

INTERNATIONAL STANDARD

**ISO
8831**

First edition
1989-06-01

Information processing systems — Open Systems Interconnection — Job transfer and manipulation concepts and services

*Systèmes de traitement de l'information — Interconnexion de systèmes ouverts —
Concepts et services pour le transfert et manipulation de travaux*



Reference number
ISO 8831:1989 (E)



Documents
Document Sharing Hub

ISO 8831 : 1989 (E)

Contents

Foreword	vi
Introduction	vii
Section 1 : General	1
1.1 Scope	1
1.2 Normative references	1
1.3 Definitions	2
1.3.1 CCR service definitions	2
1.3.2 JTM service definitions	2
1.4 Abbreviations	3
1.5 Conventions	3
Section 2 : Overview	4
2.1 Overview and general description	4
2.1.1 Overview	4
2.1.2 Work specification contents	4
2.1.3 Proformas and spawning	4
2.1.4 Source, sink and execution agencies.	5
2.1.5 OSI jobs	5
2.1.6 Processing of work specifications	5
2.1.7 Reporting and the monitor function	6
2.1.8 Commitment, concurrency and recovery	7
2.1.9 Transfer control	9
2.1.10 Report manipulation	10
2.1.11 Work manipulation	10
2.1.12 Transfer manipulation	11
2.1.13 Authorisation and accounting	11
2.1.14 Work specification identification	11
2.1.15 Reporting responsibility	12
2.1.16 Documents	13
2.1.17 JTM relays	13

© ISO 1989

All rights reserved. No part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

International Organization for Standardization

Case postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

2.2	Overview of the service	13
2.3	Basic and extended implementations	14
2.4	Model for the service specification	14
Section 3 : Definition of primitives		16
3.1	JTM service primitives	16
3.1.1	Service primitive groups	16
3.1.2	Components of service primitive groups	17
3.1.3	Parameters of CCR-related primitives	18
3.1.4	Sequence of service primitive groups	18
3.1.5	Sequence of service primitives	18
3.2	Notation for service primitive and datastructure definition	19
3.2.1	Basic datatypes	19
3.2.2	Structuring mechanisms	19
3.2.3	Definition of basic datatypes	19
3.3	JTM events and report parameters	20
3.3.1	Event categories	20
3.3.2	Reports	22
3.3.3	Diagnostic information parameter	23
3.4	Fields of the conceptual data structures	24
3.4.1	Work specifications	25
3.4.2	OSI job parameters	25
3.4.3	OSI subjob parameters	28
3.4.4	JTM action parameters	29
3.4.5	Proformas	35
3.4.6	Header lists	36
3.4.7	Transfer control records	38
3.5	JTM documents	38
3.5.1	JTM report-display document	38
3.5.2	JTM work-display document	38
3.5.3	JTM TCR-display document	39
3.6	Parameters of JTM service primitives	39
3.6.1	J-INITIATE request and confirm primitives	39
3.6.2	J-DISPOSE service primitives	39
3.6.3	J-GIVE service primitives	40
3.6.4	J-ENQUIRE indication and response primitives	41
3.6.5	J-MESSAGE request primitive	41

ISO 8831 : 1989 (E)

3.6.6	J-SPAWN request primitive	41
3.6.7	J-END-SIGNAL request primitive	41
3.6.8	J-STATUS indication and response primitives	42
3.6.9	J-HOLD, J-RELEASE, J-KILL, J-STOP indication primitives	42
3.6.10	Summary	42
Section 4 : Basic Class		44
4.1	Primitive groups and document types for Basic Class	44
4.1.1	Service primitive groups	44
4.1.2	Document types	44
4.2	Conceptual datastructures for Basic Class	44
4.2.1	Reports	44
4.2.2	Work specifications	45
4.2.3	Document movement operations	46
4.2.4	Work manipulation operations	46
4.2.5	Report movement operations	46
4.2.6	Proformas	46
4.2.7	Transparency of proformas	46
4.2.8	Transfer control records	46
4.2.9	JTM documents	47
4.3	J-INITIATE primitive group parameters for Basic Class	47
4.3.1	Top level parameters	47
4.3.2	OSI job parameters	47
4.3.3	OSI subjob parameters	47
4.3.4	JTM action parameters	47
4.3.5	Proformas	48
4.3.6	Summary of information contained in a Basic Class J-INITIATE request	48
4.4	Other primitive groups in Basic Class	49
4.4.1	J-DISPOSE	49
4.4.2	J-GIVE	49
4.4.3	Other groups	49
4.5	Summary of Basic Class primitives and parameters	49
Annexes		
A	JTM service conventions	51
A.1	Introduction	51
A.2	Scope	51
A.3	References	51

A.4	Definitions	51
A.5	Model for the JTM service	51
A.6	Service primitives	51
A.7	Time sequence diagrams	52
A.8	Conventions for naming service primitives	52
B	Registers of document types	54
B.1	Introduction	54
B.2	Purpose of the entry	54
B.3	Identification	54
B.4	Document semantics	54
B.5	Transfer syntax	54
B.6	Relation of syntax and semantics	54
B.7	Syntactic considerations	55
B.8	Contents of the register	55
C	Tutorial material	56
C.1	Introduction	56
C.2	Architecture	56
C.3	Reporting	57
C.4	Manipulation	58
C.5	J-INITIATE primitives	59
C.6	Naming	59
C.7	Authentication	59
C.8	JTM terminology	60
C.9	The OSI job parameters	60
C.10	Identification of management	61
C.11	JTM activity (subjobs)	61
C.12	Action on errors	64
C.13	Documents	65
C.14	JTM relays	67
C.15	Agency access protocols	67
C.16	Time-relations on service primitives	67
C.17	Subsets and conformance	73
C.18	Interworking of Basic Class and extended implementations	73
C.19	The JTM protocol	73
C.20	Examples of the authentication mechanism	75
D	Glossary of terms	77

ISO 8831 : 1989 (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 8831 was prepared by Technical Committee ISO/TC 97, *Information processing systems*.

Annexes A and B form an integral part of this International Standard. Annexes C, D and E are for information only.

Introduction

The purpose of the job transfer and manipulation (JTM) standard is to provide a set of communication-related services which can be used to perform work in a network of interconnected open systems. This work can include both the running of traditional background jobs and other forms of information processing.

Background jobs have, in the past, been submitted either directly on the host system where they ran, or else at a remote job entry station connected to that system. Data files, program and "JCL" would be already available on the host, or would form part of the submitted "job deck". Output would be delivered on the host system, or alternatively on a printer attached to the remote job entry station. In a network of open systems, such jobs can be submitted at any open system supporting JTM to be run on another open system, using files collected from any other open systems, with output directed to peripherals or files held on any other open systems.

The JTM protocol covers not only the movement of job-related data (input and output) between open systems, but also provides for the movement of data concerned with monitoring job-related activity, and for controlling and manipulating the progress of this activity.

This International Standard does not specify individual implementations or products, nor does it constrain the implementation of entities and interfaces within a computer system. There is, therefore, no conformance to this International Standard.

Annex A is part of this International Standard, and describes the notation used for JTM service definition. This differs from that used by lower layers of OSI only because JTM is concerned with concurrent and related activity on more than two open systems. It is a superset of the notation used in the lower layers.

Annex B is part of this International Standard, and defines the requirements of JTM for a Document Type Registration Authority. These requirements of JTM apply to private, enterprise-specific, Registration Authorities; they are, however, also expected to be satisfied by any International Standard covering this area.

Annex C is tutorial material, and is not part of this International Standard. It gives a broad introduction to JTM, and should be read first by readers unfamiliar with the JTM work.

Annex D is not part of this International Standard, and contains an alphabetic glossary of the definitions appearing in the body of this International Standard.

Information processing systems – Open systems interconnection – Job transfer and manipulation concepts and services

Section 1 : General

1.1 Scope

This International Standard is an application layer standard within the Open Systems Interconnection framework set up by ISO 7498.

It defines the concepts and services for job transfer and manipulation.

This International Standard requires that the user of JTM

- specifies the open systems where work is to be done;
- knows the local functions and facilities of the open systems where work is to be done;
- knows the control languages used to specify local work on the open systems where work is to be done.

This International Standard provides the means to

- to specify work to be done on one or more open systems. The work done at one open system can result in new work to be done at other open systems;
- monitor the execution of work previously specified;
- modify work previously specified.

This International Standard does not address the standardisation of control languages, but is also applicable to the use of a standardised control language. This International Standard does not address the standardisation of user interfaces.

1.2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this International Standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this International Standard are encouraged to investigate

the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 646 : 1983, *Information processing – ISO 7-bit coded character set for information interchange*.

ISO 2022 : 1986, *Information processing – ISO 7-bit and 8-bit coded character sets – Code extension techniques*.

ISO 2375 : 1985, *Data processing – Procedure for registration of escape sequences*.

ISO 7498 : 1984, *Information processing systems – Open Systems Interconnection – Basic Reference Model*.

ISO 8571-3 : 1988, *Information processing systems – Open Systems Interconnection – File Transfer, Access and Management – Part 3 : File Service definition*.

ISO 8649 : 1988, *Information processing systems – Open Systems Interconnection – Service definition for the Association Control Service Element*.

ISO 8822 : 1988, *Information processing systems – Open Systems Interconnection – Connection oriented Presentation Service Definition*.

ISO 8824 : 1987, *Information processing systems – Open Systems Interconnection – Specification of Abstract Syntax Notation One (ASN.1)*.

ISO 8832 : 1989, *Information processing systems – Open Systems Interconnection – Specification of the Basic Class Protocol for Job Transfer and Manipulation*.

ISO/TR 8509 : 1987, *Information processing systems – Open Systems Interconnection – Service Conventions*.

ISO 9804 : —¹⁾, *Information processing systems – Open Systems Interconnection – Service definition for the Commitment, Concurrency and Recovery service element*.

1) To be published

ISO 8831 : 1989 (E)

1.3 Definitions

1.3.1 CCR service definitions

This International Standard makes use of the following terms defined in ISO 9804:

- a) atomic action
- b) master
- c) superior
- d) subordinate
- e) commitment
- f) rollback

1.3.2 JTM service definitions

The definitions are grouped into major categories, corresponding to the subclauses of clause 2.1.

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

1.3.2.1 General

1.3.2.1.1 agency: an abstract description of those functions of a real open system which are needed to support the JTM service.

1.3.2.1.2 work specification: a conceptual data structure within the JTM service provider which specifies in a defined way the work which is to be done.

1.3.2.1.3 document: a collection of data which forms part of a work specification, and which forms a unit of interaction between the JTM service provider and an agency.

1.3.2.1.4 initiation agency: that agency which causes a work specification to be created.

1.3.2.2 Proformas and spawning

1.3.2.2.1 proforma: part of a work specification which specifies further work and is used to form a new work specification as part of the processing of earlier work.

1.3.2.2.2 spawning: the process of taking the data from a proforma and using it to produce a new work specification.

1.3.2.2.3 spawning control data: data contained in a proforma which controls the circumstances in which spawning takes place from that proforma.

1.3.2.2.4 top level proforma: a proforma which is not contained within any other proforma.

NOTE — A proforma which is not a top-level proforma can become a top-level proforma as a result of spawning from its parent.

1.3.2.3 Source, sink and execution agencies.

1.3.2.3.1 source agency: any part of an open system which can provide documents for inclusion in a work specification when required by the JTM service provider as a result of processing the work specification.

1.3.2.3.2 sink agency: any part of an open system to which documents can be passed by the JTM service provider as a result of processing a work specification.

NOTE — Source and sink agencies can obtain and dispose of documents locally, or by use of non-standard protocols, or by use of FTAM.

1.3.2.3.3 execution agency: any part of an open system which initially acts as a sink for documents, but which subsequently acts as a source of related documents produced as a result of processing the earlier documents.

1.3.2.3.4 activity (in an agency): work performed by an agency, initiated by a service primitive issued to the agency by the JTM service provider; the completion of the activity is indicated by a service primitive issued to the JTM service provider by the agency.

1.3.2.4 OSI jobs

1.3.2.4.1 initial work specification: a work specification created as a result of the issue of an initiation service primitive by an initiation agency.

1.3.2.4.2 OSI job: the total work on all open systems arising directly or indirectly from an initial work specification.

1.3.2.4.3 OSI subjob (subjob): the total work arising from the processing of a single work specification, including the spawning of further work specifications, but excluding work arising from the processing of these further work specifications.

1.3.2.4.4 OSI job submission: the use of the initiation service primitive by an initiation agency for the creation of an initial work specification.

1.3.2.4.5 OSI job submission system: the open system on which OSI job submission occurs.

1.3.2.5 Reporting and the monitor function

1.3.2.5.1 JTM report: encoded information recording the progress or failure of an OSI job, generated by the JTM service provider, possibly as the result of interaction with an agency.

1.3.2.5.2 OSI job monitors: open systems to which JTM reports about a particular OSI job are sent.

1.3.2.5.3 report work specification: the type of work specification created by the JTM service provider to move JTM reports; the target open system for these work specifications is one of the OSI job monitors.

1.3.2.6 Commitment, concurrency and recovery

1.3.2.6.1 level of commitment: a parameter which determines whether operations requested in an atomic action are completed at the time of the atomic action, or are noted (as secure data) for later performance.

1.3.2.6.2 warning diagnostic: information carried by the CCR service on an offer of commitment which reports (usually for a human being) any variations on the expected action or unexpected consequences of the action.

1.3.2.6.3 retry-later diagnostic: information carried by the CCR service on a refusal of commitment when an action cannot be completed for reasons which can be transient.

1.3.2.6.4 no-retry diagnostic: information carried by the CCR service on a refusal of commitment when an action cannot be completed, and a later retry is not proposed.

1.3.2.7 Transfer control

1.3.2.7.1 work specification transfer: an atomic action by which a work specification is created at the receiving open system and destroyed at the sending open system.

1.3.2.7.2 transfer control record: a conceptual data structure held by an open system to control the transfer of work specifications and the issue of service primitives.

1.3.2.8 Report manipulation

1.3.2.8.1 report manipulation operations: operations requiring deletion or display of reports held by an open system nominated as a monitor point by some work specification.

1.3.2.8.2 report manipulation work specification: a work specification containing report manipulation operations.

1.3.2.9 Work manipulation

1.3.2.9.1 work manipulation operations: operations which select one or more work specifications or proformas and request displaying, killing, stopping or modification.

1.3.2.9.2 work manipulation work specification: a work specification containing work manipulation operations.

1.3.2.9.3 selector: data which is used to select zero, one, or more work specifications.

1.3.2.9.4 update: data which is used to modify a selected work specification or proforma.

1.3.2.10 Transfer manipulation

1.3.2.10.1 transfer manipulation operations: operations requiring setting, displaying or checking transfer control records.

1.3.2.10.2 transfer manipulation work specification: a manipulation work specification containing transfer manipulation operations.

1.3.2.11 Authorisation and accounting

1.3.2.11.1 identification authority: a naming authority which issues identifications; these identifications can be used to determine the capabilities to be made available to a particular authenticated identification (authorisation), or can be used to levy charges (accounting), or both.

1.3.2.11.2 authenticated identification: data which is known to correctly identify the user or management who requested the work to be performed, either by the use of a password check, or by some other checking mechanism.

1.3.2.11.3 user identification: data which can be used in a particular context to identify the user on whose behalf work is being requested.

1.3.2.11.4 account identification: data which can be used in a particular context to identify the account to be debited with any charges which are levied.

NOTE — User and account identifications consist of the name of an identification authority together with one of the identifications it issues.

1.3.2.11.5 open system management identification: the name of an open system which, when authenticated, authorises JTM activities or charging related to the management of that open system.

NOTE — Examples of such activities are the holding of work specifications about to be transferred to it, or the setting of transfer control records.

1.3.2.11.6 identification authority management identification: the name of an identification authority which, when authenticated, authorises JTM activities related to control of activity initiated by user identifications issued by that authority.

1.3.2.12 Work specification identification

1.3.2.12.1 OSI job local reference: a reference for an OSI job which is unambiguous within the OSI job submission system, assigned by that open system.

1.3.2.12.2 initiating identification: an identification provided by the JTM user at submission time to identify the initiator of the OSI job.

1.3.2.12.3 OSI job name: a string provided by an initiation agency when submitting an OSI job.

1.3.2.12.4 work specification identifier: a unique reference for a work specification which includes the name of the OSI job submission system, the identification of the initiating user, the OSI job local reference and the OSI job name; where a work specification was created by spawning, the identifier also contains one or more proforma names.

1.4 Abbreviations

OSI	open systems interconnection
JTM	job transfer and manipulation
FTAM	file transfer, access and management
CCR	commitment, concurrency and recovery

1.5 Conventions

The conventions used in this service definition are contained in annex A.

Section 2 : Overview

2.1 Overview and general description

In order to define JTM services, a model of the elements involved in the services is required. This JTM model is derived from the definitions given in ISO 7498 by adding further detail to encompass the services.

2.1.1 Overview

The JTM model recognizes the existence of a number of independent application entities on different open systems which cooperate to provide the JTM service and together form the JTM service provider. The JTM model also recognizes the existence of a number of agencies which are the users of the JTM service. The conceptual interactions between the JTM service provider and an agency are defined by service primitives. The JTM service provider receives from an initiation agency enough information to create a work specification.

The work which is requested is performed by

- a) standardized functions of the JTM service provider;
- b) functions of the local system environment accessed by the movement of documents between the JTM service provider and agencies.

2.1.2 Work specification contents

A work specification contains fields which provide data for

- a) identification of the work;
- b) authorisation of the work;
- c) defining where reports on the work are to be sent;
- d) selecting the reports which are required;
- e) identifying the type of the work specification;
- f) identifying the open systems which are to perform the work;
- g) specifying the urgency of the work;
- h) holding parts of the work until specified events occur;
- i) specifying the actions to be performed by the JTM service provider to carry out the initial work;
- j) specifying further work to be carried out when the initial work is completed.

A work specification refers to documents in the local system, or to documents which are obtained by the use of ISO 8571-3; these documents are subsequently passed by the JTM service provider to the same or to some other open system for storage locally, or for disposal using ISO 8571-3. When an open system has completed the work specified by a work specification (possibly including the generation of one or more new work specifications), the work specification ceases to exist. The JTM protocol transfers work specifications between open systems in order to progress the work.

The contents of a work specification relate to the following features supported by the JTM service, and are completely defined in clause 3.4.

The **identification** field of a work specification provides a universal and unique name by which the work can be referenced for subsequent reporting and manipulation. The work specification identifier is allocated when the work specification is created following submission by the initiation agency, or when it is created as the result of processing an earlier work specification. In the latter case, the identifier will reflect the parentage of the work specification.

The **initiating identification** field, and the **time of submission** field are generated when a work specification is created as a result of submission by the initiation agency. These fields are copied when new work specifications are created by JTM as a result of processing earlier work specifications.

The **authorisation** and **accounting** fields provide data which enables open systems to allow the performance of the requested work. When a work specification references files for obtaining or disposing of documents, additional authorisations and accounts can be included for the access to the files.

OSI job monitors can be specified; these are open systems to which any selected reports are to be sent.

The **report selector** field determines (for each OSI job monitor) which categories of event are to be reported to that monitor.

Different **types** of work specification are defined, corresponding to the different types of initial work to be performed. The most important type is the document movement work specification which provides for the movement of user documents between open systems. Other types provide for the movement of reports and manipulations: they are introduced later.

The **target** open system is that open system which carries out the final processing of the initial work, and which can generate new work specifications for further work. **Relays** can be specified for use as store-and-forward sites on the way to the target.

The **urgency** for the performance of the desired work and the **holding** of it can be specified.

JTM actions specify – in relation to the various types of work specification – what specific actions the JTM service provider has to perform. They determine the interactions between the JTM service provider and its local system environment, and determine the actions to be taken by it on its own data, or with regard to the reporting of events.

The **further work** field provides data in a form which allows the creation of new work specifications by a target.

2.1.3 Proformas and spawning

A proforma can contain further proformas nested to any depth. The JTM service provider spawns work specifications using top level proforma and the spawning formed data in the proformas. The new work specifications are formed using both data in the proformas, and also data in other fields of the original work specification. An important feature of spawning is the addition to the new work specifications of documents which have become available as

a result of earlier activity. The spawning process is initiated by an execution agency (see 2.1.4), or as the result of completion of activity on a sink or execution agency (see 1.3.2.3.4).

2.1.4 Source, sink and execution agencies.

The model recognizes that the JTM service provider, when processing a work specification, can – according to information in the work specification – interact with source, sink and execution agencies.

There are service primitives for interactions between a JTM service provider and agencies. Following these interactions with a source, the activity is complete. Following these interactions with a sink or execution agency the activity can either be complete or the agency can indicate only acceptance, that is, that the request for some activity has been secured. In the latter case, completion of the activity will be signalled by the agency at some later time, using another set of service primitive interactions.

An execution agency can, on completion of an activity, indicate that a number of documents are available for collection by a work specification resulting from spawning. It can also, at any time prior to completion, and if required by the activity, use service primitives to demand spawning using specified proformas; the resulting work specifications can collect documents which have become available before the end of the activity.

At any time prior to the completion of an activity in a sink or execution agency, the following events can occur:

- a) the JTM service provider can, as a result of processing a request for work manipulation (see 2.1.11), require that the activity be killed or stopped or held or released;
- b) the JTM service provider can ask for status information about the activity;
- c) the agency, if required by the activity, can cause the JTM service provider to generate a report about some significant event in the life of the activity;
- d) the agency, if required by the activity, can ask the JTM service provider to spawn from specified proformas.

All service primitives defined for interaction with an agency relate to a single work specification, and are independent of primitives used in relation to other work specifications.

An agency which cannot support concurrent activity on behalf of several work specifications can reject an attempt by the JTM service provider to initiate new activity with the response “retry later”.

Primitives used to obtain or dispose of documents using source and sink agencies carry information relating to the activity being attempted. The information is in one of two forms:

- a) in a JTM-specific form suited to simple local access, or to remote access in a non-standardised manner; or
- b) in a form specified by ISO 8571-3 (FTAM), suitable for local or remote access where FTAM is supported.

2.1.5 OSI jobs

A complete OSI job can involve access to source agencies on the OSI job submission system, access to source agencies, sink agencies and execution agencies on other open systems, the spawning of further work specifications and their processing. The complete OSI job comprises a set of activities performed either consecutively or simultaneously, potentially involving many open systems.

2.1.6 Processing of work specifications

2.1.6.1 Normal processing

After OSI job submission, an initial work specification can cause accesses by the JTM service provider to source agencies at the OSI job submission system, together with accesses to agencies at other open systems. The processing of a work specification typically comprises the following steps, performed repeatedly if necessary:

- a) source agencies are accessed and the documents delivered by them are (conceptually) embedded in the work specification;
- b) the work specification (including any embedded documents) is transferred to another open system, using the JTM protocol;
- c) documents within the work specification are passed to one or more sink or execution agencies as specified by the work specification;
- d) new work specifications are spawned using top level proformas contained within the original work specification, either on acceptance of documents by agencies, or on demand from an activity in an agency, or following completion of the sink or execution agency activity; documents which have been made available as a result of activity in an execution agency or which are specified to be obtained from other source agencies can be included in the new work specifications;
- e) the original work specification is destroyed when all the activity resulting from the above processing is completed.

If new work specifications have been produced by the processing of a work specification then these can cause further accesses to source agencies, followed by processing, as described above.

An illustration of the processing of a work specification, the activities of the JTM service provider related to an OSI job, including reporting (see 2.1.7) and manipulation (see 2.1.11), and the relationship between JTM model elements is given in annex C.

2.1.6.2 Errors in OSI job processing

Where errors are detected in a work specification (being received by an open system) before it has taken responsibility for it (offered commitment – see 2.1.8), such errors are reported to the sending open system using facilities present in the JTM protocol. This is seen by the sending system as a failure to transfer the work specification.

Errors which are detected by an open system (with responsibility for a work specification) while processing it are treated as described below.

ISO 8831 : 1989 (E)

The normal processing of a work specification, as described in 2.1.6.1, involves obtaining documents from source agencies and disposing of documents to execution agencies and sink agencies. Errors can arise which prevent one or more of these document exchanges being completed successfully. For example, the work specification might contain the name of a document which does not exist in the filestore accessed by the specified source agency, or might contain the name of a sink or execution agency which does not exist at the specified open system. Such errors are events for which JTM reports can be generated (see 2.1.7).

An error which prevents a document being obtained does not necessarily prevent further progress of the OSI job, as further useful processing can still be possible on other documents which have been obtained successfully. The work specification can indicate that an error diagnostic is to be included in the work specification in place of the document which could not be obtained; this error diagnostic is passed to the intended sink or execution agency instead of the document. In this case, all specified interactions with agencies still take place, and the sink or execution agency decides what action to take in the event of an error diagnostic being present.

An error detected while disposing of a document to a sink or execution agency, or while attempting a JTM transfer, prevents further processing of the OSI job. In this situation, the JTM service provider can be required to retain the work specification for a certain period of time to allow the error to be corrected by a subsequent work specification manipulation (see 2.1.11). (The work specification can also specify this form of error handling when documents are being obtained from a source.) For example, an erroneous sink or source name can be changed to a correct sink or source name or an erroneous target name can be changed to a correct target name. If the error has not been corrected when the time period expires, then the JTM service provider deletes the work specification, generating an abnormal termination report (see 2.1.7). This ultimate action can be specified in the work specification as the immediate action on errors, and is the only error handling provided in the Basic Class subset.

Error handling by the open system responsible for a work specification can be summarised as follows:

- a) accessing a source agency:
 - 1) embed a diagnostic and continue;
 - 2) abnormal termination;
 - 3) suspend to allow user correction and a later attempt;
- b) accessing a sink agency:
 - 1) abnormal termination;
 - 2) suspend to allow user correction and a later attempt;
- c) attempting to transfer a work specification:
 - 1) abnormal termination;
 - 2) suspend to allow user correction and a later attempt;

In the Basic Class, only option (i) is available in each of the above cases.

2.1.7 Reporting and the monitor function

After the JTM user submits an OSI job (through an initiation agency), he can be decoupled from it. Since, in general, it can take a long time for the complete work to be performed, the JTM user has to be able to find out about the progress of the OSI job and any related work specifications. Since there might be no direct contact with the JTM user, he cannot be directly informed of the progress or failure of the OSI job. The concept of OSI job monitors is therefore introduced.

These are open systems to which report work specifications can be sent regarding the progress of the OSI job: they are specified at OSI job submission time. When a significant event occurs in the life of an OSI job, the JTM service provider can create a report and deliver it to one or more of the OSI job monitors. The originating user can either enquire from an OSI job monitor in order to discover the status of the OSI job, or can arrange for all reports to be delivered by the OSI job monitor to sink agencies of his choice. The presentation of a report in a suitable (human-readable) format is the responsibility of the sink agency, and is not standardized.

The data needed to create a report work specification, and to select events for reporting, is included in fields of the work specification being reported on.

Two forms of OSI job monitor are supported. The **primary monitor** (and the report categories to be sent to it) is determined by the JTM service provider at the OSI job submission system. This data can only be changed by the "management" of that open system. The **secondary monitors** (and the report categories to be sent to each one) are determined by the user through parameters on the initiation service primitive. The user can change this data as the work proceeds (see 2.1.11).

The responsibility for generating reports is described in 2.1.15, but the OSI job submission system is responsible for ensuring that the monitor specification and the authorisation data allow any requested reports to be delivered to the primary monitor. (The user is responsible for such actions in relation to secondary monitors; the service provider cannot guarantee delivery of reports to secondary monitors if this data is in error or becomes out-of-date.)

Report work specifications differ from other work specifications in the following respects:

- a) report work specifications do not contain any user documents; the report information is contained in them in encoded form;
- b) a report work specification is not created by spawning from a proforma; it is created from a monitor specification contained in the initial work specification and copied into all work specifications produced by spawning;
- c) the work specification identification of a report work specification is the same as that of the work specification being reported on;
- d) the processing of report work specifications never causes further reports to be generated; errors detected during their processing or transfer cause them to be discarded;
- e) a report work specification contains all the permissions (see 2.1.13) which were present in the work specification being reported on; these are, however, ignored if any attempt is made to modify the report work specification; they are present to provide information to the OSI job monitor for handling subsequent requests for report manipulation (see 2.1.10);

- f) transfer of report work specifications (and manipulations – see 2.1.11) has priority over other JTM traffic.

The following categories of event can be selected for reporting (these categories are defined in clause 3.3):

- a) the creation of an OSI job;
- b) the transfer of (responsibility for) a work specification from one open system to another open system;
- c) the creation of a new work specification by spawning;
- d) the acceptance of documents by sink and execution agencies;
- e) the normal termination of a work specification following completion of the subjob;
- f) the premature ending of a subjob due to a manipulation request (STOP);
- g) the deletion of a work specification and termination of the subjob due to a manipulation request (KILL);
- h) the modification of a subjob by a manipulation request;
- i) the detection of errors in a work specification which result in error diagnostics being embedded in it;
- j) the holding of a work specification for the correction of errors;
- k) abnormal termination of the work specification due to errors;
- l) the generation of accounting information related to the subjob from the local systems environment;
- m) the generation of information by activity in an execution agency (user messages);
- n) abnormal termination of the work specification due to a request for facilities which are not supported;
- o) an attempted modification of the work specification by a subjob lacking the necessary authorisation;
- p) warnings associated with successful completion of the JTM activity, but alerting the (human) user to possible effects or variations.

A report work specification carries one or more reports. Each report has the following parameters:

- a) the name of the open system generating the report (see 2.1.14);
- b) the identification of the subjob being reported on, and of its initiator;
- c) the type of event being reported;
- d) the date and time when the report was generated (optional);
- e) a non-standardised text message.

Reports for certain categories of event also carry the name of the target, the number of work specifications which were spawned, a diagnostic message, the current hold/released status, and identification of another subjob (violation attempts). The reports which carry these parameters are defined in clause 3.3.

2.1.8 Commitment, concurrency and recovery

2.1.8.1 Overview

In order to ensure the correct operation of the JTM service in the presence of application or communication failures (recovery), freedom from interference by other activities (concurrency control), and the ability to perform the OSI job as one or more atomic actions (commitment), the Commitment, Concurrency and Recovery (CCR) procedures specified in ISO 9804 are used.

Each JTM transfer between open systems, and each interaction with an agency, uses service primitives to which the CCR semantics are applied. The procedures specified in ISO 9804 are carried out by each open system, and are conceptually carried out by each agency.

A Tutorial Annex in ISO 9804 describes the nature of CCR, for those unfamiliar with this work. An understanding of CCR is essential to an understanding of JTM.

The CCR primitives are used to initiate an atomic action, to indicate readiness to commit to it, or refusal, to order rollback or commitment, and to negotiate restart following failure.

The CCR service provides a user data parameter on the primitive which initiates an atomic action, on the indication of readiness or refusal, and on the restart primitives.

This user data is used to

- a) indicate the minimum level of commitment which is required, and the level actually achieved; and
- b) indicate to a subordinate the character sets which the master prefers for diagnostic messages; and
- c) indicate where a refusal is retry-later or no-retry; and
- d) carry optional warning messages on C-READY, optional timer values when a later retry is proposed, and a diagnostic message when no retry is proposed.

2.1.8.2 Levels of commitment for JTM

In order to perform an OSI job, the JTM service uses one or more atomic actions, according to the level of commitment requested by the initiation agency. The initial atomic action starts at the point when the initiation agency submits the OSI job to the JTM service provider. (The initiation agency is the commitment master for this initial atomic action).

The JTM service defines three levels of commitment which can be requested by the initiation agency to control the scope of this initial atomic action. The agency specifies the minimum commitment level which it requires, and is informed of the (equal or higher) commitment level actually achieved.

ISO 8831 : 1989 (E)

2.1.8.2.1 Completion commitment (level 3)

The highest level of commitment is COMPLETION (level 3). In this case, the entire OSI job is completed as part of the initial atomic action.

When commitment takes place, all the work specified has been completed, and all work specifications created as part of the processing of this OSI job have been fully processed, and have ceased to exist.

This level of commitment is sometimes described as "on-line" processing.

2.1.8.2.2 Agency acceptance commitment (level 2)

The next lower level of commitment is AGENCY-ACCEPTANCE (level 2). In this case, commitment to the initial atomic action means that the initial OSI subjob has progressed at least to the point at which all specified source agencies have been accessed, and all interactions with sink or execution agencies have resulted in the associated documents being secured by the agency, for later completion of the OSI subjob as part of subsequent atomic actions.

2.1.8.2.3 Provider acceptance commitment (level 1)

The lowest level of commitment is PROVIDER-ACCEPTANCE (level 1). (This level is also applied to all atomic actions initiated after the initial atomic action.) At this level, the initial atomic action results in the securing of the created work specification by the JTM service provider, and in it becoming visible to other open systems. There is no guarantee that the specified source documents have been accessed, nor that any material has been passed to sink or execution agencies. This commitment level is always available to a user, even if the local system is currently unable to establish any form of communication with other open systems. It can be described as "off-line" processing.

2.1.8.2.4 Definition of agency actions

A source agency treats all commitment levels as requiring it to deliver the requested material (or to refuse commitment). The CCR atomic action timer indicates the time available for obtaining the material.

A sink or execution agency treats levels 1 and 2 as requiring it to at least secure the document and carry out the action later (or to refuse commitment). It treats level 3 as requiring it to complete the activity (or to refuse commitment). The CCR atomic action timer again provides the agency with an indication of the time within which these actions should be completed.

2.1.8.3 Timeouts

No time-outs are specified for the provision of required responses. An inability to carry out the required actions because of temporary lack of resources (congestion or concurrency control) results in local retry at that open system. The commitment master (for example, the initiation agency) can set a time-out on a local basis and issue a rollback service primitive if the timer expires; it can use CCR parameters to indicate this intent to its subordinates.

2.1.8.4 User data when starting an atomic action

This user data carries a commitment level parameter and a diagnostic code indicator parameter.

2.1.8.4.1 Commitment Level

The commitment level parameter is an integer. (See 2.1.8.2.) The lowest level of commitment is 1. The highest is 3.

If the commitment level parameter has a maximum value, all results of the action are completed before the subordinate offers commitment. If it has the minimum value, the subordinate need only note the actions to be taken (as secure data), and to perform them some time later.

NOTE — A subordinate can be able to attempt the action at a higher level of commitment than that requested by its superior, possibly involving several new subordinates. If this attempt fails (RETRY-LATER), it can be possible to rollback all the new subordinates and perform the action at the lower commitment level on a purely local basis.

This parameter specifies the minimum commitment level required by the superior. The equivalent parameter on an offer of commitment specifies the commitment level actually achieved, and equals or exceeds that of the parameter when starting the action.

An atomic action started by a subordinate carries an equal or greater level of commitment than that on the start of the action which it received from its superior.

The atomic action timer parameter (see ISO 9804) enables the subordinate to optimise its actions in relation to trying for a high commitment level first, then moving to a lower one.

2.1.8.4.2 Diagnostic code indicator

The diagnostic code indicator parameter is a list of zero one or more code specifiers. Each code specifier consists of one or more integers, or the value "ALL". Each integer is a Register Number from the ISO Register of Character Sets (see ISO 2375).

Each code specifier references the one or more graphics character sets in the Register entries with the given numbers. Each subsequent diagnostic message generated by JTM contains only the character space, together with characters from the character sets referenced by one of the code specifiers.

NOTE — Different diagnostic messages can use character sets referenced by different code specifiers.

The code specifiers appear in the diagnostic code indicator in the order of preference of the CCR master. A code specifier with the single integer 2, specifying the graphics characters of the International Reference Version of ISO 646, is always implicitly present as the least preferred option. An empty diagnostic code indicator specifies ISO 646 only.

The start of an action issued by a subordinate carries a diagnostic code indicator which is equal to that received in the start of the action from its superior.

2.1.8.5 Use of CCR user data when offering commitment

This carries a commitment level parameter, an optional warning diagnostic, and optional accounting information.

2.1.8.5.1 Commitment level

The form of the commitment level parameter is described in 2.1.8.4.1. The value on the offer of commitment is greater than or equal to the value on the start of the atomic action.

NOTE — Where a superior uses subordinates to perform part of an atomic action, the commitment level offered to its superior (if any) does not exceed that offered by any of its subordinates.

2.1.8.5.2 Diagnostic parameter

The diagnostic parameter is optional. If present, it is a warning diagnostic, consists of one or more diagnostic messages, each consisting of:

- a) the name of the generator of the diagnostic message; and
- b) a JTM diagnostic code; and
- c) human-readable message text or an FTAM datastructure carrying warnings.

The name of the generator of the diagnostic message is an application-title. Its form is outside the scope of this International Standard. It unambiguously identifies the generator of the message.

The JTM diagnostic code values are listed in the JTM Protocol Specifications.

The human readable message text consists of one or more strings of graphic characters (plus space). Each character is taken from any of the graphics sets referenced by one of the code specifiers in the diagnostic code indicator parameter of the start of the atomic action.

Preference is given to the code specifiers which appear earlier in the diagnostic code indicator. Each string is a line of text, and does not exceed 40 characters in length. It contains zero or more characters.

NOTES

- 1 The number of strings of characters is not restricted.
- 2 The human language used in the strings is chosen by the generator of the diagnostic message; the requested character sets can be used as an indication of the preferred language.

2.1.8.5.3 Accounting parameter

This parameter is optional, and provides charging information relating to the atomic action for which commitment is offered. If present, it can cause reports to be generated as described in 2.1.1.7.

Its form is a list of quadruples, consisting of

- a) an (globally unambiguous) account identification (see 2.1.1.3);
- b) a character string resource identifier;
- c) a character string charging unit;
- d) an integer charge.

The values of b) and c) are assigned by the identification authority (see 2.1.1.3) associated with the account identification.

2.1.8.6 Use of CCR user data on refusal of commitment

This carries only a diagnostic parameter and an optional accounting parameter.

2.1.8.6.1 Diagnostic parameter

The diagnostic parameter is either a retry-later diagnostic or a no-retry diagnostic.

If it is a no-retry diagnostic, it consists of one or more diagnostic messages, each consisting of

- a) the name of the generator of the diagnostic message; and
- b) a JTM diagnostic code; and
- c) human-readable message text or an FTAM datastructure carrying "no-retry" diagnostics.

If it is a retry-later diagnostic, it consists of an optional retry timer and an optional diagnostic message, possibly containing an FTAM datastructure with "retry-later" diagnostics.

The retry timer has the same form and range of values as the CCR atomic action timer. If present, it indicates that the superior should wait a specified time before attempting the atomic action again. It can be ignored by the superior.

When a superior has multiple subordinates and they each produce diagnostic parameters, the superior discards those of lower severity. The order of severity is defined as

- a) warning (lowest);
- b) retry-later;
- c) no-retry (highest).

2.1.8.6.2 Accounting parameter

This has the form described in 2.1.8.5.3

Note that although the atomic action is in this case rolled back, charges are still levied, and a report of accounting charges can be generated.

Where multiple attempts are made to complete the action, there can be multiple charges.

2.1.8.7 Use of CCR user data on restart primitives

The user data parameter is not used in the request and indication, (it is optional in CCR).

On the response and confirm it is used only if the response is RETRY-LATER or REFUSE, and carries only a diagnostic parameter and optional accounting parameter.

If the resumption point parameter is RETRY-LATER, the diagnostic parameter is optional, and is a retry-later diagnostic. If present, it indicates that the superior should wait a specified time before attempting the restart again. It can be ignored by the superior.

If the resumption point parameter is REFUSE, the diagnostic parameter is present, and is a no-retry or retry-later diagnostic. The form of the accounting parameter is described in 2.1.8.5.3. The accounting parameter in this case repeats information previously provided on a refusal of commitment.

2.1.9 Transfer control

A transfer control record is used to control which work specifications are eligible for transfer to another (specified) open system, and the number of such transfers which can be in progress simultaneously.

ISO 8831 : 1989 (E)

A transfer control record has no effect on the transfer of transfer manipulation work specifications (see 2.1.12). It can control all other types of work specification, depending on the value of the selectors.

An open system with work specifications to transfer to another open system, in the absence of a transfer control record for the destination system, attempts the transfers in an order and with a degree of concurrency decided locally.

A transfer control record, if present, causes the implementation to restrict its transfer attempts to those needed to progress work specifications with specified characteristics.

The presence of transfer control records can prevent commitment to atomic actions. Such refusals are always flagged as "retry later".

2.1.10 Report manipulation

A report manipulation work specification contains one or more report manipulation operations. There are two types of operations defined for report manipulation.

The first is the DELETE operation which causes deletion of all received reports about a specified OSI job or tree of subjobs. Its only parameter is a work specification identification which is used to refer to reports on the specified work specification or on any created from it.

The second is the DISPLAY operation. This has two parameters. The first is the same as above, a work specification identification used to refer to part or all of an OSI job. The second is a document name which is referenced by a proforma contained in the manipulation work specification. This proforma defines an arbitrarily complex subjob for disposal of the named document.

The action by the JTM service provider at the open system is to take each report which was received concerning the specified work specification(s), in the order in which they have been received, and to generate a document containing the reports. If two display operations specify the same document name, both displays are placed in the same document. The end of the activity produces spawning of new work specifications from proformas in the usual way: these collect the display document.

NOTE — Where a monitor point is requested to pass reports directly to a sink agency, storage for display or deletion by report manipulation does not occur.

Authorisation for deletion of reports requires the report manipulation work specification to contain an authority corresponding to the permission information in the report work specification which delivered the report. It is a local option whether authorisation for display is also restricted in such a way. (See 2.1.13.)

2.1.11 Work manipulation

A work manipulation work specification contains one or more JTM work manipulation operations. These provide for the operations of selection, modification, killing, stopping and displaying work specifications and associated subjobs. (The effect of these operations is described below.)

The first operation is a selection; the work specifications or proformas selected are the ones to which all operations preceding the next selection apply. The form of a work specification selector is described in 2.1.11.1.

The modification operation changes certain fields in a work specification or proforma. It can cause an interaction with source, sink or execution agencies associated with the work specification in order to effect a hold or release on activity associated with the work specification.

The kill and stop operations are only applied to a work specification, and can also cause interactions with one or more agencies in order to stop or to kill associated activity. If an uncommitted transfer is in progress, they affect the transfer.

Killing causes the JTM service provider to delete the work specification to which it is applied, and to inform any associated agency that a kill is required. This causes the agency to cease its activity, with rollback if possible. Any uncommitted transfer is rolled back.

Stopping causes the JTM service provider to inform any associated agency that the activity is to be stopped. Normal spawning will occur, and the work specification is not affected. Any uncommitted transfer is rolled back, with retry at a later time.

The display operation causes a copy of parts of the selected work specifications or proformas to be placed in a specified document. The document is collected by normal spawning as for the report manipulation display. The display also contains local status information concerning any source, sink, execution or transfer activity associated with the work specification.

Authorisation for killing, stopping, and modification requires the work manipulation work specification to contain an authority corresponding to a permission (see 2.1.13) in the selected work specification, or in the work specification containing a selected proforma. When an attempt is made to kill, stop, or modify one or more work specifications (selected by a selector), then the manipulation is applied only to those work specifications which contain the necessary permission, and the other work specifications are unchanged. When an attempt is made to display one or more work specifications, it is a local option whether the work specifications which do not contain the necessary permission are displayed or not.

When an unauthorised modification attempt is made by subjob A (say) on a work specification B (say), this results in the VIOLATION-ATTEMPT event for B, which can be reported to one or more monitor points of B.

2.1.11.1 Selectors

A selector specifies a series of tests to be applied to certain fields of a work specification.

The basic tests are of the form

field-name operator value

where the range of permitted field names and of operations is defined in section 3.

For the purposes of selection, the important fields of a work specification are regarded as forming a somewhat simpler datastructure called a **header-list**. The selector selects on the basis of a general logical expression using tests for equality or inequality of these fields against specified values.

Work specifications can also be selected according to the fields of the proformas they contain.

2.1.11.2 Updates

An update is of the form

field-name operator value

and is used in the modification operation to specify the changes to be made. It is specified in section 3. Certain field names which are available for selection cannot be modified.

The header-list conceptual data structure is again used for specifying the update, but changes to its fields produce corresponding changes in the associated work specification.

2.1.12 Transfer manipulation

A transfer manipulation work specification contains one or more transfer manipulation operations.

There are JTM transfer manipulation operations for setting, displaying and checking transfer control records.

The setting operation has a parameter which is a transfer control record.

The display operation has a parameter which identifies a transfer control record, and a second parameter which is a document name; it displays the current transfer control record in the specified document for collection by spawning.

The checking operation contains a copy of a transfer control record which the sending open system is holding in respect of transfers to the receiving open system or to an agency. The receiving open system generates a new transfer manipulation work specification containing a setting operation if this is no longer an appropriate transfer control record for the sender to be using. An open system normally issues checking operations following any period in which it is disconnected from the open systems environment.

A transfer manipulation work specification needs an authorisation element for the management of any open system referenced in the transfer control record of a setting or displaying operation, and also for the management of an open system performing a checking operation. (See 2.1.13.)

2.1.13 Authorisation and accounting

A work specification contains authorisation data. This provides a list of **authorisation elements**. The authorisation element provides an open system with an identification issued by an identification authority whose name it recognises, and a **validation field**. The validation field is either secret data (a password or key) which serves to authenticate the identification, or is a statement that the identification has already been authenticated by an open system earlier in the history of the OSI job.

In order to accept a validation field as having been "already checked", the authenticated identification of the open system which performed the check has to be available. This is obtained in a step-wise manner. The lower layer services provide to the JTM service provider the authenticated name of a calling open system. This data is recorded in the work specification, so that it contains the name of each open system through which the work specification (or one of its ancestors) has passed. Thus an open system is able to determine, by inspection of this data, whether to accept authorisation elements claiming that an identification has already

been authenticated. (The detailed operation of this procedure is specified in ISO 8832.) A further description (and examples), of this mechanism appears in annex C.

An authorisation element can identify a user, or the management of an identification authority, or the management of an open system.

The authenticated identification is used to authorize any activity which is requested, and can, in the absence of a suitable account element (see below), also be used to levy charges.

A work specification can also contain account data. This provides a list of **account elements**. An account element provides an open system with an account identifier issued by an identification authority whose name it recognises, and a validation field.

The authorisation data also names the user identifications which are given permission to make modifications to this work specification. Implicit permission is given to validated identifications representing the identification authority management issuing these identifications, and to those representing the open system management for open systems involved in the OSI job.

An open system's local management decides whether to allow queries and status displays about a work specification without the existence of a permission, or whether to treat a work specification as invisible to anyone other than those with permission to modify it.

Permission to modify a transfer control record is implicitly given to authenticated identifications representing the open system managements involved.

Arrangements between open systems for use of a common identification authority are not excluded, but are not required.

2.1.14 Work specification identification

The work specification identifier provides an unambiguous name by which the work specification can be referenced by a report or by a manipulation work specification.

The first part of a work specification identifier is the name of the OSI job submission system followed by the identification of the initiating user, the OSI job local reference and the OSI job name. The second part of the identifier is a list of (proforma name, qualifying integer) pairs. An element is added to the list whenever a new work specification is created by spawning. The proforma name is the name of the proforma used to spawn the work specification, and the qualifying integer is used to distinguish between work specifications spawned from the same proforma.

In summary, all work specifications have an identification known as the work specification identifier. A complete OSI job, that is the set of all its work specifications, can be referenced by quoting the name of the OSI job submission system and either

- a) the initiating user identification and OSI job-name; or
- b) the OSI job local reference.

An individual subjob can be referenced by quoting a complete work specification identifier. A group of subjobs with a common origin can be referenced by quoting the work specification identifier of their origin.

ISO 8831 : 1989 (E)**2.1.15 Reporting responsibility**

Reports are generated either as part of the CCR atomic action progressing the work (and with the same commitment level) – integral reporting, or as part of a separate atomic action – separate reporting. In the former case they are committed if and only if the atomic action proceeds to commitment. When generated by a commitment master, they are either not generated before commitment is offered by subordinates, or are rolled-back if the main action is rolled back (as an implementors option).

In the latter case, they are generated as a result of progressing the main action, and are not affected by rollback of that action.

Generation of reports with a commitment level of one, or generated by separate reporting, is never subject to failure. Subsequent processing of the report can result in failures causing it to be discarded. These events are not reported. At levels two and three, failures to deliver reports generated by integral reporting to agencies causes the entire atomic action to fail.

2.1.15.1 Separate reporting

Reports issued as separate atomic actions are

- a) reports of accounting information;
- b) reports of attempted security violations;
- c) reports of the failure of an atomic action, resulting in no progress of a work specification, or in its abnormal termination.

2.1.15.2 Integral reporting

Reports which are committed only if the main action commits are

- a) reports that J-INITIATE has created a new work specification;
- b) reports that spawning of a new work specification has occurred;
- c) reports that another open system has taken responsibility for progressing a work specification in a subsequent atomic action;
- d) reports that AGENCY ACCEPTANCE for the subjob has occurred;
- e) reports that an error diagnostic has been included in a work specification (for delivery to a sink agency) in place of a document;
- f) reports carrying messages produced during processing activity (and passed to JTM by a J-MESSAGE request from an execution agency);
- g) reports of normal termination of a subjob;
- h) reports of termination of a subjob, or of modification as a result of manipulation by another work specification (see section 2.1.11);
- i) warning reports.

2.1.15.3 Responsibility for separate reporting

Responsibility for issuing reports which are issued as separate atomic actions is

- a) for accounting information, that open system containing the master of the atomic action; (note that where the master is a JTM agency, the report is generated by the JTM service provider accessing the agency); the accounting information is carried to the master by the accounting parameter in the user data of the CCR primitives; the number and frequency of such reports is not Standardised;
- b) for attempted security violations, that open system with responsibility for the work specification being manipulated;
- c) for reports of failure of an atomic action, that open system containing the master of the atomic action; diagnostic messages are carried to the master by the diagnostic parameter in the user data of the CCR primitives.

2.1.15.4 Responsibility for integral reporting

Responsibility for integral reporting rests with the open system most closely involved, thus:

- a) for creation, the open system where the J-INITIATE request is issued;
- b) for spawning, the open system performing the spawning;
- c) for transfer of responsibility, the open system taking responsibility;
- d) for AGENCY ACCEPTANCE, the open system where the J-DISPOSE indication service primitives are issued;
- e) for error diagnostic embedding, the open system performing the embedding;
- f) for user messages, the open system where the J-MESSAGE service primitive was issued;
- g) for normal termination, the open system which completes the subjob;
- h) for manipulation termination, the open system performing the manipulation (and responsible for the work specification being manipulated).

2.1.15.5 Job tracing

The reporting categories of JTM have been designed to permit a sufficiently sophisticated monitor point implementation to maintain a "picture" of the progress of an OSI job, providing it receives all reports in certain categories.

Such job tracing is not required of an implementation by ISO 8832, and can be performed on a purely manual basis. On the other hand, highly sophisticated JTM implementations are not excluded.

The minimum relevant report categories are

- a) creation reports, indicating the creation of a new OSI job, with its identification, the open system responsible for it, its target, and any relays to be used on the way to the target;
- b) spawning reports, indicating the activation of a new subjob, with its identification, and the open system responsible for it;

- c) transfer of responsibility, indicating a change in the open system responsible for progressing a subjob;
- d) normal, abnormal, and manipulation termination, indicating the end of a named subjob.

It can happen that, due to congestion or long-term network failures, some reports are delayed. In general, reports can arrive out of order. The correct order of the above categories can be deduced from the identifications of the subjobs.

Ambiguity arises only when all subjobs appear to have ended, but reports of prior spawning have been delayed. The JTM service enables this ambiguity to be resolved by including with all termination reports a count of the number of subjobs which were spawned.

The JTM service does not provide any mechanism for deducing, on receipt of a subjob termination report, that all reports of accounting information, attempted security violations, error diagnostic insertion, and user messages have been received.

The above list of reporting categories is the minimum needed to determine the current location and states of the parts of an OSI job. Useful additional information is provided by:

- a) reports of AGENCY ACCEPTANCE;
- b) reports of holding and release of a work specification by modification or of "no-progress".

2.1.16 Documents

JTM defines three document types which carry JTM information. These are JTM-report-display documents (defined in 3.5.1), JTM transfer-display documents (defined in 3.5.3), and JTM work-display documents (defined in 3.5.2).

The semantics of documents of these types are fully defined in this International Standard, and a transfer syntax for such documents is fully defined by the JTM protocol.

In addition to these document types (which are generated by the JTM service provider), the JTM service is also capable of carrying any form of user document for which a transfer syntax has been defined. The definition of the syntax and semantics of other document types is outside the scope of this International Standard, but is essential to its use.

JTM recognises three sorts of document type. These are

- a) the JTM-generated document types; these are the three types listed above; and
- b) standardised document types; these are document types whose semantics and transfer syntax have been defined elsewhere, and for which there is an entry in the **ISO Register of Document Types**; and
- c) enterprise-specific document types; these are document types whose semantics and transfer syntax are determined on an enterprise-specific or bilateral basis.

The JTM protocol static conformance clause recommends that certain JTM implementations provide support for three simple forms of document, designed to carry simple lines of text, text with carriage controls, and an arbitrary length string of bits.

These document types are defined in ISO 8832. They can all be carried by FTAM where necessary.

There are no JTM conformance requirements in relation to support for other document types such as those supporting graphical transfers, office documents, or data descriptive files. Support by JTM implementations for such document types is optional.

NOTE — Annex B defines the information required in a Document Type Register Entry for the registration or distribution of definitions of enterprise-specific document types.

2.1.17 JTM relays

Each JTM work specification has a **target** open system. This is present in the original work specification when the OSI job is created. It is obtained from proformas when new work specifications are spawned, and from monitor data when reports are generated.

Each of these sources of information can additionally specify one or more **relays** to be used in transferring the work specification to the target.

NOTE — These are JTM application-layer relays, they are not communications relays.

The work specification is transferred to the first relay, then to the second, and so on, until the final relay transfers it to its target. If reporting of the TRANSFER event is selected, reports are generated for each of these transfers.

Each relay performs normal initial processing to resolve any source references for which it is the action open system, then queues the work specification for onward transfer.

It is visible for manipulation and subject to transfer control in exactly the same way as work specifications which are generated by OSI job creation or spawning, and are queued for transfer (or are being transferred).

Examples of the use of relays are given in annex C.

2.2 Overview of the service

The JTM service primitives defined in section 3 of this standard are specified in terms of the model elements described in 2.1. These JTM service primitives provide for performing all job-related activities between open systems.

They provide for

- a) the submission of an OSI job to the JTM service provider from an initiation agency; (OSI jobs can be submitted to move documents between source, sink and execution agencies; OSI jobs can also be submitted for display or manipulation of any previously submitted OSI jobs, that is, to display and modify the actions of the JTM service provider or the current activities of agencies);
- b) the delivery of documents to sink agencies, together with an interaction about the end of the activity caused by the delivery;
- c) the collection of documents from source agencies;

ISO 8831 : 1989 (E)

NOTE — The use of FTAM by source or sink agencies is fully supported.

d) the delivery of documents to execution agencies for processing, together with an interaction about the end of the activity of the agency caused by the delivery.

2.3 Basic and extended implementations

NOTE — A complete definition of Basic Class JTM services is given in section 4.

A Basic Class implementation and an extended implementation are recognised. A Basic Class implementation supports only a subset of the possible range of work specifications, and supports only a subset of the range of parameters on JTM service primitives.

Interworking between extended and Basic Class implementations is possible. In particular it is possible for work specifications generated on Basic Class implementations to be transferred to and processed by extended implementations; the converse is also possible if the work specification is within the range covered by Basic Class (see section 4 and clause C.17).

The following are the main restrictions and characteristics relating to Basic Class implementations:

- a) a work specification carries at most one document;
- b) a work specification carries at most one top-level proforma; any embedded proformas are wholly transparent to a Basic Class implementation;
- c) errors detected when collecting a document from an execution agency result in embedded diagnostics; other errors result in abnormal termination;
- d) the only reporting which can be selected is
 - 1) abnormal termination;
 - 2) normal termination;
 - 3) manipulation termination;
 - 4) user messages.
- e) accounting is not supported;
- f) commitment level 3 (COMPLETION) can not be requested;
- g) transfer control records are not supported;
- h) report manipulation is not supported;
- i) only selection (by "subjob type", "OSI jobname", and "OSI job submission system"), "brief display", "stop", and "kill" operations are supported;
- j) use of FTAM for obtaining or disposing of documents is not supported;
- k) there are no secondary monitors;
- l) the "hold" concept is not supported;
- m) the following primitives are not supported;

- 1) J-INITIATE-TCR-MAN
- 2) J-INITIATE-REPORT-MAN
- 3) J-ENQUIRE
- 4) J-HOLD
- 5) J-RELEASE
- 6) J-SPAWN.

n) recovery from applications or communications failure during JTM transfers is supported only through the CCR restart mechanisms.

NOTE — This involves repeating the entire transfer unless it has been completed; intermediate checkpointing is available in the extended JTM protocol.

Another important characteristic of a JTM implementation is the types of agency which it supports, and the range of real devices or processing functions that these are mapped onto.

A JTM implementation might support only an initiation agency or only an execution agency; it might support only a sink agency mapped to a single one of its many printers. On the other hand, it might support the mapping of all its devices and functions onto JTM agencies, and support a full range of agencies.

This agency support dimension is separate from the Basic Class and extended distinction.

2.4 Model for the service specification

JTM services are provided by JTM Specific Application Service Elements, the service elements for Association Control and for Commitment, Concurrency and Recovery and lower layer entities. These together constitute the JTM service provider. The JTM services are defined in relation to and in terms of the JTM model elements which have been described earlier.

The service primitives represent conceptual interactions between the JTM service provider and the different agencies in the model (that is, initiation, source, sink and execution agencies, constituting the users of the service).

According to further specific JTM mechanisms, such as spawning of new work specifications from proformas, a service request from one user can give rise to a set of interactions between the JTM service provider and agencies. This constitutes a one-to-many correspondence between JTM service users. (See figure 1)

An initiation agency interacts with the JTM service provider to provide all the information necessary to define an initial work specification with its proformas (that is, the whole work to be performed), and there are service primitives related to this.

The main unit of interaction between the JTM service provider and sink, source and execution agencies is the document, and a set of service primitives is provided to support these interactions for any type of document.

Another set of service primitives relate to the manipulation of work specifications.

Finally, there are a set of service primitives related to the monitoring and status concepts in the JTM model.

All these service primitives are defined in section 3. A summary is given at the end of that section. The name of each service primitive has been chosen to be in line with the concepts of the JTM model, and relates to a specific interaction between a specific

agency and the service provider. Several of these service primitives are needed to describe the complete processing of a single OSI job.

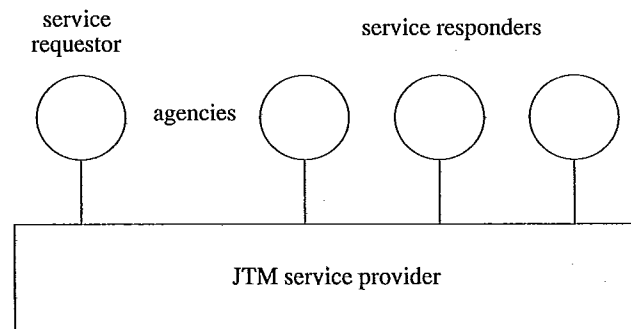


Figure 1 — JTM users

Section 3 : Definition of primitives

3.1 JTM service primitives

An agency provides an abstract representation of part of a real open system.

Each conceptual interaction between the JTM service provider and an agency is defined by the issue of a sequence of service primitives (called a **JTM service primitive group** or a **commitment group**).

This International Standard associates Commitment, Concurrency, and Recovery semantics (as defined in ISO 9804) with these service primitives. This provides a means of defining the points in the JTM protocol when externally visible events are to occur, and of stating the commitment, concurrency, and recovery capability which the total real system is to exhibit.

Provided the external behaviour corresponds with that of the model open system, no constraints are placed on the internal structure, interfaces, or procedures of the real system.

3.1.1 Service primitive groups

A JTM service primitive group is a sequence of service primitives, some of which are mapped onto the CCR service primitives of ISO 9804, and some of which have a purely JTM function.

3.1.1.1 Creation of new work specifications

There are four groups of primitives (initiated by an initiation agency) which each define the creation of a new work specification in an open system. These are:

J-INITIATE-WORK this group is used by an initiation agency to create a work specification for document movement;

J-INITIATE-WORK-MAN this group is used by an initiation agency to create a work specification for the manipulation of existing work specifications;

J-INITIATE-REPORT-MAN this group is used by an initiation agency to create a work specification for the manipulation of reports received by an OSI job monitor;

J-INITIATE-TCR-MAN this group is used by an initiation agency to create a work specification for manipulation of JTM transfer control records;

Note that where a statement applies to all the above primitive groups, the term "J-INITIATE" is used. In particular, the only difference between the parameters of these primitives is in the nature of the JTM action parameters outside of proformas, and the Basic Class restrictions.

3.1.1.2 Document movement

There are three groups of primitives initiated by the JTM service provider as the result of processing a document movement work specification, that is, as the result, directly or indirectly, of a J-IN-

ITIAL group of primitives. The first group is issued to a sink or execution agency, and the last two groups to an execution agency or source. These groups are:

J-DISPOSE this group is used by the JTM service provider to pass a document to a sink or execution agency, creating a new activity in that agency;

J-GIVE this group is used by the JTM service provider to request a document from a source or execution agency;

J-ENQUIRE this group is used by the JTM service provider to obtain from a source or execution agency a list of document names for satisfaction of references.

3.1.1.3 Other agency-initiated groups

There are three groups of primitives which are initiated by a sink or execution agency in relation to a specific activity which is in progress in the agency. These are:

J-MESSAGE (execution agency only) this group is used by the agency to cause the JTM service provider to generate a **USER-MESSAGE** report work specification;

J-SPAWN (execution agency only) this group is used by the agency to specify that demand spawning from a specified proforma is to occur;

J-END-SIGNAL this group is used by a sink or execution agency to signal the end of an activity which is in progress following an earlier **ACCEPTANCE** commitment.

3.1.1.4 Other provider-initiated groups

There are five groups of primitives which are initiated by the JTM service provider to a sink or execution agency to control an activity which is in progress following earlier **ACCEPTANCE** commitment. These are:

J-STATUS this group is used by the JTM service provider to obtain information about the progress of an activity;

J-HOLD this group is used by the JTM service provider to request temporary suspension of the progress of an activity (its only defined effect is to prevent the issue of the primitive groups of 3.1.1.3; other effects are not standardised);

J-RELEASE this group is used by the JTM service provider to cancel an earlier **J-HOLD**;

- J-KILL** this group is used by the JTM service provider to cause termination of the activity; this causes an immediate J-END-SIGNAL, and no documents are provided by an execution agency; it is desirable, but not required, for all effects of the activity being killed to be undone;
- J-STOP** this group is used by the JTM service provider to cause termination of the activity; this causes an immediate J-END-SIGNAL, but documents can be supplied; it is recommended that any work already done should not be undone, but this is not required.

3.1.2 Components of service primitive groups

3.1.2.1 Groups initiated by an agency

These groups consist of the following primitives (in order):

- a) a J-BEGIN request; followed by
- b) one of:
 - 1) J-INITIATE-WORK request and confirm; or
 - 2) J-INITIATE-WORK-MAN request and confirm; or
 - 3) J-INITIATE-REPORT-MAN request and confirm; or
 - 4) J-INITIATE-TCR-MAN request and confirm; or
 - 5) J-MESSAGE request; or
 - 6) J-SPAWN request; optionally followed by the results collection sequence specified in 3.1.2.3; or
 - 7) J-END-SIGNAL request; optionally followed by the results collection sequence specified in 3.1.2.3; followed by
- c) one of:
 - 1) a J-READY indication; or
 - 2) a J-REFUSE indication; followed by
- d) one of:
 - 1) J-COMMIT request and confirm (only following a J-READY indication); or
 - 2) J-ROLLBACK request and confirm.

The primitives in a), c), and d) carry only CCR parameters, and correspond exactly to CCR service primitives of the same name. Their sequence, semantics and parameters are completely defined by ISO 9804, except for the user data parameter (see 3.1.3). The agency is the commitment master (and issues the CCR atomic action identifier and defines the commitment level and atomic action timer) in all cases except the J-MESSAGE and J-SPAWN group. Where these groups are issued following ACCEPTANCE commitment, the agency is the commitment master, otherwise it is a sub-

master. Where the agency is a sub-master, it chooses a commitment level greater than or equal to that received. For a J-END-SIGNAL request, the agency always specifies PROVIDER-ACCEPTANCE (level 1), and does not refuse commitment if the JTM service provider offers commitment.

3.1.2.2 Groups initiated by the JTM service provider

These groups consist of the following primitives (in order):

- a) a J-BEGIN indication; followed by
- b) one of:
 - 1) J-DISPOSE indication and response; optionally followed by the results collection sequence specified in 3.1.2.3; or
 - 2) J-GIVE indication and response; or
 - 3) J-ENQUIRE indication and response; or
 - 4) J-STATUS indication and response; or
 - 5) J-HOLD indication; or
 - 6) J-RELEASE indication; or
 - 7) J-KILL indication; or
 - 8) J-STOP indication; followed by
- c) one of:
 - 1) a J-READY request; or
 - 2) a J-REFUSE request; followed by
- d) one of:
 - 1) J-COMMIT indication and response (only following a J-READY request); or
 - 2) J-ROLLBACK indication and response.

The primitives in a), c) and d) carry only CCR parameters, and correspond exactly to CCR service primitives of the same name. Their sequence, semantics and parameters are completely defined by ISO 9804, except for the user data parameter (see 3.1.3). The JTM service provider is the commitment master (and issues the CCR atomic action identifier) if and only if it has already given ACCEPTANCE commitment to the J-INITIATE, J-SPAWN or J-END-SIGNAL group which generated the work specification causing the primitives to be issued, otherwise it is a submaster. When the JTM service provider is the master it always requests commitment level 1.

3.1.2.3 Results collection sequence

This sequence consists of one or more repetitions of

- a) J-ENQUIRE indication and response; or
- b) J-GIVE indication and response

ISO 8831 : 1989 (E)

3.1.3 Parameters of CCR-related primitives

The parameters of the CCR-related primitives identified in 3.1.2.1 and 3.1.2.2 are specified in ISO 9804. The user-data parameter specified in ISO 9804 is used to carry JTM-specific CCR-related parameters as defined below.

3.1.3.1 J-BEGIN request and indication

These primitives carry:

- a) commitment level (see 2.1.8.4.1);
- b) diagnostic code indicator (see 2.1.8.4.2).

3.1.3.2 J-READY request and indication

These primitives carry:

- a) commitment level (see 2.1.8.5.1);
- b) optional warnings (see 2.1.8.5.2);
- c) optional accounting information (see 2.1.8.5.3).

3.1.3.3 J-REFUSE request and indication

These primitives carry:

- a) a diagnostic (see 2.1.8.6.1);
- b) optional accounting information (see 2.1.8.6.2).

3.1.3.4 J-RESTART request and indication

The user data parameter is not used.

3.1.3.5 J-RESTART response and confirm

The user data parameter is used if the response is RETRY-LATER or REFUSE, and contains

- a) a diagnostic (see 2.1.8.7);
- b) optional accounting information (see 2.1.8.7).

3.1.4 Sequence of service primitive groups

The following groups are issued at any time:

J-INITIATE-WORK
J-INITIATE-WORK-MAN
J-INITIATE-REPORT-MAN
J-INITIATE-TCR-MAN
J-DISPOSE
J-GIVE
J-ENQUIRE

There is an additional restriction on the J-GIVE and J-ENQUIRE groups when issued to an execution agency. In this case they are only issued (and complete) between a J-END-SIGNAL or J-SPAWN request, and the corresponding J-READY or J-REFUSE indication or between a J-DISPOSE response where COMPLETION commitment is offered and the corresponding J-COMMIT or J-ROLLBACK indication.

The following groups are issued between a J-DISPOSE indication and the corresponding J-READY or J-REFUSE request:

J-MESSAGE
J-SPAWN

The following groups are issued between a J-COMMIT response in a J-DISPOSE group which gave ACCEPTANCE commitment, and the J-BEGIN request commencing the J-END-SIGNAL for the activity:

J-MESSAGE
J-SPAWN
J-END-SIGNAL

A J-END-SIGNAL group is not started if any other group is incomplete in relation to the same activity, nor is a J-READY or J-REFUSE request to a J-DISPOSE indication issued if a J-MESSAGE or J-SPAWN sequence is incomplete.

The following groups are started if and only if no other group is in progress in relation to the same activity:

J-STATUS
J-HOLD
J-RELEASE
J-KILL
J-STOP

In addition to these primitives, J-RESTART is defined with the same semantics and parameters as C-RESTART, and can occur as defined in ISO 9804.

3.1.5 Sequence of service primitives

An arbitrarily complex JTM activity involves many primitive groups. The possible sequences of all the primitives in these groups can be determined, for any specific JTM activity, by applying the following rules:

- a) the sequence within a primitive group is determined by 3.1.2;
- b) the sequence of primitive groups issued by or to a single JTM entity for a single JTM subjob is determined by 3.1.4;
- c) the sequence of primitives with CCR semantics (at the same or at different open systems) is determined by the CCR Service definition;
- d) where data provided by one primitive (e.g. a J-GIVE response) is needed for the issue of another primitive (e.g. a J-DISPOSE indication), then the sequence of these primitives is determined by this requirement;
- e) there are no other constraints on the sequence of the service primitives.

An example applying these rules to specific JTM activity is provided in annex C.

NOTE — An implementation can support only a single agency (initiation, sink, source, or execution), or can support several agencies of the same or different types. Where a type of agency is not supported, service primitive groups interacting with that type of agency do not occur on that open system.

3.2 Notation for service primitive and datastructure definition

This clause specifies the notation used for defining the parameters of the JTM service primitives which are not CCR-related. (The parameters of the CCR-related primitives have a less complex structure, and are defined in 3.1.3.)

Each JTM service primitive parameter or field in a JTM conceptual datastructure is either a basic datatype or a structured datatype. Each structured datatype is constructed using one of the structuring mechanisms in 3.2.2 below applied to either basic datatypes or to further structured datatypes.

3.2.1 Basic datatypes

There are six basic datatypes used by JTM. These are

- a) literals;
- b) integer datatypes;
- c) string datatypes;
- d) name datatypes;
- e) referenced datatypes;
- f) document datatypes.

A literal is specified by giving the literal name in capitals. Each literal is an unambiguous identification whose semantics are entirely defined in this service definition. For example,

MOVE HIGH CREATION

All other basic datatypes are written with a lower case name enclosed in angle brackets, with an “=I”, “=S” or “=OS”, “=N” or “=NA” or “=ND”, “=R”, and “=D” included to denote the basic datatypes b) to f) above. For example:

integer	<qualifying integer = I>
string	<OSI job name = S> <source name = S>
name	<authority name = NA> <open system name = N> <document type = ND>
referenced	<security parameters = R>
document	<basic document = D>

The definition of these basic datatypes is given in 3.2.3 below.

3.2.2 Structuring mechanisms

There are four mechanisms (and corresponding notation) for forming a structured datatype. These are

- a) an ordered sequence of datatypes; the datatypes are listed; for example, the datatype <OSI-subjob parameters> is defined as

<subjob name-list>
<target list>
<urgency>
<holds>
<subjob type>

each of which is another structured datatype;

- b) zero one or more repetitions of a single datatype, where order is important and duplicates are allowed (a list); the single datatype is included in square brackets followed by “*L”; for example, the datatype <document list> is defined as

[<basic document = D>]*L

- c) zero one or more repetitions of a single datatype, where order is not significant, and where duplicate values can not occur (a collection); the single datatype is included in square brackets followed by “*C”; for example, the datatype <permissions> is defined as

[<permission element>]*C

- d) a choice from two or more alternatives, often (but not always) literals; the alternatives are separated by vertical bars and included in round brackets; for example, the datatype <urgency> is defined as

(HIGH | MEDIUM | LOW)

A structured datatype is defined by placing its name on the left of a “:=” symbol, with the structure notation on the right. For example

<permissions>:= [<permission element>]*C

3.2.3 Definition of basic datatypes

3.2.3.1 Literal and integer datatypes

The meaning of each literal is defined as it occurs.

An integer is any value from zero upwards, and is not bounded.

3.2.3.2 String datatypes

A string datatype is either a sequence of characters plus space taken from any character set registered for use as a G0, G1, G2, or G3 set in the ISO Register of Character Sets, or is an octet string. In the former case “=S” is used, in the latter case “=OS” is used.

The JTM protocol conformance requirements specify that an implementation

- a) supports the specification of “=S” string fields using at least

- 1) the ISO 646 International Reference Version character set; and

- 2) hexadecimal notation for the encoding of the string in accordance with ISO 2022; and