

INTERNATIONAL STANDARD

ISO
8571-1

First edition
1988-10-01



INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION
МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ

Information processing systems — Open Systems Interconnection — File Transfer, Access and Management —

Part 1 : General introduction

*Systèmes de traitement de l'information — Interconnexion de systèmes ouverts — Gestion,
accès et transfert de fichier —*

Partie 1 : Introduction générale

ISO 8571-1 : 1988 (E)

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by the technical committees are circulated to the member bodies for approval before their acceptance as International Standards by the ISO Council. They are approved in accordance with ISO procedures requiring at least 75 % approval by the member bodies voting.

International Standard ISO 8571-1 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

Users should note that all International Standards undergo revision from time to time and that any reference made herein to any other International Standard implies its latest edition, unless otherwise stated.

ISO 8571 consists of the following parts, under the general title *Information processing systems — Open Systems Interconnection — File Transfer, Access and Management*

- *Part 1 : General introduction*
- *Part 2 : Virtual Filestore Definition*
- *Part 3 : File Service Definition*
- *Part 4 : File Protocol Specification*

Annexes A and B are for information only.

Contents

	Page
0 Introduction	1
1 Scope and field of application	1
2 References	1
3 Reference model definitions	1
4 Service conventions definitions	2
5 FTAM definitions	2
5.1 General	2
5.2 Architectural	2
5.3 Filestore schema	3
5.4 Filestore access	3
5.5 File structure	3
5.6 Constraint set	3
5.7 Document types	4
6 Abbreviations	4
Section one: FTAM general concepts	
7 OSI architectural background	5
8 Nature of the file service	6
8.1 Control of file activity	6
8.2 Asymmetry of the dialogue	6
8.3 External file service and internal file service	6
8.4 Service classes and functional units	7
9 Functions associated with the file service	7
9.1 Control of actions	7
9.2 Accounting	8
9.3 Concurrency control	8
9.4 Access control	9
9.5 Commitment	10
10 Service providers supporting FTAM	10
10.1 ACSE - application contexts and the FTAM environment	10
10.2 Presentation Service	10
10.3 Session Service	10
Section two: Virtual Filestore — General Concepts	
11 Virtual filestore	12

ISO 8571-1 : 1988 (E)

11.1	Need for a filestore model	12
11.2	Mapping the virtual filestore definition	12
11.3	Form of the virtual filestore	13
11.4	Attribute dynamics	14
11.5	Filestore schema	14
12	File structures	14
12.1	Categories of Structure	14
12.2	File access structure	15
12.3	Presentation structure	15
13	Constraint sets	16
14	Document types	16
Section three: Overview of the file service and file protocol		
15	File service	17
15.1	FTAM regime initialization phase	17
15.2	Filestore management phase	17
15.3	File selection phase	17
15.4	File management phase	18
15.5	File open phase	18
15.6	Data access phase	18
15.7	File close phase	18
15.8	File deselection phase	18
15.9	FTAM regime termination phase	18
16	Mechanisms in the file protocol	18
16.1	Protocol state machine	18
16.2	Grouping of protocol data units	18
16.3	Transparency	18
16.4	Checkpoint insertion	19
16.5	Diagnostics and results	19
16.6	Docket handling and non-volatile storage	19
16.7	Error recovery mechanisms	19
Annexes		
A	Examples of the use of FTAM	20
B	Summary of information objects identified in ISO 8571	25

Figures

1	Flow of information between RSE and OSIE.....	5
2	Logical and actual flows of information in file transfer	6
3	Example of the dialogue between file entities.....	6
4	Structuring a file protocol entity	7
5	Restrictions on the possible actions	8
6	Mapping between real systems and open systems	12
7	Virtual filestore schema	13
8	An example of a file access structure	14
9	Transmission of an unstructured text file	15
10	File service regimes and related primitives.....	17
11	Sending a file to a remote system	21
12	Remote database access	22
13	Use of FTAM in a LAN fileserver	23
14	File management.....	24

Information processing systems — Open Systems Interconnection — File Transfer, Access and Management —

Part 1 : General introduction

0 Introduction

ISO 8571 is one of a set of International Standards produced to facilitate the interconnection of computer systems. Its relation to other International Standards in the set is defined by the Reference Model for Open Systems Interconnection (ISO 7498). The Reference Model subdivides the area of standardization for interconnection into a series of layers of specification, each of manageable size.

The aim of Open Systems Interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of computer systems:

- a) from different manufacturers,
- b) under different managements,
- c) of different levels of complexity,
- d) of different ages.

ISO 8571 defines a file service and specifies a file protocol available within the application layer of the Reference Model. The service defined is of the category Application Service Element (ASE). It is concerned with identifiable bodies of information which can be treated as files, and may be stored within open systems or passed between application processes.

ISO 8571 defines a basic file service. It provides sufficient facilities to support file transfer, and establishes a framework for file access and file management. ISO 8571 does not specify the interfaces to a file transfer or access facility within the local system.

ISO 8571 consists of the following four parts:

- Part 1: General introduction
- Part 2: Virtual Filestore definition
- Part 3: File Service definition
- Part 4: File Protocol specification

The definitions in this part of ISO 8571 are used in the subsequent parts of ISO 8571 which specify the virtual filestore, services and protocols.

This part of ISO 8571 contains the following annexes which do not form part of the standard:

- Annex A - Examples of the use of FTAM
- Annex B - Summary of objects identified

1 Scope and field of application

This part of ISO 8571 provides a general introduction to the concepts and mechanisms specified in parts 2 to 4 of ISO 8571.

2 References

ISO 7498, *Information Processing Systems - Open Systems Interconnection - Basic Reference Model*.

ISO 8326, *Information Processing Systems - Open Systems Interconnection - Basic Connection Oriented Session Service Definition*.

ISO/TR 8509, *Information Processing Systems - Open Systems Interconnection - Service Conventions*.

ISO 8571, *Information Processing Systems - Open Systems Interconnection - File transfer, access and management*.

- Part 2: Virtual Filestore definition.
- Part 3: File Service definition.
- Part 4: File Protocol specification.

ISO 8649, *Information Processing Systems - Open Systems Interconnection - Service definition for the Association Control Service Element*.

ISO 8822, *Information Processing Systems - Open Systems Interconnection - Connection-oriented Presentation Service Definition*.

ISO 8824, *Information Processing Systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1)*.

ISO 8825, *Information Processing Systems - Open Systems Interconnection - Specification of Basic encoding rules for Abstract Syntax Notation One (ASN.1)*.

ISO 9804, *Information Processing Systems - Open Systems Interconnection - Definition of Application Service Elements - Commitment, Concurrency and Recovery*.¹⁾

ISO 9834-2, *Information Processing Systems - Procedures for specific OSI registration authorities - Part 2: Registration of Document Types*.¹⁾

3 Reference model definitions

ISO 8571 is based on the concepts developed in ISO 7498 and makes use of the following terms defined in it:

- a) application-entity;
- b) application process;
- c) application service element;
- d) (N)-connection;
- e) open system;
- f) (N)-protocol;

¹⁾ At present at the stage of draft; publication anticipated in due course.

ISO 8571-1 : 1988 (E)

- g) (N)-protocol-control-information;
- h) (N)-protocol-data-unit;
- i) (N)-service-access-point;
- j) (N)-service-access-point-address;
- k) (N)-service-data-unit;
- l) sublayer;
- m) (N)-user-data;
- n) user element.

4 Service conventions definitions

ISO 8571 makes use of the following terms defined in ISO/TR 8509 as they apply to the file service:

- a) confirm;
- b) indication;
- c) primitive;
- d) request;
- e) response;
- f) service provider;
- g) service user.

5 FTAM definitions

Unless otherwise stated, all terms apply to the view of a system presented for the purpose of open interconnection. This implies that the terms relate to a virtual filestore rather than to any real filestore (see clause 7).

The definitions are grouped into major categories, and ordered alphabetically within each category.

For the purpose of ISO 8571, the following definitions apply:

5.1 General

5.1.1 empty file: A file whose file contents consist of only a root node with no associated data unit, and no node name.

5.1.2 file access: The inspection, modification, replacement or erasure of part of a file's contents.

5.1.3 file contents: The data units, node names and structuring information contained in the file, which may be manipulated during the file open regime; the file attributes do not form part of the file's contents.

5.1.4 file management: The creation and deletion of files, and the inspection or manipulation of the file attributes.

5.1.5 file transfer: A function which moves a part or the whole of a file's contents between open systems.

5.1.6 hierarchical file model: A model of the internal structure of a file which takes the form of a tree of nameable file access data units.

5.1.7 real file: The named collection of information and its attributes which reside in a real system and to which the references to virtual files made in the OSIE are mapped.

5.1.8 real filestore: An organized collection of files, including their attributes and names, which reside in a real system and to which the virtual file references in the OSIE are mapped.

5.1.9 virtual file; file: An unambiguously named collection of structured information having a common set of attributes.

5.1.10 virtual filestore: An abstract model for describing files and filestores, and the possible actions on them. Where no ambiguity exists, the term is shortened in ISO 8571 to "filestore".

5.2 Architectural

5.2.1 accounting regime: The period during which a particular set of accounting information applies.

5.2.2 commitment unit: A set of filestore actions which either succeeds, the effect then being made visible to other processes, or fails completely with no effect visible to other processes.

NOTE - Nullification of the actions in a commitment unit is possible at any time up to the completion of the commitment unit.

5.2.3 docket: That collection of information which can be associated with a file service regime and which must be preserved if error recovery is to be possible.

5.2.4 external file service: File Transfer, Access and Management as seen by the file service user.

5.2.5 file service user: That portion of the application entity which conceptually invokes the FTAM service.

5.2.6 Initiator: That file service user which requests FTAM regime establishment.

5.2.7 internal file service: The service used by the file error recovery protocol machine to transmit both file error recovery protocol control information and normal file protocol control information.

5.2.8 open systems interconnection environment: The set of definitions of the standardized services, protocols and data structures which enable the interconnection of systems.

5.2.9 phase: The period of time in which protocol exchanges have a particular purpose, such as establishing or releasing an application context; for each phase a set of valid messages is defined in terms of state transitions.

NOTE - An entity is in one phase at any time.

5.2.10 presentation data value: The unit of information, specified at the abstract syntax level, which is transferred by the presentation service.

5.2.11 real system environment: The implementation aspects which support an application process within a real system.

5.2.12 receiving entity; receiver: The entity which receives part or all of the file's contents during the file data transfer regime.

5.2.13 regime: The period during which the entity is in a subset of its possible states for which particular actions are permitted.

NOTE - Regimes may be nested.

5.2.14 responder: That file service user which accepts an FTAM regime establishment requested by the initiator.

5.2.15 rollback: The nullification of uncommitted actions.

5.2.16 sending entity; sender: The entity which sends part or all of the file's contents during the file data transfer regime.

5.2.17 service element: A unit of standardization specifying a complete group of functions.

5.2.18 service primitive: The smallest defined interaction between the user and the provider of a communication service.

5.2.19 symbiotic service element: A service element which will support the operation of some second service by adopting its semantics and including parts of its abstract syntax at defined points in the abstract syntax of the first service's protocol control information.

5.3 Filestore schema

5.3.1 activity attributes: The attributes describing the activity of using the file service. The attributes are local to one FTAM regime (and any regime nested within it).

5.3.2 attribute: A piece of information stating a property of something, taking one of a set of defined values, each value having a defined meaning.

5.3.3 file attributes: The name and other identifiable properties of a file.

NOTE - The same value of a file attribute is observed at a particular time by any user of the file service, even when more than one user is active at that time.

5.4 Filestore access

5.4.1 access context: The specification of an algorithm defining a subset of the structuring information and user information in a file's contents, when reading the file for transfer or access.

5.4.2 data element: The smallest piece of data whose identity is necessarily preserved when transferred by the Presentation Service. A data element can convey file contents information, file structuring information or protocol control information.

5.4.3 data unit: The smallest unit of a file's contents which the filestore actions can manipulate. Each data unit is associated with a node of the file access structure. A data unit is a series of data elements.

5.4.4 file access data unit: A unit of the file access structure on which the actions of transfer, deletion, extension, replacement or insertion can be performed. A file access data unit contains zero or more data units.

5.4.5 filestore action: One of the actions specified as part of the definition of the virtual filestore.

5.4.6 file access structure: The data structure of a file that relates the file access data units, allowing their identification, description and manipulation.

5.5 File structure

5.5.1 arc: A directed link between two nodes.

5.5.2 arc length: A positive integer expressing the difference in levels between a child node and its parent node.

5.5.3 child (of a node): A node at which an outbound arc of the node concerned terminates.

5.5.4 leaf: A node of a tree that has no outbound arcs.

5.5.5 level (of a node): The sum of the arc lengths from the root to the node concerned.

5.5.6 long arc: An arc with an arc length greater than one.

5.5.7 node: The elementary component from which a tree is built up.

5.5.8 ordered tree: A tree in which there is a defined ordering of the children of each node in the tree.

5.5.9 parent (of a node): The node from which the inbound arc of the node concerned originates.

5.5.10 path: A sequence of arcs which links one node to another node in such a way that each arc is traversed in its defined direction.

5.5.11 root: The unique node of a tree that has no inbound arcs; it is at level 0.

5.5.12 sister (of a node): A node which shares the same parent node as the node concerned.

5.5.13 subtree: A part of a tree comprising an arbitrary node as the subtree root node and all the other nodes which can be reached by a path from the subtree root node.

5.5.14 traversal sequence: An ordering of the nodes in a tree such that each node occurs once and only once, and which is determined by an algorithm applicable to all possible trees.

NOTE - In general, many different trees may generate the same traversal sequence.

5.5.15 tree: A connected structure in which each node is linked to other nodes by directed arcs in such a way that one node has no inbound arcs and all other nodes have exactly one inbound arc.

5.6 Constraint set

5.6.1 constraint set: A set of restrictions and refinements of a general file model which specifies a less general model tailored to the needs of a particular class of applications.

5.6.2 file model: A model of the access structure of a file's contents.

5.6.3 flat (constraint set): A constraint set which, when applied to the general hierarchical file model, generates an access structure that consists of two levels, at the levels zero and one, and that may have data units at only the leaf nodes and has no data unit at the root node.

ISO 8571-1 : 1988 (E)

5.6.4 general hierarchical file model: a model in which the file access data units are organized in a hierarchical tree.

5.6.5 hierarchical (constraint set): A constraint set which, when applied to the general hierarchical file model, generates an access structure which is still hierarchical, but in which the form of the node descriptions and data units is restricted.

5.6.6 unstructured (constraint set): A constraint set which, when applied to the general hierarchical file model, generates an access structure that consists only of the root node with one data unit.

5.7 Document types

5.7.1 concatenation (of documents): The combination of two documents to form a single resultant document.

5.7.2 document: A collection of information with known abstract syntaxes and partially known semantics, and a known set of possible transfer syntaxes.

5.7.3 document type: The specification of a class of documents, which states their necessary semantics, abstract syntaxes, transfer syntaxes and dynamics.

5.7.4 dynamics (of a document): The concatenation and simplification properties of a document.

5.7.5 relaxation (of a document): The process of deriving one document from another by making the parameters describing it less restrictive.

5.7.6 simplification (of a document): The process of deriving one document from another of a different type by discarding structural information.

6 Abbreviations

The following abbreviations are used in parts 2 to 4 of ISO 8571:

ACSE	association control service element
ASE	application service element
CCR	commitment, concurrency and recovery
DU	data unit
EFS	external file service
FADU	file access data unit
FERPM	file error recovery protocol machine
FPDU	file protocol data unit
FPM	file protocol machine
FTAM	file transfer, access and management
Id	identifier
IFS	internal file service
OSI	open systems interconnection
OSIE	open systems interconnection environment
PCI	protocol control information
PDU	protocol data unit
PDV	presentation data value
PSDU	presentation service data unit
RSE	real system environment

Section one: FTAM general concepts

7 OSI architectural background

The aim in the standardization of a file service is to allow the Open Systems Interconnection of file users, who wish to transfer, access or manage information held by systems which behave as if they store files. Anything which appears as an open system, and conforms to the specified file protocols in the role of responder, is considered to provide a virtual filestore.

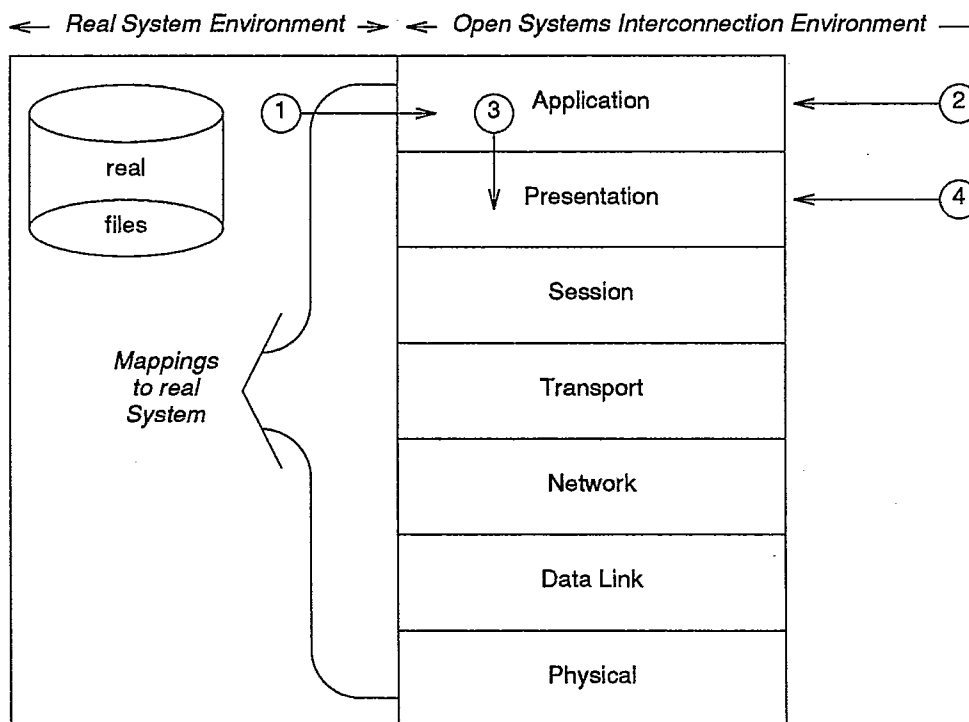
As in all the Open Systems Interconnection standards, the architectural divisions used here reflect classifications of protocol. They do not in any way constrain the way that possible underlying implementations are to be structured. The only point at which conformance can be tested is the point of interconnection.

The various parts of ISO 8571 are part of a larger set of Open Systems Interconnection standards, which are interrelated by the OSI Basic Reference Model (ISO 7498). The aspects of interconnection protocols common to many different fields of use are defined in separate standards, and the logical relation of the various elements is provided by supporting definitions of the services available.

In understanding the intent of OSI, it is important to distinguish carefully between the interrelated set of standards and the implementation, in hardware and software, of a real open system using the protocols specified in the standards. This distinction leads to the identification of two environments:

a) the implementation aspects, in terms of the real facilities and resources which support an application process within a real system, constitute the Real System Environment (RSE);

b) the set of definitions of the standardized services and protocols and data structures which enable the interconnection of systems constitute the Open Systems Interconnection Environment (OSIE). The stylized aspects of the behaviour of an application process visible in the Open Systems Interconnection Environment constitute the application entity; the activities of the application entities are supported by the layered communication functions defined in the OSI reference model.



NOTES

- 1 The flow of activations between the RSE and the OSIE represents the mapping of the intention to communicate onto the application entity user element.
- 2 The flow between application entities (both the application PCI and any user data) is a logical flow of information in an agreed abstract syntax, which is independent of any particular transfer syntax.
- 3 The flow between the application and presentation entities is a logical flow of information, including both user data and application protocol control information in one or more agreed presentation contexts.
- 4 The flow between the presentation entities is of data encoded in the negotiated transfer syntaxes, and of presentation protocol control information.

Figure 1 — Flow of information between RSE and OSIE