomy for Information Systems Engineering'* is to provide a support tool for activities such as:

- promoting a common understanding of ISE, its concepts and its terminology throughout Europe;
- promoting understanding of the scope and objectives of ISE standardisation throughout Europe;
- facilitating co-operative working in the field of ISE between different organisations in Europe and worldwide;
- promoting the European view of ISE to the rest of the world;
- facilitating the translation of user requirements to areas of ISE standardisation and certification;
- identifying areas of standards and profiles needs, gaps and overlaps;
- assisting CEN/TC 311 in the exercise of its mission and the fulfilment of its objectives.

The document has other potential uses, for example:

- assisting those standards bodies which monitor the work of CEN/TC 311;

- identifying areas of R&D project need, gaps, relationships and overlaps.

The *'Conceptual Model and Taxonomy for Information Systems Engineering'* will benefit Europe's ISE providers and users by helping CEN/TC 311 to encourage and enable the provision of consistent, coherent and complete standards that meet Europe's business, economic, political, cultural and legislative needs.

1.3.1 Structure

Sections 1 to 5 contain introductory material. The body of the document is divided into two parts:

Part 1 The ISE conceptual model and taxonomy

Essential reading for anyone using the document for reference purposes.

Section 6 Theoretical foundation of the ISE conceptual model

The theoretical foundation on which the ISE conceptual model and taxonomy is based.

Section 7 The conceptual model of ISE

The ISE conceptual model defined and described.

Section 8 The ISE taxonomy

ISE taxonomies defined and described.

Part 2 Using the ISE conceptual model and taxonomy

Useful reading for anyone applying the ISE conceptual model and taxonomy in standards-related and other tasks.

Section 9 Using the language

Using the ISE conceptual model and taxonomy as a language to facilitate communication.

Section 10 Using the pictures

Using the pictures from the ISE conceptual model for scoping and positioning tasks.

Section 11 Using the taxonomies

Using the ISE taxonomies for classification and comparison tasks.

Section 12 Supporting standards management

Using the ISE conceptual model and taxonomy in tasks relating to the management of standards and standardisation.

The annexes contain background material about ISE, examples of user requirements, a list of some standards committees working in the ISE domain and a bibliography.

1.4 Summary

This section summarises the content of Parts 1 and 2 of the document.

1.4.1 The ISE conceptual model and taxonomy

To produce an ISE conceptual model adequate for the purposes listed in section 1.3, four systems have to be considered:

- the information system;
- the organisation system in which the information system is used;
- the ISE system which develops the information system;
- the ISE technology system which develops the technology used by the ISE system.

The four systems in their market context are connected in a cycle of needs and satisfaction of needs (the top level of the ISE conceptual model—see Figure 1-1). The satisfaction of needs is achieved by products and services which are qualified by attributes or qualities. Important qualities from the point of view of ISE are those of the target information system, for example, its dependability, its efficiency and its testability. Such qualities, seen as requirements arising from the organisation system and its market context, affect the way in which information systems are engineered. The way information systems are engineered in turn places requirements on the ISE technology system to provide appropriate products and services to support that engineering.

The ISE taxonomies—the classification of ISE concepts—are derived from the conceptual model. The taxonomies are another way of stating the conceptual model in terms of the relationships ‘is a kind of’ (specialisation) and ‘consists of’ (containment). Stated informally, examples from the taxonomy of ISE entities are:

- an ISE system consists of a production sub-system and a control sub-system;
- a production sub-system consists of production products, production processes and production resources;
- a formalism is a kind of ISE technology product.

This kind of taxonomy can be used in standards management activities involving the classification and comparing of standards.

A second taxonomy, based on the qualities of ISE entities, is useful for classifying such things as information systems.

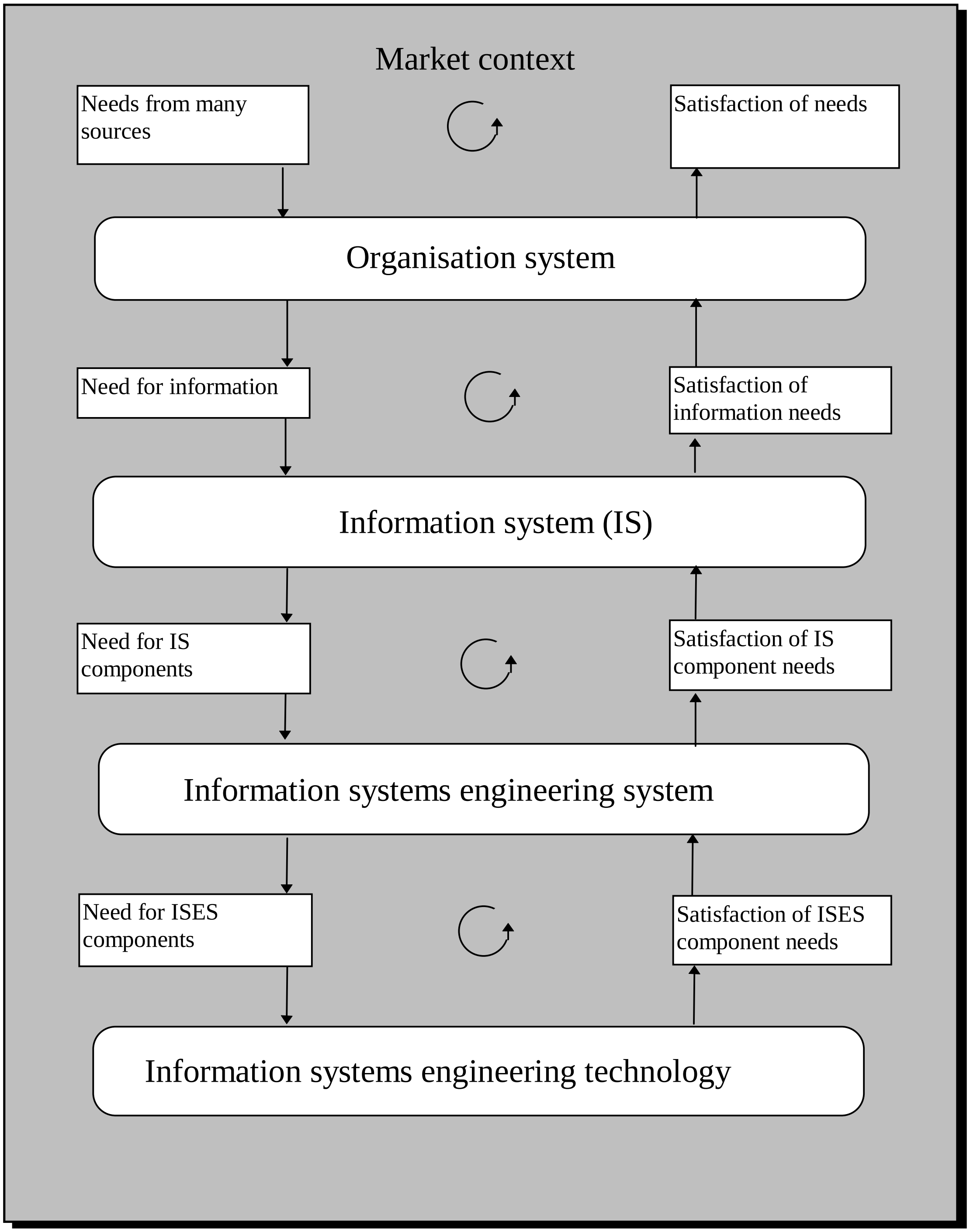


Figure 1-1 The cycle of needs

1.4.2 Using the ISE conceptual model and taxonomy

There are three main kinds of use to be made of the ISE conceptual model and taxonomy:

1 **A language to facilitate communication:** This involves utilising the framework, concepts and terminology of the ISE conceptual model in discussions relating to ISE. Here the ISE conceptual model and taxonomy provides a common language and shared world view or understanding that will facilitate communication, debate and discussion of issues. Examples of this kind of use are in:
- promoting a common understanding of ISE, its concepts and its terminology throughout Europe;
- promoting understanding of the scope and objectives of ISE standardisation throughout Europe;
- facilitating co-operative working in the field of ISE between different organisations in Europe and worldwide;
- promoting the European view of ISE to the rest of the world.

2 **Pictures to facilitate elucidation of relationships:** A different kind of use concerns the various kinds of pictures that can be constructed from the text of the conceptual model. It is possible to position research projects, standardisation work items, models, frameworks, etc. in such pictures and thereby shed light on the scope and relationships of the items. A picture used for positioning need not be formal and the positioning process will not be rigorous. The picture provides a catalyst to discussion, not an end in itself. Examples of this kind of use are in:

- facilitating the translation of user requirements to areas of ISE standardisation and certification;
- identifying areas of standards and profiles needs, gaps and overlaps.

3 **Matrices or models to facilitate formal comparisons:** The more formal use concerns the construction of matrices from the ISE taxonomies and placing items for comparison etc. in the cells of the matrices. Another formal use involves the construction of a model of, for example, a standard, and comparing it for consistency and divergence with an equivalent model developed from the ISE conceptual model. Examples of this kind of use are in:

- facilitating the translation of user requirements to areas of ISE standardisation and certification;
- identifying areas of standards and profiles needs, gaps and overlaps.

Figure 1-2 shows how a picture taken from the conceptual model can be used to make a preliminary positioning of standards in the ISE domain (see item 2 above).

Standards work positioned in Figure 1-2 are:

1	ISO 9000/EN 29000
2	EDI
3	OSI
4	ETSI
5	IRDS
6	Security
7	SPICE
8	POSIX
9	PCTE
10	Euromethod
11	Guidelines for selection of CASE tools (ISO/IEC JTC1 SC7)
12	EPHOS
13	Quality of service
14	DISC Framework for User Requirements

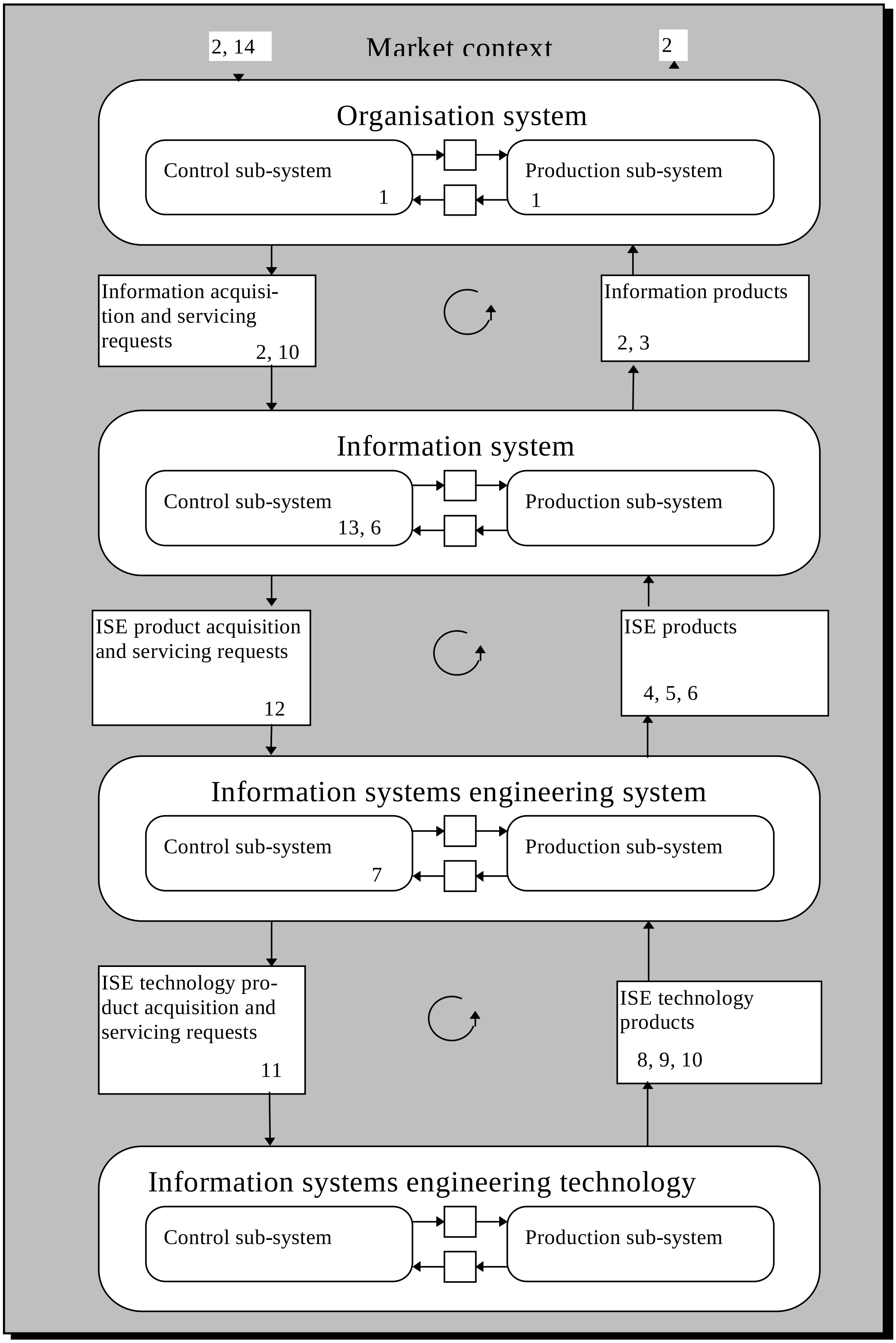


Figure 1-2 Positioning other standards work in the ISE domain

2. Normative references

No normative references are relevant to this document.

3. Definitions

For the purposes of this document, the following definitions apply. Unless otherwise noted, the definitions are specific to this document.

3.1 acquirer: Obtainer of a product or service.

3.2 acquisition: The process of obtaining, by purchase or by in-house development, a product or service to meet an identified need (where purchase is involved, sometimes called *procurement*).

3.3 activity: Element of a process, performed by an actor when attempting to effect a change of some state.

3.4 actor

1) A person (animate) or a device (inanimate) capable of acting and thereby changing states of perceived things.

2) A resource (person, computer system or group thereof) performing within the organisation system, information system, information systems engineering system or ISE technology system or in relationship with one or more of those systems. [Based on Euromethod]

3.5 analysis: The process of translating an information problem or need into a statement of requirements sufficient for the design of a solution to the problem or need.

3.6 area: A conceived part of the world such that for any conceived thing it may be decided whether it does or does not belong to that conceived part of the world.

3.7 assessment: The process of judging any thing in respect of an established norm, involving testing, measuring, evaluating, correcting and improving.

3.8 CASE tool: a sub-set of information systems engineering technology products (both Computer Aided Software Engineering tool and Computer Aided System Engineering tool).

3.9 computer: A number of hardware and software components with the systemic property of being capable of rendering data storage and processing services.

3.10 computational sub-system: A sub-system having a computer processor as its main actor.

3.11 concept: A general idea of something formed by mentally combining all specific parts and characteristic features; an abstract notion, a theoretical construct. [Collins English Dictionary]

3.12 conceptual model

1) A consistent collection of sentences expressing the necessary propositions that hold for a universe of discourse. [ISO TR 9007]

2) An abstract, stable, maintained architecture for a domain, which positions the domain in its context, delimits the boundaries of the domain, identifies the elements of the domain, specifies their principal relationships and attributes and categorises the elements into classes. [ISO/IEC JTC1/SC7]

3) A purposely abstracted, unambiguously expressed, description of part of the world, expressed by sentences in defined languages.

3.13 construction: The process of translating a specification into items ready for installation. [Based on Euromethod]

3.14 control sub-system: The sub-system which controls (directs, assesses, supports) the resources, products and processes of a system.

3.15 customer: Organisation that recognises the need for an information product or service, ISE product or service or ISE technology product or service and decides to purchase it from or contract it to a supplier. [Based on Euromethod]

3.16 data: A re-interpretable representation of information or observations in a formalised manner suitable for communication, interpretation or processing by automatic means. [Based on ISO 2382-1]

3.17 deliverable: Product in a certain state exchanged between a provider and an acquirer.

3.18 design: The process of translating a statement of requirements into a specification of design sufficient for the construction of a product to realise that design.

3.19 development: A work area aiming at developing (creating, adapting, extending, evolving) products and services.

3.20 **direction:** The process of guiding a process or sub-system to achieve a specified goal or purpose, involving planning, instructing, monitoring and adjusting.

3.21 **element:** A thing belonging to a specific area (sometimes called component).

3.22 **engineering:** The systematic, disciplined application of knowledge, methods and experience to the production of a complex product.

3.23 **environment:** A complete set of IT resources (tools, platforms and additional infrastructure) available to one or more people or work groups in an ISE system.

3.24 **event:** Something that happens (the time of happening might or might not be known in advance); an important fact; the result of a process, activity or task. [Based on Harrap's French Dictionary]

3.25 **formalism:** A language or other notation, such as a natural language, formal language, mark-up language, graphical language or diagramming notation.

3.26 **function:** A set of actions performed by a system or by a device in pursuit of a purpose or goal.

3.27 **information**

1) Knowledge concerning objects, such as facts, events, things, processes or ideas, including concepts, that within a certain context has a particular meaning. [ISO 2382-1]

2) Knowledge available from a person's memory and/or perception of the world and/or extraction from records and/or reception of a message, such that a rational decision can be made in connection with some (contemplated) action.

3.28 **information product**: An element of an organisation system; deliverable of an information system; data or information.

3.29 **information system**

1) An information processing system, together with associated organisational resources such as human, technical and financial resources, that provides and distributes information. [ISO 2382-1]

2) Aspect of an organisation which provides, uses and distributes information. It is considered to include the associated organisational resources such as human, technical and financial resources. It is thus effectively a human system, possibly containing a computer system that automates selected elements of the information system. [Euromethod]

3) A system considered in terms of the information flows across its boundaries and between its elements and of the information it must contain in order to exhibit the behaviour required of it and fulfil its organisational objective. [Defence Research Agency, UK]

4) A conception of how the information-oriented aspects of an organisation are composed (actors, resources, etc.) and how these operate, thus describing the (explicit and/or implicit) information-providing arrangements existing within that organisation. Information systems support information requirements and communication fulfilments within organisations.

3.30 **information system adaptation:** Any kind of modification (correction, enhancement, improvement, etc.) and automation of an information system to fulfil the ever changing needs of an organisation. [Euromethod]

3.31 **information system development:** A process aiming at changing information systems. Information system development may include problem analysis, business analysis, system requirements analysis, system architectural design, software detailed design, software coding, testing, installation, writing materials, training of actors, etc. [Based on Euromethod]

3.32 **information systems engineering:** The systematic, disciplined, application of knowledge, methods and experience to the provision and support of information systems, bridging strategic goals/requirements and operational tools. [CEN/TC 311]

3.33 **information systems engineering product:** An element of an information system; deliverable of an information systems engineering system.

3.34 **information systems engineering system:** The system which carries out the provision (development and support) of an information system.

3.35 **information systems engineering technology product:** An element of an ISE system; deliverable of an ISE technology system.

3.36 information systems engineering technology system: A specialised instance of an ISE system which serves to provide ISE systems and their components.

3.37 installation: The process of making a system operational within a target domain. [Euromethod]

3.38 interoperation: The activity whereby two or more processes or systems use the same product, data or information (sometimes called interworking).

3.39 language: A set of conventions for expressing conceptions.

3.40 method: Formalised knowledge that can be communicated in a reproducible way and which allows one or more people to organise work to achieve a certain class of goals within a certain class of problem situations. A method contains usually concepts, one or more languages, assumptions, rules, heuristics, procedures, guidelines, subsidiary methods and techniques. A method describes a way to conduct a process. [Euromethod]

3.41 metric: A system or standard of measurement; a measure. [OED]

3.42 observation: The manifested result of a measurement or a calculation based on data and observations or an act of informing.

3.43 organisation: A grouping of actors, together with a collection of resources, such that (a) a common goal is pursued or some other characteristic coherence is displayed, and (b) interactions occur that are based on communication. A social system where action is performed within the frame or more or less well established norms or rules of behaviour, having the systemic property of being able to act as one whole.

3.44 organisation system

1) Human system, i.e. structured set of people possibly using machines (including computers), co-ordinating their efforts towards a common goal. [Euromethod]

2) A system (or group of systems and users) consisting of a set of actors fulfilling roles by participating in actions in order to meet its objective. [Defence Research Agency, UK]

3) A conception of how an organisation is composed (actors, resources, etc.) and how it operates (goals, business rules, interactions with its environment, etc.).

3.45 platform: A set of services implemented in hardware and software, including operating system and communications services.

3.46 process

1) Transformation of a system aiming at some goal, ideally contributing to the overall goal of the system. [Euromethod]

2) A predetermined course of events defined by its purpose or by its effect, achieved under given conditions. [ISO 2382-1]

3.47 product: The result of processes. [Euromethod]

3.48 production sub-system: The sub-system which performs deliverable development processes under the control of a control system. [Based on Euromethod]

3.49 provider: System which delivers a required product or service to an acquiring system (sometimes called a *supplier).*

3.50 quality

1) The totality of characteristics of an entity that bear on its ability to satisfy stated and implied needs. [ISO 8402-91]

2) A distinguishing aspect of an element associated with the nature of how and why it belongs to a particular area (sometimes called an a*ttribute, characteristic, property* or *feature*).

3.51 quality assurance: All the planned and systematic activities implemented with the quality system and demonstrated as needed to provide adequate confidence that an entity will fulfil requirements for quality. [ISO 8402-91]

3.52 quality management: All activities of the overall management function that determine the quality policy, objectives and responsibilities and implement them by means such as quality planning, quality control, quality assurance and quality improvement within the quality system. [ISO 8402-91]

3.53 repository: A database specialised for an ISE system, where ISE information for an IS development is stored and dynamically accessed by co-operating actors.

3.54 **representation space:** A multi-dimensional space where the dimensions represent categories and where instances of entities can be positioned as single points at the places in the space where their categorisations intersect.

3.55 **requirement:** Essential condition that a system has to satisfy. [Euromethod]

3.56 **resource:** A thing essential to the execution of a process.

3.57 **role:** A set of actions performed by a human actor in pursuit of a purpose or goal.

3.58 **service**

1) (product) The results generated by activities at the interface between the supplier and the customer and by supplier internal activities, to meet the customer needs. [Euromethod]

2) (process) A process, activity or task delivering a result (tangible or intangible product) for the benefit of some other entity requiring that result.

3.59 **specification:** Definitive description of a system for the purpose of developing or validating a system. [ISO 2382-20]

3.60 **supplier:** Organisation to which an engineering project decided by the customer is contracted (see *Provider*). [Based on Euromethod]

3.61 **support:** The process of sustaining a product in an operational state, including sourcing, helping, training and problem investigation.

3.62 **system**

1) Representation of something which is identifiable, and which evolves within and acts on an environment, according to some goal with respect to the environment. [Euromethod]

2) The conception of an area (the system domain) having its elements seen to be related to form a whole and having at least one systemic property. Systems can be represented as models using some (modelling) language.

3.63 **target domain:** Part of an organisation for which an engineering project should be performed. [Euromethod]

3.64 **target system:** The system which is the object of any process, activity or task undertaken by another system.

3.65 **task:** Element of an activity, performed by an actor. [Euromethod]

3.66 **taxonomy:** Classification, especially in relation to general laws or principles. [Oxford English Dictionary]

3.67 **technique:** Set of heuristics and procedures which explain how to create a set of one or more products in a specified formalism. [Based on Euromethod]

3.68 **technology:** The application of practical or mechanical sciences to industry or commerce; the methods, theory and practices governing such application. [Collins English Dictionary]

3.69 **tool:** Anything used as a means of performing an operation or achieving an end. [Collins English Dictionary]

3.70 **user:** A person who makes use of an information product, an ISE product or an ISE technology product; a sub-type of actor. [Based on Euromethod]

4. Symbols

4.1 Conceptual model notations and conventions

The following notations and conventions apply to diagrammatic illustrations of the conceptual model (see section 7):

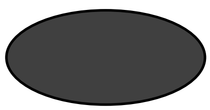
- Processes are shown as boxes with rounded corners:

Process

- Products are shown as square-cornered boxes:

Product

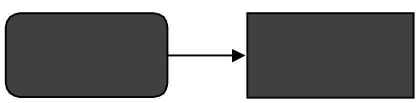
- Resources are shown as ellipses:



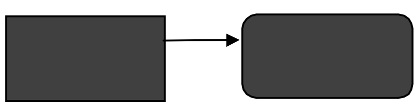
Resource

- The following associations (relationships) are denoted:

Produces/receives (processes produce products, products feed processes), e.g.

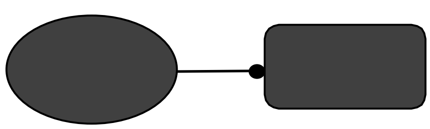


Process produces product

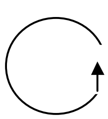


Product feeds process

Performs, e.g.

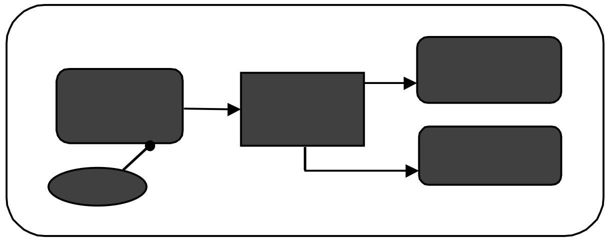


Actor performs process



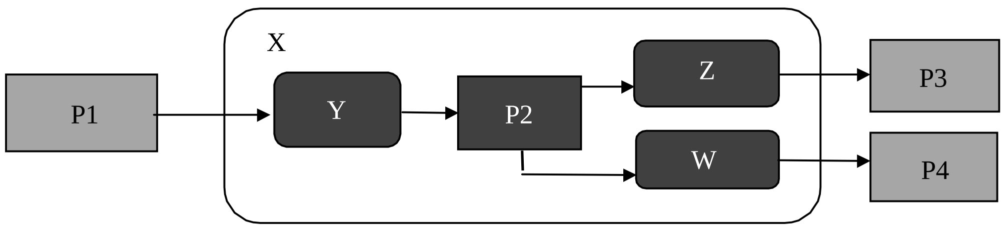
Cyclic process(es)

Decomposition is denoted by boxes within a box:



Processes, products or resources may be labelled by a name inside the box. If the name is absent, the box denotes some process, product or resource which it is not relevant to name.

Products can only be decomposed into other products; processes may be decomposed into a set of interconnected processes and products. Arrows can cross the boundaries of boxes to indicate, for example that a product external to process X feeds a process internal to X.



These notations privilege two main kinds of entities—*processes* and *products*—and three kinds of relations between them—*produces, feeds* and *is decomposed into*. The third kind of entity—*resource*—is primarily of interest for its sub-type *actor* which has the relation *performs* with the process it is associated with. These are the modelling concepts that are most relevant to this document, but they are not exhaustive. In particular, the text accompanying the figures references attributes (qualities) of products, processes and resources, as well as other concepts and relationships. For this reason, it is the text of section 7 that is definitive of the ISE conceptual model. The diagrams, which are necessarily partial, are illustrative rather than definitive.

4.2 Taxonomy notation

The taxonomies in section 8.2 are expressed using a formal syntax based on Backus Naur notation (see the SGML standard ISO 8879). In the taxonomies, an undefined element means that this is as far as the decomposition needs to go for the purpose of this document.

The conventions of the formal syntax are as follows:

|	is used to separate alternatives (‘or’)
&	is used to separate co-existent components (‘and’)
?	means the component is optional
*	means the component can occur zero or more times (‘optional and repeatable’)
+	means the component must occur one or more times (‘obligatory and repeatable’)
( )	are used for grouping as in mathematics

5. Abbreviations and acronyms

3GL	Third Generation Language
4GL	Fourth Generation Language
API	Applications Programming Interface
BSI	British Standards Institution
CAD	Computer Aided Design
CALS	Computer aided Acquisition and Logistics Support
CASE	Computer Aided Software Engineering
	Computer Aided System Engineering
CCTA	The Government Centre for Information Systems (UK)
CD	Committee Draft
CDIF	CASE Data Interchange Format
CEC	Commission of the European Communities
CEN	Comité Européen de Normalisation
CENELEC	Comité Européen de Normalisation Electrotechnique
CMM	Capability Maturity Model
CORBA	Common Object Request Broker Architecture
DCE	Distributed Computing Environment
DD	Draft for Development
DIS	Draft International Standard
DoD	Department of Defense (US)
ECMA	European Computer Manufacturers Association
ECU	European Currency Unit
EDI	Electronic Data Interchange
EFTA	European Free Trade Area
EIA	Electronic Industries Association (US)
EN	European Norm
ENS	European Nervous System
ENV	European Norm Vornorm
ER	Entity Relationship
ERA	Entity Relationship Attribute
ESA	European Space Agency
ETSI	European Telecommunications Standards Institute
EU	European Union
EUROCAE	EURopean Organisation for Civil Aviation Electronics
EWOS	European Workshop on Open Systems
FoF	Framework of Frameworks
FRISCO	FRamework of Information System COncepts
FUR	Framework for User Requirements
GATT	General Agreement on Tariffs and Trade
GOSIP	Government Open Systems Interconnection Profile (UK)
ICT	Information and Communications Technology
IDA	Interchange of Data between Administrations

IEC	International Electrotechnical Commission
IEE	Institution of Electrical Engineers (UK)
IEEE	Institute of Electrical and Electronics Engineers (US)
IFIP	International Federation for Information Processing
IRDS	Information Resource Dictionary System
IS	Information System
	International Standard
ISE	Information Systems Engineering
ISES	Information Systems Engineering System
ISETS	Information Systems Engineering Technology System
ISO	International Organisation for Standardisation
IT	Information Technology
ITSEC	IT Security
ITT	Invitation To Tender
ITU	nternational Telecommunications Union
JTC	Joint Technical Committee
MS	Mathematical Statistics
MTBF	Mean Time Between Failures
MTTF	Mean Time To Failure
MTTR	Mean Time To Repair
ODP	Open Distributed Processing
OED	Oxford English Dictionary
OMG	Object Management Group
OO	Object Oriented
OS	Organisation System
OSF	Open Software Foundation
OSI	Open Systems Interconnection
PC	Personal Computer
PCTE	Portable Common Tools Environment
PT	Project Team
QMS	Quality Management System
RACE	Research and technology development in Advanced Communication technology in Europe
R&D	Research and Development
SC	Sub Committee
SEDDI	Software Engineering Data Definition and Interchange
SME	Small / Medium sized Enterprise
SNMP	Simple Network Management Protocol
SOGITS	Senior Officials Group on IT Standardisation
SOGT	Senior Officials Group on Telecommunication
SQL	Structured Query Language
TC	Technical Committee
TEN	Trans European Network
TMN	Telecommunications Management Network
TR	Technical Report

VAN	Value Added Network
VAT	Value Added Tax
WD	Working Draft
WG	Working Group

PART 1

The ISE Conceptual Model and taxonomy

Part 1 contains essential material for anyone using the document for reference purposes.

6. Theoretical foundation of the ISE conceptual model

6.1 Systems, processes, products and resources

Information systems engineering is modelled as a series of related systems in order to deal with the complexity of the domain.

A system is a representation or conception of something which is identifiable, the elements of which are seen to be related to form a whole. Systems evolve within and act on an environment according to some goal with respect to that environment.

Any system can be modelled as an organised set of processes, products and resources:

- A process is a set of inter-related activities that meets an objective or function within the system. The elements of a process are called activities; the elements of an activity are called tasks. Processes, activities and tasks have the potential to be automated—to be executed by a machine resource.

- A product is a result of a process. It may be any tangible artefact, e.g. a deliverable, file, document, data item or information item. A product may also be intangible, e.g. a decision, event or a change in the state of something resulting from the execution of a process. For any process, there may be one or more products serving as inputs and one or more products serving as outputs. Products may be transitory (serving only as communication between processes) or persistent (retained in product stores). An output product may be unchanged by the process (e.g. retrieved data) or changed by the process (e.g. updated data).

- A resource is an object that participates in specific relations with processes, facilitating the execution of a process, e.g. a processor (which may be human or machine), fuel, money or building. A processor is commonly called an actor. An actor is a person or device capable of acting and thereby changing the state of perceived things. To perform any process, one or more actors are needed. The set of actions performed by an actor to achieve a defined outcome is known as a role (human) or a function (machine). Any process can be viewed as an actor performing the process by executing a process definition (a program for a computer or a procedure for a person). In executing the process, actors interact with other resources such as fuel, money, storage, etc.

The term *service* in its normal usage is ambiguous. It can mean both a process (activity or task) and the result (product) of a process. In the ISE conceptual model and taxonomy the term *product* is used to denote both product in the conventional sense and service in the sense of a result delivered by a process (e.g. a consultant's recommendations, a repair to a faulty component). The term *service* is used to denote the particular kind of process, activity or task undertaken on behalf of another entity that delivers a beneficial result to that entity (e.g. consulting, supporting, repairing).

Every process, product and resource has attributes (characteristics, properties, features, qualities) which may be measured.

A system must have a clearly definable boundary. The system's processes, products and resources belong inside that boundary. Everything outside that boundary is called the system context. An interface is a part of the boundary over which an input or output product may pass, together with the rules governing such inputs and outputs and their interpretation (i.e. protocols, rules for invoking services, data structures, etc.).

System control is modelled as an exchange of information products between a control sub-system and a production sub-system. Control processes are directing, assessing and supporting. Production processes are analysing, designing, constructing and, installing—collectively called developing. Maintaining is not treated as a distinct process, but rather as a special case of developing which may involve some or all of the development processes.

6.2 Data, information and information systems

A system interacts with its context by means of input and output products. It processes input products and produces output products in accordance with its *purpose* (goal or function). The Oxford English Dictionary (*OED*) defines 'purpose' as:

> *"The object for which anything is done or made, or for which it exists; the result or effect intended or sought; end, aim"*

The input and output products of a system (or a process) include data products and information products. ISO defines 'data' as:

> *"A re-interpretable representation of information in a formalised manner suitable for communication, interpretation or processing."*

In itself, data is without meaning. Data becomes meaningful when it is *interpreted* in the light of a specific purpose. The OED defines 'interpret' as:

> *"To expound the meaning of; to make out the meaning of; to bring out the meaning of"*

Data so interpreted is information. ISO defines 'information' as:

> *"Knowledge concerning objects, such as facts, events, things, processes or ideas, including concepts, that within a certain context have a particular meaning."*

Information causes a change in the state of the receiving processor—human or machine. Information looks like data, and is data, but it is a special kind of data in that it has the ability to effect a change—in the case of a human resource, to create knowledge, understanding or a decision. Information is bound to its act of informing and to the effect of informing.

Data can be reused, but it can only be information the first time it is used because the resulting change causes the system to have a slightly different purpose; the second time the data is accessed, the potential for change has already been realised. In human terms, once particular data has been interpreted and information is known, further receipt of the same data does not increase the knowledge (although the data may be reaccessed for different purposes, such as reconfirming or reassessing the information)[2]. Research data, for example, is often used information; if such data is interpreted in the light of a new purpose, it can sometimes provide new information.

Data is without sense in itself, but when related to a purpose, data interpretation may extract information. The information sent from one information system is not the same as is received by the second because it is interpreted in the light of different purposes. The fact that the same data can be interpreted differently means that care has to be taken to ensure that the data is adequate for its purpose (in completeness or precision) and that it is not used out of context.

---

[2]Note, however, that the idea that information is reduced to data when used is counter to normal IT usage.

Information products are therefore different from data products. An information product will be bound to its specific purpose, while a data product can be re-interpreted by another purpose. However, a data product useful for a system having one purpose is not necessarily of any value for a system having another purpose. Equally, two systems with different purposes could generate different information from the same input data.

A system which receives data products as input, which interprets that data and outputs information products is an information system. For example, a word processing package does not provide any interpretation of its input data (text and graphics). It is therefore a data processing system and not an information system. It becomes part of an information system when it is viewed in conjunction with the human resource which creates its output. A person interacting with a word processor (software, hardware and communications) to output information is an information system. The person who interacts with the word processor to create the output product is an actor of the information system (i.e. inside the system); the person who interprets (reads, uses) the output product is a user of the information system (i.e. outside the system).

6.3 Systems and sub-systems

ISE is concerned with four systems:

- the information system (IS);
- the organisation system (OS) in which the information system is used;
- the ISE system (ISES) which develops the information system;
- the ISE technology system (ISETS) which develops the technology used by the ISE system.

An information system is a sub-system of an organisation system. In its widest sense, the term information system can be used to denote the full extent of an organisation's information flows and usage considered as a whole. In the narrower sense it is often used to denote a computerised information (sub-)system. These relationships can be expressed as follows (see Figure 7-3):

- an organisation system is a system;
- an information system is a system; a specific information system is also a sub-system of some specific organisation system;
- a computerised information system is a system; it is also a sub-system of an information system in the wider sense (and hence a sub-system of an organisation system);
- a computerised application is a system; it is also a sub-system of some specific computerised information system (and hence a sub-system of some wider information system and of some organisation).

An ISE system is a kind of organisation system. A specific ISE system may be a sub-system of an organisation system with some other purpose (e.g. an IT department within a manufacturing or service industry business) or it may constitute a separate organisation system whose purpose is to provide ISE products and services to other organisations. Similarly an ISE technology system is a kind of organisation system. A specific ISET system may be a sub-system of an ISE system or it may constitute a separate organisation system whose purpose is to provide ISE technology products and services to other organisations. Like other organisation systems, the ISES and ISETS require information system support; if the ISES and ISETS are embedded in larger organisation systems, their ISs may be largely common with the ISs for the organisation as a whole.

It is important to stress that the separation of the four systems in the ISE conceptual model is an abstraction, made to facilitate analysis and description. In practice the relationships between the systems may be complex, with systems of one type partially, wholly or deeply embedded within other systems. For example, the modern trend is very much towards merging instances of IT systems, either giving the IS some ISES capability or making the IS a specialised version of the ISE system.

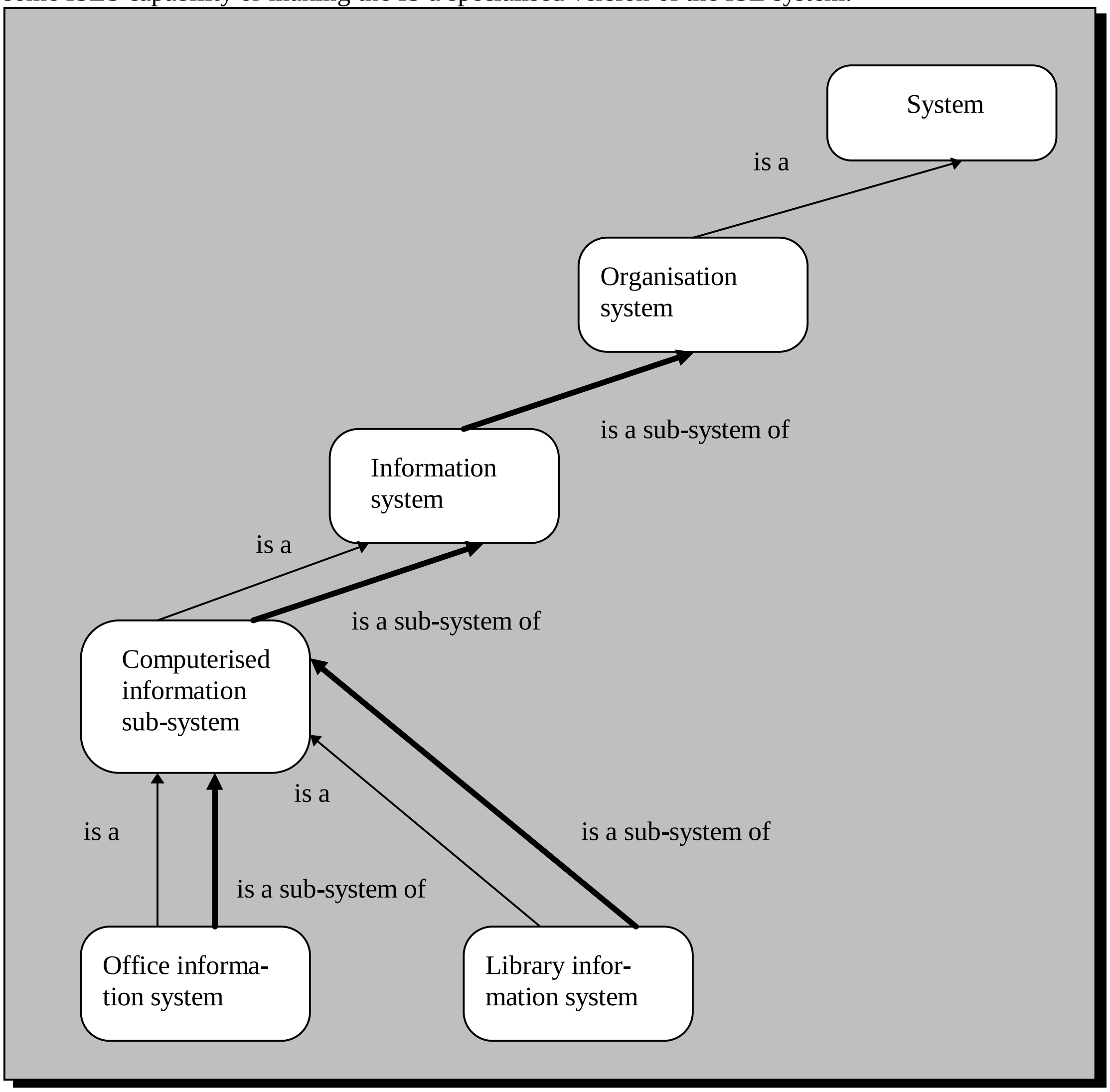


Figure 6-1 Organisation and information systems and sub-systems

7. The conceptual model of ISE

The top level of the conceptual model is defined in section 7.1. This level defines the four systems relevant to ISE and the interactions between those systems. The breakdown of each of the systems into a production sub-system and a control sub-system is described in sections 7.2 and 7.3. The next level of decomposition of the conceptual model (section 7.4) introduces more detail of particular processes, products and resources of each of the four systems.

Section 7.5 defines the qualities of processes, products and resources that are particularly relevant to information systems engineering.

7.1 The top level of the ISE conceptual model

CEN/TC 311 has defined ISE as:

> *"the systematic, disciplined application of knowledge, methods and experience to the provision and support of information systems, bridging strategic goals/requirements and operational tools."*

The provision of an information system encompasses social, business, engineering and technology considerations relating to:

- the identification, understanding and formulation of organisations' information-related activities and needs;
- the acquisition of information systems to meet these needs;
- the implementation and evolution of the information systems;
- the support of operational information systems and their users to ensure that the organisations' needs continue to be met;
- the activities and products associated with acquiring, developing and supporting the technology used in the engineering of information systems.

7.1.1 The systems and context of ISE

The four systems with which ISE is concerned—the OS, the IS, the ISES and the ISETS—and their context are shown in Figure 7-1.

7.1.1.1 The organisation system and its context

An organisation is a human system, consisting of a structured set of people, possibly using machines (including computers), co-ordinating their efforts towards a certain goal. The organisation has strategy, organisational structure, assets, resources, systems, processes and information as its architectural elements.

The organisation system and its market context are the *raison d'être* of the other three systems, with processes that concern the identification of the organisation's need for information, the satisfaction of that need and the evolution of the need leading to a further cycle of information supply.

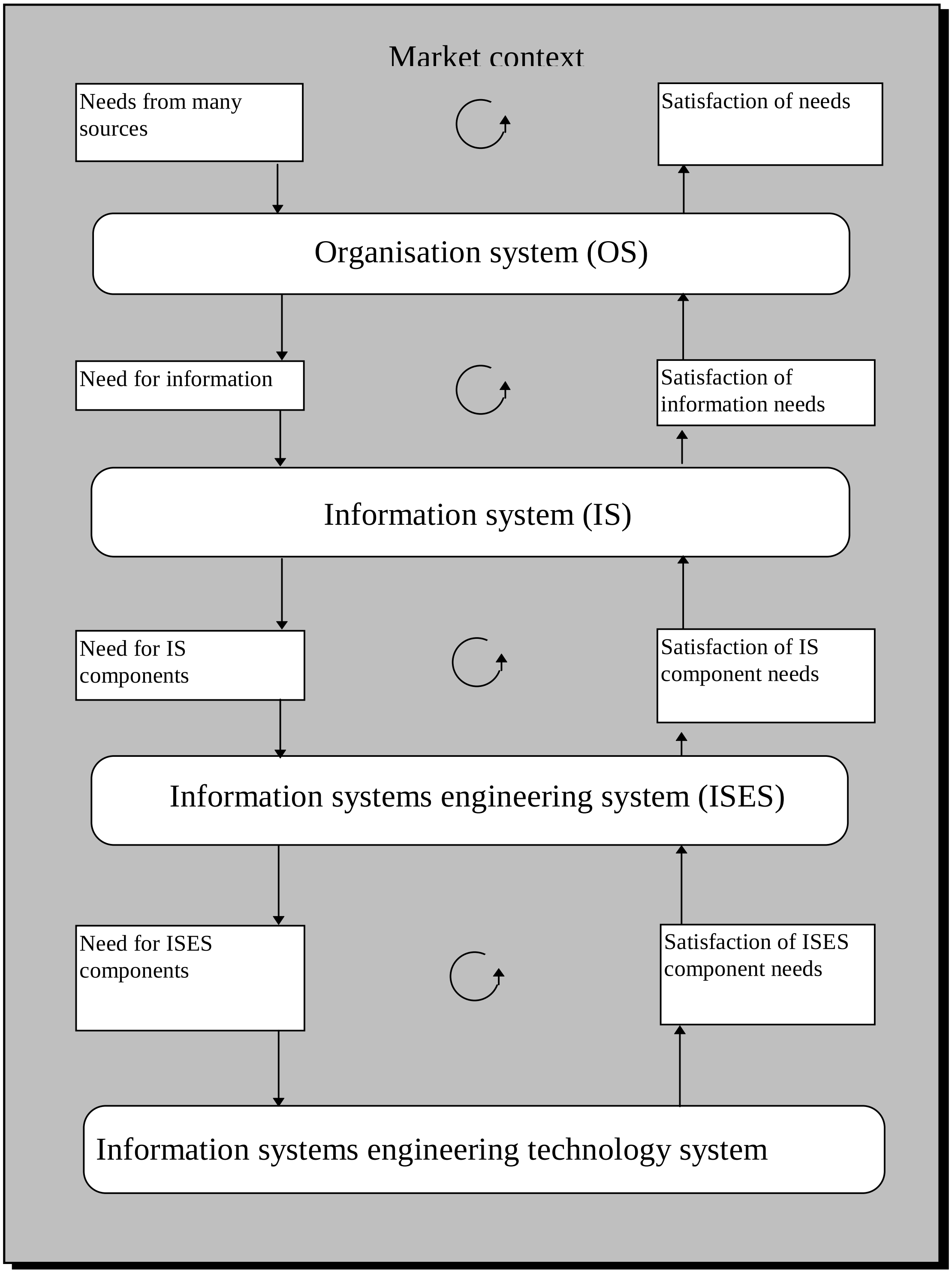


Figure 7-1 The systems and context of ISE

The organisation's requirements for information arise from both internal processes and from direct and indirect interactions with its market context, including other organisations. Typical examples of external interactions giving rise to information requirements are: government and EU regulations, and the trade cycle of a company. The trade cycle, for example, involves transactions with suppliers and with customers, with agents and distributors, with transport organisations, with banks, with customs, with support agencies such as chambers of commerce, with information providers such as market research organisations. More indirectly, changes in market conditions, political situations and any number of other factors, can also exert pressures on the company and necessitate access to existing or new information services, sources and facilities.

7.1.1.2 The information system

An information system handles the collection, storage, processing, distribution and display of information and data for a given area of one or more organisation systems. The actors constituting the processor components of the information system may be human or automated (a computer, a microchip, etc.) or both. An information system may range in size and complexity from a computer-based military command and control system to a manual directory system based on a card index.

7.1.1.3 The ISE system

An ISE system is a system which provides the information system to the organisation system. Provision encompasses the analysis, design, construction, installation and support (management) of an IS. Any extension, adaptation or evolution of the IS is carried out by the ISE system. An instance of an ISE system may relate to part or all of the life cycle of the information system, from the initial strategic planning through to the retirement of the information system or its evolution to a new form and purpose.

7.1.1.4 The ISE technology system

An ISE technology system is a specialised instance of an ISE system which serves to define, build, support and extend ISE systems. The end products of an ISE technology system include such elements of an ISE system as techniques, languages, methods, metrics, tools, standards and procedures.

7.1.2 Interactions between the systems

No system is self-contained: each interacts with other systems, for example as provider, acquirer, trading partner. Attention to the interfaces between systems (and components of systems) is essential for the successful development of global information infrastructures. Examples of critical interfaces are:

- the acquirer/provider (customer/supplier) interface;
- computer sub-system/telecommunications sub-system interface.

The interactions (interfaces) between the four systems of the ISE domain are an essential part of ISE. At the highest level of the ISE conceptual model, the interaction to be considered is the cycle of need and satisfaction of need that takes the form of product acquisition/supply and service acquisition/supply (see Figure 7-2):[3]

- the market context requires products and services (including data and information) of the OS which in turn supplies those products and services;

[3] Note that service is used in its sense of a process that delivers a result (product) to the acquiring organisation.

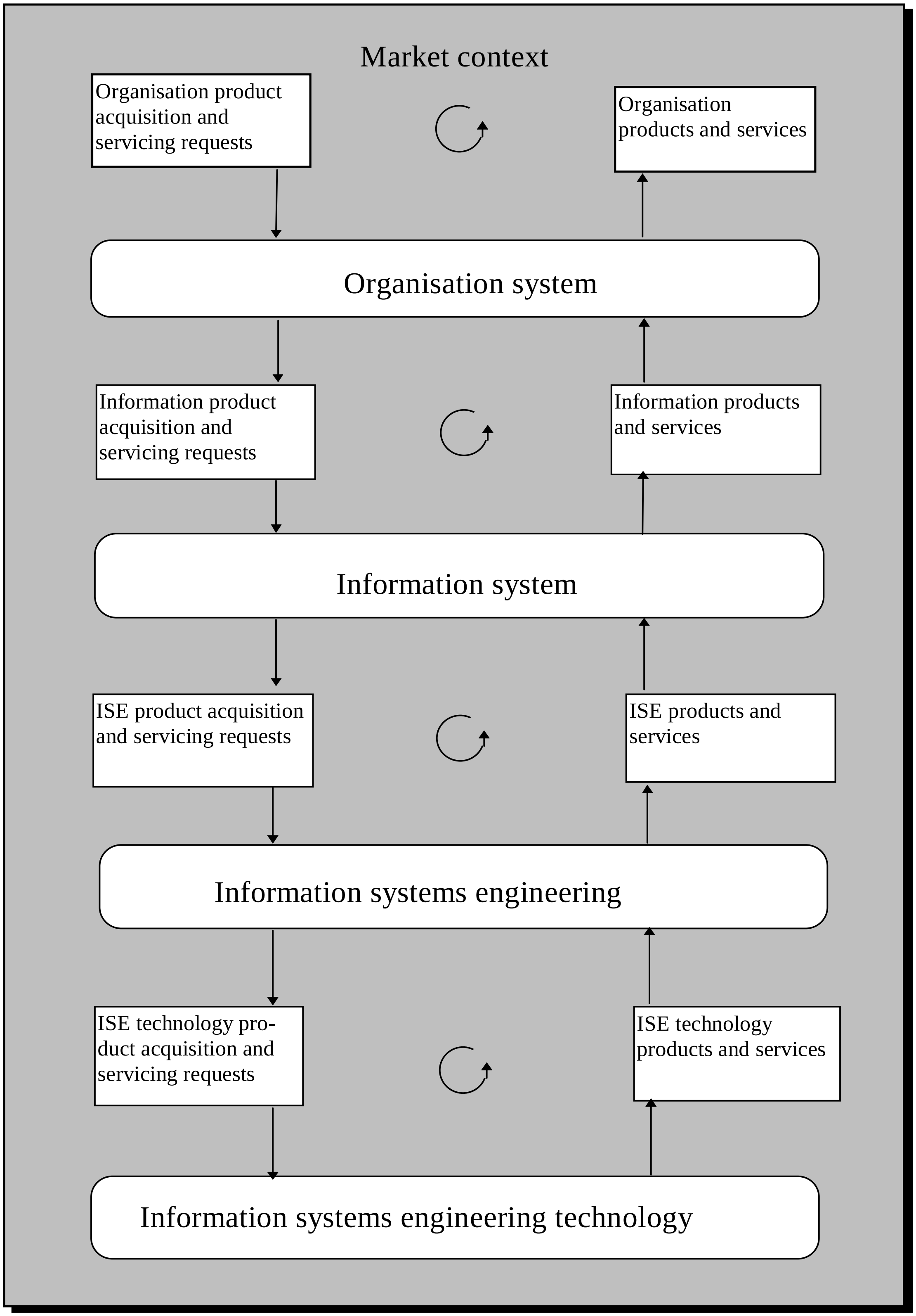


Figure 7-2 Interactions between the systems

- the organisation system requires information products and services of the information system which in turn supplies those products and services;
- the information system requires ISE products and services of the ISE system which in turn supplies those products and services;
- the ISE system requires ISE technology products and services of the ISE technology system which in turn supplies those products and services.

The provider of the product or service may be part of the acquiring organisation (in-house acquisition) or external to it (out-sourced acquisition). Where acquisition involves purchase, the following transactions take place between the system requiring the product or service and the system providing it:

- the tendering process in which bids are solicited and evaluated and a provider is selected;
- the development process during which the purchaser agrees the product or service specification with the provider, monitors progress, reviews and verifies deliverables and agrees any contractual changes;
- the completion process during which the purchaser accepts the final deliverables and the agreement (contract) is completed.

7.2 Production sub-systems

Products, processes and resources of each of the four systems can be categorised as belonging to a production sub-system or a control sub-system.

For example, the production sub-system of an organisation system consists of those processes which add value to the inputs to the organisation and generate outputs in accordance with the goals of the organisation. In a car-making organisation, production processes transform input raw materials and component parts into output finished cars, and orders into deliveries. A series of other organisational processes constitute the control sub-system which serves to manage the organisation's value adding activities—ranging from the setting of strategic goals through to the effort to achieve those goals through careful monitoring of production processes, products and resources. See Figure 7-3.

The sub-systems that are of particular concern in the provision of information systems are:

- the production sub-systems of the ISE system and the ISE technology system (described in this section);
- the control sub-systems of the information system, the ISE system and the ISE technology system (described in section 7.3).

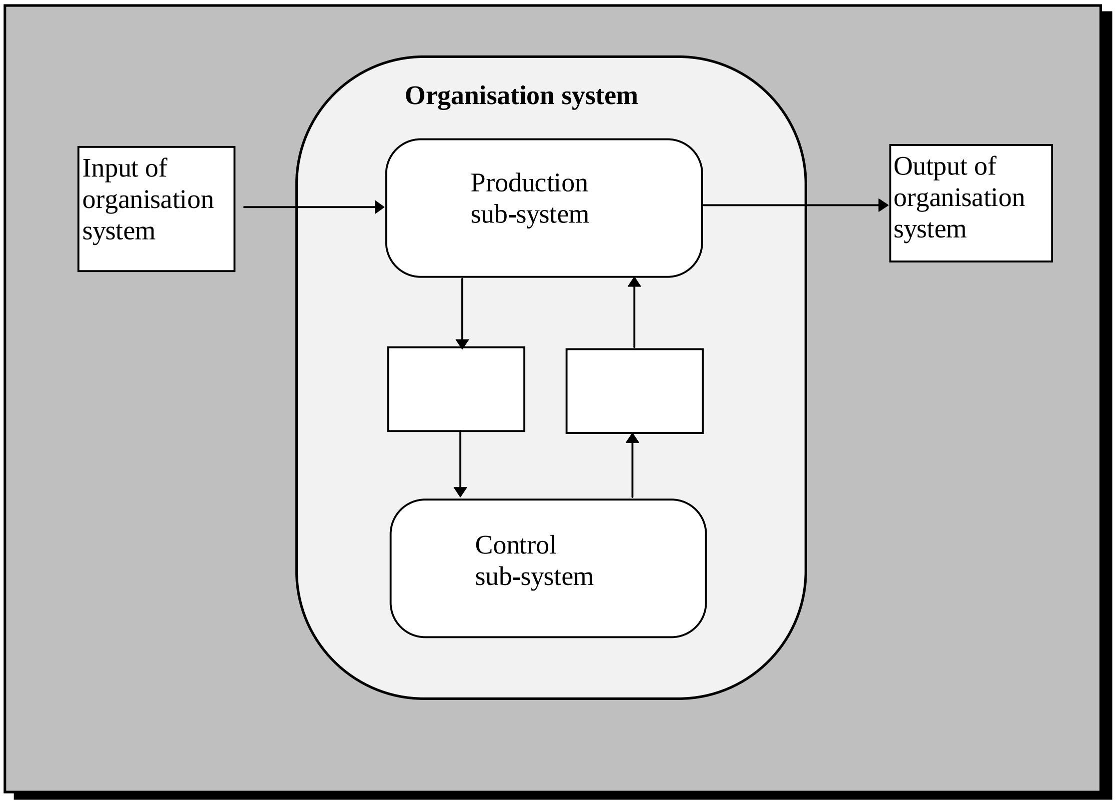


Figure 7-3 Production and control sub-systems

A production sub-system generates the output products of the ISE or ISE technology project and any interim products such as requirements and design specifications. Production activities fall into the following process categories: analysis, design, construction and installation (see Figure 7-4). Collectively these processes may be called development, change, extension, evolution or adaptation depending on the focus or extent of the particular ISES or ISETS project.

Testing, which is sometimes thought of as a production process, is actually a control process applied to the output products of each production process (see section 7.3).

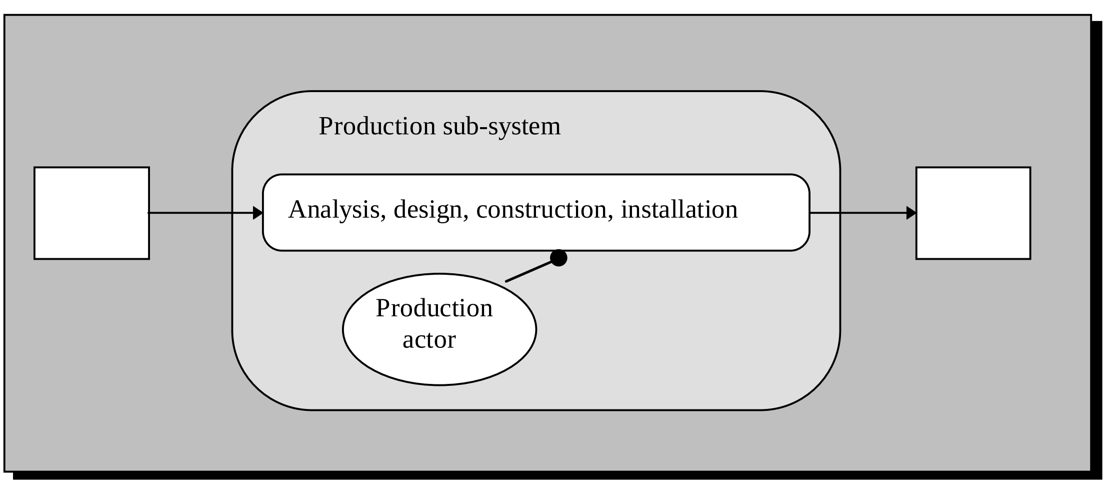


Figure 7-4 Production processes

7.2.1 Analysis

The input to analysis is an expression of particular problem situations and needs of the acquiring system, arising from such things as changed business conditions, new technology potential, evolving management policy, revised government regulations, etc.

Analysis includes activities undertaken by the acquirer and the provider of the ISE or ISE technology product or service. Acquirers undertake or sub-contract activities such as feasibility studies, identification of functional and non-functional requirements arising from different sources (e.g. organisational policy, end user requirements, interface requirements), problem analysis, identification of options, investigation of technical, financial and time constraints, early sub-set and implementation priorities, foreseeable modifications and enhancements, design hints, guidelines and constraints, acceptance criteria and the preparation of a statement of requirements suitable for issuing an invitation to tender (ITT) or for initiating in-house development work.

Analysis also includes the activities involved in the provider's response to an ITT, including studying the ITT, considering how to meet the requirements in the light of constraints on both sides, estimating, costing and responding.

Interim products of analysis include a feasibility study, an ITT and a response to an ITT (a proposal or bid). The final output of analysis is a description (model, specification, representation) of the essential conditions that the target system or component has to satisfy. It is a specification of requirements which also expresses the constraints on possible solutions, e.g. information and organisational resource constraints (human, machine and financial).

7.2.2 Design

The input to the design process is a specification or statement of requirements together with any modifications formally requested, for example, by contract negotiations or by management review.

The design process translates the statement of requirements into a representation of the architectural, data and procedural elements of the target system or component. Typically design will involve an iterative process of increasingly detailed designing, starting with the high level design of the functions to be performed by the target system or component and leading to detailed designs of the elements which will realise the high level functions (hardware, communications, software, etc.).

The output of design is a comprehensive description (build specification) of the system or component sufficient for the intention of the design to be realised by construction.

7.2.3 Construction

The input to the construction process is a specification or representation of a design sufficient for the construction of the target system or component.

The construction process includes the acquisition of components from external sources (out-sourcing, sub-contracting), as well as the internal building of components (programming in the case of software components). The integration of components is a construction activity.

The output of ISES construction is an ISE product (software, hardware, firmware, communications component, document, etc. or an assembly of some or all of these).

The output of ISETS construction is an ISE technology product (formalism, technique, method, metric, infrastructure component, tool, document, etc. or an assembly of some or all of these).

7.2.4 Installation

The input to the installation process is an ISE product or an ISE technology product together with a definition of the procedures to be followed to install and make the product operational.

The installation process takes the developed product and establishes it in its operational environment. Physical connections are made, interfaces are established, human actors are trained (see section 7.3.3.3) and the product is tested in operational mode (see section 7.3.2.1).

The output of installation is an accepted product ready for operational use.

7.2.5 Adaptations of ISs and ISESs

The extension, adaptation or evolution of an IS or ISES constitutes a special case of development, involving some or all of the processes of analysis, design, construction and installation. Legacy is about combining different systems while change is about the evolution from one system to another over time. Changes can be classified as follows:

- **Changes to meet new functional requirements:** Organisations change and the result is often demands for changes in information systems and their components. Since the pressure for organisational change is increasing, the frequency of change due to new user requirements can be expected to follow suit.

- **Adaptation to changes in computer platform:** Changes in the computational environment of an IS or ISES also happen frequently. New network software may be installed, or new versions of operating systems may require adaptations of applications. Functional requirements for improved integration of systems also often have consequences at the computer platform level, with the result that systems are modified in order to meet both new functional and computational requirements.

- **Porting to a new environment:** This is a special kind of adaptation to changes in the computational environment, where major parts of the environment are replaced. Definitions of machine-independent languages and protocols have helped make porting into a task that is often successfully carried out.

- **Re-engineering of software:** When computer sub-systems become dominated by ‘spaghetti code’ and demands for changes continue to arrive, the program code may be restructured or totally replaced with new code, often in a more modern and higher level language. Since the specification of the new system is often a copy and extension of the old one, the requirements specification of the new system may be fairly easy to define and may be stable during the restructuring or replacement of the code. This is contrary to development by analysis and design, where the uncertainty and instability of the requirements specification can cause many problems, requiring substantial project management effort.

7.2.5.1 Control sub-systems

The control sub-systems of an IS, ISES and ISETS are relevant to ISE. A control sub-system manages a production sub-system by directing, assessing and supporting the processes, products and resources of the production sub-system.

Aspects of a control system include control of:

- projects: project management;
- project risk: risk management;
- product access: security management;
- product integrity: change and configuration management;
- data: data management;
- documents: documentation management;
- sourcing: acquisition (procurement) management;
- quality of products, processes, resources: quality management;
- value of resources: financial management (accounting).

Aspects of direction, assessment and support are relevant to all these categories of control.

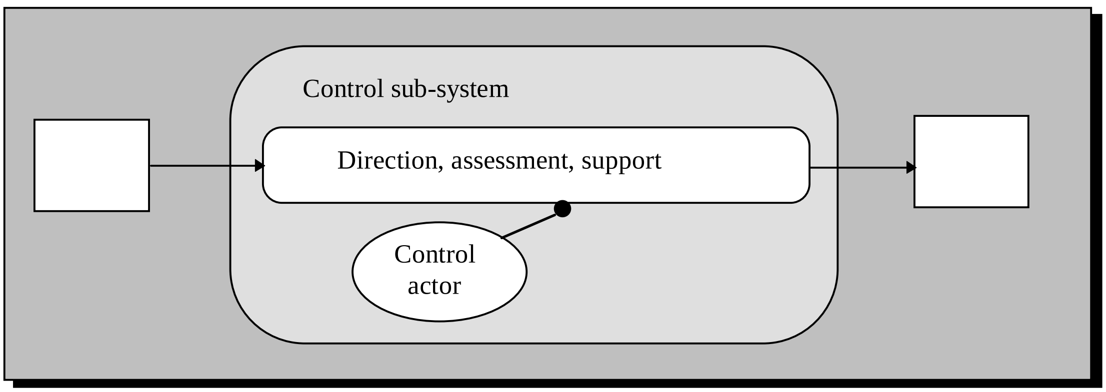


Figure 7-5 Control processes

7.2.6 Direction

The process of direction concerns the guidance of a production sub-system to achieve a specified goal or purpose within the constraints imposed by the environment (contractual conditions, management policy, etc.). Direction activities are planning, instructing, monitoring and adjusting.

7.2.6.1 Planning

Planning involves the estimating, negotiation and agreement of the resources, timescales, methods, activities, milestones, deliverables etc. of a production instance (e.g. a development project).

7.2.6.2 Instructing

Instructing concerns the issuing of process, activity and task definitions to the actors of the production sub-system. These will take the form of verbal or written instructions for human actors and software for computational actors. Instructing also concerns the acquisition of components of both the production and the control sub-systems for a particular production instance, e.g. the purchase of IT infrastructure and tools (requests for ISE technology products and services).

7.2.6.3 Monitoring

Monitoring involves the continuous verification that all the actors of the production and control sub-systems are actually achieving their purpose with respect to their roles and commitments, within the constraints of law and regulation and with proper usage of resources.

7.2.6.4 Adjusting

Adjusting concerns the issuing of revised instructions to the actors and the reallocation of resources sufficient to put the production instance back on course for achieving its purpose.

7.2.7 Assessment

Assessment takes place in the context of established norms.

Products may be assessed for conformance to specifications and for conformance to standards; human resources may be assessed against standards for their competence, particularly in using relevant techniques and methods. The availability of good review and testing techniques, test specifications and test suites for products and for resources is a quality prerequisite.

Assessment of quality (conformance to requirements) may take place within a system, for example, when a development engineer tests system components and integrated systems, or one system may be assessed by another system, for example, when a purchaser of an IS product assesses the capability of different providers or an independent certification body audits a provider's quality management system (QMS).

Processes can be examined to:

- characterise current practices, identifying strengths, weaknesses and risks inherent in the process;
- determine to what extent the practices are effective in achieving process goals;
- determine the extent to which the processes conform to a set of baseline practices.

Capability assessment is the assessment of process components and their maturity. Process maturity is a system's ability consistently to follow and improve its processes. Process capability is the range of results expected from following a process (a predictor of future project outcomes), while process performance is the actual results achieved from following a process. Capability determination is normally undertaken by independent bodies and by purchasers wishing to assess the level of maturity of particular providers; process improvement is the concern of providers who wish to evaluate their processes and improve their capability maturity.

The main activities of assessment are testing, measuring and evaluating.

7.2.7.1 Testing

Testing involves recording the output of an entity in response to a particular input stimulus or set of stimuli. Testing is an essential activity in relation to design (to establish performance and check conformance to specification), in relation to construction (to prove correct transfer of the design and eliminate defects) and in relation to installation (to prove correct operations and facilitate acceptance by the acquirer). Testing techniques include inspection, review, verification and validation. Testing also plays a part in support (see section 7.3.3).

7.2.7.2 Measuring

The activities of evaluating, improving and estimating all require some form of measurement of processes and products. Sizing techniques and measures facilitate the evaluation of the size and complexity of information systems, improving the estimating and costing of development effort. Provision of appropriate well-chosen metrics can be used to demonstrate to customers and third parties that the ISES process is managed and mature.

7.2.7.3 Evaluating

Evaluation is the activity of taking the results of testing and measuring and forming a judgement as to their significance in the light of guiding norms. This activity is normally performed by human actors as it involves the application of discrimination and judgement. However, some knowledge-based computational systems might be said to be capable of evaluating.

7.2.8 Support

Support is the process of preserving the product in an operational state, by providing on-going activities (services) such as supply of resources, help desks, additional training and problem investigation. The input to the support process is an operational system or component together with any constraints or controls imposed by the organisation on its operations.

As a result of evaluation, instructions may be issued for correction or improvement.

7.2.8.1 Sourcing

Sourcing is the activity of identifying, acquiring and providing the resources needed to maintain a product in an operational state.

7.2.8.2 Helping

The activity of helping is often provided by means of a help desk or a telephone help line. Questions from human actors are answered and minor problems are resolved.

7.2.8.3 Training

Training is the activity of transferring knowledge of, for example, process, activity and task definitions to human actors by informing, demonstrating and practising (equivalent to the activity of installing software on a machine resource).

7.2.8.4 Investigating

Investigating is the activity of locating the cause of problems in operational systems, using testing techniques etc., to facilitate the initiation of the necessary development processes to carry out correction or repair.

7.3 The processes, products and resources of the four systems

The next level of decomposition of the ISE conceptual model develops the processes, products and resources of the four systems. Particular points relevant to ISE and not already discussed in sections 7.2 and 7.3 are covered here.

7.3.1 Processes, products and resources of the OS

Organisation system processes directly relevant to ISE are:

- the analysis of the OS, determining information and information flow requirements;
- the acquisition of information products and the ISs which provide those products;
- the use of information products and information systems.

Key products of the acquisition process are requests for the provision of products and services. Where acquisition involves purchase, products include invitations to tender, contracts, orders, delivery and acceptance documentation and invoices. Actors involved in acquisition fulfil roles such as procurement manager, sourcing engineer, quality manager and contracts manager.

The key product of the use process is information. Actors involved in use can be playing almost any role in the organisation.

7.3.2 Processes, products and resources of the IS

IS control processes include the management of the operations of information systems (sometimes called service management or facilities management). This is becoming an increasingly important process as information systems interact across organisational system boundaries and the corresponding control systems have to co-operate (see section 9.3). System management involves support and other types of control activity, including security management, change management, service level management, charging, capacity management, operations management, availability management, problem management, acquisition management and customer management.

The input products of an information system are data, information and ISE products. The output products are information and requirements for ISE products (e.g. software) and services (e.g. correcting errors and faults in the system).

The actors of an information system are people and/or computational sub-systems performing within the information system or in relationship with it, for example:

- the users and operators of a computer system;
- the users of manuals and documents;
- the computer system itself;
- managers concerned with the effectiveness and efficiency of the information system;
- people or organisations concerned with the performance of the information system (e.g. the customers of the organisation);
- external computer systems having relationships with the information system.

7.3.3 Processes, products and resources of the ISES

Some processes have the potential to be automated—to be executed by a machine resource. ISE process automation involves the use of ISE technology products, including CASE *(Computer Aided Software/System Engineering*) tools, databases and development environments (see section 7.4.4).
The process of developing information systems has two main kinds of result:

- **Products that can be delivered by a supplier:** ISES processes result in the generation of products which are, in the main, characterised by being data or information products. Examples of ISE products are an enterprise model, a design specification, a quality plan, source code, executable objects. Examples of the elements of ISE products are entity type, flow, critical success factor, event, activity, program, module, relationship, data store, trigger, operating system file. Together such elements constitute the ISE information base. Databases specialised for the storage of ISE information are commonly called repositories (see section 7.4.4.8). The end products of an ISE system are the components of an information system, such as hardware, software, firmware, communications components, human procedure definitions and various kinds of documentation. The end products and some or all of the intermediate products are delivered to an acquiring organisation (customer).

- **Organisational and human changes in the acquiring organisation:** Computerised tools for use by an individual user, such as a spreadsheet program, require little more than individual training to become effective. When computer-based systems involving many people are engineered, the corresponding human and organisational development requires much more effort. For such a system to be a success, everyone has to accommodate their work style and attitude to a common agreement. A shared database requires everyone to enter data in a disciplined manner and understand the purpose of the data entry. In addition, introduction of shared databases usually alters the conditions of work tasks in such a way that tasks change or are transferred to other people or departments, with corresponding re-distribution of resources and authority. These organisational and human changes have to be planned and carried out to a large extent by the users themselves, with support from management.

Roles performed by ISES actors include information (system) engineer, analyst, designer, programmer, test engineer and project manager. The internal organisation of ISES departments in Europe usually has smaller barriers between operators, programmers, analysts and other specialists than in Japan and USA. This cultural difference has implications for role definitions, recruitment and training. Highly educated personnel with technical knowledge and a broad background can be appropriate for most tasks in European ISES organisations.

In terms of other types of resource, the development of information systems requires time and money just like any other business process. However, while many other business processes are able to adhere to plans and budgets, information systems development suffers from poor planning and from too little knowledge of techniques and procedures for effective estimating of resource requirements.

Tools and environments can improve productivity, provided they support relevant techniques and methods used in the ISES and provided they are open and able to be modified to suit changes in the ISES (see section 7.4.4).

The knowledge of the human actors of the ISES (maintained by ongoing training) is the most important resource in information systems development and human resource qualities are therefore critical (see section 7.5.4).

7.3.4 The processes, products and resources of the ISETS

As an ISE technology system is a specialised instance of an ISE system, the observations made in section 7.4.3 regarding processes and resources are applicable here.

The end products of the ISE technology system include:

- development models;
- techniques;
- methods;
- formalisms;
- standards and procedures;
- metrics;

- tools;
- repositories;
- environments.

Written standards, procedures, development models, techniques and methods constitute process definitions for the human actors of an ISE system. Embodied in software (e.g. in CASE tools and in automated environments) they constitute process definitions for the computational actors of an ISE system.

7.3.4.1 Development models

The development process is usually organised in stages. These stages are by no means always discrete and linear: stages may overlap, they may be revisited many times, they may run concurrently. Different development models[4] (waterfall, spiral, prototyping, step-wise refinement, etc.) are defined by ISE technology systems and used by ISE systems.

7.3.4.2 Techniques

Recognised ways of carrying out particular activities are called techniques. Examples are organisation modelling, data flow modelling, entity relationship (ER) modelling, object modelling, structured programming, function point analysis, Fagan inspections. Some techniques are appropriate for one process category only, while others may be used in more than one process. Techniques may incorporate formalisms (see section 7.4.4.4).

7.3.4.3 Methods

A method (methodology) is usually an integrated set of ISE guidelines, techniques and the products produced by those techniques. Euromethod defines a method as:

> *"formalised knowledge that can be communicated in a reproducible way and which allows one or more people to organise work to achieve a certain class of goals within a certain class of problem situations. A method usually contains concepts, one or more languages, assumptions, rules, heuristics, procedures, guidelines, subsidiary methods and techniques. A method describes a way to conduct a process to achieve a desired outcome."*

Methods systematically prescribe the activities and products that constitute particular ISE processes; examples are MERISE, AXIAL, SDM, DAFNE, IEM, Multiview, object-oriented (OO) analysis and design, PRINCE, SSADM. Methods may apply to the production sub-system (for example, analysis and design methods) or to the control sub-system (for example, project management, configuration management and testing methods).

7.3.4.4 Formalisms

Languages and other notations are collectively called formalisms; examples are the Petri net notation, ERA notations. Formalisms include natural and formal languages, mark-up languages, graphical languages and diagramming notations.

Programming languages are a kind of ISE formalism; examples are Ada and C++. Programming interfaces define the interfaces between system components and their supporting infrastructure (the

---

[4] Sometimes called lifecycles.

platforms or sets of resources on which the software will run). An API (*Applications Programming Interface*) specifies a mapping between a particular programming language and the features of a particular service; examples are APIs for operating system functions, for graphics systems and for databases.

7.3.4.5 Standards and procedures

Standards and procedures to be followed by ISE actors in the execution of an ISE project are produced by the ISE technology system. Standards may be *de jure* standards, publicly available specifications or private standards (not used outside a particular organisation).

7.3.4.6 Metrics

To monitor and improve processes and/or products, metrics are used to quantitatively evaluate the actual qualities and predict future performance. Examples of metrics include size of code, effort deployed to complete tasks and activities, number of errors found in a particular piece of work. Analysis of trends over time will indicate if actions are needed to maintain or improve any level of performance. Refined measurements may further provide information on which factors have the greatest influence on processes and/or products and assist in calibrating beneficial actions.

7.3.4.7 Tools

Examples of the tools used as resources by ISE systems are planning tools, project management tools, acquisition management tools, quality management tools, statistical tools, configuration management tools, access controllers, analysis and design tools, test tools, programming tools, language compilers, debuggers, performance profilers, text management tools, graphic/multimedia tools, scheduling tools and support tools. Automated tools may take the form of software applications, commonly known as CASE tools. CASE tools for software engineering are a sub-set of tools for information systems engineering.

7.3.4.8 Repositories

ISE information is frequently stored electronically in repositories. A repository is a specialised database where all the information for an ISE project is stored and dynamically accessed by work groups or tools. Repositories can hold both production information used in system development and control information used for measuring such things as performance and return on investment. The information stored may be in the form of documents, specifications, source code, object code, test cases, test data and a myriad of other items essential to an ISE project.

7.3.4.9 Environments

An environment is the complete set of IT resources of an ISE system that is available to one or more people or work groups. An environment consists of tools which automate processes, platforms (which are common to information systems in general) and additional infrastructure that is often referred to as a framework for the environment.

A platform is a set of services, implemented in hardware and software, that have been found over many years to be useful in the development of most information systems. It comprises operating system and communications services as well as computational services (provided by mainframe, mini computers, client/server systems, (networked) workstations and PCs, closely or loosely coupled computers, etc.).

An environment framework provides many services useful in ISE systems, a subset of which comprises a repository. The environment framework can be thought of as a more up-to-date specification of platform services.

An environment will support particular ISE techniques, methods and formalisms. An off-the-shelf environment can also affect the choice of method, through cost influences such as licences, availability of existing tools for reuse and training of programmers and other users.

7.4 The qualities

All systems and their components (processes, products and resources) have attributes, which are here called ‘qualities’.

The qualities considered for the purpose of the ISE conceptual model are particular ISE product, process and resource qualities, namely:

- ISE product qualities;
- information product qualities;
- process qualities;
- human resource qualities.

7.4.1 ISE product qualities

ISE product qualities can be categorised as direct qualities or indirect qualities. Both sets of qualities can be used to group and classify information systems (see section 11).

7.4.1.1 Direct qualities

These qualities are the non-functional aspects of the system requirement and they influence the way in which ISE products are developed (production and control processes). They also influence the design; the need to incorporate these features into a design can lead to additional functionality to handle the features.

The direct qualities are dependability, changeability, manageability, openness, efficiency, learnability, usability, testability and certifiability.

1 Dependability

Dependability quantifies the reliance on the quality of the service the information system delivers. From the system user point of view this property can be decomposed into four attributes: reliability, availability, safety and security. For repairable systems an alternative set of attributes can be obtained if availability is replaced by repairability.

Dependability in relation to software is called software integrity. Software integrity is directly related to reliability, safety and security but is different from repairability and availability.

a Reliability

This quality quantifies the ability of an ISE product to deliver a non-interrupted service. It is the probability that an item can perform a required function under given conditions for a given time interval, e.g. that during a certain period of time and under specified conditions an information system can perform the functions described in its requirements specification. Reliability can be mathematically described by a probability distribution of the time of system functioning until the first failure. Metrics used in this context are Mean Time To Failure (MTTF) and Mean Time Between Failures (MTBF).

b *Repairability*

This quality relates to repairable products and quantifies the ability of the product to be repaired, i.e. its ability under given conditions to be retained in or restored to a state in which it can effectively perform the required function. More precisely it is the probability that a given active repair action for an item under given conditions of use can be carried out within a stated time interval when the repair is performed under stated conditions and using stated procedures and resources. Repairability can be mathematically described by a probability distribution of the time necessary for the repair after failure. A metric used in this context is Mean Time To Repair (MTTR).

c *Availability*

This quality relates to repairable products and quantifies the user's expectation to find the product functioning. It is the probability that a product will be able to perform its designated function when required for use. Availability depends on reliability and repairability.

d *Safety*

This quality describes the product characteristic of doing no harm to its environment. Safety is the expectation that a product will not lead to a state in which human life, limb, health or environment are endangered. It is quantified by the probability of provoking a harmful event.

e *Security*

This quality describes product resistance to external threats against its integrity. Security is resilience to vandalism and human malfeasance, usually taking the form of fault prevention by access control against deliberate intrusion from implanted or external sources.

2 Changeability

Aspects of changeability are reconfigurability, preparedness and modifiability. These qualities contribute to the flexibility of the ISE product and its ability to be migrated to new environments and evolved to meet new requirements.

a *Reconfigurability*

This quality expresses the possibility of influencing an information system's structure and/or functions during system operation. The reconfigurability of a system is linked with the service discontinuity during modifications of its structure or functions. If a system is capable of changing its properties without degradation of the quality of its service it can be categorised as reconfigurable on-line. If the modification of its properties necessitates temporary interruption of its service, it is reconfigurable off-line.

The notion 'modification of structure or function' is an arbitrary issue. It is sometimes difficult to distinguish between system reconfiguration (major changes of function and/or structure) and system parameterisation (minor changes of system function or structure).

b *Preparedness*

Preparedness is the readiness of an ISE product to be modified. Some future modifications may be anticipated at the time of development, and products may be designed to facilitate these changes. For example, a computer system may be prepared for extension with new types of data by locating data definitions, data operations and data presentation in one place in the code. Preparedness may be assessed in accordance with the kinds of new requirement that the product is prepared for, the extent to which the existing product has to be modified and the extent of the additions that have to be made.

c Modifiability

Modifiability is an ISE product's ability to be modified to meet user requirements for which no preparation was made. A large proportion of development effort is spent meeting requirements for changes that were never foreseen. These modifications often require restructuring of a product. Good modifiability is when user-requested changes are carried out promptly and restructuring of functionality and integration with other applications happens smoothly. Backlogs of user requests for change may be an indication of poor modifiability. ISE products that are difficult to change can easily lose their usefulness.

Because modifiability relates to unanticipated changes, predictions of modifiability have to depend on characteristics of design and the empirical coupling of these characteristics with observed modifiability of other products. However, there is currently only weak empirical support for stating that specific characteristics improve modifiability. One tentative correlation for computational sub-systems is that construction by higher level languages in a more modular fashion leads to improved modifiability. More research is needed, both to find the important characteristics and to relate them to modifiability.

3 Manageability

Manageability is a set of qualities expressing the facility of gathering information about an ISE product, monitoring it and controlling it during all or part of its lifecycle.

4 Openness

Portability and interoperability together determine the openness of an information system or its components.

a Portability

This is a set of qualities which bears on the ability of an ISE product to be transferred from one environment to another.

b Interoperability

This is a set of qualities which bears on the ability of an ISE product to exchange data and information with another ISE product. Interoperability depends on a range of features, from a physical communications link to carry the data/information through to a common interpretation of the form (syntax) and content (semantics) of the data/information by the sending and receiving systems.

5 Efficiency

Efficiency is the set of attributes that bears on the relationship between the level of performance of an ISE product and the amount of resources used, under stated conditions.

6 Learnability

Learnability is the set of attributes of an information system or its components that determines the effort required by a human being to learn to use it. Systems intended for public use need to be intuitively usable, while systems for complex tasks like CAD (*Computer Aided Design*) may require extensive training and still be productive. Learnability can be measured by time to learn and by counting the number of errors made. Thinking aloud sessions, where users say what they think when trying to use a system, give a qualitative evaluation of learnability.

7 Usability

Usability relates to the facility of manipulation by the end user (the effort needed for use once the ISE product has been learned). It is assessed through the user efforts required to fulfil a specified task using the product: the less effort required, the better the usability of the product. Usability may be measured by such things as the time needed to carry out a task, by the number of key-strokes needed, the numbers of errors made. Qualitative evaluation is achieved through asking the users about the burden they experience when using the product.

Usability does not necessarily imply learnability. For example, a computer-based application that is controlled by means of key-stroke commands may be harder to learn than pointing and clicking in a menu. However, when the key combinations are known, the time to carry out a task may be shorter than if the user had to choose alternatives from a menu.

8 Testability

Testability is the inherent property of a testable entity which facilitates the verification of its defined qualities and performance. This leads to the diagnosis of malfunctions in terms of characteristic transfer functions, in a timely and affordable manner. A testable entity is an entity whose transfer functions can be defined and verified via test points which are accessible to the test facilities provided either internally or externally. A transfer function is the relationship between an output response of a testable entity and any input stimulus or set of stimuli that produced the response.

Because quality cannot be tested into a product during construction, designing for testability is a basic requirement. To achieve a high level of testability for a system as a whole, each of its replacement component parts should be testable entities.

9 Certifiability

Compliance is a set of qualities of an ISE product which makes it adhere to standards, conventions or regulations in laws and similar prescriptions.

Certifiability of an information system expresses the possibility of obtaining a formal statement concerning the compliance of system operation with respect to its specified requirements. The need to formally certify a system considerably influences the choice of methods for its production and is strongly dependent on the end user.

Certifiability is also related to a set of formal standards (the ISO 9000, EN 29000 series) which permits the certification of an organisation's quality management system (see section 7.3.2).

7.4.1.2 Indirect qualities

The qualities identified in section 7.5.1.1 group together those ISE products which may share common kinds of requirement and development style. There is a further set of external factors which may influence the selection of ISE methods and tools. These ‘indirect qualities’ are physical constraints, platform, external interfaces, replication and application domain.

1 Physical constraints

This set of attributes is immediately linked with the ISE product’s external function and with the environment in which it will be used. It represents those parts of the non-functional requirements which are not directly linked to the service perceived by the product’s users.

The set of constraints is extremely context and environment-dependent and can be related to hardware features such as size, shape, power consumption, price, colour, choice of materials and temperature range. The set of constraints can also include resource restrictions such as ‘maximum processor utilisation and memory utilisation must not exceed 50% during any acceptance test’.

2 Platform (hardware and software)

The hardware and software used in ISES production processes determines the possible tools and methods which are practicable. This platform might include mainframes with conventional operating systems, client/server systems, lock-step coupled processors for dependability, closely coupled computers, loosely coupled computers or deeply embedded computers.

A development environment can also affect the choice of method, through cost influences such as licences, availability of existing tools for reuse and programmer training.

3 External interfaces

The use of specific external transducers and communications protocols often calls for development of hardware and basic software. This may have an impact on the schedule, test coverage and debugging environment. If standard interfaces and protocols can be adopted, there is usually a significant gain in development efficiency. Examples are data communication (digital data links), analogue-digital converters, human-computer interfaces, digital plant control, radar and sonar.

4 Replication

An ISE product may be needed in a single instance or may be needed in multiple copies. Supplying multiple installations with the same or similar software, for example, may involve special-purpose production methods and tools for appropriate parameterisation, replication and validation. The management of such deliveries has to be dealt with in the early stages of development in order to facilitate post-delivery support, including change management of the installed product.

5 Application domain

The area in which the ISE product will operate within the organisation system has many implicit influences on the requirement. The application domain often involves specific standards which have a strong influence on the required ISE approach, particularly configuration management. It may also imply some of the direct qualities, e.g. certifiability in avionics, dependability in the automotive world.

The application domain determines constraints such as availability, repairability, power consumption, mass and volume. It can also impose restrictions concerning context of use (e.g. military or civil), safety/security, legal obligations, industrialisation, economic issues (e.g. selling prices), changeability and openness. These constraints may have a high priority during analysis and design. They have a very strong influence on the quality that must be achieved.

7.4.2 Information product qualities

At the level of ISE systems and ISE technology systems it is natural to regard information as something stored in a data structure (this is reflected in the ISO definition of information###see section 3). At the level of information systems and organisation systems it is more natural to regard information as something that creates a decision or other instantaneous effect. It is this latter view that underlies this section.

Once used, information has lost its ability to inform and only the features of data are left (see section 6.2). Normally a database consists of used information awaiting a new purpose to become information once again. As a result, consideration of data quality tends to look back to the origins of the data while consideration of information quality tends to look forward to the effect of the information.

Required information products delivered from one information system to another may have different qualities for the two information systems because of their difference of purpose.

7.4.2.1 Validity

The quality of validity (correctness) concerns how well data reflects its origins with respect to its original purpose. Data that is not perfectly valid obviously contains some defects. Primary types of defect are missing data, biased data, data with noise, data of unclear origin, data with redundancy that cannot be eliminated.

When existing data is used for new (secondary) purposes, new defects often show up. Precautions can be introduced to increase the quality of data when it is used for secondary purposes—purposes often not considered when the data was originated.

A special kind of data is created with no purpose—often as a side effect of the creation of data which does have a purpose. It is kept because it is easier to keep than to eliminate or because someone believes it may be useful in the future. Correctness attributes are possible for such data but are of little interest for as long as the data remains unused.

7.4.2.2 Relevance

Data is said to be relevant for a given purpose if it can be interpreted to provide information satisfying that purpose. After its use in this way, it disappears or is stored as supplementary data. The result of the satisfaction of the purpose is the manifestation of a decision, a state transition or knowledge.

Imperfect information obviously contains some defects. Primary types of defect are missing information, biased information, information with noise, information of unclear origin, information with redundancy, erroneous information, deliberately false information, a mixture of false and true information, an inseparable mix of relevant information and irrelevant data.

7.4.2.3 Communicability

Communicability is the ability of the data or information to be transferred from one processor to another without loss or distortion. Where the data is to function as information, the meaning (semantics) of the information must be communicated (interchanged) as well as its form.

7.4.2.4 Integrity

Integrity is the quality of data that has not been modified or destroyed, either maliciously or accidentally.

7.4.2.5 Confidentiality

Confidentiality is the degree to which information or data is not to be made available or disclosed to unauthorised individuals, entities or processes.

7.4.2.6 Availability

Availability is the quality of information or data being accessible and usable on demand by an authorised individual, entity or process.

7.4.2.7 Privacy

Privacy is the degree to which information or data is subject to the right of individuals to control or influence what information or data related to them may be collected and stored and to whom that data or information may be disclosed.

7.4.2.8 Sensitivity

Sensitivity is the degree to which information or data must be protected because its unauthorised disclosure, alteration, destruction or loss will cause perceivable damage to something or someone.

7.4.3 Process qualities

There is no comprehensive body of scientific knowledge on which to base a complete definition of process qualities###this is an area for further research. However, work has been done by the Carnegie Mellon Software Engineering Institute in the area of process maturity (known as the capability maturity model—CMM). A preliminary definition has also been attempted for process interoperability and monitorability.

7.4.3.1 Maturity

A key quality of a process is its maturity. Five stages of maturity may be usefully distinguished: initial, repeatable, defined, managed and optimising. From the point of the acquirer of a product or service, knowledge of providing systems' process maturity may be relevant in deciding which provider is most likely to deliver a satisfactory product or service.

1 Initial

This quality is where processes are ad hoc, undefined and uncontrolled, and where success is entirely due to individual talent and efforts. Staff will argue against discipline under the guise of individual creativity. Managers will sacrifice standards and practices to other priorities—usually time deadlines. This level of maturity may, of course, be appropriate for very small (one person) projects.

2 Repeatable

This quality is where basic project management is in place, even in a crisis. Project management, acquisition management, quality management, and change management are implemented. A stable, managed environment ensures that previously successful processes are repeatable. Sufficient process capability exists for meeting deadlines.

3 Defined

This quality is where processes are fully defined and applied. Defined techniques and methods are in place. There is a resource whose responsibility is to focus on process improvement, including the application of technology. Training needs are identified, provided for and tracked.

4 Managed

This quality is where the focus is on product and process quality, processes are measured and analysed, and full quality management is in place. Statistical process control principles are used to address special causes of process variation. Detailed process and product data is available and quality targets are set and tracked. Processes can be performed within narrowly defined quantitative limits and targets are predictable.

5 Optimising

This quality is where continuous process improvement is undertaken, and where process change management, technology innovation and defect prevention are in place. Remaining causes of poor performance are identified and eliminated, while process capability is continuously raised. New technologies are prototyped, piloted and, if successful, used.

7.4.3.2 Monitorability

Monitorability depends on the amount of control information that the process is able to create.

7.4.3.3 Interoperability

Interoperability is the ability of two or more processes to use (share) the same data or information. Interoperability depends on a range of features, from a physical communications link to carry the data or information (cables, air waves, microwaves, light, etc.) through to a common interpretation of the form (syntax) and content (semantics) of the data or information by the sending and receiving processes. (See also item 4b of section 7.5.1.1.)

7.4.4 Human resource qualities

The capability of the human resource has been identified as the most important factor in ensuring the development of satisfactory products in the available time (Boehm, '*Improving Software Productivity*'). The ability to develop information systems may be increased through formal education, professional experience and informal means.

Since human skills and knowledge are at least as important for the productivity of OSs, ISs, ISESs and ISETSs as tools, environments and project organisation, human qualities need to be considered in the ISE conceptual model[5].

7.4.4.1 Experience

Experience is time spent in the practice of a particular technique or role. The greater the experience, the greater the accumulated knowledge and skills are assumed to be.

The effective development of ISs tends to require a wide range of experience, for example, social scientists and management scientists able to apply theories and practical methods for organisational change and personnel development, as well as the more traditional experience of analysts, designers and programmers.

7.4.4.2 Aptitude

Aptitude is the innate ability to perform well in a particular task or role, for example, an aptitude for logical thinking, an aptitude for leading other people or an aptitude for good human relations.

7.4.4.3 Education

Education is the level of formal education or training (inculcation of knowledge and skills) successfully undertaken relevant to the performance or a particular task or role.

7.4.4.4 Competence

Competence is the ability to apply experience, aptitude, skill and knowledge to the performance of a task or role such that a satisfactory outcome is normally achieved.

7.4.4.5 Integrity

Integrity is the ability to consistently and reliably apply professional standards of ethics in the performance of a task or role.

7.4.4.6 Motivation

Motivation is the ability to focus efforts on the particular task or role with the determination to achieve improved or high standards of output.

[5] When human resource qualities are evaluated in practice, it must be done within the relevant laws and regulations.

8. The ISE taxonomy

In this section the ISE conceptual model is expressed in the form of taxonomies, that is, classifications of the concepts of ISE. This section does not introduce new concepts: rather it expresses the concepts developed in section 7 in a form more suitable for uses such as those connected with standards management (see section 12).

8.1 Taxonomy background

Taxonomy as a technique and scientific discipline is historically linked with natural sciences such as zoology or botany where it concerns a systematic study of the classification of species. In this document, the use of a taxonomy has the same objectives as in natural sciences. It is aimed at expressing the classification of the concepts of the domain of ISE.

Two premises underlie the ISE taxonomies:

1. that the classification of the concepts of ISE can be expressed using the containment relationship ‘consists of’ (e.g. ‘an ISE system consists of an ISES production sub-system and an ISES control sub-system’) and the specialisation relationship ‘is a kind of’ (e.g. ‘analysis is a kind of production process’);

2. that every entity of interest can be described sufficiently by a set of concepts chosen at the same level of abstraction; and that where this is the case, every entity can positioned at a single point in a multi-dimensional representation space.

8.2 The ISE taxonomies

For reasons of brevity and accuracy the taxonomies are expressed here using a formal syntax based on Backus Naur notation (see section 4). In section 8.3 one of the taxonomies is shown in a less formal and less complete but more easily readable form.

In the following taxonomies, an undefined element means that this is as far as the decomposition needs to go for the purpose of this document.

8.2.1 Taxonomy of ISE entities

The following taxonomy of the entities of ISE expresses the classification of those entities.

[Note: attributes are not yet added to this taxonomy###they position at the lowest points of decomposition]

ISE domain:=	((ISES+) & (ISETS+) & (IS+) & (OS+) & (ISETS-ISES interface product+) & (ISES-IS interface product+) & (IS-OS interface product+) & (OS-OS context interface product+))
ISES:=	(ISES production sub-system & ISES control sub-system)

ISES production sub-system:=	((ISES production process+) & (ISES production product+) & (ISES production resource+))
ISES production process:=	(analysis | design | construction | installation)
ISES production product:=	(analysis product | design product | construction product | installation product)
ISES production resource:=	(analysis resource | design resource | construction resource | installation resource)
ISES control sub-system:=	(ISES project management & ISES risk management & ISES security management & ISES change management & ISES data management & ISES documentation management & ISES acquisition management & ISES quality management & ISES service level management & ISES capacity management & ISES operations management & ISES availability management & ISES problem management & ISES customer management & ISES financial management & other)
ISES project management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES risk management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES security management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES change management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES data management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES documentation management :=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES acquisition management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES quality management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES service level management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES capacity management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES operations management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES availability management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES problem management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES customer management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES financial management:=	((ISES control process+) & (ISES control product+) & (ISES control resource+))
ISES control process:=	(direction | assessment | support)
ISES control product:=	(direction product | assessment product | support product)

ISES control resource:=	(direction resource | assessment resource | support resource)
ISETS:=	(ISETS production sub-system & ISETS control sub-system)
ISETS production sub-system:=	((ISETS production process+) & (ISETS production product+) & (ISETS production resource+))
ISETS production process:=	(analysis | design | construction | installation)
ISETS production product:=	(analysis product | design product | construction product | installation product)
ISETS production resource:=	(analysis resource | design resource | construction resource | installation resource)
ISETS control sub-system:=	(ISETS project management & ISETS risk management & ISETS security management & ISETS change management & ISETS data management & ISETS documentation management & ISETS acquisition management & ISETS quality management & ISETS service level management & ISETS capacity management & ISETS operations management & ISETS availability management & ISETS problem management & ISETS customer management & ISETS financial management & other)
ISETS project management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS risk management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS security management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS change management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS data management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS documentation management :=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS acquisition management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS quality management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS service level management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS capacity management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS operations management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS availability management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS problem management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS customer management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))

ISETS financial management:=	((ISETS control process+) & (ISETS control product+) & (ISETS control resource+))
ISETS control process:=	(direction | assessment | support)
ISETS control product:=	(direction product | assessment product | support product)
ISETS control resource:=	(direction resource | assessment resource | support resource)
IS:=	(IS production sub-system & IS control sub-system)
IS production sub-system:=	((IS production process+) & (IS production product+) & (IS production resource+))
IS control sub-system:=	(IS project management & IS risk management & IS security management & IS change management & IS data management & IS documentation management & IS acquisition management & IS quality management & IS service level management & IS capacity management & IS operations management & IS availability management & IS problem management & IS customer management & IS financial management & other)
IS project management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS risk management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS security management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS change management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS data management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS documentation management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS acquisition management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS quality management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS service level management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS capacity management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS operations management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS availability management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS problem management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS customer management:=	((IS control process+) & (IS control product+) & (IS control resource+))
IS financial management:=	((IS control process+) & (IS control product+) & (IS control resource+))

IS control process:=	(direction | assessment | support)
IS control product:=	(direction product | assessment product | support product)
IS control resource:=	(direction resource | assessment resource | support resource)
OS:=	(OS production sub-system & OS control sub-system)
OS production sub-system:=	((OS production process+) & (OS production product+) & (OS production resource+))
OS control sub-system:=	(OS project management & OS risk management & OS security management & OS change management & OS data management & OS documentation management & OS acquisition management & OS quality management & OS service level management & OS capacity management & OS operations management & OS availability management & OS problem management & OS customer management & OS financial management & other)
OS project management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS risk management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS security management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS change management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS data management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS documentation management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS acquisition management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS quality management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS service level management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS capacity management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS operations management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS availability management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS problem management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS customer management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS financial management:=	((OS control process+) & (OS control product+) & (OS control resource+))
OS control process:=	(direction | assessment | support)

OS control product:=	(direction product | assessment product | support product)
OS control resource:=	(direction resource | assessment resource | support resource)
direction:=	(planning & instruction & monitoring & adjusting)
assessment:=	(testing & measuring & evaluating)
support:=	(sourcing & helping & training & investigating)
resource:=	(actor | fuel | money | building | other)
actor:=	(human actor | machine actor)
ISETS-ISES interface product:=	(ISE technology product-service request | ISE technology product)
ISE technology product:=	(development model | technique | method | formalism | standard | procedure | metric | tool | repository | environment | other)
ISES-IS interface product:=	(ISE product-service request | ISE product)
ISE product:=	(software | hardware | firmware | communications component | procedure | documentation | other)
IS-OS interface product:=	(information product-service request | information product)
information product:=	(data | information)
OS-OS context interface product:=	(OS product-service request | OS product)
OS product:=	(data | information | other)

8.2.2 Taxonomy of ISE qualities

The following taxonomy of the qualities of various ISE entities expresses a framework for the classification of those entities (see section 11).

ISE domain quality:=	(ISE product quality | information product quality | process quality | human resource quality)
ISE product quality:=	(direct quality | indirect quality)
direct quality:=	(dependability | software integrity | changeability | manageability | openness | efficiency | learnability | usability | testability | certifiability)
dependability:=	(reliability & repairability & availability & safety & security)
software integrity:=	(reliability & safety & security)
changeability:=	(reconfigurability | preparedness | modifiability)
indirect quality:=	(physical constraint | application domain | platform | external interface | replication)

information product quality:=	(validity | relevance | communicability | integrity | confidentiality | availability | privacy | sensitivity)
process quality:=	(maturity | monitorability | interoperability)
maturity:=	(initial | repeatable | defined | managed | optimising)
human resource quality:=	(experience | aptitude | education | competence | integrity | motivation)

8.3 An alternative representation of the taxonomy of ISE entities

This section shows the first of the two ISE taxonomies (see section 8.2.1) expressed less formally as a ‘family tree’.

- ISE domain
  - ISES
    - ISES production sub-system
      - ISES production process
        - analysis
        - design
        - construction
        - installation
      - ISES production product
        - analysis product
        - design product
        - construction product
        - installation product
      - ISES production resource
        - processor (actor)
        - money
        - fuel
        - building
        - other
    - ISES control sub-system
      - ISES control process
        - direction
          - planning
          - instructing
          - monitoring
          - adjusting
        - assessment
          - testing
          - measuring
          - evaluating
        - support
          - sourcing
          - helping
          - training
          - investigating
      - ISES control product
        - direction product

            - assessment product
            - support product
        - ISES control resource
            - processor (actor)
            - money
            - fuel
            - building
            - other
- ISETS
    - ISETS production sub-system
        - ISETS production process
            - analysis
            - design
            - construction
            - installation
        - ISETS production product
            - analysis product
            - design product
            - construction product
            - installation product
        - ISETS production resource
            - processor (actor)
            - money
            - fuel
            - building
            - other
    - ISETS control sub-system
        - ISETS control process
            - direction
                - planning
                - instructing
                - monitoring
                - adjusting
            - assessment
                - testing
                - measuring
                - evaluating
            - support
                - sourcing
                - helping
                - training
                - investigating
        - ISETS control product
            - direction product
            - assessment product
            - support product
        - ISETS control resource
            - processor (actor)
            - money
            - fuel
            - building
            - other
- IS
    - IS production sub-system

        - IS production process
        - IS production product
        - IS production resource
            - processor (actor)
            - other
    - IS control sub-system
        - IS control process
            - direction
                - planning
                - instructing
                - monitoring
                - adjusting
            - assessment
                - testing
                - measuring
                - evaluating
            - support
                - sourcing
                - helping
                - training
                - investigating
        - IS control product
            - direction product
            - assessment product
            - support product
        - IS control resource
            - processor (actor)
            - money
            - fuel
            - building
            - other
- OS
    - OS production sub-system
        - OS production process
            - analysis
            - design
            - construction
            - installation
        - OS production product
            - analysis product
            - design product
            - construction product
            - installation product
        - OS production resource
            - processor (actor)
            - money
            - fuel
            - building
            - other
    - OS control sub-system
        - OS control process
            - direction
                - planning
                - instructing

                - monitoring
                - adjusting
            - assessment
                - testing
                - measuring
                - evaluating
            - support
                - sourcing
                - helping
                - training
                - investigating
        - OS control product
            - direction product
            - assessment product
            - support product
        - OS control resource
            - processor (actor)
            - money
            - fuel
            - building
            - other
- ISETS-ISES interface product
    - ISE technology product-service request
    - ISE technology product
        - development model
        - technique
        - method
        - formalism
        - standard
        - procedure
        - metric
        - tool
        - repository
        - environment
        - other
- ISES-IS interface product
    - ISE product-service request
    - ISE product
        - software
        - hardware
        - firmware
        - communications component
        - procedure
        - documentation
        - other
- IS-OS interface product
    - information product-service request
    - information product
        - data
        - information

PART 2

USING THE ISE CONCEPTUAL MODEL AND TAXONOMY

Part 2 contains informative material for anyone wishing to use the ISE conceptual model and taxonomy in any of the following ways:

1 **A language to facilitate communication:** This involves utilising the framework, concepts and terminology of the ISE conceptual model in discussions relating to ISE. Here the conceptual model and taxonomy provides a common language and shared world view or understanding that will facilitate communication, debate and discussion of issues. See section 9.

2 **Pictures to facilitate elucidation of relationships:** A different kind of use concerns the pictures that can be constructed from the text of the conceptual model. It is possible to position research projects, standardisation work items, models, frameworks, etc. in such pictures and thereby shed light on the scope and relationships of the items. A picture used for positioning need not be formal and the positioning process will not be rigorous. See section 10.

3 **Matrices to facilitate formal comparisons:** A more formal use concerns the construction of matrices from the ISE taxonomies and placing items for comparison etc. in the cells of the matrices. See section 11.

These three different kinds of use are all relevant to the application of the ISE conceptual model and taxonomy to standards management tasks. See section 12.

9. Using the language

This section gives examples of the use of the ISE conceptual model and taxonomy as a language for discussing issues in ISE. This kind of use can be of value in activities such as:

- promoting a common understanding of ISE, its concepts and its terminology throughout Europe;
- promoting understanding of the scope and objectives of ISE standardisation throughout Europe;
- facilitating co-operative working in the field of ISE between different organisations in Europe and worldwide;
- promoting the European view of ISE to the rest of the world.

The issues discussed in this section using the language of the ISE conceptual model and taxonomy are:

- interoperability;
- data quality;
- distributed systems.

9.1 The interoperability issue

This section originated in a trial of the ISE conceptual model and taxonomy for generating and structuring ideas about data interchange as the means of satisfying the important requirement of interoperability.

9.1.1 Definition of interoperability

Interoperability is the ability of two or more processes to use the same product (which may be data or information).

Interoperability is concerned with crossing boundaries. Its importance, especially for defining requirements for standards and functional profiles (and hence for CEN/TC 311), is that it measures the avoidance of potential problems arising from both the construction of systems from existing components and the decomposition of systems into components in order to construct them.

When talking about interoperability, we often talk of ‘sharing’ rather than ‘using’, especially if talking about one process, so that the possibility of concurrent usage is made explicit. Using includes processes having the same understanding of the product either as data or as information. It includes all processes reading the same product and some processes writing parts of the product (under suitable concurrency controls). It includes using the same representation of the product or copies of the representation.

The processes may be in the same system or in different systems at the same level, e.g. in two organisation systems or two ISE systems. Interoperability issues are essentially the same in both cases (different systems can be treated as parts of a larger composite system) but reference to different systems emphasises the fact that interoperability is about overcoming problems arising from integration of systems implemented separately to meet different user requirements and design criteria.[6]

The product may be used as either data or information, i.e. either as the representation itself or as the knowledge that is represented. The distinction between data and information is not a simple fixed binary

[6] On the other hand, we think of providing an interface product for a system at the next higher level as meeting its requirements rather than another case of interoperability.

property. A product is shared as data if the processes have a common understanding of its syntax and as information if the processes have a common understanding of the context of the product and of the semantics that are not captured by syntax. In this definition of the distinction, two financial packages may share an amount of money as data or as information, depending on whether it is represented, for example, as an Integer data type or as a Money data type, respectively. It is convenient to talk of sharing data in an inclusive sense of possibly sharing some or all of the associated information as well.

9.1.2 Requirements for interoperability

Interoperability is an important user requirement at all four system levels.

9.1.2.1 Organisation system

No organisation is self-contained: each one interacts with other organisations, e.g. as supplier, producer or trading partner. For organisation systems, interoperability is treated as being domain specific. For example, two banks may make payments between customers of each bank. This is carried out by transferring data denoting the amount of the payment, and contextual information such as identification of the two bank customers. The two banks have to share a common model for cash payments which resolves such issues as coding of the currency used, how to determine the exchange rate and when to apply it.

The classification of organisations by domain, e.g. into vertical markets such as financial services, health care and manufacturing, enables the central concerns on interoperability between organisations to be addressed. It also generates secondary interoperability problems because domains cannot be defined to be completely independent. For example, a trading organisation will interact with other organisations through a variety of financial services that have to be integrated with the main line of business: buying and selling, payment of employees, payment of taxes (income tax, corporation tax, VAT, customs duty, etc.), provision of state or private pensions for employees, provision of health care for employees. Such things as agreed generic models of the international trade cycle and business transaction models are required that are applicable both within and across vertically classified domains.

9.1.2.2 Information system

The most important quality for interoperability between organisation systems is dependability. Since interoperability has to be based on a precise definition of the shared model of information, it is therefore necessary and sufficient that it should be implemented by automated components of information systems used by the organisation systems. Thus the requirement for interoperability at organisation level can be translated directly into a requirement at the information system level. In other words, what has been said under organisation system also applies under information system.

But the information system brings in new requirements for interoperability, namely the requirement for domain independent infrastructure to support the sharing of data. This can be divided into:

- databases for directly sharing the same data;
- mechanisms for data (or information) interchange (exchange, transfer) for copying shared data for separate use.

Direct sharing of data through the corporate database was once the proposed universal solution of the problem of interoperability, but is best thought of as a special case with more mature technology than is available for the general case. The architecture of databases is well understood and is hidden behind a single API, e.g. SQL. The architecture for data interchange is less well understood (there is no satisfactory treatment in any reference model or the like) and standards concentrate on one aspect, viz. the interchange or transfer format used to copy the data in a file or over a communications link. This is certainly the case at the ISES level, e.g. with CDIF (*CASE Data Interchange Format*) and programming languages, and is also largely true for the IS level, e.g. with EDI. Questions such as data validity are not being addressed (see section 9.2.1).

Direct sharing of data through databases has to be included in the general interoperability architecture for two reasons:

- The sharing of data may not lead to complete sharing of information, since there may be incompatibilities in the schemas or views by which processes access the data. This distinction between data representation and the information added by a schema applies to data interchange as well.
- The common data representation, schemas, etc. that are needed for data interchange enable copied data to be stored in a database where it can be manipulated directly by data interchange tools, e.g. for validating exported data, editing exported data to fit its new context (for different representations of basic data types, for different naming conventions, avoidance of name clashes, etc.), selecting a subset for import, and adding contextual information for controlling import.

9.1.2.3 ISE system

Interoperability requirements for information systems become more specialised when they apply to the ISE system. They also become more difficult to meet since an ISE system is normally constructed from independently supplied components (environments, repositories, tools) without the benefit of mature supported standards to ensure interoperability.

The requirement can be described more usefully at the level of decomposition into the production and control sub-system and in terms of the interface products supplied by the ISE technology system. Separate production and control processes such as analysis and project management are normally supported by a single tool from one supplier.[7] The main concern for interoperability is that products of one tool can be used by another tool, either to enable replacement of the first tool or to integrate products from a sub-contractor who uses a different tool.

Interoperability becomes more complex when two processes interact. This problem is best understood for processes of the production sub-system. The central concern has been the ability of one stage of the production process to use as input the output product of the preceding stage in the production process. Interoperability requirements are met in different ways for different transitions:

- **Analysis and design.** This is the main focus of the related CDIF and SEDDI (*Software Engineering Data Definition and Interchange*) standards. The approach taken is to define a modelling formalism, models of analysis and design techniques in that formalism, and a related formalism for interchanging data that is modelled in the first formalism. This work is helping to crystallise separate techniques for analysis and design, such as logical data modelling and data flow modelling, and to normalise the differences between schools such as ER and OO modelling.

- **Design and construction.** This is normally handled either automatically within a CASE tool (automatic generation of 3GL or 4GL programs) or as a manual process (coding). The reverse process of reverse engineering or re-engineering is handled similarly by a tool or a manual process. In all cases, interoperability can be provided satisfactorily because there are mature standards for

[7] This may change as each area advances and provides more options for automation of alternative techniques.

3GLs and well understood ways of sharing source files, whether they are byte- or record-oriented. 4GLs are not standardised but form part of the proprietary solution to interoperability within a tool.

- **Construction and installation**. The interoperability issue can be reduced to portability and interoperability across platforms. There has been some progress in standards here, e.g. OSF's Architecture Neutral Distribution Format, but the more common approach is the repetition of an automated build process for each of a small number of ubiquitous classes of platform. This process is made reliable by using control processes for configuration management.

The requirement for interoperability between production and control processes is also important but is less well understood. Solutions to date usually depend on failure to separate the control and production processes in the first place: we have just seen an example with configuration management. There is a draft CDIF subject area for project management but this may be more the result of trying to extend beyond analysis and design than recognition of relationships between production and project management processes.

To indicate the range of requirements, let us consider testing. This is a control process for quality management, but is often considered as a production process because of its close links. It uses construction and installation processes to generate versions of the software being tested, probably in a variant form for debugging, monitoring execution recording test results, monitoring performance, etc. Generation is made reliable (the tested product is essentially identical to the released product) by using the same configuration management processes as those used by the production processes (another interaction between production and control processes). Testing is based on manual or automatic generation of test cases from requirements or design products. If the preferred testing tool is based on different models of analysis and design than those used in the production processes, products of the production processes may have to be transformed for use as input for test generation.

Categorisation of the processes, i.e. agreement on their contents and boundaries, is less well advanced for the control sub-system than the production sub-system, so it is not yet possible to discuss interoperability between these processes.

More detailed analysis of requirements may be made by consideration of ISE technology system end products.

9.1.2.4 ISE technology system

The ISE technology system is only of interest for the view of the ISE system provided by the end products of the technology system.

Development models and methods are too high level (vague), so far, to be useful for analysis of interoperability requirements.

Techniques and formalisms define the data and information content of products to which they apply. Detailed analysis of interoperability within an ISE system depends on the itemisation of the techniques and formalisms used by the system and identification of their definitions, e.g. the school of analysis and the specification of the analysis techniques. (Many techniques are available in different dialects, e.g. SADT and IDEF0.) Several techniques and formalisms for construction (3GL programming languages) are already well defined. Those for analysis and design are becoming better defined through standards such as CDIF and SEDDI. Those for installation and operation are less well defined.

Repositories, environments, platforms and tools are resources of the ISE system and mainly of interest in providing openness. In so far as they may contain data that is to be shared by ISE system processes, they must meet the interoperability requirements of those processes. In particular, the *raison d'être* of a repository is to support interoperability and many tools provide an internal repository for some processes.

9.1.3 Conclusion

This trial provides the initial confirmation of the usefulness of the ISE conceptual model and taxonomy for analysis of interoperability. As the more detailed levels of the conceptual model and taxonomy become stable they can be used for more detailed analysis. Alternatively, direct study of interoperability could help in the development of the conceptual model and taxonomy.

The analysis here supports the value of a more detailed study on interoperability and the definition of a reference model or architecture for interoperability. The information system and ISE system are the major objects of study but much can be gained from a better understanding of the issues that are common to both systems.

Such a study is of particular importance for Europe since cultural differences and national boundaries have generated many interoperability problems, for example those associated with national analysis and design methods such as SSADM, DAFNE and MERISE.

9.2 The data quality issue

The issue of data quality is new to ISE and awaits formulation[8]. Data quality has close connections with the issue of interoperability, some of which are explored in this section.

In Europe large volumes of data are being created—especially concerning social and health subjects—intended to provide important information to many information systems in administrations and in other public and private sector organisations. With the increasing reliance on the interchange of data between information systems across organisational and national boundaries has come an increasing recognition of the nature and importance of issues of data quality.

Data quality is as important to businesses as it is to administrations and the healthcare sector. Various companies have estimated their costs due to poor data quality as millions of ecus annually. For example, inaccurate mailing lists cause problems of wasted resources and effort alongside reduced customer satisfaction; inaccurate stock databases cause companies to over-order or under-order; database merges following company acquisitions often cause huge problems.

Ensuring data integrity involves security issues (e.g. password protection and virus checking), human procedure issues (e.g. quality control procedures for human data processing), data correctness issues (e.g. data errors such as duplicate or missing values), database design issues (e.g. data normalisation) and data interchange issues (e.g. ensuring the correct transfer of data between computational sub-systems).

Data quality also involves legal issues: poor data quality is not only bad for the efficiency and effectiveness of an organisation, there can also be legal repercussions. Data protection laws in many countries confer a legal responsibility on data owners for the accuracy and security of their data. A recent study by the European Commission revealed that up to one in ten products in supermarkets may be mispriced, with discrepancies between prices advertised on the shelves and those charged at the checkout

---

[8] Data quality can be defined as the validity of data with respect to its original frame of reference and its frames of reference for any secondary purposes.

due to updates of product databases being out of step with price marking on shelves. These price errors are costing consumers millions of ecus. Over-charging, even unintentionally, is a criminal offence, and there has been a seventeen-fold increase in the number of stores prosecuted for misleading prices over the last five years.

9.2.1 The interchange of data between ISs

This section argues that a valid transfer of data between two information systems is not possible based on ISE standards available today.

In all normal cases communication between two information systems is executed by exchange of data, the data sent being the same as that received. The two communicating ISs necessarily have different purposes, for if they had the same purpose, they would naturally be regarded as one IS.

Data is without sense in itself, but when related to a purpose, data interpretation may extract information. The information sent from one information system is not the same as is received by the second because it is interpreted in the light of different purposes.

As defined in section 7.5.2, data defects can be categorised as defects in validity and defects in relevance. Validity concerns the correctness of data; relevance concerns its interpretation when related to a purpose.

Data that is perfectly valid and relevant for the sending information system can, for the receiving information system, have both validity and relevance defects. New defects in relevance are caused by the change in purpose, while new defects in validity are caused by imperfect transfer of data, for example because the documentation is not fully understood by the receiving IS.

9.2.2 Mathematical statistics and ISE

ISE standards such as the IRDS Export/Import Facility and IRDS Content Modules allow the interchange of data between computational sub-systems within a single framework of concepts, consisting of a form of extended format description (see section 9.1.2.2). However, these standards do not facilitate the interchange of information about known defects in data validity from the sending to the receiving system. Thus data is of less worth (validity) after interchange.

The framework of concepts provided by these ISE standards is also insufficient for the receiving computational sub-system to interpret the relevance of data in an unambiguous way. To find the right concepts for working with incomplete data it is necessary, in fact, to look to mathematical statistics (MS). Aspects of data quality have, under different guises, been an integral part of mathematical statistics for more than a century. MS offers a mathematical and conceptual basis for the treatment of data with uncertainty, biases and gaps.[9] Utilisation of the precise formulations of MS could benefit ISE by making it less dependent on human thinking.

The system concepts of data, interpretation, information and purpose facilitate a mapping to the concepts of mathematical statistics (MS). The following example is in the language of MS:

> *"For a particular purpose, a hypothesis is tested by a chi square test, using a set of observations with known scaletypes of the variables. Appropriate assumptions are made about the unknown aspects of distribution. The test statistic is calculated by a procedure, also calculating the corresponding P-value for a two-sided test. Then the hypothesis is*

[9] But note that the language of MS is very different from that of ISE and MS theory may yet prove to be insufficient for ISE.

> *accepted or rejected. A slightly different hypothesis based on different assumptions is tested using a gamma test on the same observations, using another procedure to calculate the test statistic and P-value. Then this hypothesis is accepted or rejected, possibly with a different result from the first test"*

A similar example in the language of ISE is:

> *"A set of data of known origin is interpreted by an interpretation method based on certain assumptions about unknown factors, extracting the information relevant to a particular purpose. Then an appropriate decision can be made. A different interpretation method based on other assumptions for a different purpose is used to interpret the same data, extracting information, possibly giving a different decision."*

In a computational system, the interpretation method will be embodied in a software program. Thus two computational systems will have different programs for interpreting the same data in a manner relevant to the purpose of the particular system. In other words, two systems with different purposes will generate different information from the same input data.

The mapping between the two examples may be made as follows (it is not implied that the mapped terms are direct synonyms):

MS	**ISE**
observations with known scaletype	data with known origin
chi square test, procedure	interpretation method
assumptions about distribution	assumptions about unknown factors
test statistic and P-value	information
acceptance or rejection of hypothesis	decision

Although a kind of mapping of concepts can be made, a comparison between the ISE standards and mathematical statistics shows major differences. First it is regarded as necessary in MS to know a reference population or similar sort of reference universe for any sort of data. Secondly it is regarded as necessary to declare the known errors, missing data items and biases in relation to population and in relation to sample spaces for single variables. No obvious place is found in existing ISE standards for these concepts. Existing standards describe the format in which data is transferred but say nothing about the reference universe or the defects.

This does not matter when the goals of two communicating ISs are so nearly related that the data is naturally interpreted in the same way. In this case, nothing is to be gained by describing the reference universe and defects. It is when correct data processing and interpretation becomes difficult that a need arises to introduce the concepts of mathematical statistics into ISE. To date this has not been done, but if the necessary standards were developed it would make it possible for a receiving IS to automatically interpret incoming data, where today only assistance by a human resource can ensure that the data is understood.

Not all of the MS approach to data quality can be immediately taken over into ISE: a phased approach will be needed. For example, if fundamental concepts in MS such as reference universe, noise, bias and other defect indicators were to be taken into account in database technology, this technology would probably have to be fundamentally redesigned.

9.2.3 Research and standardisation projects

Where only data of poor relevance is available, as has been largely the case to date, it does not really matter if there are also problems of validation. If a human resource is needed as part of an IS in order to interpret data, nothing is lost if the same human resource must evaluate the validation. Where data of low relevance is concerned, the problem of valid transfer is of minor importance.

The establishing of good quality population registers in many European countries has changed this situation. Exact reference can be meaningful and a description of defects can be utilised by a receiving IS. The effort expended by CEN/TC 251 in standardising hospital informatics points Europe even further in this direction. The problems of data transfer between hospitals, and between hospitals and practitioners, largely arise from an incomplete definition of the data transferred. This means that the data becomes less valid simply by transfer to another IS with another conceptual frame of reference. As a result, the data also becomes less relevant.

Some data is reused for innumerable purposes, for example, population data in countries with a complete population register, data about companies from business registers and databases. Special attention is normally paid to the quality of such data and there is a great deal of experience about measures to take to achieve high quality which could be collected together into a set of guidelines. Guidelines on how to increase the quality of data—particularly when it is used for secondary purposes unknown when the data was created—could be of great economic benefit to Europe because secondary purposes are often in the areas of production control, planning and research. Data is cheap when reused, and the benefits of reuse naturally increase with improved quality.

Because of the level of control of what happens in European society, it is now possible to create data of high validity and relevance. Europe has an advanced position in this area, and a project to study the description of quality of data and its impact on different aspects of ISE (especially on database technology and text processing) is required to determine whether new standards are required or existing standards can be modified. Potential areas of standardisation are the description of data for interchange purposes together with quality management procedures, techniques and methods for controlling data and information quality.

9.3 The distributed systems issue

This section looks at some of the consequences of information distribution from the point of view of their overlap with or impact on ISE. There are a number of reasons for the importance of the issue of distribution, including the fact that non-trivial implementations of ISE and ISE technology systems are frequently themselves distributed systems. Distributability is identified in the ISE conceptual model as one of the significant qualities of information systems while the ISO definition of an information system as

> *"an information processing system, together with associated organisational resources such as human, technical and financial resources, that provides and distributes information."*

emphasises the role of ISs as distributors of information.

9.3.1 OSI and ODP

The ISO Reference Model for OSI (*Open System Interconnection*) is well known. OSI was originally conceived as a programme of standardisation with the objective of allowing the interconnection of computer equipment from many different suppliers. The seven layers of the model structure the functions necessary for a system to have the quality of openness, from the physical interfaces through communications networks to the representation of the transmitted data and the services that effect the transfer of data and information between systems (message handling, distributed transaction processing, distributed directories, etc.). Standards for each of the layers are available or are under development.

Besides a seven layer communication structure, ISO OSI defines standards for management and security. In order to use OSI standards for the management of telecommunication networks, OSI functions have been decomposed into TMN (*Telecommunications Management Network*) standards.

As OSI concentrates on interconnections, it has many limitations. It is difficult to describe the standardised operation of a number of components. Such a description is needed when one system acts as an agent for others, or when the co-ordinated action of a number of systems must be performed according to a fixed set of rules in order to achieve some purpose.

The objective of ODP (*Open Distributed Processing*) is to provide a framework for the standardisation of distributed systems. The emphasis is put on logical relationships between components of the system. The work on ODP brings together the fields of OSI protocols, management, databases and graphics. The target is to provide an architecture within which the functioning of distributed systems can be expressed covering both the individual components and the overall system design of which they form a part.

Systems in ODP are considered from five different viewpoints, using an object-oriented methodology: enterprise (equivalent to organisation), information, computation, engineering and technology. Key developments in ODP are definitions of abstract interfaces, transparencies, aspects and properties of distributed systems, modelling concepts, management and conformance specifications. ODP architecture will allow the general expression of requirements for consistency between the human interface, the programming interface which supports the application and the OSI protocols which convey information between systems.

As well as ODP as developed by ISO there are two other responses to the trend towards a global distributed processing model and standardised platforms. These are the Distributed Computing Environment (DCE) of the Open Software Foundation (OSF) and the Common Object Request Broker Architecture (CORBA) of the Object Management Group (OMG).

9.3.2 The management of distributed information systems

As defined in the ISE conceptual model, the four systems of ISE have both production and control sub-systems. The control sub-system manages the production sub-system by directing, assessing and supporting the processes, products and resources of the production sub-system. There is a wide range of control activities associated with the different stages of the IS lifecycle. For example, in the design, construction and installation stages control of projects, risk, change, documentation, quality, etc. is required. During the operational (use) stage the management of such things as applications, communications services and networks is required. One of the goals of ISE is to work towards the integration, consistency and standardisation of the different aspects of control sub-systems in all stages of the lifecycle.

A further issue is the convergence of information and telecommunications systems. The telecommunications infrastructure, with its services, provides the platform for distributed information systems. In the EU the RACE R&D programme has been concerned with the development of infrastructures, methods and techniques to facilitate the availability of integrated telecommunications services while the standards body ETSI has focused on the translation of the RACE work into standards. The European telecommunications infrastructure provides the means for organisations to utilise virtual private networks.

In the contemporary heterogeneous, multi-vendor environment, control sub-systems must interact across organisational boundaries. Increasingly complex interactions between information systems on different platforms and/or in different domains are needed, leading to a need for interoperability between different standards tracks in the areas of information systems, information systems engineering and telecommunications (e.g. ISO OSI, SNMP and TMN).

The introduction of pan-European telecommunications infrastructure requires a consistent approach in the integration of distributed systems and networks management. The systems will be composed of component distributed systems bound together by a backbone of very capable communication facilities and selected support services. Each of the component systems will be controlled and operated by a different administration. The divergence is from a traditional management approach (e.g. centralised or hierarchical) to one that may be described as federated. Each organisation retains the responsibility for managing the system within its organisation's scope, while allowing interaction with other systems in a carefully prescribed way.

The analysis of requirements of the operation and administration of distributed systems in an open environment should provide the basis for establishing a framework. The framework should be scalable and extensible to allow operation in various domain relationships ranging from centralised to negotiated federations. The framework should include a reference model and architecture, service definitions, metrics and methods for applying them to specific implementations. This work would form part of the larger interoperability study proposed in section 9.1.3.

Standards and profiles are developed within the context of a framework. Profiles provide more functionality than standards. Profiles are developed by consortia (NMF, Omnipoint, TINA-C) and governments (GOSIP—UK *Government OSI Profile*). EWOS (*European Workshop for Open Systems*) is concerned with developing system and network management profiles for the EU. The effective utilisation of distributed system management profiles in the EU will be an important area for ISE.

10. Using the pictures

A second use of the ISE conceptual model and taxonomy concerns the pictures that can be constructed from the text of the conceptual model (for example, the figures in section 7[10]). This kind of use can be of value in activities such as:

- facilitating the translation of user requirements to areas of ISE standardisation and certification;
- identifying areas of standards and profiles needs, gaps and overlaps.

It is possible to position research projects, standardisation work items, models, frameworks, etc. in such pictures and thereby shed light on the scope and relationships of the items. A picture used for positioning need not be formal and the positioning process will not be rigorous—items rarely belong just to one place. Indeed the value of using a picture in this way can come from the debate provoked by an attempt to position an item. Once this value is extracted, attempts to produce formal mappings will probably not be needed. The picture provides a catalyst to discussion, not an end in itself.

As an example of this kind of use, this section positions some standards-related work developed elsewhere on a picture taken from the ISE conceptual model.

Positioning in the picture (Figure 10-1) has been done informally by judging the main focus of particular item and placing it in the relevant place in the picture. Evidently, where particular aspects of the ISE conceptual model are not shown in this picture (for example, the qualities are not shown) positioning in relation to those aspects is omitted.

Standards work positioned in Figure 10-1 are:

1	ISO 9000/EN 29000
2	EDI
3	OSI
4	ETSI
5	IRDS
6	Security
7	SPICE
8	POSIX
9	PCTE
10	Euromethod
11	Guidelines for selection of CASE tools (ISO/IEC JTC1 SC7)
12	EPHOS
13	Quality of service
14	DISC Framework for User Requirements

[10] Note that other pictures (and diagrammatic models) can legitimately be constructed from the text of section 7. The pictures in section 7 are not the only ones that can be used should different pictures be more appropriate.

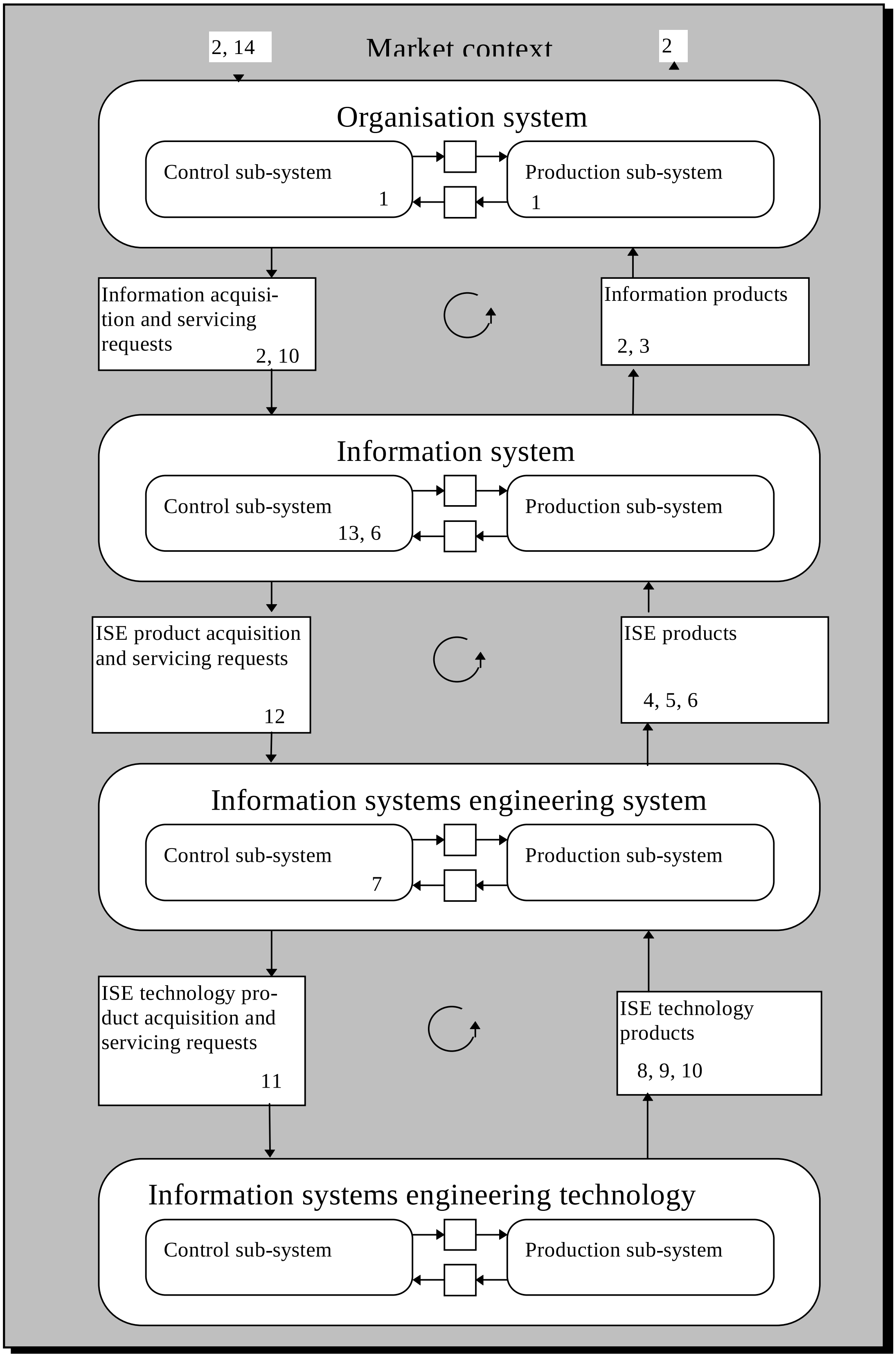


Figure 10-1 Positioning other standards work in the ISE domain

11. Using the taxonomies

The taxonomies of ISE entities and qualities defined in section 8 can be used in more formal activities of classification and comparison involving the use of matrices. This kind of use can be of value in activities such as:

- facilitating the translation of user requirements to areas of ISE standardisation and certification;
- identifying areas of standards and profiles needs, gaps and overlaps.

11.1 Classification of information systems

The classification of information systems can be undertaken based on the premise that every information system is described sufficiently by a set of qualities chosen at the same level of abstraction, each of them measurable for every system analysed. Where this is the case, every system can be represented by a single point in a multi-dimensional representation space. The taxonomy to be used in delimiting such a representation space is defined in section 8.2.2. This taxonomy structures the non-functional qualities, but the functionality of ISs can be used as a further dimension of the representation space if required.

The similarity between systems can be measured by the distance separating them. Clusters of systems having similar qualities are represented by clouds of points in the representation space. The existence of those clouds suggests the possibility of reducing the variety of systems by dividing the space into a set of non-overlapping volumes. Every volume contains one cloud of points and designates a class of systems.

Every cloud is represented by one system which corresponds to a particular point of the representation space. The point chosen for representation will model the qualities of all systems belonging to the class.

The classification process is easier for discrete value qualities than for continuous qualities. Equally, the complexity of the classification procedure for discrete-valued qualities decreases with the cardinality of the value set. The simplest classification procedure is obtained if every quality has a discrete, binary value set, but this is not always achievable or useful in practice (e.g. software integrity levels).

There is also the difficulty caused by the number of features determining the dimensionality of the representation space. Large numbers of qualities (and hence dimensions) are difficult to understand and learn; on the other hand, an excessively reduced set of qualities will cause the model to be too far from reality and to have no practical utility. The complexity of classification increases with the number of dimensions.

Although most real systems exhibit all the qualities to some extent, the significant thing here is whether the quality in a particular system is sufficiently important to influence the ISES. Thus, as far as possible a binary metric should be used in each dimension and the dimensions selected for a particular analysis should be carefully chosen and limited.

11.2 Classification of the other concepts of the ISE domain

To use the ISE conceptual model and taxonomy for formal classification purposes it is necessary to be able to define a representational space that can hold all the concepts of ISE. For this reason the decomposition of the ISE concepts given in the taxonomy of ISE entities (section 8.2.1) is needed for the dimensions of a representation space, together with or independently of any quality dimensions used to group or position particular instances of ISE entities.

11.3 Developing a comparison matrix

Positioning or mapping a standard, user requirement or research project in relation to the ISE conceptual model and taxonomy involves the following steps (described in terms of the mapping of a standard). Note, however, that the quality (consistency and coherence in particular) of the item in question will largely determine how easy the positioning is to carry out and how useful the results are.

1. Study the content of the standard.

2. If the focus of the standard lies within the ISE domain, identify the concepts which the standard addresses and map these to the ISE conceptual model. The results of this mapping may be represented in diagrammatic form, showing where concepts of the standard overlay those of the conceptual model (see section 10).

3. If a more detailed mapping is required, produce a matrix of relevant entities and qualities from the ISE taxonomies (restricting the number of dimensions to those that are essential, for the sake of ease of comprehension). Mark those cells of the matrix to which the standard maps.

12. Supporting standards management

The ISE conceptual model and taxonomy can function as a tool for standards management, to be used, for example, in assisting CEN/TC 311 in the exercise of its mission and the fulfilment of its objectives. This section discusses objectives of standards management that can be supported by such a tool, including:

1. Relating standards to each other:

    - standards which are related to each other should not contain unnecessary overlap;
    - standards which are related to each other should be consistent with each other;
    - standards which are related to each other should have an interface which is clearly identified and defined;
    - a product based on a group of two or more standards should be able to be developed without prejudice to any member of the group of standards.

    See section 12.1.

2. Determining standards coverage:

    - the need for a new standard to fill a gap left by existing standards should be easily identified;
    - standards committees should be scoped in such a way as to reveal overlaps and gaps and to facilitate the allocation of new work items.

    See section 12.2.

3. Relating user requirement categories to standards. See section 12.3.

The scope of ISE, ISE standardisation and CEN/TC 311 is discussed in section 12.4.

12.1 Relating standards to each other

This standards management objective is concerned with how pairs of standards relate to each other### whether they overlap, whether they are consistent and whether they have a common interface.

- **No unnecessary overlap between standards:** Positioning a standard in conjunction with other standards is the ISE conceptual model (using pictures and/or matrices) will identify the existence of overlaps. However, such positioning does not identify the nature of the overlap, nor does it reveal whether the overlap is necessary or not. An overlap may take the form of data generated by one standard which is used by another standard; this situation is better treated as an interface (see below). It may take the form of the same kind of data being generated by each standard (and subsequently used by neither) or the same kind of data being used by both standards (and generated by neither). These overlaps must be assessed to establish whether they are necessary for both standards and whether their form in the two standards is compatible. In the area of safety-critical software standards, for example, many standards could be merged (e.g. IEC 880 in the nuclear field and DO-178B in the civil avionics field). However, this does not happen because the standards have a different community of users, even though they address identical issues. Even if an overlap is necessary, it is important to ensure that it is handled in a way which at best ensures (and at least enables) compatibility between pairs of products based on the overlapping standards.

- **Consistency between related standards:** Consistency is the degree of uniformity and freedom from contradiction within or among entities or their parts. It is important to ensure that the relationship of consistency between any pair of standards is handled in a way which at best ensures (and at least enables) compatibility between pairs of products based on the two standards.

- **Defined interfaces between standards:** Given the disparate nature of ISE standards, the concept of an interface between a pair of standards must be carefully distinguished from the concepts of overlap and relationship. An interface between two standards exists if data generated by one standard is needed as input to another standard. Any such interface needs to be clearly identified and defined (in the later of the two standards at least).

- **Product based on two or more standards:** This is the most difficult objective for the standards management function to achieve. At the same time it relates to one of the major objectives of ISE standardisation, namely, interoperability (see section 9.1). Defining the relationship of standards helps identify groupings of standards which are likely to be used in the development of a particular product. The only way to ensure compatibility on this level is to adopt a consistent way of defining standards. This is particularly important for the data aspects of each standard.

The discovery of overlap may highlight the need for a clearly defined interface, or it may represent unfortunate duplication which can, and probably will, cause consistency problems. In order to investigate a suspect situation in more detail, it may be necessary to develop a model of each standard in the pair in a form which enables more detailed comparison (typically a process model and/or a data model). Adopting a process modelling approach (where applicable) facilitates the elimination from the modelling of any data that is not explicitly used by any of the processes in the standard.

When models are available for both standards, the lists of process and data types can be examined for common entities. Terminology is the cause of many difficult problems during this comparison. Common process and data types should be noted and steps taken to ensure that these are handled in the same way in both standards. Relationships between pairs of common data types should also be noted and a common approach agreed. Since common processes will each be related to one or more data types, it will again be necessary to ensure a common approach across the two standards.[11]

When consistency on the data type and process level has been achieved, the questions of overlap and interface can be tackled. The concern here is the problem of implementing a product which conforms to both standards. It is the common data types which have to be considered. It may be possible to regard one standard as the owner of the data type and the other as the user (i.e. the first standard generates the data and the other standard uses it).

The availability of the ISE conceptual model and taxonomy contributes to problem prevention rather than problem solving by facilitating consistency between independently developed standards and reducing the need for the detailed modelling and comparison of pairs of standards. The questions to ask are:

- do all the processes of this standard have an identifiable place within instantiations of the ISE conceptual model?
- does the ISE conceptual model have an equivalent for each of the data items mentioned in the standard and for the processes needed to use that data?

12.1.1 Example of related standards

Figure 12-1 illustrates the relationship between some software engineering standards and the quality of reliability (see section 7.5.1.1, item 1a). The standards referenced in the diagram are ISO 9126 *Software quality characteristics*, DO-178B *Software considerations in airborne systems and equipment*

[11] ISO draft standard 11404, *Language independent datatypes,* is relevant here.

certification, ITSEC *Information technology security evaluation criteria* and pr EN50128 *Railway applications: software for railway control and protection systems*.

Comparison of these standards and their coverage of the concepts of reliability and maintainability, together with the fact that the standards are not able to interface with existing standards for software engineering, reveals tasks to be addressed by standards management. These tasks concern the consistency and coverage of standards dealing with software engineering techniques and with software and system dependability qualities.

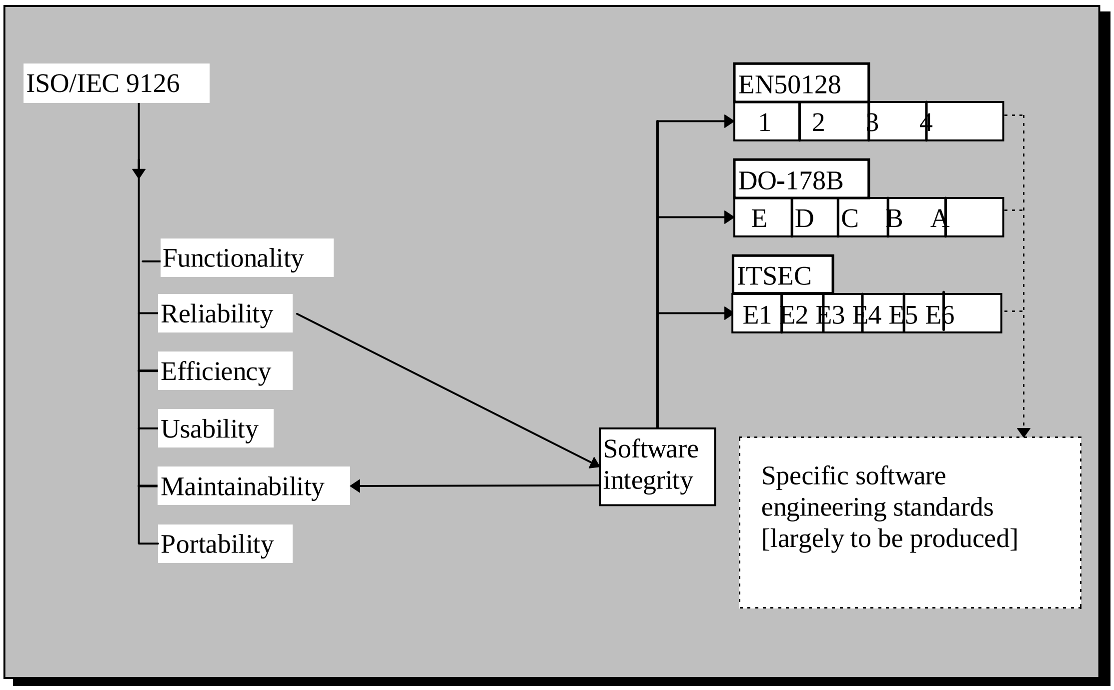


Figure 12-1 A mapping of standards concerned with reliability

As noted in section 7, reliability is best measured in terms of the mean time between failures (or something similar). Unfortunately, during the process of software development it is impossible to predict reliability, even though such a prediction can be made for hardware sub-systems based upon known component reliability figures. Even with hardware, it may not be possible to produce accurate reliability figures due to uncertainties in the reporting mechanisms.

Several standards concerned with safety have a requirement to give high assurance on the reliability of the software component of a system even though it cannot be quantified. This is overcome by introducing the concept of software integrity (see section 7.5.1.1, item 1) which is variously placed on a scale of between four and six levels. The standards specify that to conform to a specific software integrity level, certain software engineering requirements must be met. For instance, for level A airborne civil avionics software conforming to DO-178B, the testing must include modified condition decision coverage.

The advantage of this approach to software reliability is that producers know which software engineering measure to apply in order to claim that the software is fit-for-purpose. The disadvantage is that the specific measure of reliability may not be obtained or may involve greater cost by obtaining a higher figure.

The mapping of the software integrity level onto specific software engineering requirements should ideally be by reference to specific software engineering standards, establishing an interface between the standards. However, at least for safety issues, existing software engineering standards do not give the level of assurance needed. So, for example, DO-178B does not refer to other software engineering standards, but instead defines the requirements explicitly, including the requirement for modified condition decision coverage noted above.

12.2 Determining standards coverage

This objective concerns the identification of gaps where standards and standardisation responsibilities are needed. Information on existing, emerging and proposed standards can play a useful role here. Examination of the standards will reveal facets of ISE where there is no coverage by any standard. In such cases, the first question to ask is whether a standard is needed (is there an established user requirement which would be satisfied by the availability and use of the standard). The second question is whether a new standard should be developed or whether an existing standard should be enhanced. The third question is whether the new standard falls within the scope of an existing committee and if not, whether a new committee or a change to the scope of an existing committee is needed.

- **New standards to fill gaps:** Positioning each existing and developing standard in the ISE conceptual model and taxonomy will identify gaps. It should also assist in the decision-making to determine whether each gap should be filled by initiating a new standard or by extending an existing standard.

- **Scoping standards committees:** On the basis of the structured scopes for a group of related committees, it should be possible to determine whether a more detailed analysis is necessary in order to identify gaps and overlaps. The resulting picture will facilitate the allocation of new work items to particular committees.

12.3 Relating user requirement categories to standards

User requirements are rarely expressed in terms of requirements for standards but in terms of problem areas confronting the user (for examples of such concerns, see Annex B). It is the task of standards developers and management###in deciding or prioritising, for example, work items###to relate such requirements to actual and potential standardisation activities that could contribute to the solution of the problems users are experiencing or the issues they would like to address. The problem for standards managers is how to translate user concerns into specific user requirements that can in turn be translated into specifications of ISE standards to meet those requirements.

Some work has already been done (by DISC###the IT division of the British Standards Institution) on how to map business needs arising in organisation systems to information service requirements. This addresses part of the ISE domain (the organisation system and the components of the information system) but further work remains to be done for this method to be adequate for relating user requirements to the totality of ISE standards.

12.3.1 The DISC **'Framework for User Requirements'**

The DISC *'Framework for User Requirements'* (FUR) was developed by a working group of the BSI DISC Business Strategy Forum to provide a framework in which users can analyse their requirements and express them in terms of services which can be provided by information technology. In the FUR, users are equated with businesses. The FUR adopts the premise that users cannot be expected to state their requirements directly in terms of needs and priorities for individual technical standards, but rather that a bridge is needed—some concepts which form a common language between users and those responsible for the development of standards and technology. The perceived beneficiaries are both the user organisations (businesses) and those involved in developing information technology, components and associated standards.

A complementary *'Framework for Standards and Technology'* was envisaged, this being expected to emerge from work done elsewhere.

The FUR does not prescribe how the analysis of requirements is done, expecting that different businesses will have different starting points and choose different approaches. However it is considered that organisations should be able to relate to the concepts of business functions, business data and business systems and identify instances of these. Thus the FUR does not prescribe the processes by which the organisation system and its use of information services are engineered, but does provide a structure in which the organisation system and its requirements can be expressed.

The principal concepts of the FUR, and its relationship to the *'Framework for Standards and Technology'* and the ISE conceptual model and taxonomy, are shown in Figure 12-2.

In terms of the ISE conceptual model, the FUR—including the information services plus the components which provide them (where these are not out-sourced)—corresponds to the organisation system. The collection of information services plus whatever provides them corresponds to the information system. The information system engineering system is not shown nor are any of its processes prescribed.

In the FUR, a strategic business factor is treated as a characteristic of an organisation (these might better be expressed in terms of strategic business objectives and policies). The organisation has a number of business functions to achieve its objectives in a manner consistent with its policies. A business function is a discrete group of related activities, a convenient grouping of functionality that supports one aspect of the mission of the organisation. Business functions may or may not be equivalent to a department role, and are essentially independent of the organisation structures that undertake them.

These functions are achieved by a number of business activities (a concept not shown in the diagram), supported by business systems operating on and delivering business data. A business system is a system which performs one or more of the tasks of one or more business functions by transforming a set of inputs, using a set of rules and procedures, to produce a set of outputs. Business data as defined in the FUR is that information which is needed for the activities of the business. It can encompass raw facts, structural data, information (processed data), knowledge (interpreted information) and wisdom (knowledge combined with experience).

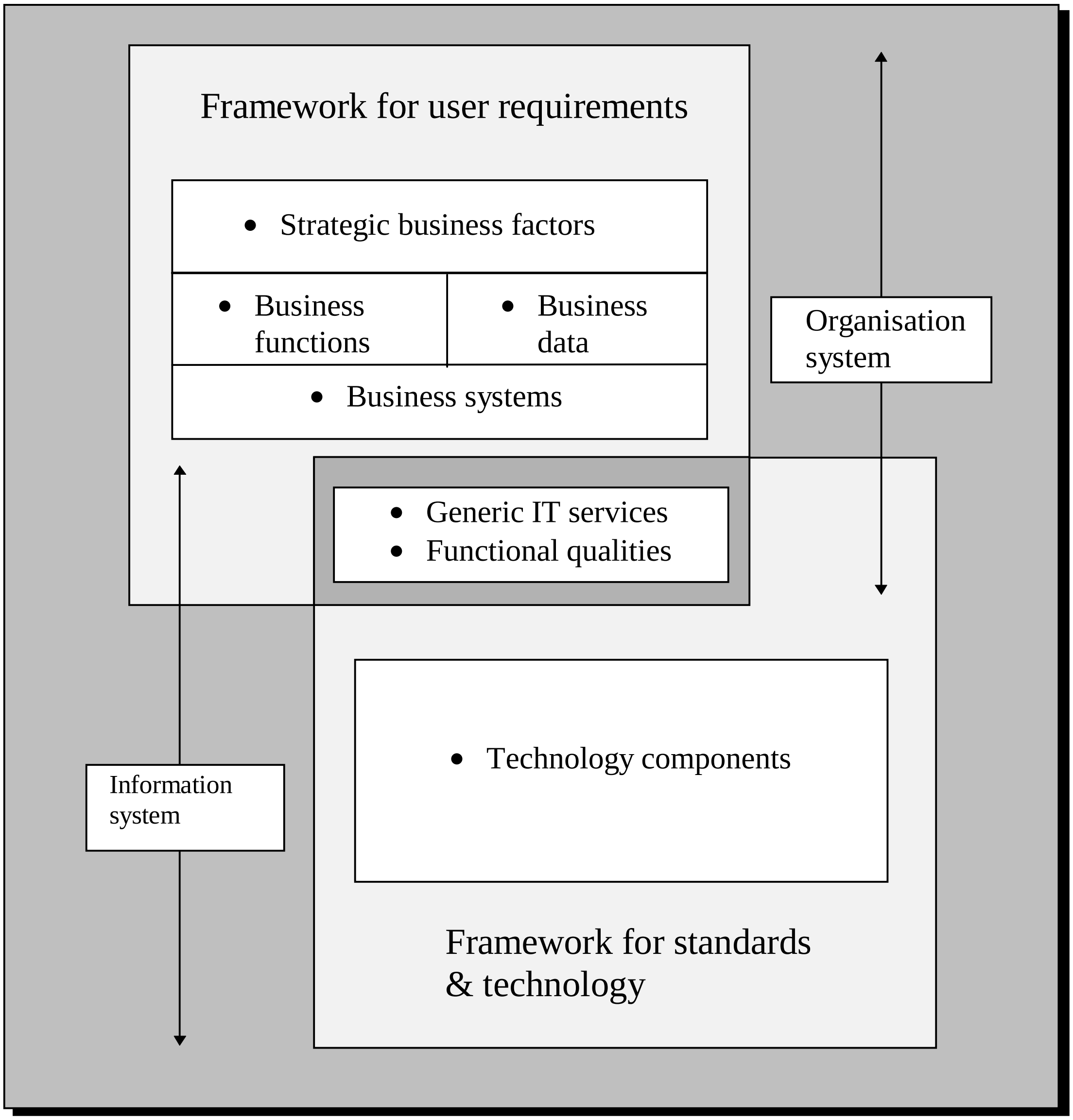


Figure 12-2 The DISC 'Framework for User Requirements'

A business system makes use of information services. These must not only deliver the functionality required but have functional qualities appropriate to the requirements of the business function, attributes which determine the levels of service (e.g. of availability and response time) and the types of information operated on.

In the FUR it was envisaged that the information services provided by IT could be mapped onto one or more of a set of defined generic IT services (e.g. a message distribution service), comprehensible from both user and supplier viewpoints, which could thus provide a link between the two frameworks; an initial set of such services is included in the document. Using these has in practice led to difficulties and this aspect of the FUR needs further work. Other ways of relating information services to generic information systems have been used.

The FUR has contributed to a greater or lesser extent to a number of processes for capturing user requirements (e.g. in X/Open). In the UK it has been used both by DISC as the basis for a survey to capture user priorities to influence the standards programme and by CCTA as the basis of a process in a project to capture the requirements of government departments for future information technology. These exercises have shown that with some minor modifications the framework can successfully be used to analyse the objectives and the functions of organisations, and as a result to identify business systems and their functional and qualitative requirements on IT support. Further work is now needed to consolidate changes and address residual difficulties identified as a result of its use.

Work was done in DISC on a *'Framework of Frameworks'* (FoF) with the objective of enabling various frameworks, models and areas of standardisation to be related and so to assist the management and prioritisation of activities, and to assist standards developers to relate their work to that in other areas. The picture in the FUR was developed and extended to provide more detail on the IS and to cover areas in the ISES and ISETS. Various pertinent frameworks, models and work items were discussed and positioned in it and some ideas for further work outlined. There is considerable overlap between the objectives of FoF and those of the ISE conceptual model and taxonomy. Ideas and material from FoF have been carried across into this document.

12.3.2 Mapping user requirements to the ISE conceptual model

It is important that the presentation of the user requirement can be clearly related to the scope of the relevant standards. The comparison of a mapping of standards with a mapping of categories of user requirements to the ISE conceptual model and taxonomy provides the basis for determining correspondences and gaps. Narrative statements of the comparison should play a supplementary role to any structured presentation (it should not be necessary to prepare detailed models). The results should be comprehensible to individuals not involved in the details of the standards.

The steps to be taken to map user requirements to the ISE conceptual model are essentially those described in section 11.3:

1. Study the content of the concern or issue;

2. If the focus of the concern lies within the ISE domain, identify the concepts which the concern expresses and map these to the conceptual model. The mapping cannot be done until the concern has been expressed in terms of requirements for ISE products or services: domain knowledge and expertise is essential for this step. The results of this mapping may be shown in diagrammatic form by showing where concepts of the concern overlay those of the conceptual model;

3. If the concern maps directly and solely to the concepts covered by the FUR, the FUR approach can be used to relate the user requirements to information services. However, the further mapping to relevant standards or areas of potential standardisation must still be done.

4. If a more detailed mapping is required, produce a matrix of relevant ISE concepts and qualities from the ISE taxonomies (restricting the number of dimensions to those that are essential, for the sake of ease of comprehension). Mark those cells of the matrix to which the concern maps.

12.4 Scoping ISE

12.4.1 ISE

ISE is concerned with:

- the organisation system and its context in respect of the analysis of information flows, information requirements and all related factors necessarily considered in the provision of an information (and data) solution[12];
- the information system in respect of its specification (components, qualities, etc.), its acquisition, its development and the support and control of its operations;
- the information systems engineering system in all its aspects;
- the output products of the ISE technology system (there is potential for it to be concerned with the production and control sub-systems of the ISETS, but as yet this area is ill defined and probably best treated by extension of the production and control sub-systems of the other systems).

12.4.2 ISE standardisation

It is clear that ISE—and hence ISE standardisation—covers a wide scope, from the analysis of user requirements onwards, from human, information technology and telecommunications considerations, and so on. The difficulty is to limit it not to cover every standard in the ICT field.

One possible area to consider for exclusion from ISE standardisation is that of the interworking standards of the operational system. However this is debatable. If engineering a system for which these standards are not laid down, then it is an engineering task to select or develop standards for that system (so ISE does produce them). If engineering a system for which they are laid down, then the standards give guidance and constraints on how the engineering is done and on which components are selected. Therefore, while devising them is not in the scope of the ISE system, they are actually deliverables to the ISE system—and are deliverables of an ISE technology system. So any general purpose interworking standardisation programme could be considered as part of an ISE programme. Take the analogy of screw threads in mechanical engineering—a thread standard is an engineering standard and devising a set of such standards is part of the field of engineering.

The scope of ISE standardisation must cover all aspects of ISE products (information system components) and all standards for ISE product interoperability. Data standards must also be covered, although the nature of the data content (what it means as information to the end user) will in general be determined by the overall analysis of the organisation, not the design of the IS and its components.

It is important to recognise that many committees work within the field of ISE standardisation (see Annex C) rather than trying to restrict the domain to an area that can be the focus of only one standards committee.

12.4.3 CEN/TC 311

As CEN/TC 311's focus is a strategic, top-down view of ISE standardisation to ensure that Europe's requirements are met in a beneficial and consistent manner by standards development work being done elsewhere (i.e. it is largely concerned with studies and investigations rather than with making standards), it is in fact beneficial for it to have a wide scope. The wide scope will ensure the freedom to investigate new areas and cross-boundary topics as existing and emerging European needs dictate. Individual task groups can be set up with the right experts to address identifiable subjects, under the aegis of CEN/TC 311.

---

[12] The requirements of an organisation in its market context cannot be analysed from top down for information system requirements as a process distinct from the engineering of the organisation in other respects. Decisions on the locations of warehouses, the implementation of 'just in time' manufacturing, and so on, necessarily interact with supporting information system services and infrastructure. ISE, therefore, does not cover the whole of organisational analysis, but it is a component of and contributes to organisational engineering.

With the acceptance of a wide scope, the concern for standards management is to ensure that CEN/TC 311 has a good mechanism for prioritising its work so that it is not overwhelmed by the scale of the domain of ISE and to ensure that CEN/TC 311 does not duplicate work being done elsewhere.

1. Annex A ISE in context

This annex outlines, as background material, the understanding of information systems that underlies this document—their nature, their role in the modern world and the impact of that nature and role on how they are developed and supported. The ISE conceptual model must be sufficiently abstract and flexible to support this current understanding as well as any evolved understanding that may emerge during the lifetime of the model.

The importance of information systems to Europe is brought home by the Bangemann report '*Europe and the Global Information Society*':

> *"Throughout the world, information and communications technologies are generating a new industrial revolution already as significant and far-reaching as those of the past.*
>
> *It is a revolution based on information, itself the expression of human knowledge. Technological progress now enables us to process, store, retrieve and communicate information in whatever form it may take—oral, written or visual—unconstrained by distance, time and volume.*
>
> *This revolution adds huge new capacities to human intelligence and constitutes a resource which changes the way we work together and the way we live together.*
>
> *Europe is already participating in this revolution, but with an approach that is still too fragmentary and which could reduce expected benefits. An information society is a means to achieve so many of the Union's objectives. We have to get it right, and get it right now."*

1.1 Information systems

Many of the activities of an organisation can be described as an information demand and supply cycle as shown in Figure A-1.

The information sources required to meet the needs of a particular organisational activity may be:

- homogeneous or heterogeneous;
- located on a single site or distributed across many sites (local and remote);
- internal and/or external to the organisation.

An information system (IS) unites one or more such sources, together with the necessary processing capability (human or machine), into a single logical entity that supplies the demanded information.

Information systems have the defining characteristic of supporting the information and communication requirements of an organisation or a constellation of organisations.

There are many different types of information system, including database systems, knowledge-based systems, decision support systems, expert systems, document retrieval systems, information disclosure systems, information browsing systems, real-time information systems, office information systems, management information systems, command and control systems, process control systems, mail/messaging systems, and many more.

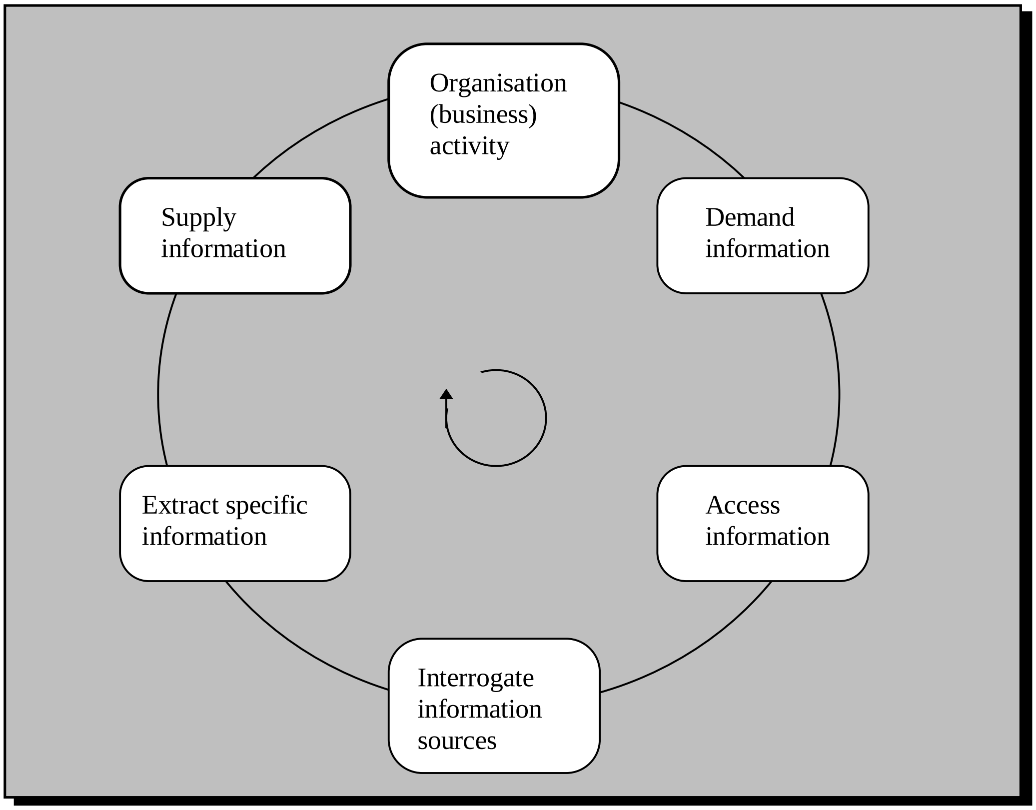


Figure 1-1 The information demand and supply cycle

Information systems are all pervasive. Information systems are required, acquired and used by an organisation or a group of co-operating, associated or related organisations. The acquiring organisation(s) may be of any size—small, medium or large—and may operate in any sector. Organisations range in duration and significance: a project, a business, a hospital, a university, a ship and an army are all examples of organisations.

Organisations use information systems to provide services which assist them in their operations and which support them in meeting their strategic goals. The effective exploitation of such systems—especially where information and communications technology (ICT) components are used—can give an organisation competitive advantage, improved productivity and, indeed, the opportunity for radical redesign of operations both within the organisation and across whole supply chains.

Increasingly, information systems do not exist in isolation but communicate with each other, crossing organisational, cultural and national borders. The implications of this fact are particularly important in Europe as the process of creating the European Union gathers pace and organisations in different countries collaborate in ways that require the sharing, exchange and distribution of information. Where this communication is automated, for example where EDI (*Electronic Data Interchange*) is used to transfer information from one organisation's computer systems to another, questions of information and data interpretation, integrity, reliability and validation arise. Initiatives such as IDA (*Interchange of Data between Administrations*), ENS (*European Nervous System*) and TEN (*Trans European Network*) make this a key issue for Europe.

Information systems with open (public) access are creating new kinds of organisation. The Internet is an example *par excellence* of this phenomenon: every user becomes an integral part of a global 'virtual' organisation with none of the hierarchical control structure and rules traditionally associated with human organisations.

1.2 The engineering (provision) of information systems

An organisation has certain strategic objectives and policies. It establishes a structure of functions to achieve its objectives in a way consistent with its policies. These functions are achieved by a number of operations supported by information system services operating on and delivering information. The services must not only deliver the functionality required but be supplied with qualities appropriate to those required by the function (e.g. availability and response time) and to those required by the overall policies of the organisation (e.g. a security policy).

The requirements of an organisation and its end users lead to the need for and acquisition of information systems with particular qualities and particular components to support these requirements. These requirements may arise from internal needs, but more often they arise from pressures, needs and constraints generated by the outside world—from other organisations and entities interacting directly with the organisation or acting indirectly by creating market forces etc. to which the organisation must respond.

Information systems have always existed in organisations. Information and communications technologies have enabled improved implementation of ISs, automation of many processes and invention of new services. When an IS is found to deliver poor service, or when better technology appears, or when a new requirement for information arises, an IS development process is initiated involving the engineering of IS components and their integration into effective operational systems.

The components of the IS—some implemented in ICT hardware, some in software, some in other technologies, some human—must all fit together and interwork; standards to ensure this is possible will normally be utilised. The choice of these standards will in part reflect overall policies for the information system (e.g. a policy of open procurement). All information systems need to be developed by means of a suitably controlled process, in which the use of well-tried standards, tools and components can be of great assistance.

Information systems involve human and organisational issues as well as equipment and ICT issues. When any one of these aspects is changed, the other aspects are influenced too. The provision of information systems therefore includes training, motivation and other stimulation, changes of work tasks and responsibilities, resource allocation and other organisational developments, as well as the design and construction of computer sub-systems. Scientific knowledge and methods for personnel and organisational development can be found in the social sciences. Development of computer sub-systems can benefit from knowledge and methods from the various engineering disciplines. The challenge in information systems engineering is to find the right blend of socially and technically oriented methods.

The term engineering is used here in its widest sense to cover the provision of information systems. This provision entails not only the development of the system and system components but also the operational provision of the resulting information services (hence management of the system, service level agreements, etc.) and, indeed, all the elements of the IS life cycle.

1.3 Information system adaptations

An organisation and its IS requirements are not static. The more an organisation can retain flexibility in its exploitation of information systems, the more it will be able to maintain competitive advantage and exploit existing investment as its objectives, activities and environment evolve. What is needed is effective management of change in which an organisation's information systems not only assist change but can be evolved smoothly to support rather than impede change.

The geographic distribution of the organisation, its suppliers and its customers—and the emergence of the 'virtual' organisation linked solely by its common ISs—is another input to these needs. There are requirements to be able to design distributed systems and to federate existing ones. Company acquisitions are a particular occasion for the latter, but the need to share information between different departments is a more frequent if less dramatic cause.

Not only does this lead to requirements on the flexibility of how IS components are used and interconnected, but also to a need for requirements to be expressed at some level in common terms and, where possible, to call up a common set of building blocks using common engineering techniques. Otherwise it is not possible for suppliers to aggregate those requirements pertinent to a common product, whether component, system or sub-system.

Evolution imposes constraints as well as opportunities on investment protection and exploitation. Few if any organisations are in a green field situation in their use of information systems, or their analysis of needs. Likewise few if any IS suppliers are in a green field situation. Exploitation of existing investment and the integration of legacy systems, even where not ideal, is essential. Hence the provision of information systems is an essentially evolutionary process, with existing components being used or enhanced where practicable.

Many information systems are redesigned, undergo re-engineering or are subject to major modifications, often including integration with other systems. Empirical studies indicate that such effort accounts for more than half of all development projects. The term used by Euromethod for the development of an information system is 'IS-adaptation', reflecting the reality of how ISs are engineered.

1.4 The importance of information systems engineering

With the growing reliance of organisations of all kinds on modern information systems, the quality of those systems becomes ever more significant. The key factor to ensure that an information system is fit for its purpose is its functionality, but non-functional qualities can be equally important, e.g. openness, usability and dependability. The quality of information systems necessary for Europe to get its response to the information revolution right simply cannot be achieved by *ad hoc* and amateur development of information systems: it is dependent on skilled, professional practices—on information systems engineering.

Information systems engineering is the provision of information systems of required quality in organisations. This provision involves the processes of analysing the organisations' information system requirements in the context of strategic and operational goals, producing designs to meet those requirements, building systems that realise those designs, installing tested systems, supporting operational systems and their users and enhancing or evolving systems as the organisations' strategies, goals and information system requirements evolve. Information systems engineering is required to support business goals of improved quality, better customer responsiveness, lower costs and faster time to market. Increasingly businesses are reorganising around their information flows to reduce the cost of duplicated effort and inefficient or redundant activities. This reorganisation, involving the study of information flows within one or more organisations and the redesign of business processes (often called business process re-engineering), has information systems engineering at its heart.

As information systems contribute directly to the achieving of both strategic and operational goals of organisations, nations and communities—embracing goals such as efficiency, productivity, competitiveness, responsiveness and customer satisfaction—the effectiveness of the engineering that creates, supports and evolves those systems can be seen to be crucial.

1.5 The importance of ISE standards

Through ISE standardisation, organisational effectiveness and business competitiveness can be greatly increased. The key to the successful exploitation of standards is a clear understanding of the different kinds of benefits that the adopting organisation may derive. Factors which may motivate European organisations to adopt ISE standards range from bare necessity to sophisticated self-interest, for example:

- treating standards as representative of best practice, where the organisation uses standards to effect process improvement and better quality products and to achieve financial benefit;
- achieving ISE maturity, where the organisation has, of its own volition, integrated a broad range of ISE standards into its corporate strategy and is highly sophisticated in their application;
- being able to work effectively within the organisation and with other organisations, where the organisation benefits from interoperability of ISE procedures, products and technology;
- being able to reduce risk and increase the probability of getting process and product right first time, where the organisation's energy is freed from having to resolve the same situations over and over again;
- being able to meet regulatory requirements, where the organisation is obliged to comply with international procurement legislation;
- being able to use 'state of the art' defence in liability claims, where the organisation adopts relevant standards as a safeguard against litigious customers and sub-standard products.

The growth of cross-border activity across Europe and the investment required for information system development lead irrevocably to the need for harmonised approaches to the engineering of those systems. The use of good ISE standards in Europe will:

- contribute to the removal of barriers to trade and overcome language and cultural barriers, enabling organisations to compete on equal terms throughout Europe;
- support the establishment of information systems needed to implement the single European market;
- provide long term economic benefits as European companies influence the development of products world-wide;
- reduce the risks associated with ISE products, with benefits to both the acquirers and the providers of Europe's information systems;
- contribute to the efficiency of ISE in Europe in order to increase Europe's competitiveness in the global market.

1.6 The role of CEN/TC 311

CEN/TC 311's scope is standardisation in the field of information systems engineering. It aims to support Europe's business, economic, political, cultural and legislative needs by identifying the role of and need for existing or new agreed standards in the field of ISE and encouraging and enabling their preparation and application.

The *'Conceptual Model and Taxonomy for ISE'*, mandated by SOGITS (SOGITS N 695.2 SOGT 93/45.2), is the baseline document for the work of CEN/TC 311 and may further serve as a reference document for others working in the field of ISE and ISE standardisation.

CEN/TC 311 has defined ISE as:

> *"the systematic, disciplined application of knowledge, methods and experience to the provision and support of information systems, bridging strategic goals/requirements and operational tools."*

It has further defined standardisation in the field of ISE as:

> *"concerning the standards needed to integrate the professional methods, techniques, tools and system components used in the planning, management, development, support and operation of information systems to achieve stated and implied requirements."*

There has been a rapid growth in the number, scope and scale of ISE standards projects, brought about by such pressures as:

- the explosion in the use, range and complexity of information systems;
- the expansion of markets world-wide;
- the re-engineering of businesses to benefit from the power of ICT;
- the increasing role of the customer in the engineering of information systems;
- the increasing impact of EU legislation.

However, existing standards are very uneven in their coverage. Very few standards are available for the early parts of the information system lifecycle while many are available for programming languages. Some, such as ISO 9001, span the whole lifecycle.

CEN/TC 311 takes a top-down, consensus-driven approach to ISE standardisation, looking from a specifically European standpoint to:

- identify market needs and priorities;
- identify existing ISE standards activity—*de facto* or *de jure*;
- whenever a European need is established, identify relevant standards;
- identify gaps in ISE standards coverage, ensure that these gaps are pointed out to relevant bodies and encourage standards development activity;
- wherever a need is established for European standards and no other course is open, propose the initiation of work.

The Bangemann Report identifies two features that are *"essential to the deployment of the information infrastructure needed by the information society"*: one is a seamless interconnection of networks and the other that the services and applications that build them should be able to work together (interoperability). It also predicts that *"open systems standards will play an important role in building information infrastructure"* and that standards bodies should *"establish priorities based on market requirements."* CEN/TC 311's strategy fits well with the high-level approach recommended by the Bangemann report.

2. Annex B Areas of user requirement

This annex identifies some differing areas of user requirement that need to be related to potential and actual areas of ISE standardisation.

ISE standards have a vital role to play in helping organisations achieve their strategic and operational goals. In particular, they are a powerful weapon in the drive to contain costs, improve quality, increase customer satisfaction, interact effectively with the outside world and integrate information systems into corporate strategy.

Some user concerns that could be mitigated by the availability in Europe of good ISE standards are:

- facilitating project collaboration;
- removing barriers to trade;
- facilitating ICT exploitation by small/medium sized companies;
- optimising ISE techniques;
- improving the benefits of ISE tools;
- improving product and process quality;
- using measurement as a quality improvement tool;
- improving evaluation methods and their specification in contractual situations;
- estimating and predicting.

2.1 Collaboration

Collaboration is an essential element of many ISE projects—between different providers and between providers and their customers. In the aerospace industry, for example, individual organisations no longer have—or, indeed need to have—the resources necessary to bear the full weight of many projects and their supporting R&D.

Equally important is the need to improve communication and collaboration between providers and their customers—to clarify the roles and obligations of each side and to address new approaches. Major procurement initiatives are underway, such as Euromethod in Europe and the US Department of Defense's CALS (*Computer-aided Acquisition and Logistics Support*) programme. Euromethod deals with contracting and management issues (processes) while CALS focuses on technology issues (products). Fully electronic business interfaces are envisaged, with customer and provider databases interconnected, with production information freely exchanged and with paperless project management. This vision entails a new supplier/customer integration which can give significant business advantage.

The requirement arising from these circumstances is the ability to interchange data and information without loss of form or meaning between different organisation systems, between different information systems and between different ISE systems using different ISE technology products. This ability is now crucial to the effective collaboration of companies with staff of different culture, training, skills and nationalities—and is in turn dependent on the existence of suitable data and information interchange and quality standards.

2.2 Trade barriers

With the globalisation of trade and the freeing of trade restrictions arising from the GATT agreement, all ISE providers face increasing competition and the need to find markets outside their national boundaries. This pressure gives rise to the need to trade, operate and collaborate with organisations in other countries without undue obstacles.

The problem in Europe is that, as things stand, free trade in ISE is not just difficult but virtually impossible. Each country, each business sector, indeed each company, operates with its own vocabulary, and its own methods, techniques and tools, with almost no compatibility between them and an equally small chance of successful interworking. Considerable effort is required to turn this situation around, with agreed, harmonised standards having a major role to play.

2.3 Small/medium enterprises

ICT is no longer the privilege of large companies: a growing number of small and medium sized enterprises (SMEs) also need ICT in order to retain their competitiveness in the expanding market and—simply—to survive. As the majority of European businesses fall into the category of SME, addressing their welfare is of prime importance to Europe.

Forms and methods of contact and co-operation are taking on interesting new dimensions, for example, technology transfer and international co-operation in production, sub-contracting, transport, etc. Working patterns and technical possibilities are changing. End users are able to access information and services through existing communications networks such as value added networks (VANs) and the Internet.

Interactive information network services create totally different ways to function and to behave (for example, remote/tele-working and electronic commerce). New market structures are emerging through the realisation of the concept of a network economy. This development will have a profound influence on SMEs. The availability of, awareness of and use of facilitating ISE standards to ensure affordable, easy access to new information technology and systems by SMEs is essential.

2.4 ISE techniques

ISE providers looking to improve their engineering processes have to confront the fact that there are so many different, incompatible, ISE techniques that might be used and so little objective evidence as to which technique might be best in which circumstance.

Examples of ISE techniques include entity/relationship modelling, data flow modelling, process modelling, state/event modelling, conceptual modelling and object-oriented extensions and variations. Different techniques use different formalisms, including natural and formal languages, mark-up languages, graphical languages and diagramming notations. This confusion is compounded by lack of agreement about basic concepts, with different techniques making different assumptions about such concepts as data, process, state, object and relationship.

Other engineering disciplines have long since progressed to the stage where basic concepts and techniques have been agreed and where design and construction information can be communicated through the shared use of standard formats. The goal for ISE is to achieve what is now taken for granted in many other branches of engineering, where designs and plans are understandable around the world because they use accepted standard symbols and rules.[13]

---

[13]This does not mean standardising on a single technique, but rather providing standards for techniques that are widely used (cf. programming language standardisation).

2.5 ISE tools

The market for ISE tools is growing rapidly and is potentially huge. However, it is a market that is already overcrowded with products—most of them expensive to implement and incompatible with other ISE technology.

This incompatibility leads to users being locked into the first product they buy or suffering from disconnected ‘islands’ of technology. No one vendor appears able to provide cost-effective and broad-ranging tools across the whole development cycle and across different operating systems. As a result, users are not able to achieve the interworking of different tools that is essential for effective ISE processes.

But the situation is changing. A major thrust of provider and purchaser strategies towards open, integrated ISE technology is under way. As the necessary standards come into force, tools providers will no longer find it so easy to control their customers’ future by locking them in to proprietary technology. Users will get better choice and lower cost products. Providers, too, will benefit from a broader market and from lower cost of entry.

2.6 Quality

The key element in many information systems is the software. It is software that gives a system its power and flexibility but it is, unfortunately, also the source of many problems, causing inefficiency and loss of competitiveness and possibly compromising public and private safety and security. There is widespread agreement that providers need to demonstrate that they can produce products of an appropriate quality in order to satisfy their users’ requirements.

An essential step in achieving software quality is to understand, control and improve the processes used in building that software—both production processes and management processes. This is no easy task. Software development processes are characterised by being complex, collaborative, distributed and difficult to manage, with added problems caused by the fact that the end product is an intellectual rather than a manufactured product and that the requirements for this product are often largely unknown until a demonstrator, or the real thing, begins to exist and can be tried out.

The key to improving ISE quality are standards and procedures for ensuring compliance with these standards. In this context the following trends are important:

- a quality framework conforming to ISO 9001; the guidance provided in ISO 9000-3 is vital here;
- use of demanding software engineering standards, as required in the safety and security sectors, which could provide high assurance at a non-trivial cost, e.g. DO-178B (civil avionics) and ITSEC;
- evaluation of ISE capability on a sliding scale by the production of an internationally agreed version of the Carnegie Mellon University Software Engineering Institute’s capability maturity model (CMM);
- procedures for third party assessment and certification agreed throughout Europe.

2.7 Measurement

To improve quality, it is first necessary to be able to say what quality is and to be able to measure it. Information systems have many different qualities. Reliability, ease of use, portability, cost and performance are all qualities that may be more or less important to different users at different times.

Given careful definition, some of these qualities may be measured directly. Testing coverage or execution speed, for example, are relatively easily measured; usability may be defined in terms of the ability of an untrained user to carry out a set of defined functions in a given time using the available documentation.. The effectiveness of the software inspection process can be metricated, giving some predictive capability.

Other attributes cannot be easily measured directly, such as modifiability, but can be partially quantified via other attributes such as program size and complexity, or the availability of appropriate design documentation.

However, although measurements can be taken, evidence suggests that most providers in the ISE domain do not effectively or regularly measure their products and processes. Standards defining the qualities and related measurements and measurement techniques are needed to provide a consistent measurement environment and to encourage the adoption of metrics programmes. Without a consistent environment, measurements can be taken but they are just data: it is very difficult to use them to support rational decision making. When measurements are taken under controlled conditions, they become information and may justifiably support reasoning from cause to effect. Measurement can then provide the necessary valid basis for quality improvement.

2.8 Evaluation

There are various ways to test, measure, compare and evaluate qualities of systems. In a contract between an acquirer and a provider of ISE products, there needs to be agreement on how to evaluate the system before the system is made. Equally, when a software provider offers products on the market, the provider should be able to document the qualities of the software and should include information on how these qualities have been evaluated.

Key questions are:

- what is evaluated;
- who does the evaluating and how neutral they are;
- under which proximity to reality does the evaluation take place;
- what is the source of knowledge for the evaluation;
- how is the evaluation carried out.

Evaluation issues arising from these questions are potential aspects of procurement standards dealing with contracts and/or marketing information:

- is the evaluation of process or product; if product, is the evaluation of a model or real product, of parts or the totality;
- who are the evaluators: users, developers, third party;
- where is the evaluation carried out: at provider or user site;
- will the evaluation use generated or real test data;
- will the evaluation be based on mental and/or behavioural knowledge and will the population be specified;
- will the results be informally discussed, qualitatively evaluated or quantitatively measured.

2.8.1 What is evaluated

Information system qualities are obvious objects of evaluation. However, there are other options too:

- **Process or product**: During information system development and during efforts to improve ISE systems, aspects of the process may need evaluation, e.g. measurement of time consumption compared to progress of production, discussion about the way decisions are made.

- **Model or product:** In many cases, it is unnecessary or impossible to evaluate qualities of the real system and models of the system are evaluated instead. For example, a data model is discussed in a reference group, the number of keystrokes needed to carry out a task is counted in a prototype, a simulation model is built to test a system's transaction capacity.

- **Parts or totality:** Even if parts of a system can be evaluated independently, this guarantees nothing for the functionality of the composite, integrated, system.

2.8.2 Neutrality of the evaluators

Initial tests of computer sub-systems are carried out by the developers, while users may want to evaluate a system in acceptance tests to decide its usefulness. Users, of course, comply with the principle that it is the customers' right to decide what is appropriate for them. In other situations, evaluation with scientific rigour may be preferred.

Users and providers can have conflicting interests concerning the outcome of some evaluations. Therefore, it can be more appropriate to have a neutral third-party organisation carry out the evaluation.

2.8.3 Proximity to reality

Some software qualities may be reliably evaluated on the developers' computers, while other qualities need to be tested at the user site. Testing a system in context at the user site may unveil problems other than those revealed by tests carried out at the provider's site. For example, when a user serves customers or clients, the intersection of work tasks when interacting with clients may require other functionality than can be specified and evaluated in laboratories. Test data may be generated by the providers. However, experience shows that real data at the user site can uncover inappropriate algorithms and data structures, even when the system has performed excellently on test data.

2.8.4 Source of knowledge

When evaluating qualities of systems in use, the evaluators may observe what users do, e.g. through logging user actions at the computer. On a macro level, it is possible, for example, to calculate costs and benefits. These are behavioural sources of knowledge.

The mental source of knowledge is what users think. Their thoughts may be verbalised through an interview or a questionnaire. User satisfaction is a way of measuring how users conceive the system compared to their expectations.

In scientific evaluations, a combination of knowledge from both behavioural and mental sources generally improves the reliability of the evaluation, compared to evaluations which rely on only one of the sources.

When individual users or user organisations constitute sources of knowledge for an evaluation, it is vital that the population is described. It may be of little relevance to know that a system functions perfectly for 28 year old male engineers, when the system is going to be installed on public terminals in a job centre.

2.8.5 Method of evaluation

The way information systems are evaluated may vary according to the method employed and the people carrying out the evaluation. Many qualities can be evaluated through informal discussion between the parties involved. This is often the only feasible way to evaluate process qualities, e.g. management style or division of tasks and responsibilities.

Since information systems and ISE systems are social constructs, qualitative methods from the social and humanistic sciences may be applied. Examples are:

- video analysis of thinking aloud sessions;
- interviews followed up by group discussions about tailoring a computer sub-system to different departments.

Quantitative measurements may also be carried out on, for example, time consumption, error frequency, answers in survey instruments.

2.9 Predictability

Improving achievable time and price on information system contracts yields obvious benefits but, for many, it is not the main challenge. What really needs to be addressed is how to provide customers—reliably—with what was agreed at the outset. A track record in this area is very unusual, very attractive to customers and the source of considerable competitive edge.

The inability of providers to deliver reliably to specification, to time and to price—in other words, to deliver a quality product—stems not from an inability to plan but rather from an inability to predict. Luck is still a key ingredient in many ISE products.

The most unpredictable element has always been whether or not components, large or small, will come together or ‘integrate’ successfully. Providers are still unable to integrate components with any degree of confidence as to whether it will take five minutes or five weeks. Unfortunately, integration, at whatever level or scale, is at the centre—indeed it is the essence—of systems building. If providers are unable to integrate system components reliably and predictably, then they cannot reliably and predictably build information systems.

The key to predictable and reliable integration is compilation systems which provide the necessary consistency checks (e.g. Ada compilation systems) combined with dependable, repeatable and enforceable design standards.

3. Annex C Standards committees related to ISE

Standards committees working within, interfacing with or overlapping with the information systems engineering domain include the following (the list is not exhaustive):

ISO/IEC JTC1	
SC1	Vocabulary
SC7	Software Engineering
SC14	Data Element Principles
SC18	Document Processing and Related Communication
SC21	Information Retrieval, Transfer and Data Management
for OSI	
SC22	Programming Languages, Environments and Systems
Software Interfaces	
SC24	Computer Graphics and Image Processing
SC27	IT Security Techniques
SC30	OPEN edi
IEC	
TC56	Dependability
TC65	Industrial Process, Measurement and Control
TC93	Design Automation
ISO	
TC145	Graphical Symbols
TC159	Ergonomics
TC176	Quality Management and Quality Assurance
TC184	Industrial Automation Systems and Integration
CEN	
TC251	Medical Infomatics
TC310	Advanced Manufacturing Technologies
TC311	Information Systems Engineering
CENELEC	
SC9XA	(WG1 and WG5) Railway Safety—Software and Systems
TC74X	Safety in IT Equipment
Associations	
ECMA TC33	PCTE
EIA	CDIF
Institutions and Agencies	
ESA	Software Engineering
IEE	Various committees
IEEE	Various committees
ITU	Various committees

4. Annex D Bibliography

Documents and publications consulted in the preparation of this document are listed below.

- AECL CANDU, *Standard for software engineering of safety critical software,* Revision 0, Ontario, December 1990.
- Asmis G J K, Kendall J D and Parnas D L, *Reviewable development of safety critical software*, International Conference on Control & Instrumentation in Nuclear Installations, Glasgow, May 1990.
- Asmis G J K, Tezel H O and Kendall J D, *The Canadian process for the regulatory approval of safety critical software in nuclear power reactors*, International Conference on Control & Instrumentation in Nuclear Installations, Glasgow, May 1990.
- Bangemann et al, *Europe and the global information society*, *Recommendations to the European Council*, Brussels, May 1994.
- Bertalanffy L von, *General System Theory*, George Braziller, New York, 1968.
- BIS Strategic Decisions, *Overcoming barriers to the adoption of open systems, a study for the Commission of the European Communities,* November 1992.
- Boehm J, *Improving Software Productivity*, Computer, September 1987.
- Boman M, *EC from A to Z, Abbreviations*, DS/INF, 1993, ISBN 87-7310-123-0.
- Born G, *Process Management to Quality Improvement*, Wiley, Chichester, 1994, ISBN 0-471-94283-9.
- BS 4778, *Quality vocabulary*, Part 3 *Availability, reliability and maintainability terms*, 1991.
- BSI, *Guide to a framework for user requirements for information technology*, DD 210, 1992.
- BSI, *Quality Assurance, Handbook 22,* fourth edition, 1990.
- CEN/BT WG63, *Final Report*, 1993.
- CEN/CENELEC, *Internal regulations*, *Part 3: Rules for the drafting and presentation of European Standards* (PNE-Rules), 1991.
- CEN/PT001, *Europe's Requirements for ISE Standards*, Final Report, BC-IT-014 SI, Brussels, May 1991.
- CEN/TC 311, *A note on the title and scope of CEN/TC 311 Information Systems Engineering*, N 033, 1993.
- CEN/TC 311, *Call for Experts*, N 101, 1994.
- CEN/TC 311, *Draft Consolidated and Standing Document of CEN/TC 311*, N 067, 1994.
- CEN/TC 311, *Draft Standardization Mandate Forwarded to CEN/CENELEC/ETSI in the field of Information Technology and Telecommunications: Information Systems Engineering,* N 050, January 1994.
- CEN/TC 311, *Report to BTS7*, N 054, January 1994.
- CEN/TC 311, *Response to ITSTC N1691 CENELEC comments on ITSTC N1601, response by CEN/TC 311*, N 104, August 1994.
- CEN/TC 311, *Standardisation in the field of Information Systems Engineering in Europe*, *Profile of CEN/TC 311*, N 087R, 1994.

- CEN/TC 311, *Work Programme*, N 088R, June 1994.
- COMPLEMENT Project, *Taxonomy for RT&E Systems*, ESPRIT, February 1994.
- Department of Defense, *Trusted Computer Systems Evaluation Criteria*, DOD 5200.28-STD, USA, December 1985.
- Department of Health, *The application of GLP principles to computer systems*, Good Laboratory Practice Advisory Leaflet no. 1, DoH, UK, 1989.
- DIN, *Grundsaetze für Rechner in Systemen mit Sicherheitsaufgaben, Principles for computers in safety-related systems*, Preliminary Standard, DIN V VDE 0801, January 1990.
- DO-178B, *Software Considerations in Airborne Systems and Equipment Certification*, issued in the USA by the Requirements and Technical Concepts for Aviation (document RTCA SC167/DO-178B) and in Europe by the European Organization for Civil Aviation Electronics (EUROCAE document ED-12B), December 1992.
- Dyer M, *The cleanroom approach to quality software development*, Wiley, 1992, ISBN 0-471-54823-5.
- ECMA/TR55, *Reference model for frameworks of software engineering environments*, 1993.
- EN 45001, *General criteria for the operation of testing laboratories*, CEN, 1989.
- Euromethod Project, *Euromethod Document Set*, Version 0, June 1994.
- Gilb T and Graham D, *Software Inspection*, Addison-Wesley, 1993, ISBN 0-201-63181-4.
- *Grundlegende Sicherheitbetrachtungen für MSR-Schutzeinrichtungen, Fundamental safety aspects to be considered for measurement and control equipment*, January 1989.
- Hammer M and Champy J, *Reengineering the Corporation*, Nicholas Brealey Publishing, London, 1993, ISBN 1-86373-505-4.
- Hausen H L, Cacutalua N and Welzel D. *A method for software assessment and certification—the informal model—refinement of the phase one model*, February 1992.
- Heym M and Österle H, *A Semantic Data Model for Methodology Engineering*, Proceedings of the 5th International Workshop on Computer-Aided Software Engineering, IEEE, 1992.
- Heym M and Österle H, *Reference Model for Methods Classification*, IWI, St Gallen, 1991.
- Holscher H and Rader J, *Microcomputers in Safety Technique*, Tüv Study Group on Computer Safety, Verlag Tüv Bayern, Tüv Rheinland, 1986.
- IEC 50, *International electrotechnical vocabulary*, chapter 191, *Dependability and quality of service.*
- IEC 880, *Software for computers in the safety systems of nuclear power stations*, 1986.
- IEC/SC65A/(Secretariat 122), *Software for computers in the application of industrial safety-related systems*, Draft for comment, December 1991.
- IEC/SC65A/WG10/*111 Functional safety of programmable electronic systems: Generic aspects*, September 1989.
- IEE, *Guidelines for Assuring Testability*, London, 1988, ISBN 0-86341-129-0.
- IEE, *Guidelines for the Documentation of Computer Software for Real Time and Interactive Systems*, 2nd edition, London, 1990, ISBN 0-86341-129-0.
- IEEE, Software Engineering Standards Long-Range Planning Study Group, *Master Plan for Software Engineering Standards*, Version 1.0, December 1993.
- IEEE, *Standard Glossary of Software Engineering Terminology*, IEEE Std 610.12-1990.

- ISO 8402, *Quality vocabulary*, 1991.
- ISO 8879, *Information processing; text and office systems; Standard Generalized Markup Language (SGML),* 1986.
- ISO 9001, EN 29001, *Quality systems; model for quality assurance in design, development, production, installation and servicing*, 1994.
- ISO 9000-3, *Guidelines for the application of ISO 9001 to the development, supply and maintenance of software*, 1991.
- ISO/IEC 2382-1, *Information technology; vocabulary; part 1: fundamental terms*, 1993.
- ISO/IEC 2382-20, *Information technology; vocabulary; part 20: system development*, 1990.
- ISO/IEC 9126, *Information technology; software product evaluation—quality characteristics and guidelines for their use*, 1991.
- ISO/IEC 10027, *Information technology; information resource dictionary system (IRDS) framework*, 1990.
- ISO/IEC 10032, *Information technology; reference model of data management*, 1995.
- ISO/IEC 12119, *Information technology; software packages—quality requirements and testing*,1994.
- ISO/IEC CD 14662, *Information technology; open edi reference model*, 1995.
- ISO/IEC DIS 10746, ITU-T X.900 *Information technology; basic reference model of open distributed processing (RM-ODP)*, 1994.
- ISO/IEC DIS 11404, *Information technology; language independent data types*, 1994.
- ISO/IEC JTC1/SC7, *Information technology; standardization framework for software engineering*, N 984R, 1992.
- ISO TR 9007, *Concepts and terminology for the conceptual schema and information base*, 1989.
- ITSEC, *Information Technology Security Evaluation Criteria, Provisional Harmonised Criteria*, Version 1.2, CESG, Cheltenham, 1991.
- Kitson D H and Humphrey W S, *The Role of Assessment in Software Process Improvement*, SEI-89-TR-3, Software Engineering Institute, Carnegie Mellon University, 1989.
- Management Consultants Association, *Quality Assurance—Guidelines for Management Consultancy*, December 1990.
- McDermid J A (Editor), *Software Engineer's Reference Book*, Butterworth-Heinemann, Oxford, 1991, ISBN 0-750-961040-9.
- Millis B G, *FoF and the CEN/TC 311 Work Programme*, London, 1994.
- Millis B G, *Towards an overall information technology standardization framework or framework of frameworks*, Issue 1.0, London, December 1992.
- Morvan P, *Dictionnaire de l'Informatique###concepts, matériels, langages*, Larousse, Paris, 1989.
- National Computing Centre, *Introduction to PRINCE*, Manchester, 1990, ISBN 1-85554-007-X.
- Nocentini S, *Il Sistema di Qualita del Software*, ETAS Libri, Milano, December 1993.
- Ögren, Staffan, *Informations - och data-kvalitet*, SISU, Electrum 212, 16440 Kista, Sverige.
- Olle T W, Hagelstein J, Macdonald I G, Rolland C, Sol H G, Van Assche F J M and Verrijn-Stuart A A, *Information Systems Methodologies*, Addison-Wesley, 1991.

- Paulk M C, Curtis B and Chrissis M B, *Capability Maturity Model for Software*, CMU/SEI-91-TR-91, Software Engineering Institute, Carnegie Mellon University, August 1991.
- prEN 50128, *Railway applications: Software for railway control and protection systems*, Version 1.0, CENELEC, February 1994.
- PSI, *Guideline Standard Operating Procedures for Good Statistical Practices in Clinical Research.*
- Resnick R and Taylor D, *The Internet Business Guide*, Samms Publishing, USA, 1994.
- Rushby J, *Formal Methods and the Certification of Critical Systems.*
- SCOPE Project, *Guide to software product evaluation—the evaluator's guide,* working draft, August 1992.
- SOGITS/SOGT, *Mandate for Information Systems Engineering*, SOGITS N695.2 SOGT 93/45.2, 1993.
- Thornton J, *Information Systems Engineering*, DISC, Blackwells, London, 1994, ISBN 0-580-22693-X.
- TickIT, *Guide to Software Quality Management System Construction and Certification using EN 29001*, DTI, London, 1992, ISBN 0-95-19309-0-7.
- Wichmann, B A (Editor), *Software in Safety-Related Systems*, Wiley, Chichester, 1992, ISBN 0-471-934747, also published by the British Computer Society.
- Yeh R T, *Software Development as a Wicked Problem*, International Journal of Software Engineering and Knowledge Engineering, Vol. 1-2, 1991.