INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

X.861

(12/97)

SERIES X: DATA NETWORKS AND OPEN SYSTEM COMMUNICATION

OSI applications – Transaction processing

Open Systems Interconnection – Distributed transaction processing: Service definition

ITU-T Recommendation X.861

(Previously CCITT Recommendation)

ITU-T X-SERIES RECOMMENDATIONS

DATA NETWORKS AND OPEN SYSTEM COMMUNICATION

PUBLIC DATA NETWORKS	
Services and facilities	X.1–X.19
Interfaces	X.20–X.49
Transmission, signalling and switching	X.50–X.89
Network aspects	X.90–X.149
Maintenance	X.150–X.179
Administrative arrangements	X.180–X.199
OPEN SYSTEM INTERCONNECTION	
Model and notation	X.200–X.209
Service definitions	X.210–X.219
Connection-mode protocol specifications	X.220–X.229
Connectionless-mode protocol specifications	X.230–X.239
PICS proformas	X.240–X.259
Protocol Identification	X.260–X.269
Security Protocols	X.270–X.279
Layer Managed Objects	X.280–X.289
Conformance testing	X.290–X.299
INTERWORKING BETWEEN NETWORKS	
General	X.300–X.349
Satellite data transmission systems	X.350–X.399
MESSAGE HANDLING SYSTEMS	X.400–X.499
DIRECTORY	X.500–X.599
OSI NETWORKING AND SYSTEM ASPECTS	
Networking	X.600–X.629
Efficiency	X.630–X.639
Quality of service	X.640–X.649
Naming, Addressing and Registration	X.650–X.679
Abstract Syntax Notation One (ASN.1)	X.680–X.699
OSI MANAGEMENT	
Systems Management framework and architecture	X.700–X.709
Management Communication Service and Protocol	X.710–X.719
Structure of Management Information	X.720–X.729
Management functions and ODMA functions	X.730–X.799
SECURITY	X.800–X.849
OSI APPLICATIONS	
Commitment, Concurrency and Recovery	X.850–X.859
Transaction processing	**X.860–X.879**
Remote operations	X.880–X.899
OPEN DISTRIBUTED PROCESSING	X.900–X.999

For further details, please refer to ITU-T List of Recommendations.

ITU-T RECOMMENDATION X.861

OPEN SYSTEMS INTERCONNECTION – DISTRIBUTED TRANSACTION PROCESSING: SERVICE DEFINITION

Summary

This Recommendation defines in an abstract way the Distributed Transaction Processing service within the Application layer in terms of the actions and events of the service primitives, the parameter data associated with each service primitive's action and event, and the relationship between, and the valid sequences of these actions and events.

Source

ITU-T Recommendation X.861 was revised by ITU-T Study Group 7 (1997-2000) and was approved under the WTSC Resolution No. 1 procedure on the 12th of December 1997.

FOREWORD

ITU (International Telecommunication Union) is the United Nations Specialized Agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of the ITU. The ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Conference (WTSC), which meets every four years, establishes the topics for study by the ITU-T Study Groups which, in their turn, produce Recommendations on these topics.

The approval of Recommendations by the Members of the ITU-T is covered by the procedure laid down in WTSC Resolution No. 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. The ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, the ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

CONTENTS

Introduction

This Recommendation is one of a set of Standards produced to facilitate the interconnection of computer systems. It is related to other Recommendations and International Standards in the set as defined by the Reference Model for Open Systems Interconnection (see ITU-T Rec. X.200 | ISO/IEC 7498-1). The Reference Model subdivides the area of standardization for interconnection into a series of layers of specification, each of manageable size.

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

a) from different manufacturers;

b) under different management;

c) of different levels of complexity; and

d) of different technologies.

The ITU-T X.860-Series Recommendations and ISO/IEC 10026 defines an OSI TP Model, an OSI TP service and specifies an OSI TP Protocol available within the Application Layer of the OSI Reference Model.

The OSI TP service is an Application Layer service. It is concerned with information which can be related as distributed transactions, which involve two or more open systems.

This Recommendation defines a basic OSI TP service. It provides sufficient facilities to support transaction processing, and establishes a framework for coordination across multiple TP resources in separate open systems.

This Recommendation does not specify the interface to local resources or access facilities that are provided within the local system. However, future enhancement of the Recommendation may deal with these issues.

Recommendation X.861

OPEN SYSTEMS INTERCONNECTION – DISTRIBUTED TRANSACTION PROCESSING: SERVICE DEFINITION[1]

(revised in 1997)

1 Scope

This Recommendation defines in an abstract way the Distributed Transaction Processing service within the Application Layer in terms of:

a) the actions and events of the service primitives;

b) the parameter data associated with each service primitive's action and event; and

c) the relationship between, and the valid sequences of these actions and events.

It does not specify individual implementations or products, nor does it constrain the implementation of entities or interfaces within a computer system.

2 Normative references

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; all users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published.

2.1 Identical Recommendations | International Standards

– ITU-T Recommendation X.200 (1994) | ISO/IEC 7498-1:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: The Basic Model*.

– ITU-T Recommendation X.210 (1993) | ISO/IEC 10731:1994, *Information technology – Open Systems Interconnection – Basic Reference Model: Conventions for the definition of OSI services*.

– ITU-T Recommendation X.217 (1995) | ISO/IEC 8649:1996, *Information technology – Open Systems Interconnection – Service definition for the association control service element*.

– ITU-T Recommendation X.650 (1996) | ISO/IEC 7498-3:1997, *Information technology – Open Systems Interconnection – Basic Reference Model: Naming and addressing*.

2.2 Paired Recommendations | International Standards equivalent in technical content

– ITU-T Recommendation X.860 (1997), *Open Systems Interconnection – Distributed transaction processing: model*.

ISO/IEC 10026-1:1992, *Information technology – Open Systems Interconnection – Distributed Transaction Processing – Part 1: OSI TP Model*.

– ITU-T Recommendation X.862 (1997), *Open Systems Interconnection – Distributed transaction processing: protocol specification*.

ISO/IEC 10026-3:1996, *Information technology – Open Systems Interconnection – Distributed Transaction Processing - Part 3: Protocol specification*.

---

[1] Recommendation X.861 and ISO/IEC 10026-2 "Information technology – Open Systems Interconnection – Distributed Transaction Processing – Part 2: OSI TP Service" were developed in close collaboration and are technically aligned.

3 Definitions

For the purposes of this Recommendation, the definitions given in ITU-T Rec. X.860 | ISO/IEC 10026-1 and the following definitions apply.

3.1 dialogue establishment indication outstanding: A dialogue state in which a TP-BEGIN-DIALOGUE indication with the Confirmation parameter set to "always" has been issued but has not yet been responded to by a TP-BEGIN-DIALOGUE response.

3.2 dialogue establishment request outstanding: A dialogue state in which a TP-BEGIN-DIALOGUE request with the Confirmation parameter set to "always" has been issued but has not yet been responded to by a TPBEGIN-DIALOGUE confirm.

3.3 dialogue termination indication outstanding: A dialogue state in which a TP-END-DIALOGUE indication with the Confirmation parameter set to "true" has been issued while there is no *user error request outstanding*, but has not yet been responded to by a TP-END-DIALOGUE response, or by a TP-U-ERROR request.

3.4 dialogue termination request outstanding: A dialogue state in which a TP-END-DIALOGUE request with the Confirmation parameter set to "true" has been issued, but has not yet been responded to by a TP-END-DIALOGUE confirm, or by a TP-U-ERROR indication.

3.5 exclusive branch: A transaction branch on which one of the following is true:

– the dialogue is with the superior, *ready can be sent*, and either:

  - *ready cannot be received*; or
  - there is *tree checking* at the node;

– the dialogue is with a subordinate, *ready can be sent*, *ready cannot be received*, and either:

  - the Read-only functional unit is not selected and the Early-exit functional unit is not selected; or
  - there is *tree checking* at the node; or

– the dialogue is with a subordinate and the coordination level is "one-phase commitment" and either:

  - the Read-only functional unit is not selected and the Early-exit functional unit is not selected; or
  - there is *tree checking* at the node;

and if the Unchained Transactions functional unit is selected, none of the following service primitives have been issued during the current transaction: TP-READY indication, TP-READ-ONLY indication, TP-ONE-PHASE indication, or TP-EARLY-EXIT indication.

3.6 handshake indication outstanding: A dialogue state in which one of the following service primitives:

– TP-HANDSHAKE indication;

– TP-HANDSHAKE-AND-GRANT-CONTROL indication,

has been issued while there is no *user error request outstanding*, but has not yet been responded to by one of the following service primitives (respectively):

– TP-HANDSHAKE response;

– TP-HANDSHAKE-AND-GRANT-CONTROL response;

or by a TP-U-ERROR request, or, if *the dialogue is coordinated*, by a TP-EARLY-EXIT request or a TP-EARLY-EXIT indication or any *rollback-initiating service primitive*.

3.7 handshake request outstanding: A dialogue state in which one of the following service primitives:

– TP-HANDSHAKE request;

– TP-HANDSHAKE-AND-GRANT-CONTROL request,

has been issued, but has not yet been responded to by one of the following service primitives (respectively):

– TP-HANDSHAKE confirm;

– TP-HANDSHAKE-AND-GRANT-CONTROL confirm,

or by a TP-U-ERROR indication, or, if *the dialogue is coordinated*, by a TP-EARLY-EXIT request or a TP-EARLY-EXIT indication or any *rollback-initiating service primitive*.

3.8 ready can be received (on a branch): If the branch is with a superior and *superior can send ready* on the branch, or the branch is with a subordinate and *subordinate can send ready* on the branch.

3.9 ready can be sent (on a branch): If the branch is with the superior and *subordinate can send ready* on the branch or the branch is with a subordinate and *superior can send ready* on the branch.

3.10 rollback-initiating indication: An indication or confirm that triggers a rollback; it is one of the following service primitives:

– TP-ROLLBACK indication;

– TP-U-ABORT indication with the Rollback parameter set to "true";

– TP-P-ABORT indication with the Rollback parameter set to "true";

– TP-BEGIN-DIALOGUE confirm with the Rollback parameter set to "true".

3.11 rollback-initiating request: A request that triggers a rollback; it is one of the following service primitives:

– TP-ROLLBACK request;

– TP-U-ABORT request for a *coordinated dialogue* not issued during the *termination phase of a transaction* and neither a TP-READ-ONLY indication nor a TP-EARLY-EXIT indication has been issued for the dialogue.

3.12 rollback-initiating service primitive: A service primitive that triggers a rollback; it may be either a *rollback-initiating request* or a *rollback-initiating indication*.

3.13 static one-phase exclusive branch: An *exclusive branch* on which the coordination level is "one-phase commitment".

NOTE – This is only defined for a branch to a subordinate.

3.14 subordinate dialogue: A dialogue with a subordinate.

3.15 subordinate can send ready (on a branch): If the branch is with a subordinate and the Commit functional unit is selected and the Dynamic Commit functional unit is not selected, or the Dynamic Commit functional unit is selected and the Subordinate-may-send-ready parameter of TP-BEGIN-DIALOGUE request was set to "true".

3.16 subordinate subtree: A subtree of a subordinate.

3.17 superior can send ready (on a branch): If the branch is with the superior and the Dynamic Commit functional unit is selected and the Superior-may-send-ready parameter of TP-BEGIN-DIALOGUE indication was set to "true".

3.18 superior dialogue: The dialogue with the superior.

3.19 termination phase of a transaction; termination phase: The phase of a transaction between initiation of commitment or rollback and the end of the transaction.

This phase is entered, for a given TPSUI, upon issuance of a *transaction completion request* or a TP-EARLY-EXIT request or any *rollback-initiating service primitive*.

For a TPSUI which does not have a *dialogue establishment indication outstanding*, this phase is exited upon issuance of a TP-COMMIT-COMPLETE indication or a TP-UNKNOWN-COMPLETE indication or a TP-ROLLBACK-COMPLETE indication.

For a TPSUI which does have a *dialogue establishment indication outstanding* when the termination phase is entered (this can only happen when a TP-ROLLBACK indication is issued), this phase is exited by a TP-BEGIN-DIALOGUE response with the Result parameter set to "rejected(user)" or by a TP-P-ABORT indication for the dialogue; if the dialogue is accepted during the termination phase, the termination phase is exited by the subsequent TP-ROLLBACK-COMPLETE indication.

3.20 TPSUI owes a TP-DONE request: This obligation is created by the issuing of certain indications and confirmations during termination of a transaction; the TPSUI must issue a TP-DONE request before the transaction can be completed.

3.21 transaction completion request: A service request which triggers completion (rather than rollback) of a transaction; it is one of the following service primitives:

– TP-COMMIT request;

– TP-READ-ONLY request;

– TP-ONE-PHASE request.

3.22 transaction subordinate: A TPSUI which is the subordinate for a *coordinated dialogue*.

3.23 transaction superior: A TPSUI which is the superior for one or more *coordinated dialogues*.

3.24 transaction tree constraint: A constraint that cannot be checked at a single node.

3.25 tree checking: There is tree checking at a node if there is a transaction branch from the superior on which either:

– *ready can be sent* and *ready can be received* and the Check-ready-directions parameter of the TP-BEGIN-DIALOGUE indication or the TP-BEGIN-TRANSACTION indication for the current transaction was absent or set to "true"; or

– *ready can be sent*, and *ready cannot be received*.

There may also be tree checking at a node (including at a root node) as the result of a local decision.

NOTE – This would typically occur as the result of configuration information on a real open system.

3.26 two-phase expected branch: A transaction branch on a *coordinated dialogue* with a subordinate, on which *ready can be received* and either:

– none of the One-phase, Read-only, and Early-exit functional units are selected; or

– the Unchained Transactions functional unit is selected, there is *tree checking* at the node, and none of the following service primitives have been issued during the current transaction: TP-ONE-PHASE indication, TP-READ-ONLY indication, or TP-EARLY-EXIT indication.

3.27 user error indication outstanding: A state of a dialogue with the Polarized Control functional unit selected. In this state, a TP-U-ERROR indication, issued while the recipient had control of the dialogue and has neither a *handshake request outstanding* nor a *dialogue termination request outstanding*, has not yet been responded to by a TP-GRANT-CONTROL request, or, if *the dialogue is coordinated*, by a TP-EARLY-EXIT request or a TP-EARLY-EXIT indication or any *rollback-initiating service primitive*.

3.28 user error request outstanding: A state of a dialogue with the Polarized Control functional unit selected. In this state, a TP-U-ERROR request, issued without having control of the dialogue and without having either a *handshake indication outstanding* or a *dialogue termination indication outstanding*, has not yet been responded to by a TP-GRANT-CONTROL indication, a TP-HANDSHAKE indication, a TP-HANDSHAKE-AND-GRANT-CONTROL indication, a TP-END-DIALOGUE indication with the Confirmation parameter set to "true", or, if *the dialogue is coordinated*, by a TP-EARLY-EXIT request or a TP-EARLY-EXIT indication or any *rollback-initiating service primitive*.

4 Abbreviations

Abbreviations used in this Recommendation are defined in ITU-T Rec. X.860 | ISO/IEC 10026-1 (OSI TP Model), except for the following which are used in some tables:

cnf confirm service primitive;

ind indication service primitive;

req request service primitive;

rsp response service primitive.

5 Conventions

5.1 Service conventions

This Recommendation defines services for Distributed Transaction Processing guided by the descriptive conventions defined in ITU-T Rec. X.210 | ISO/IEC 10731.

However, the terms "request" and "indication" are sometimes used in the following ways:

a) a single request may result in multiple indications (an example is that a single TP-COMMIT request may result in TP-PREPARE indications to each direct subordinate TPSUI);

b) several requests may result in a single indication (an example is that a single TP-COMMIT-COMPLETE indication may be issued to a superior TPSUI only after TP-DONE requests have been issued by this TPSUI and by all subordinate TPSUIs in the transaction tree);

c) the convention that a request primitive results in an indication primitive of the same name, is not always followed (for example, the issuance of a TP-COMMIT request may cause a TP-PREPARE indication to be issued).

NOTE – In this Recommendation, requests and responses are described as being issued by the TPSUI whereas indications and confirms are described as being issued by the TPSP.

For a given primitive, the presence of each parameter is described by one of the following values:

blank Not applicable

M Presence is mandatory

U Presence is a user option

O Presence is a provider option

C Presence is conditional

In addition the notation (=) indicates that a parameter value is semantically equal to the value of the parameter of the preceding primitive in the table.

5.2 Usage of the term transaction

In this Recommendation, the term "transaction" is used to denote a distributed provider-supported transaction.

5.3 Usage of italics for notations

In this Recommendation, the following notations, defined in clause 3 or in ITU-T Rec. X.860 | ISO/IEC 10026-1, appear in italics:

– *commitment hinterland*;

– *coordinated dialogue; dialogue is coordinated*;

– *dialogue establishment indication outstanding*;

– *dialogue establishment request outstanding*;

– *dialogue termination indication outstanding*;

– *dialogue termination request outstanding*;

– *exclusive branch*;

– *handshake indication outstanding*;

– *handshake request outstanding*;

– *ready can be received*;

– *ready can be sent*;

– *rollback-initiating indication*;

– *rollback-initiating request*;

– *rollback-initiating service primitive*;

– *static one-phase exclusive branch*;

– *subordinate can send ready*;

– *subordinate dialogue*;

– *subordinate subtree*;

– *superior can send ready*;

– *superior dialogue*;

– *termination phase* (*of a transaction*);

– *transaction tree constraint*;

– *there is tree checking*;

– *TPSUI owes a TP-DONE request*;

– *transaction completion request*;

– *transaction hinterland*;

– *transaction subordinate*;

– *transaction superior*;

– *two-phase expected branch*;

– *user error indication outstanding*;

– *user error request outstanding*.

6 Overview of the OSI TP service

The Distributed Transaction Processing service and its supporting protocol are concerned with creating an environment in which two or more users may interact to:

a) establish dialogues;

b) invoke services of specific user application service elements, subject to the constraints of the TPSP;

c) delimit provider-supported transactions;

d) coordinate work for application-supported transactions or provider-supported transactions;

e) prepare for commitment, and commit or roll back a provider-supported transaction;

f) heuristically place bound data either in the final or initial state;

g) report errors;

h) terminate dialogues allowing all resources allocated to these dialogues to be freed;

i) terminate dialogues abnormally;

j) synchronize processing by handshaking;

k) support chained or unchained sequences of provider-supported transaction branches for a dialogue.

A node crash may result in the TPSP issuing certain TP service primitives more than once (i.e. TP-COMMIT indication, TP-ROLLBACK indication, and TP-HEURISTIC-REPORT indication). The TPSP and the TPSUI are both aware of the node crash through local means.

7 Service facilities

7.1 Functional unit descriptions

The following functional units are defined:

a) **Dialogue**: The Dialogue functional unit supports the basic services required to establish a dialogue between two TPSUIs within which U-ASE primitives may be invoked, signal user-initiated errors and terminate the dialogue. The user or the provider may signal abnormal termination.

b) **SharedControl**: The Shared Control functional unit supports both TPSUIs having control of the dialogue at the same time and allows them to issue request primitives subject only to the normal sequencing constraints of the primitives. For example, data may be transferred by both TPSUIs at the same time.

c) **PolarizedControl**: The Polarized Control functional unit allows only one TPSUI to have control of the dialogue at any point in time. Many request primitives may be issued only by the TPSUI which has control of the dialogue. This restriction is in addition to the normal sequencing constraints for the primitives. For example, a handshake may only be requested by the TPSUI which has control of the dialogue.

d) **Handshake**: The Handshake functional unit allows the TPSUIs to synchronize their processing with one another.

e) **Commit**: The Commit functional unit allows reliable commitment and rollback of transactions.

f) **ChainedTransactions**: The Chained Transactions functional unit supports coordination of both TPSUIs with a chained sequence of transaction branches. The coordination level of the dialogue will always be "commitment" or "one-phase commitment". The subordinate TPSUI will always be a participant in the same transaction as the superior TPSUI.

g) **UnchainedTransactions**: The Unchained Transactions functional unit supports coordination of both TPSUIs with an unchained sequence of transaction branches. The superior determines when the coordination level of the dialogue is "commitment" or "one-phase commitment". At a given point in time, the two TPSUIs may be participants in the same transaction, in different transactions, or one or both TPSUIs may not be involved in a transaction.

h) **Dynamic Commit**: The Dynamic Commit functional unit allows the reliable commitment of a transaction to be organized in a way which is not limited by the orientation of the supporting dialogue tree.

i) **Unchecked Tree**: The Unchecked Tree functional unit allows the suppression of TPSP checks on the construction of transaction trees when the Dynamic Commit functional unit is selected.

j) **Implicit Prepare**: The Implicit Prepare functional unit allows the signal that the transaction is to be completed to be carried in application semantics such that an explicit TP service indication may not be required.

k) **Read-only**: The Read-only functional unit allows a TPSUI which has completed processing of all work related to a transaction to request that it withdraw from participation in the transaction if it has not modified its bound data.

l) **Early-exit**: The Early-exit functional unit allows a TPSUI to indicate that it is unable to contribute to the work of a transaction, its bound data has not been modified, and that the TPSUI has no preference as to whether the transaction commits or is rolled back.

m) **One-phase Commit**: The One-phase Commit functional unit allows a TPSUI that has no requirement for reliable reporting of the outcome of a transaction to request one-phase termination of that transaction.

n) **Completion Diagnostics**: The Completion Diagnostics functional unit allows a TPSUI to signal information related to the completion of a transaction to its superior TPSUI in the transaction tree, including the severity and reason for a rollback request.

o) **Heuristic Containment Required**: The Heuristic Containment Required functional unit allows a TPSUI to require its subordinate to contain heuristic conditions; as a result the TPSUI will not receive heuristic reports from the subordinate.

The Dialogue functional unit shall always be selected.

For a given dialogue, the Shared Control and Polarized Control functional units are mutually exclusive. One and only one of these two functional units shall be selected.

For a given dialogue, the One-phase Commit functional unit may be selected alone (static one-phase commit) or together with both of the Commit and Dynamic Commit functional units (dynamic one-phase commitment).

For a given dialogue, the Chained Transactions and Unchained Transactions functional units are mutually exclusive. If either or both of the Commit or One-phase Commit functional units are selected, one and only one of the Chained Transactions and Unchained Transactions functional units shall be selected. If neither the Commit nor the One-phase Commit functional unit is selected, neither one of the Chained Transactions or Unchained Transactions functional units shall be selected.

For a given dialogue, if the Commit functional unit is selected, then one or more of the following functional units may also be selected in any combination: Implicit Prepare, Read-only, Early-exit, Completion Diagnostics, Heuristic Containment Required.

For a given dialogue, if the One-phase Commit functional unit is selected and the Commit functional unit is not selected, then one or more of the following functional units may also be selected in any combination: Implicit Prepare, Read-only, Early-exit, Completion Diagnostics, Heuristic Containment Required.

For a given dialogue, only if the Commit and Unchained Transactions functional units are selected may the Dynamic Commit functional unit be selected. If the Dynamic Commit functional unit is selected, then the Unchecked Tree functional unit may also be selected.

NOTE – There are multi-dialogue constraints on functional units, which apply when transaction branches are created by TP-BEGIN-DIALOGUE request or TP-BEGIN-TRANSACTION request. For example, the static one-phase commit capability may be selected for a subordinate transaction branch only if there is no superior transaction branch. Such constraints are defined in the descriptions of the relevant services.

With the rules given above, the following dialogue types are valid:

The dialogue functional unit must always be selected. Then, following one of the vertical arrows, if any, means that the selection process is continued. If the selection process stops (i.e. there is no further choice or one chooses not to proceed), then the resulting dialogue type is given on the right hand side.

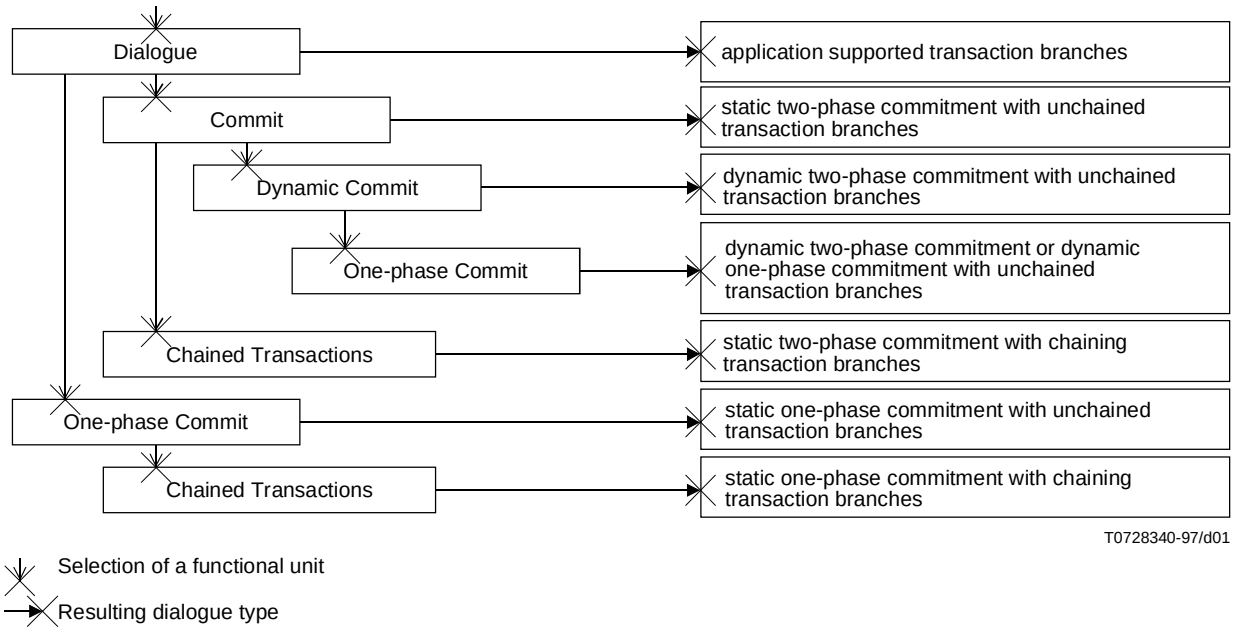

Figure 1/X.861 – Combination of functional units and resulting dialogue types

7.2 Services contained in functional units

Table 1 lists the functional units and the associated services.

Table 1/X.861 – Functional units and their services

Functional unit	Services
Dialogue	TP-BEGIN-DIALOGUE TP-END-DIALOGUE[a)] TP-U-ERROR TP-U-ABORT TP-P-ABORT
Shared Control	(No associated services)
Polarized Control	TP-GRANT-CONTROL TP-REQUEST-CONTROL
Handshake	TP-HANDSHAKE TP-HANDSHAKE-AND-GRANT-CONTROL[b)]
Commit	TP-DEFERRED-END-DIALOGUE TP-DEFERRED-GRANT-CONTROL[b)] TP-PREPARE TP-READY TP-COMMIT TP-DONE TP-COMMIT-COMPLETE TP-ROLLBACK TP-ROLLBACK-COMPLETE TP-HEURISTIC-REPORT
Chained Transactions	(No associated services)
Unchained Transactions	TP-BEGIN-TRANSACTION
Dynamic Commit	(No associated services)
Unchecked Tree	(No associated services)
Implicit Prepare	(No associated services)
Read-only	TP-READ-ONLY TP-UNKNOWN TP-UNKNOWN-COMPLETE
Early-exit	TP-EARLY-EXIT TP-UNKNOWN TP-UNKNOWN-COMPLETE
One-phase Commit	TP-DEFERRED-END-DIALOGUE TP-DEFERRED-GRANT-CONTROL[b)] TP-PREPARE TP-COMMIT[c)] TP-DONE TP-COMMIT-COMPLETE TP-ROLLBACK TP-ROLLBACK-COMPLETE TP-HEURISTIC-REPORT TP-ONE-PHASE TP-UNKNOWN TP-UNKNOWN-COMPLETE
Completion Diagnostics	TP-COMPLETION-REPORT
Heuristic Containment Required	(No associated services)

a) This service shall not be used if the Chained Transactions functional unit is selected.

b) This service may be used only if the Polarized Control functional unit is also selected.

c) TP-COMMIT request shall be used by a node when the One-phase Commit functional unit is selected on the *superior dialogue* and the node has bound data even if the Commit functional unit is not also selected.

7.3 Service for modelling data transfer

Table 2 shows the service for modelling data transfer.

TP-DATA is not a service in the normal sense. It represents the capability of a TPSUI to invoke specific U-ASE services on a dialogue, constrained by the TPSP.

Table 2/X.861 – Service for modelling data transfer

Data transfer	TP-DATA

7.4 Structure of service descriptions

7.4.1 "Purpose" subclause

The "Purpose" subclause describes, in a few words, the purpose of the service.

7.4.2 "Service and parameters" subclause

The "Service and parameters" subclause describes the service primitives and their parameters.

The constraints or conditions on the presence or values of these parameters are described in this subclause.

7.4.3 "Sequences of primitives" subclause

The "Sequence of primitives" subclause is included for certain services; it shows the relationship in time between the service request and the resulting indication, and, if applicable, the subsequent response and the resulting confirm.

7.4.4 "TPSUI conditions" subclause

The "TPSUI conditions" subclause applies to certain requests and responses only; it specifies prerequisites for the respective request or response to be issued by the TPSUI. TPSUI conditions cannot be monitored by the TPSP, nevertheless it is vital for orderly cooperation of the TPSUI and for atomicity that they are obeyed.

TPSUI conditions include:

– the state of bound data;

– the success of synchronization.

7.4.5 "TPSP constraints" subclause

The "TPSP constraints" subclause applies to all service primitives. For request and response service primitives, it specifies prerequisites for issuance by the TPSUI that are enforced by the TPSP. For indication and confirm service primitives, it specifies constraints on the issuance of the service primitives by the TPSP. Constraints on the values of parameters for service primitives are described separately in the "Service and parameters" subclause for each service.

In general, the constraints are based on information associated with the state of the TPSUI at the time the service primitive is issued. Constraints for service primitives that are associated with a particular dialogue relate only to that dialogue unless the constraints explicitly reference other dialogues or attributes that are not related to a particular dialogue.

Information on which constraints are based includes:

– functional units selected for a dialogue;

– superior or subordinate status;

– control of the dialogue;

– coordination level;

– state of bound data;

– transaction state;

– sequence of service primitives and associated parameter values.

7.4.6 "Effects of a service primitive" subclause

The "Effects of a service primitive" subclause describes any effects on the characteristics of the dialogue or the transaction resulting from the issuance of a service primitive.

Effects include:

– initiating or terminating the dialogue or the transaction;

– control of the dialogue;

– superior or subordinate status;

– change of the coordination level;

– in the case of a request or response, the issuance of resulting service primitives.

NOTE – Effects of a service primitive on certain lower layers facilities (e.g. Session tokens) are described in ITU-T Rec. X.862 | ISO/IEC 10026-3.

7.4.7 "Collisions" subclause

There is a collision of two requests if the requests have been issued:

– on opposite sides of the same dialogue; and

– before the indication resulting from the request issued on the other side is either issued or suppressed.

The "Collisions" subclause describes any effects on a service request or response caused by collision with a service primitive issued by the partner TPSUI.

NOTE – Thus a collision subclause does not appear for an indication or confirm primitive.

In general, the effects of a collision involving a particular service are described in the "Collisions" subclause for that service.

These effects include:

– suppression of an indication;

– generation of a different indication.

7.5 Effects of dialogue termination

Whenever a dialogue is terminated for a particular TPSUI, no further service primitives are issued to the TPSUI for the dialogue, except TP-HEURISTIC-REPORT indication, which may be issued during the *termination phase of the transaction*.

For a particular TPSUI, a dialogue is terminated by one of the following service primitives:

– TP-END-DIALOGUE request with the Confirmation parameter set to "false";

– TP-END-DIALOGUE indication with the Confirmation parameter set to "false";

– TP-END-DIALOGUE response;

– TP-END-DIALOGUE confirm;

– TP-BEGIN-DIALOGUE response with the Result parameter set to "rejected(user)";

– TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" or "rejected(user)";

– TP-U-ABORT request;

– TP-U-ABORT indication;

– TP-P-ABORT indication;

– TP-COMMIT-COMPLETE indication when a TP-DEFERRED-END-DIALOGUE request or indication has been issued;

– TP-UNKNOWN-COMPLETE indication when a TP-DEFERRED-END-DIALOGUE indication has been issued and a TP-EARLY-EXIT request has not been issued;

– TP-UNKNOWN-COMPLETE indication when a TP-DEFERRED-END-DIALOGUE request has been issued and a TP-EARLY-EXIT indication has not been issued.

Suppression of subsequent service primitives is not described in the collisions subclauses.

8 Service primitives and their parameters

The OSI TP Service is invoked using a sequence of OSI TP service primitives.

Table 3 lists:

a) the service primitives of the OSI TP Service;

b) for each service primitive, whether the service primitive is associated with a particular dialogue or with the TPSUI as a whole;

c) the subclause in which the service primitive is described; and

d) the parameters associated with each service.

Blanks in the parameters column indicates that the service primitive has no parameters.

Table 3/X.861 – OSI TP service primitives

Services	Primitives	Scope	Subclause	Parameters
TP-BEGIN-DIALOGUE	req/ind/rsp/cnf	Dialogue	10.2	Initiating-AP-Title Initiating-API-Identifier Initiating-AE-Qualifier Initiating-AEI-Identifier Initiating-TPSU-Title Recipient-AP-Title Recipient-API-Identifier Recipient-AE-Qualifier Recipient-AEI-Identifier Recipient-TPSU-Title Functional-Units Quality-of-Service Application-Context-Name Begin-Transaction Confirmation Result Diagnostic Rollback Superior-may-send-ready Subordinate-may-send-ready Check-ready-directions User-Data

Table 3/X.861 – OSI TP service primitives *(continued)*

Services	Primitives	Scope	Subclause	Parameters
TP-END-DIALOGUE	req/ind/rsp/cnf	Dialogue	10.3	Confirmation
TP-U-ERROR	req/ind	Dialogue	10.4	
TP-U-ABORT	req/ind	Dialogue	10.5	Rollback User-Data
TP-P-ABORT	ind	Dialogue	10.6	Diagnostic Rollback
TP-GRANT-CONTROL	req/ind	Dialogue	12.2	
TP-REQUEST-CONTROL	req/ind	Dialogue	12.3	
TP-HANDSHAKE	req/ind/rsp/cnf	Dialogue	13.2	Confirmation-Urgency
TP-HANDSHAKE-AND-GRANT-CONTROL	req/ind/rsp/cnf	Dialogue	13.3	Confirmation-Urgency
TP-BEGIN-TRANSACTION	req/ind	Dialogue	14.5	Check-ready-directions
TP-DEFERRED-END-DIALOGUE	req/ind	Dialogue	14.6	
TP-DEFERRED-GRANT-CONTROL	req/ind	Dialogue	14.7	
TP-PREPARE	req	Dialogue	14.8	Data-Permitted
TP-PREPARE	ind	Dialogue	14.9	Data-Permitted
TP-READY	ind	Dialogue	14.10	
TP-COMMIT	req	TPSUI	14.11	
TP-COMMIT	ind	TPSUI	14.12	
TP-DONE	req	TPSUI	14.13	Heuristic-Report Severity Completion-Data
TP-COMMIT-COMPLETE	ind	TPSUI	14.14	
TP-ROLLBACK	req	TPSUI	14.15	
TP-ROLLBACK	ind	TPSUI	14.16	Severity Diagnostic
TP-ROLLBACK-COMPLETE	ind	TPSUI	14.17	
TP-HEURISTIC-REPORT	ind	Dialogue	14.18	Heuristic-Report
TP-READ-ONLY	req	TPSUI	14.19	Confirmation-Urgency
TP-READ-ONLY	ind	Dialogue	14.20	
TP-EARLY-EXIT	req	TPSUI	14.21	Severity User-Data
TP-EARLY-EXIT	ind	Dialogue	14.22	Severity User-Data
TP-ONE-PHASE	req	TPSUI	14.23	
TP-ONE-PHASE	ind	Dialogue	14.24	

Table 3/X.861 – OSI TP service primitives *(concluded)*

Services	Primitives	Scope	Subclause	Parameters
TP-UNKNOWN	ind	TPSUI	14.25	
TP-UNKNOWN-COMPLETE	ind	TPSUI	14.26	
TP-COMPLETION-REPORT	ind	Dialogue	14.27	Severity Completion-Data Diagnostic
NOTE – The method for identifying the appropriate dialogue for the service primitives which are associated with a particular dialogue is a local matter.				

9 Data transfer

9.1 Overview of data transfer

Data transfer is performed within the framework of OSI TP by issuance of the service primitives offered by one or more U-ASEs. To specify the coordination between these service primitives and OSI TP service primitives, these U-ASE service primitives are modelled as TP-DATA.

NOTE – TP-DATA may not only be used to model data transfer but also to model any other U-ASE services that may be constrained by the TPSP (see ITU-T Rec. X.862 | ISO/IEC 10026-3 for constraints on such services).

9.2 Data transfer service –TP-DATA

9.2.1 Purpose

This service represents the capability of a TPSUI to transfer data. From the standpoint of the TPSP, it is used to specify the coordination between data transfer and other OSI TP services.

This service is never invoked as such, but is used in the OSI TP Service Definition to represent any U-ASE service primitive within the OSI TP framework.

This service is associated with one particular dialogue.

9.2.2 Primitives and parameters

Table 4 lists the TP-DATA primitives.

Table 4/X.861 – TP-DATA primitives and parameters

TP-DATA		
Parameters defined in the U-ASE	req	ind

NOTE – TP-DATA is modelled as an unconfirmed service. This is not meant to exclude the possibility of other types of services (e.g. confirmed services).

9.2.3 TPSP constraints on TP-DATA request

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall have control of the dialogue; or, if the Polarized Control functional unit is selected, the *dialogue shall be coordinated* and a TP-PREPARE indication with the Data-Permitted parameter set to "true" shall have been issued during the current transaction.

The requestor shall not have a *handshake request outstanding*.

The requestor shall not have a *user error indication outstanding*.

The requestor shall have neither a *dialogue termination request outstanding* nor a *dialogue termination indication outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

9.2.4 TPSP constraints on TP-DATA indication

The recipient shall not have a *dialogue establishment request outstanding*.

If the Polarized Control functional unit is selected:

– the recipient shall not have control of the dialogue; or

– the *dialogue shall be coordinated* and a TP-PREPARE request with the Data-Permitted parameter set to "true" shall have been issued during the current transaction.

The recipient shall not have a *handshake indication outstanding*.

The recipient shall not have a *user error request outstanding*.

The recipient shall not have a *dialogue termination indication outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

9.2.5 Collisions

A TP-DATA indication is not issued to a TPSUI if there is a collision of the TP-DATA request and a TP-U-ERROR request.

A TP-DATA indication is not issued for a *coordinated dialogue* after a TP-EARLY-EXIT request or a *rollback-initiating service primitive*.

A TP-DATA indication is not issued for a *coordinated dialogue* after a *transaction completion request*; either:

– a TP-ROLLBACK indication is issued unless a *rollback-initiating service primitive* has already been issued for the current transaction; the Diagnostic parameter will be set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected; or

– a TP-P-ABORT indication will be issued with the Diagnostic parameter set to "user-protocol-error" if the transaction can not be rolled back.

10 The Dialogue functional unit

10.1 Overview of the Dialogue functional unit

The Dialogue functional unit supports the basic services required to establish a dialogue between two TPSUIs within which U-ASE primitives may be invoked, signal user-initiated errors, and terminate the dialogue. The user or the provider may signal abnormal termination.

The Dialogue functional unit shall always be selected.

10.2 Dialogue Establishment service – TP-BEGIN-DIALOGUE

10.2.1 Purpose

This optionally confirmed service is used to establish a dialogue with a new TPSUI.

This service is associated with one particular dialogue.

10.2.2 Primitives and parameters

Table 5 lists the TP-BEGIN-DIALOGUE primitives and their parameters.

Table 5/X.861 – TP-BEGIN-DIALOGUE primitives and their parameters

TP-BEGIN-DIALOGUE				
Parameters	**req**	**ind**	**rsp**	**cnf**
Initiating-AP-Title		O		
Initiating-API-Identifier		O		
Initiating-AE-Qualifier		O		
Initiating-AEI-Identifier		O		
Initiating-TPSU-Title	U	C(=)		
Recipient-AP-Title	M			
Recipient-API-Identifier	U			
Recipient-AE-Qualifier	U			
Recipient-AEI-Identifier	U			
Recipient-TPSU-Title	U			
Functional-Units	M	M(=)		C
Quality-of-Service	U			
Application-Context-Name	M			
Begin-Transaction	C	C(=)		
Confirmation	M	M(=)		
Result			M	M
Diagnostic				C
Rollback				M
Superior-may-send-ready	C	C(=)		
Subordinate-may-send-ready	C	C(=)		
Check-ready-directions	C	C(=)		
User-Data	U	C(=)	U	C(=)

10.2.2.1 Initiating-AP-Title, Initiating-API-Identifier, Initiating-AE-Qualifier, and Initiating-AEI-Identifier are parameters optionally provided by the TPSP. They give information about the application-entity-invocation of the TPSUI that has issued the TP-BEGIN-DIALOGUE request.

These parameters are of type AP-Title, API-Identifier, AE-Qualifier, and AEI-Identifier, respectively, as defined in ITU-T Rec. X.650 | ISO/IEC 7498-3.

10.2.2.2 Initiating-TPSU-Title is an optional parameter which is provided by the TPSUI. It denotes the TPSU and identifies the type of TPSUI which has issued the TP-BEGIN-DIALOGUE request.

10.2.2.3 Recipient-AP-Title, Recipient-API-Identifier, Recipient-AE-Qualifier, and Recipient-AEI-Identifier are parameters which are provided by the initiating TPSUI in order to give information about the recipient application-entity-invocation at which the remote TPSUI will be located.

These parameters are of type AP-Title, API-Identifier, AE-Qualifier, and AEI-Identifier, respectively, as defined in ITU-T Rec. X.650 | ISO/IEC 7498-3.

10.2.2.4 Recipient-TPSU-Title is an optional parameter provided by the initiating TPSUI in order to identify the type of TPSUI with which the initiating TPSUI wants to establish a dialogue.

10.2.2.5 Functional-Units defines, in the request/indication, the functional units which may be used during the life of the dialogue. The combination of functional units specified by this parameter follows the rules according to which functional units may be combined, described in 7.1; the Unchecked Tree functional unit is not specified in this parameter unless both the Superior-may-send-ready and the Subordinate-may-send-ready parameters are set to "true". In the confirm, Functional-Units is present if and only if the Result parameter is set to "rejected(provider)" and the Diagnostic parameter is set to "functional-unit-not-supported"; in this case, it defines the functional units that the recipient application-entity-invocation may support for a dialogue.

10.2.2.6 Quality-of-Service is an optional parameter that specifies the quality of service required for the dialogue. It is of type Quality of Service as defined in ITU-T Rec. X.217 | ISO/IEC 8649.

NOTE – Quality-of-Service parameters are currently being studied.

10.2.2.7 Application-Context-Name is a parameter that specifies the application context to be used for the dialogue. It is of type Application Context Name as defined in ITU-T Rec. X.217 | ISO/IEC 8649.

10.2.2.8 Begin-Transaction is mandatory when the Unchained Transactions functional unit is selected, and is absent otherwise. This parameter is used to specify whether a transaction branch is initiated on the dialogue. It shall take one of the following values:

a) "false", when the subordinate TPSUI will not initially be a participant in a transaction;

b) "true", when the subordinate TPSUI will initially be a participant in a transaction.

10.2.2.9 Confirmation is used to specify whether confirmed dialogue establishment is required. It shall take one of the following values:

a) "always", when confirmed dialogue establishment is required;

b) "negative", when the requestor only requires notification of rejection of the dialogue.

10.2.2.10 Result is used to specify the result of the dialogue establishment attempt. It shall take one of the following values:

a) "accepted", when the Confirmation parameter was set to "always" and the recipient has accepted the dialogue;

b) "rejected(provider)", when the TPSP has rejected the dialogue. The value "rejected(provider)" is only valid on the confirm service primitive;

c) "rejected(user)", when the recipient has rejected the dialogue.

10.2.2.11 Diagnostic is a conditional parameter which is present in the confirm if the Result parameter is set to "rejected(provider)" and is absent otherwise. It describes the type of error which caused the dialogue to be rejected. It shall take one of the following values:

a) "recipient-unknown" when the parameters identifying the recipient application-entity-invocation do not identify a known application-entity-invocation;

b) "recipient-tpsu-title-unknown" when the TPSP cannot find the requested TPSU-Title at the recipient;

c) "tpsu-not-available(permanent)" when the dialogue request is recognized as being valid, but the addressed TPSU is not available due to a permanent failure. It is not worth trying again until the failure has been repaired;

d) "tpsu-not-available(transient)" when the dialogue request is recognized as being valid, but the addressed TPSU is not available due to a transient condition. It might be worth retrying with a reasonable expectation of success;

e) "recipient-tpsu-title-required" when the recipient application-entity-invocation requires the presence of the Recipient-TPSU-Title and this parameter was not provided in the TP-BEGIN-DIALOGUE request;

f) "functional-unit-not-supported" when one or more of the functional units selected in the TP-BEGIN-DIALOGUE request are not supported by the recipient application-entity-invocation for the dialogue;

g) "functional-unit-combination-not-supported" when the combination of functional units, selected in the TP-BEGIN-DIALOGUE request is not supported by the recipient application-entity-invocation for the dialogue;

h) "subordinate shall be commit master" when the recipient application-entity-invocation does not support the Subordinate-may-send-ready parameter being set to "true";

i) "subordinate shall be commit slave" when the recipient application-entity-invocation does not support the Superior-may-send-ready parameter being set to "true";

j) "no-reason-given".

NOTE – It is recognized that, with respect to diagnostic values, work is still in progress to provide an integrated treatment across all the layers of the OSI Reference Model.

10.2.2.12 **Rollback** is a parameter of the confirm primitive. It shall take one of the following values:

a) "true", if the transaction in which the recipient is a participant is being rolled back; this value has the same usage and semantics as the TP-ROLLBACK indication. This value occurs on a TP-BEGIN-DIALOGUE confirm if the *dialogue is coordinated*, the Result parameter is set to "rejected(provider)" or "rejected(user)", and the TP-BEGIN-DIALOGUE confirm is issued after a *transaction completion request*.

NOTE – The Rollback parameter will be set to "true" after a TP-READ-ONLY request or a TP-ONE-PHASE request if the TPPM has bound data or there is bound data in the *subordinate subtree*.

b) "false", otherwise.

10.2.2.13 **Superior-may-send-ready** is mandatory when the Dynamic Commit functional unit is selected and is absent otherwise. This parameter is used to specify whether the superior will issue a ready signal or one-phase signal to the subordinate as soon as it is able. It shall take one of the following values:

a) "false", when the superior will not send ready;

b) "true", when the superior will send ready as soon as it is able.

10.2.2.14 **Subordinate-may-send-ready** is mandatory when the Dynamic Commit functional unit is selected and is absent otherwise. This parameter is used to specify whether the subordinate will issue a ready signal or one-phase signal to the superior as soon as it is able. It shall take one of the following values:

a) "false", when the subordinate will not send ready;

b) "true", when the subordinate will send ready as soon as it is able.

10.2.2.15 **Check-ready-directions** is mandatory when the Dynamic Commit and Unchecked Tree functional units are selected and the Begin-Transaction parameter is set to "true", and is absent otherwise. It shall take one of the following values:

a) "true", when the TPSP shall check that the settings in the *subordinate's subtree* of the Superior-may-send-ready and Subordinate-may-send-ready parameters can not lead to a deadlocked transaction tree;

b) "false", when the TPSUIs will take responsibility for ensuring that created transaction trees will be viable.

10.2.2.16 **User-Data** is an optional parameter that may be present in the request/indication and/or in the response/confirm. This parameter may be present in the response/confirm only if the Result parameter is set to "accepted" or "rejected(user)".

In the request and indication, this parameter may identify user-specific semantics associated with the dialogue establishment attempt, for example, security-related information for validation, or additional information regarding the particular activity to be commenced.

In the response and confirm, this parameter may identify user-specific semantics associated with the acceptance or rejection of the dialogue by the recipient TPSUI.

10.2.3 Sequence of primitives

The time sequence diagram of Figure 2 shows the dialogue establishment sequence of primitives when the service is not confirmed.

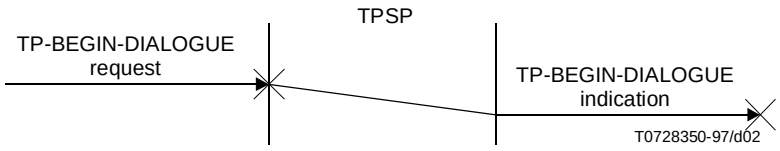

Figure 2/X.861 – Unconfirmed TP-BEGIN-DIALOGUE sequence of primitives

The time sequence diagram of Figure 3 shows the dialogue establishment sequence of primitives when the service is confirmed.

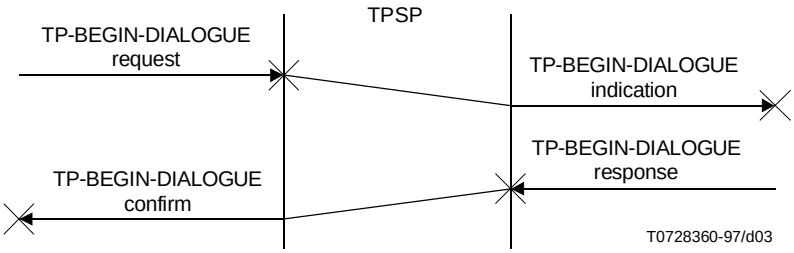

Figure 3/X.861 – Confirmed TP-BEGIN-DIALOGUE sequence of primitives

10.2.4 TPSP constraints on TP-BEGIN-DIALOGUE request

The requestor shall not have a *dialogue establishment indication outstanding* with the superior.

This service primitive shall be issued as the first service primitive for the particular dialogue and shall not be issued more than once for a particular dialogue.

If the TP-BEGIN-DIALOGUE request is used to establish a *coordinated dialogue*, the current transaction shall not be in the *termination phase*.

If the dialogue will have a coordination level of "one-phase commitment", then there shall not be superior transaction branch.

If this will be a *static one-phase exclusive branch*, then there shall not be a *two-phase expected branch*.

If this will be a *two-phase expected branch*, then there shall not be a *static one-phase exclusive branch*.

If this will be an *exclusive branch*, then there shall not be an existing *exclusive branch*.

If the Dynamic Commit functional unit is selected and the Begin-Transaction parameter is set to "true", then if *there is tree checking*, then the Check-ready-directions parameter shall be absent or set to "true".

10.2.5 Effects of a TP-BEGIN-DIALOGUE request

The requestor is the superior of the recipient.

The requestor has control of the dialogue.

If the Chained Transactions functional unit is selected, or if the Unchained Transactions functional unit is selected and the Begin-Transaction parameter is set to "true", then:

a) if the Commit functional unit is selected, the coordination level of the dialogue is initialized to "commitment";

b) if the One-phase Commit functional unit is selected and the Commit functional unit is not selected, the coordination level of the dialogue is initialized to "one-phase commitment",

otherwise the coordination level of the dialogue is initialized to "none".

If the *dialogue is coordinated*, the requestor becomes a participant in a new transaction if it is not already a participant in a transaction.

If the TPSP rejects the dialogue, it issues a TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" to the requestor and does not issue a TP-BEGIN-DIALOGUE indication.

10.2.6 TPSP constraints on TP-BEGIN-DIALOGUE indication

The recipient shall not have any other dialogues.

This service primitive shall be issued as the first service primitive for that particular TPSUI and shall not be issued more than once for a particular TPSUI.

10.2.7 Effects of a TP-BEGIN-DIALOGUE indication

The recipient is a subordinate of the requestor.

If the Chained Transactions functional unit is selected, or if the Unchained Transactions functional unit is selected and the Begin-Transaction parameter is set to "true", then:

a) if the Commit functional unit is selected, the coordination level of the dialogue is initialized to "commitment";

b) if the One-phase Commit functional unit is selected and the Commit functional unit is not selected, the coordination level of the dialogue is initialized to "one-phase commitment",

otherwise the coordination level of the dialogue is initialized to "none".

If the *dialogue is coordinated* and the Confirmation parameter is set to "negative", the recipient becomes a participant in the same transaction as the requestor by manipulating bound data or issuing any service primitives other than a TP-BEGIN-DIALOGUE response with the Result parameter set to "rejected(user)"; otherwise if the *dialogue is coordinated* and the Confirmation parameter is set to "always", the recipient becomes a participant in the same transaction as the requestor by issuing a TP-BEGIN-DIALOGUE response with the Result parameter set to "accepted".

If the Polarized Control functional unit is selected, the recipient does not have control of the dialogue.

10.2.8 TPSUI conditions on TP-BEGIN-DIALOGUE response

If the *dialogue is coordinated*, the responder shall not have manipulated bound data.

10.2.9 TPSP constraints on TP-BEGIN-DIALOGUE response

A TP-BEGIN-DIALOGUE indication shall have been issued to the responder.

Bound data handled by the TPSP shall not be changed prior to the issuance of a TP-BEGIN-DIALOGUE response.

The TPSUI shall not have issued any requests or responses to any dialogues (including TP-BEGIN-DIALOGUE requests).

If the Confirmation parameter of the TP-BEGIN-DIALOGUE indication is set to "negative", a TP-BEGIN-DIALOGUE response with the Result parameter set to "accepted" shall not be issued.

NOTE – Indications may be issued for the dialogue before TP-BEGIN-DIALOGUE response (for example a TP-DATA indication).

10.2.10 Effects of a TP-BEGIN-DIALOGUE response

If the Result parameter is set to "accepted", the dialogue is established.

If the Result parameter is set to "rejected(user)", the dialogue is terminated.

If the *dialogue is coordinated* and the Result parameter is set to "accepted", the recipient is a participant in the same transaction as the requestor.

NOTE – If a TP-ROLLBACK indication has been issued before a TP-BEGIN-DIALOGUE response with the Result parameter set to "rejected(user)", no TP-DONE request is owed and TP-ROLLBACK-COMPLETE indication is not issued.

10.2.11 TPSP constraints on TP-BEGIN-DIALOGUE confirm

No indications or confirms shall have been issued for the dialogue.

10.2.12 Effects of a TP-BEGIN-DIALOGUE confirm

If the Result parameter is set to "accepted", the dialogue is established.

If the Result parameter is set to "rejected(provider)" or "rejected(user)", the dialogue is terminated:

a) if the value of the Rollback parameter is "true", then the transaction is being rolled back. This value has the same usage and semantics as a TP-ROLLBACK indication; the *TPSUI owes a TP-DONE request*;

b) if the TP-BEGIN-DIALOGUE confirm is issued with the Rollback parameter set to "false", for the only *coordinated dialogue*, before the transaction is in the *termination phase*, there is no longer a distributed provider-supported transaction; in such a situation, the TPSUI shall either continue with the current transaction by initiating new transaction branches, or terminate the transaction through the use of a *transaction completion request* or TP-ROLLBACK request.

10.2.13 Collisions

A TP-BEGIN-DIALOGUE confirm is not issued to a TPSUI which has already terminated the dialogue by issuing a TP-U-ABORT request, or to a TPSUI which has received a TP-P-ABORT indication.

10.3 Dialogue Termination service – TP-END-DIALOGUE

10.3.1 Purpose

This optionally confirmed service is used to terminate a dialogue. Either the superior or subordinate may terminate a dialogue.

This service is associated with one particular dialogue.

10.3.2 Primitives and parameters

Table 6 lists the TP-END-DIALOGUE primitives and their parameter.

Table 6/X.861 – TP-END-DIALOGUE primitives and their parameter

TP-END-DIALOGUE				
Parameters	req	ind	rsp	cnf
Confirmation	M	M(=)		

Confirmation is a parameter that indicates whether the TP-END-DIALOGUE request will be confirmed. It shall take one of the following values:

a) "true", when confirmation is required and dialogue termination is conditional (confirmed dialogue termination service);

b) "false", when dialogue termination is unconditional (unconfirmed dialogue termination service).

10.3.3 Sequence of primitives

The time sequence diagram of Figure 4 shows the dialogue termination sequence of primitives when the service is not confirmed.

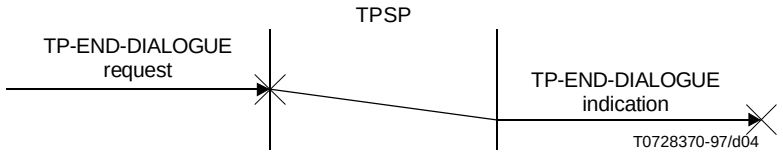

Figure 4/X.861 – Unconfirmed TP-END-DIALOGUE sequence of primitives

The time sequence diagram of Figure 5 shows the dialogue termination sequence of primitives when the service is confirmed.

In Figure 5, TP-END-DIALOGUE response and TP-END-DIALOGUE confirm may be replaced by TP-U-ERROR request and TP-U-ERROR indication, respectively, if the dialogue termination is rejected.

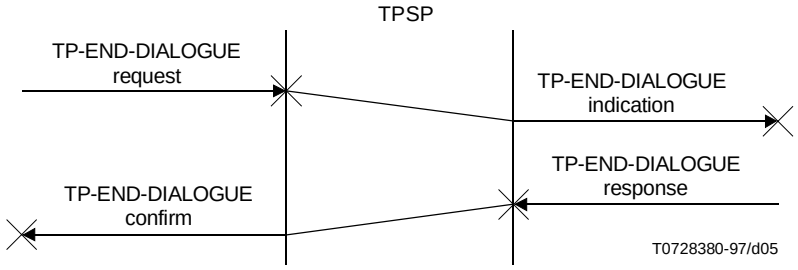

Figure 5/X.861 – Confirmed TP-END-DIALOGUE sequence of primitives

10.3.4 TPSP constraints on TP-END-DIALOGUE request

The requestor shall have neither a *dialogue establishment request outstanding* nor a *dialogue establishment indication outstanding*.

The requestor shall have control of the dialogue and shall not have a *user error indication outstanding*.

The requestor shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The coordination level of the dialogue shall be "none".

The requestor shall have neither a *dialogue termination request outstanding* nor a *dialogue termination indication outstanding*.

10.3.5 Effects of a TP-END-DIALOGUE request

If the Confirmation parameter is set to "false", the dialogue is terminated.

10.3.6 TPSP constraints on TP-END-DIALOGUE indication

The recipient shall have neither a *dialogue establishment request outstanding* nor *a dialogue establishment indication outstanding*.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue.

The recipient shall have neither a *handshake indication outstanding* nor a *dialogue termination indication outstanding*.

The coordination level of the dialogue shall be "none".

10.3.7 Effects of a TP-END-DIALOGUE indication

If the Confirmation parameter is set to "false", the dialogue is terminated.

If the Confirmation parameter is set to "true" and the recipient had a *user error request outstanding*, the recipient has control of the dialogue.

10.3.8 TPSP constraints on TP-END-DIALOGUE response

The requestor shall not have a *handshake request outstanding*.

The requestor shall have a *dialogue termination indication outstanding*.

10.3.9 Effects of a TP-END-DIALOGUE response

The dialogue is terminated.

10.3.10 TPSP constraints on TP-END-DIALOGUE confirm

The recipient shall not have a *handshake indication outstanding*.

The recipient shall have a *dialogue termination request outstanding*.

10.3.11 Effects of a TP-END-DIALOGUE confirm

The dialogue is terminated.

10.3.12 Collisions

If the Shared Control functional unit is selected, a TP-END-DIALOGUE indication with the Confirmation parameter set to "true" is not issued to a TPSUI if there is a collision of the TP-END-DIALOGUE request and a TP-U-ERROR request.

A TP-END-DIALOGUE indication is not issued to a TPSUI which has issued a TP-BEGIN-TRANSACTION request; instead, a TP-P-ABORT indication with the Diagnostic parameter set to "begin-transaction-end-dialogue-collision" is issued.

A TP-END-DIALOGUE indication with the Confirmation parameter set to "true" is not issued to a TPSUI which has issued a TP-END-DIALOGUE request with the Confirmation parameter set to "true"; instead, a TP-P-ABORT indication with the Diagnostic parameter set to "end-dialogue-collision" is issued.

A TP-END-DIALOGUE confirm is not issued to a TPSUI which has already terminated the dialogue by issuing a TP-U-ABORT request, or to a TPSUI which has received a TP-P-ABORT indication.

10.4 User Error Reporting service – TP-U-ERROR

10.4.1 Purpose

This service is used to notify a partner TPSUI of a processing error occurrence; it also serves as a negative response to the handshake service, to the handshake with grant-control service, and to the confirmed dialogue termination service.

The description of the error may be communicated by the TPSUI, using TP-DATA.

This service is associated with one particular dialogue.

10.4.2 Primitives and parameters

Table 7 lists the TP-U-ERROR primitives.

Table 7/X.861 – TP-U-ERROR primitives

TP-U-ERROR		
No parameters	req	ind

10.4.3 Sequence of primitives

The time sequence diagram of Figure 6 shows the TP-U-ERROR sequence of primitives.

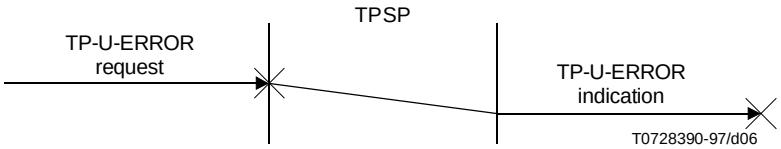

Figure 6/X.861 – TP-U-ERROR sequence of primitives

The time sequence diagram of Figure 7 shows a TP-U-ERROR request issued as a negative confirmation of a handshake service.

In Figure 7, the TP-HANDSHAKE may be replaced by a TP-END-DIALOGUE with Confirmation set to "true" or, if the Polarized Control functional unit has been selected, a TP-HANDSHAKE-AND-GRANT-CONTROL.

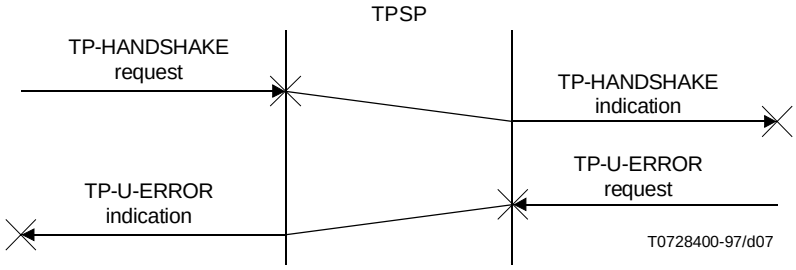

Figure 7/X.861 – TP-U-ERROR as the negative confirmation to a handshake service

10.4.4 TPSUI conditions on TP-U-ERROR request

If the requestor has a *handshake indication outstanding*, synchronization was not successful.

10.4.5 TPSP constraints on TP-U-ERROR request

The requestor shall not have a *dialogue establishment indication outstanding*.

If the Polarized Control functional unit is selected, the requestor shall not have a *handshake request outstanding*.

The requestor shall not have a *dialogue termination request outstanding* unless the Shared Control functional unit is selected and the requestor has a *handshake indication outstanding*.

The requestor shall have neither a *user error request outstanding* nor a *user error indication outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If *the dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

10.4.6 Effects of a TP-U-ERROR request

If the Polarized Control functional unit is selected, and there was a *handshake indication outstanding* or a *dialogue termination indication outstanding*, the requestor has control of the dialogue.

10.4.7 TPSP constraints on TP-U-ERROR indication

The recipient shall not have a *dialogue establishment request outstanding*.

If the Polarized Control functional unit is selected, the recipient shall have neither a *handshake indication outstanding* nor a *dialogue termination indication outstanding*.

The recipient shall have neither a *user error request outstanding* nor a *user error indication outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If *the dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

10.4.8 Effects of a TP-U-ERROR indication

If the Polarized Control functional unit is selected and the recipient has control of the dialogue, then the recipient shall surrender control by issuing a TP-GRANT-CONTROL request.

If the Polarized Control functional unit is selected, and a *handshake request is outstanding* or a *dialogue termination request is outstanding*, the recipient does not have control of the dialogue.

10.4.9 Collisions

A TP-U-ERROR indication is not issued to a TPSUI which has issued a TP-U-ERROR request while not having control of the dialogue.

A TP-U-ERROR indication is not issued for a *coordinated dialogue* if a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued during the current transaction.

A TP-U-ERROR indication is not issued for a *coordinated dialogue* after *a transaction completion request* or a TP-PREPARE request has been issued; either:

– a TP-ROLLBACK indication is issued unless a *rollback-initiating service primitive* has already been issued for the current transaction; the Diagnostic parameter will be set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected; or

– a TP-P-ABORT indication will be issued with the Diagnostic parameter set to "user-protocol-error" if the transaction cannot be rolled back.

10.5 User Abort service – TP-U-ABORT

10.5.1 Purpose

This service is used to abort a dialogue. Use of this service may cause loss of indications and/or confirms to both TPSUIs.

This service is associated with one particular dialogue.

10.5.2 Primitives and parameters

Table 8 lists the TP-U-ABORT primitives and their parameters.

Table 8/X.861 – TP-U-ABORT primitives and their parameters

TP-U-ABORT		
Parameters	req	ind
Rollback User-Data	U	M C(=)

10.5.2.1 **Rollback** is a parameter of the indication primitive. It shall take one of the following values:

a) "true", if the transaction in which the recipient is a participant is being rolled back; this value has the same usage and semantics as the TP-ROLLBACK indication.

   This value shall occur if all of the following conditions are satisfied:

   – the *dialogue is coordinated*;

   – there is no *dialogue establishment indication outstanding*;

   – a TP-COMMIT indication or a TP-UNKNOWN indication or a *rollback-initiating service primitive* has not been issued for the current transaction;

   – there has not been a collision of a TP-BEGIN-TRANSACTION request and the TP-U-ABORT indication;

   – a TP-EARLY-EXIT request or TP-READ-ONLY request has not been issued;

   – a TP-EARLY-EXIT indication or TP-READ-ONLY indication has not been issued for the dialogue.

   This value may also occur whenever the TPSP is in a state that requires the current transaction to be rolled back;

b) "false", if no rollback of a transaction occurs or if a rollback is already in progress.

   This value shall occur if any of the following conditions are satisfied:

   – the dialogue has a coordination level of "none";

   – the *dialogue is coordinated*, but the recipient has a *dialogue establishment indication outstanding*;

   – the *dialogue is coordinated*, but a TP-EARLY-EXIT indication or a TP-READ-ONLY indication has been issued for the dialogue, or a TP-COMMIT indication or a TP-UNKNOWN indication or a *rollback-initiating service primitive* has been issued for the current transaction.

   This value may also occur whenever the TPSP is in a state such that the rollback of the current transaction is not required.

   NOTE – The TPSP is in such a state when a TP-U-ABORT request disrupts a TP-BEGIN-TRANSACTION request or when a TP-U-ABORT request is issued by a subordinate during a rollback procedure and received by the superior after TP-ROLLBACK-COMPLETE indication (see ITU-T Rec. X.862 | ISO/IEC 10026-3 for more information).

10.5.2.2 **User-Data** is an optional parameter which may be used to convey user-specific semantics associated with the abort of the dialogue.

10.5.3 Sequence of primitives

The time sequence diagram of Figure 8 shows the TP-U-ABORT sequence of primitives.

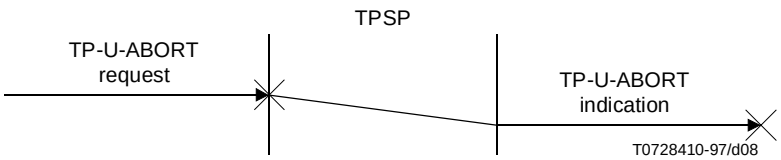

Figure 8/X.861 – TP-U-ABORT sequence of primitives

10.5.4 TPSP constraints on TP-U-ABORT request

The requestor shall not have a *dialogue establishment indication outstanding*.

For a *coordinated dialogue*, TP-U-ABORT request shall only be issued either:

a) before the current transaction is in the *termination phase*; or

b) during the termination of the current transaction, after the issuance of any of the following service primitives:

   – TP-ROLLBACK request;

   – TP-ROLLBACK indication;

   – TP-EARLY-EXIT request;

– TP-EARLY-EXIT indication;

– TP-UNKNOWN indication;

– TP-U-ABORT indication issued for another *coordinated dialogue*;

– TP-P-ABORT indication issued for another *coordinated dialogue*;

– TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" or "rejected(user)" for another *coordinated dialogue*;

and prior to the issuance of the subsequent TP-DONE request.

10.5.5 Effects of a TP-U-ABORT request

The dialogue is terminated. However, if a TP-U-ABORT request is issued for a *coordinated dialogue* with a subordinate during the *termination phase of a transaction*, or after a TP-PREPARE request has been issued for the dialogue, a TP-HEURISTIC-REPORT indication may still be issued for the dialogue.

NOTE – With the Implicit Prepare functional unit, heuristic decisions may be taken resulting from a prepare which is implicit in application semantics; the TPSP cannot detect such prepares and thus cannot warn of heuristic hazards when failures occur.

A TP-U-ABORT request, if used for a *coordinated dialogue*, causes a rollback of the current transaction unless one of the following is true:

– a TP-READ-ONLY indication or TP-EARLY-EXIT indication has been issued for the dialogue; or

– the TP-U-ABORT request is issued during the *termination phase of the current transaction*.

A TP-U-ABORT request, if used during the *termination phase of the transaction* for a dialogue with the Chained Transactions functional unit selected after a TP-COMMIT indication has been issued, and on which no TP-DEFERRED-END-DIALOGUE service has been issued, causes the subsequent transaction to rollback.

If the *dialogue is coordinated* and the transaction is in the *termination phase*, the *TPSUI owes a TP-DONE request*.

10.5.6 TPSP constraints on TP-U-ABORT indication

The recipient shall not have a *dialogue establishment request outstanding*.

If a TP-ROLLBACK indication has been issued, the recipient shall not have a *dialogue establishment indication outstanding*.

10.5.7 Effects of a TP-U-ABORT indication

The dialogue is terminated. However, if a TP-U-ABORT indication is issued for a *coordinated dialogue* with a subordinate during the *termination phase of a transaction* or after a TP-PREPARE request has been issued for the dialogue, a TP-HEURISTIC-REPORT indication may still be issued for the dialogue.

A TP-U-ABORT indication with the Rollback parameter set to "true" causes a rollback of the current transaction.

If a TP-U-ABORT indication is issued for a dialogue with the Chained Transactions functional unit selected, between a TP-COMMIT indication and the corresponding TP-COMMIT-COMPLETE indication, and no other *coordinated dialogues* remain after the TP-COMMIT-COMPLETE indication has been issued, there is no longer a provider-supported transaction; in such a situation, the TPSUI shall either continue with the (then) current transaction by initiating new transaction branches, or terminate the transaction through the use of *a transaction completion request* or TP-ROLLBACK request.

If a TP-U-ABORT indication is issued with the Rollback parameter set to "false", for the only *coordinated dialogue*, before the *transaction is in the termination phase*, there is no longer a distributed provider-supported transaction; in such a situation, the TPSUI shall either continue with the current transaction by initiating new transaction branches, or terminate the transaction through the use of *a transaction completion request* or TP-ROLLBACK request.

If the *dialogue is coordinated* and the transaction is in the *termination phase*, *the TPSUI owes a TP-DONE request*.

10.5.8 Collisions

A TP-U-ABORT indication is not issued to a TPSUI which has already terminated the dialogue by issuing a TP-END-DIALOGUE request with the Confirmation parameter set to "false", or by issuing a TP-U-ABORT request, or has received a TP-P-ABORT indication.

10.6 Provider Abort service – TP-P-ABORT

10.6.1 Purpose

This service is used by the TPSP to notify the TPSUIs of the occurrence of a failure which caused the dialogue between them to be terminated. This service may cause loss of indications and/or confirms to both TPSUIs.

This service is associated with one particular dialogue.

10.6.2 Primitives and parameters

Table 9 lists the TP-P-ABORT primitive and its parameters.

Table 9/X.861 – TP-P-ABORT indication and its parameters

TP-P-ABORT	
Parameters	ind
Diagnostic Rollback	M M

10.6.2.1 **Diagnostic** is a parameter which describes the type of error which has occurred. It shall take one of the following values:

a) "permanent-failure" when a permanent error condition has been encountered. It is not worth trying again until the failure has been repaired;

b) "transient-failure" when a transient condition has been encountered, for example, congestion. It might be worth retrying with a reasonable expectation of success;

c) "protocol-error" when a protocol error has been encountered;

d) "begin-transaction-reject" when the TP-BEGIN-TRANSACTION indication is not issued because the recipient is already involved in a transaction or because of a local condition;

e) "end-dialogue-collision" when two TP-END-DIALOGUE requests with the Confirmation parameter set to "true" have collided;

f) "begin-transaction-end-dialogue-collision" when a TP-BEGIN-TRANSACTION request and a TP-END-DIALOGUE request have collided;

g) "user-protocol-error" when user data has been received or a TP Service request has been issued by a partner TPSUI after a *transaction completion request* has been issued, and the transaction can no longer be rolled back.

NOTE 1 – Such TP services could be one of TP-U-ERROR request, TP-HANDSHAKE request, TP-DEFERRED-END-DIALOGUE request. The issuance of a TP-P-ABORT indication in this case implies that a ready or equivalent signal has already been sent, as otherwise a TP-ROLLBACK indication would be issued with the Diagnostic parameter set to "user-data-transaction-completion-collision".

NOTE 2 – It is recognized that, with respect to diagnostic values, work is still in progress to provide an integrated treatment across all the layers of the OSI Reference Model.

10.6.2.2 **Rollback** is a parameter of the indication primitive. It shall take one of the following values:

a) "true", if the transaction in which the recipient is a participant is being rolled back; this value has the same usage and semantics as the TP-ROLLBACK indication.

This value shall occur if all the following conditions are satisfied:

- the *dialogue is coordinated*;
- a *dialogue establishment indication is not outstanding*;
- a TP-COMMIT indication or a TP-UNKNOWN indication or a *rollback-initiating service primitive* has not been issued for the current transaction;
- there has not been a collision of a TP-BEGIN-TRANSACTION request and the TP-P-ABORT indication;
- a TP-EARLY-EXIT request or TP-READ-ONLY request has not been issued;
- a TP-EARLY-EXIT indication or TP-READ-ONLY indication has not been issued for the dialogue.

This value may also occur whenever the TPSP is in a state that requires the current transaction to be rolled back.

b) "false", if no rollback of a transaction occurs or if a rollback is already in progress. This value shall occur if any of the following conditions are satisfied:

– the dialogue has a coordination level of "none";

– the *dialogue is coordinated*, but the recipient has a *dialogue establishment indication outstanding*;

– the *dialogue is coordinated*, but a TP-EARLY-EXIT indication or a TP-READ-ONLY indication has been issued for the dialogue, or a TP-COMMIT indication or a TP-UNKNOWN indication or a *rollback-initiating service primitive* has been issued for the current transaction.

This value may also occur whenever the TPSP is in a state such that the rollback of the current transaction is not required.

NOTE – The TPSP is in such a state when a TP-P-ABORT indication disrupts a TP-BEGIN-TRANSACTION (see ITU-T Rec. X.862 | ISO/IEC 10026-3 for more information).

10.6.3 Sequence of primitives

The time sequence diagram of Figure 9 shows the TP-P-ABORT sequence of primitives.

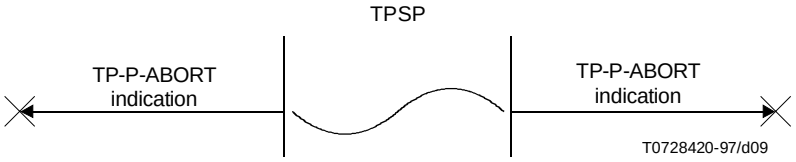

Figure 9/X.861 – TP-P-ABORT sequence of primitives

10.6.4 Effects of a TP-P-ABORT indication

The dialogue is terminated. However, if a TP-P-ABORT indication is issued for a *coordinated dialogue* with a subordinate during the *termination phase of a transaction* or after a TP-PREPARE request has been issued for the dialogue, a TP-HEURISTIC-REPORT indication may still be issued for the dialogue.

NOTE – With the Implicit Prepare functional unit, heuristic decisions may be taken resulting from a prepare which is implicit in application semantics; the TPSP cannot detect such prepares and thus cannot warn of heuristic hazards when failures occur.

A TP-P-ABORT indication with the Rollback parameter set to "true" causes a rollback of the current transaction.

If a TP-P-ABORT indication is issued for a dialogue with the Chained Transactions functional unit selected, between a TP-COMMIT indication and the corresponding TP-COMMIT-COMPLETE indication, the TPSP will trigger a rollback of the next transaction unless the deferred end dialogue service has been issued.

If a TP-P-ABORT indication is issued with the Rollback parameter set to "false", for the only *coordinated dialogue*, before the transaction is in the *termination phase*, there is no longer a distributed provider-supported transaction; in such a situation, the TPSUI shall either continue with the current transaction by initiating new transaction branches, or terminate the transaction through the use of *a transaction completion request* or a TP-ROLLBACK request.

If the *dialogue is coordinated* and the transaction is in the *termination phase*, the *TPSUI owes a TP-DONE request*.

11 The Shared Control functional unit

11.1 Overview of the Shared Control functional unit

In the Shared Control functional unit both TPSUIs have control of the dialogue and so may issue request primitives subject only to the normal sequencing constraints of the primitives. For example, data may be transferred by both TPSUIs at the same time.

The Shared Control and Polarized Control functional units are mutually exclusive for a given dialogue.

There are no service primitives associated with the Shared Control functional unit.

12 The Polarized Control functional unit

12.1 Overview of the Polarized Control functional unit

In the Polarized Control functional unit at most one TPSUI has control of the dialogue at any point in time.

When a dialogue is established, control rests with the dialogue initiator. Subsequently, control may be passed by use of a TP-GRANT-CONTROL request or a TP-DEFERRED-GRANT-CONTROL request, and may be requested by a TP-REQUEST-CONTROL request. In addition, when a TP-U-ERROR request is issued in response to a TP-HANDSHAKE indication, or a TP-END-DIALOGUE indication with the Confirmation parameter set to "true", or a TP-HANDSHAKE-AND-GRANT-CONTROL indication, control is passed automatically to the issuer of the TP-U-ERROR request. In the case of rollback or use of the TP-EARLY-EXIT service, at completion of the transaction, control is with the TPSUI which had it at the beginning of the transaction.

Many request primitives may be issued only by the TPSUI which has control of the dialogue. This restriction is in addition to the normal sequencing constraints for the primitives. For example, data may only be transferred by the TPSUI that has control of the dialogue or by a TPSUI that has been issued a TP-PREPARE indication with the Data-Permitted parameter set to "true".

The Shared Control and Polarized Control functional units are mutually exclusive, for a given dialogue.

12.2 Grant Control service – TP-GRANT-CONTROL

12.2.1 Purpose

This service is used by a TPSUI to grant control of the dialogue to the partner TPSUI.

This service is associated with one particular dialogue.

12.2.2 Primitives and parameters

Table 10 lists the TP-GRANT-CONTROL primitives.

Table 10/X.861 – TP-GRANT-CONTROL primitives

TP-GRANT-CONTROL		
No parameters	req	ind

12.2.3 Sequence of primitives

The time sequence diagram of Figure 10 shows the TP-GRANT-CONTROL sequence of primitives.

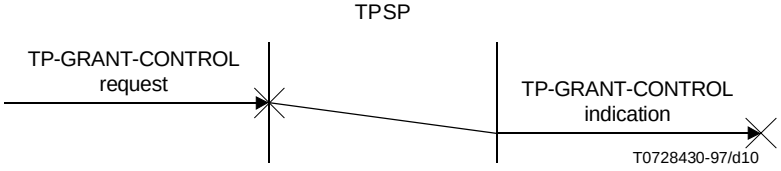

Figure 10/X.861 – TP-GRANT-CONTROL sequence of primitives

12.2.4 TPSP constraints on TP-GRANT-CONTROL request

The Polarized Control functional unit shall be selected for the dialogue.

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall have control of the dialogue.

The requestor shall not have a *handshake request outstanding*.

If the *dialogue is coordinated*, a TP-PREPARE request or a TP-EARLY-EXIT indication shall not have been issued during the current transaction.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The requestor shall not have a *dialogue termination request outstanding*.

12.2.5 Effects of a TP-GRANT-CONTROL request

The requestor no longer has control of the dialogue.

12.2.6 TPSP constraints on TP-GRANT-CONTROL indication

The Polarized Control functional unit shall be selected for the dialogue.

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall not have control of the dialogue.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The recipient shall not have a *dialogue termination indication outstanding*.

12.2.7 Effects of a TP-GRANT-CONTROL indication

The recipient has control of the dialogue.

12.2.8 Collisions

A TP-GRANT-CONTROL indication is not issued for a *coordinated dialogue* if a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued during the current transaction.

12.3 Request Control service – TP-REQUEST-CONTROL

12.3.1 Purpose

This service is used by a TPSUI to request control of the dialogue. However, control is not actually transferred by this service and the TPSUI to which the indication is issued is not obliged to relinquish control.

This service is associated with one particular dialogue.

12.3.2 Primitives and parameters

Table 11 lists the TP-REQUEST-CONTROL primitives.

Table 11/X.861 – TP-REQUEST-CONTROL primitives

TP-REQUEST-CONTROL		
No parameters	req	ind

12.3.3 Sequence of primitives

The time sequence diagram of Figure 11 shows the TP-REQUEST-CONTROL sequence of primitives.

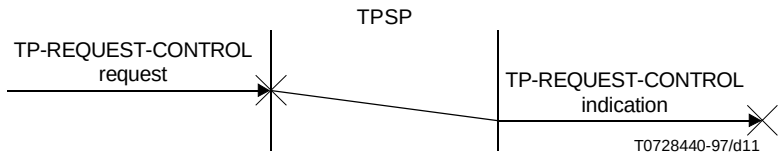

Figure 11/X.861 – TP-REQUEST-CONTROL sequence of primitives

12.3.4 TPSP constraints on TP-REQUEST-CONTROL request

The Polarized Control functional unit shall be selected for the dialogue.

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall not have control of the dialogue.

The requestor shall not have a TP-HANDSHAKE-AND-GRANT-CONTROL indication outstanding.

The requestor shall not have a *user error request outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The requestor shall not have a *dialogue termination indication outstanding*.

12.3.5 TPSP constraints on TP-REQUEST-CONTROL indication

The Polarized Control functional unit shall be selected for the dialogue.

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall have control of the dialogue.

The recipient shall not have a *user error indication outstanding*.

If the *dialogue is coordinated*, a TP-PREPARE request or a TP-EARLY-EXIT indication shall not have been issued during the current transaction.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The recipient shall not have a *dialogue termination request outstanding*.

12.3.6 Collisions

A TP-REQUEST-CONTROL indication is not issued if one of the following service primitives has been issued:

– TP-GRANT-CONTROL request;

– TP-HANDSHAKE-AND-GRANT-CONTROL request; or

– TP-END-DIALOGUE request with the Confirmation parameter set to "true".

A TP-REQUEST-CONTROL indication is not issued for a *coordinated dialogue* if one of the following service primitives has been issued:

– TP-COMMIT request;

– TP-PREPARE request;

– TP-READ-ONLY request;

– TP-EARLY-EXIT request;

– TP-ONE-PHASE request; or

– any *rollback-initiating service primitive*.

13 The Handshake functional unit

13.1 Overview of the Handshake functional unit

The Handshake functional unit allows partner TPSUIs to synchronize their processing with one another, and possibly transfer control.

13.2 Handshake service – TP-HANDSHAKE

13.2.1 Purpose

This service is used by partner TPSUIs to synchronize their processing with one another.

This service is associated with one particular dialogue.

13.2.2 Primitives and parameters

Table 12 lists the TP-HANDSHAKE primitives and their parameter.

Table 12/X.861 – TP-HANDSHAKE primitives and their parameter

TP-HANDSHAKE				
Parameter	req	ind	rsp	cnf
Confirmation-Urgency	C			

Confirmation-Urgency applies only if the Shared Control functional unit has been selected. It is provided by the requestor to specify the urgency with which the confirmation is required. This parameter shall take one of the following values:

a) "urgent" when the TPSUI requests minimal delay in receiving the confirm primitive;

b) "normal" when the TPSUI has no particular delay requirement in receiving the confirm primitive. In this case, the communication flow may be optimized by the TPSP.

13.2.3 Sequence of primitives

The time sequence diagram of Figure 12 shows the TP-HANDSHAKE sequence of primitives.

In Figure 12 the TP-HANDSHAKE response and the TP-HANDSHAKE confirm may be replaced by a TP-U-ERROR request and a TP-U-ERROR indication, respectively, if synchronization is unsuccessful.

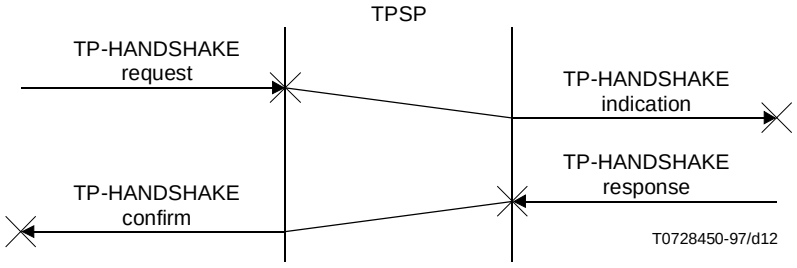

Figure 12/X.861 – TP-HANDSHAKE sequence of primitives

13.2.4 TPSP constraints on TP-HANDSHAKE request

The Handshake functional unit shall be selected for the dialogue.

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall have control of the dialogue.

The requestor shall not have a *handshake request outstanding*.

The requestor shall not have a *user error indication outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The requestor shall have neither a *dialogue termination request outstanding* nor a *dialogue termination indication outstanding*.

13.2.5 TPSP constraints on TP-HANDSHAKE indication

The Handshake functional unit shall be selected for the dialogue.

The recipient shall not have a *dialogue establishment request outstanding*.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue.

The recipient shall not have a *handshake indication outstanding*.

If the Shared Control functional unit is selected and the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The recipient shall not have a *dialogue termination indication outstanding*.

13.2.6 Effects of a TP-HANDSHAKE indication

If a *user error request was outstanding*, the recipient has control of the dialogue. In this case, no response is required and the synchronization is considered unsuccessful.

13.2.7 TPSUI conditions on TP-HANDSHAKE response

Synchronization is successful.

13.2.8 TPSP constraints on TP-HANDSHAKE response

The responder shall not have a *dialogue establishment indication outstanding*.

A TP-HANDSHAKE indication shall be outstanding.

13.2.9 TPSP constraints on TP-HANDSHAKE confirm

A TP-HANDSHAKE request shall be outstanding.

The recipient shall not have a *dialogue establishment request outstanding*.

13.2.10 Collisions

If the Shared Control functional unit is selected, a TP-HANDSHAKE indication is not issued to a TPSUI if there is a collision of the TP-HANDSHAKE request and a TP-U-ERROR request.

If the Shared Control functional unit is selected, a TP-HANDSHAKE indication is not issued for a *coordinated dialogue* after a *transaction completion request* or a TP-PREPARE request; either:

– a TP-ROLLBACK indication is issued unless a *rollback-initiating service primitive* has already been issued for the current transaction; the Diagnostic parameter will be set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected; or

– a TP-P-ABORT indication will be issued with the Diagnostic parameter set to "user-protocol-error" if the transaction cannot be rolled back.

A TP-HANDSHAKE indication or confirm is not issued for a *coordinated dialogue* if a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued during the current transaction.

NOTE – If the Shared Control functional unit is selected, then a TP-HANDSHAKE request colliding with a TP-HANDSHAKE indication from a remote partner is not regarded as a collision, but as two separate uses of the service. Either may be successful or fail independently of the other.

13.3 Handshake and Grant Control service – TP-HANDSHAKE-AND-GRANT-CONTROL

13.3.1 Purpose

This service is used by partner TPSUIs to synchronize their processing with one another and to transfer control.

This service is associated with one particular dialogue.

13.3.2 Primitives and parameters

Table 13 lists the TP-HANDSHAKE-AND-GRANT-CONTROL primitives and their parameter.

Table 13/X.861 – TP-HANDSHAKE-AND-GRANT-CONTROL primitives and their parameter

TP-HANDSHAKE-AND-GRANT-CONTROL				
Parameter	req	ind	rsp	cnf
Confirmation-Urgency	M			

Confirmation-Urgency is a parameter which is provided by the requestor to specify the urgency with which the confirmation is required. This parameter shall take one of the following values:

a) "urgent" when the TPSUI requests minimal delay in receiving the confirm primitive;

b) "normal" when the TPSUI has no particular delay requirement in receiving the confirm primitive. In this case, the communication flow may be optimized by the TPSP.

13.3.3 Sequence of primitives

The time sequence diagram of Figure 13 shows the TP-HANDSHAKE-AND-GRANT-CONTROL sequence of primitives.

In Figure 13, the TP-HANDSHAKE-AND-GRANT-CONTROL response and the TP-HANDSHAKE-AND-GRANT-CONTROL confirm may be replaced by a TP-U-ERROR request and a TP-U-ERROR indication, respectively, if synchronization is unsuccessful.

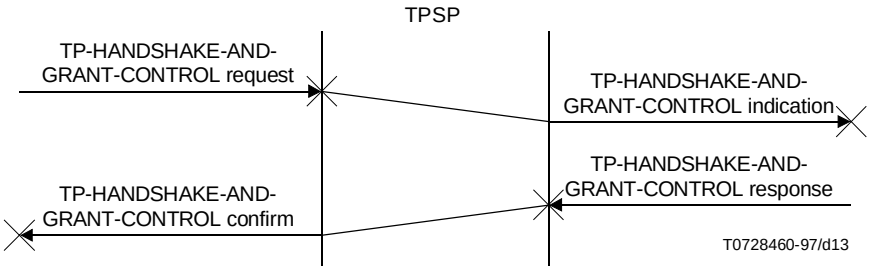

Figure 13/X.861 – TP-HANDSHAKE-AND-GRANT-CONTROL sequence of primitives

13.3.4 TPSP constraints on TP-HANDSHAKE-AND-GRANT-CONTROL request

The Handshake and Polarized Control functional units shall both be selected for the dialogue.

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall have control of the dialogue.

The requestor shall not have a *handshake request outstanding*.

The requestor shall not have a *user error indication outstanding*.

If the *dialogue is coordinated*, a TP-PREPARE request or a TP-EARLY-EXIT indication request shall not have been issued during the current transaction.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The requestor shall not have a *dialogue termination request outstanding*.

13.3.5 Effects of a TP-HANDSHAKE-AND-GRANT-CONTROL request

The requestor no longer has control of the dialogue.

13.3.6 TPSP constraints on TP-HANDSHAKE-AND-GRANT-CONTROL indication

The Handshake and Polarized Control functional units shall both be selected for the dialogue.

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall not have control of the dialogue.

The recipient shall not have a *handshake indication outstanding*.

If the *dialogue is coordinated*, none of the following services shall have been issued during the current transaction:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

If the *dialogue is coordinated*, the current transaction shall not be in the *termination phase*.

The recipient shall not have a *dialogue termination indication outstanding*.

13.3.7 Effects of a TP-HANDSHAKE-AND-GRANT-CONTROL indication

The recipient has control of the dialogue.

13.3.8 TPSUI conditions on TP-HANDSHAKE-AND-GRANT-CONTROL response

Synchronization is successful.

13.3.9 TPSP constraints on TP-HANDSHAKE-AND-GRANT-CONTROL response

The responder shall not have a *dialogue establishment indication outstanding*.

A TP-HANDSHAKE-AND-GRANT-CONTROL indication shall be outstanding.

13.3.10 TPSP constraints on TP-HANDSHAKE-AND-GRANT-CONTROL confirm

A TP-HANDSHAKE-AND-GRANT-CONTROL request shall be outstanding.

The recipient shall not have a *dialogue establishment request outstanding*.

13.3.11 Collisions

A TP-HANDSHAKE-AND-GRANT-CONTROL indication or a TP-HANDSHAKE-AND-GRANT-CONTROL confirm is not issued for a *coordinated dialogue* if a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued during the current transaction.

14 The commitment-related functional units

14.1 Introduction

The commitment-related functional units are:

– the Commit functional unit;

– the Dynamic Commit functional unit;

– the Unchecked Tree functional unit;

– the Read-only functional unit;

– the Early-exit functional unit;

– the Implicit Prepare functional unit;

– the One-phase Commit functional unit;

– the Completion Diagnostics functional unit;

– the Heuristic Containment Required functional unit;

– the Chained Transactions functional unit; and

– the Unchained Transactions functional unit.

NOTE – The allowed combinations of functional units are defined in 7.1.

The Commit functional unit or the One-phase Commit functional unit shall be selected on a dialogue if the TPSUI requires transaction branches to be supported on the dialogue.

If the Chained Transactions functional unit is selected, the dialogue will always support transaction branches, thus its coordination level is always "commitment" or "one-phase commitment". During the lifetime of the dialogue, the superior and the subordinate TPSUIs will be in the same transaction tree.

If the Unchained Transactions functional unit is selected, the dialogue may support transaction branches. The superior determines when the coordination level of the dialogue is "commitment" or "one-phase commitment". At a given point in time, the two TPSUIs may be participants in the same transaction, in different transactions, or one or both TPSUIs may not be involved in a transaction.

14.2 Summary

14.2.1 Overview of the Commit functional unit

14.2.1.1 Introduction

The Commit functional unit shall be selected if the TPSUI wants the TPSP to coordinate the maintenance of the ACID properties. If the Dynamic Commit functional unit is not selected, then the static commitment procedures apply as described below.

The static commitment procedures are based on two-phase commitment. In the first phase, all TPSUIs in the transaction tree are brought to a state called READY, in which all processing and data transfer for the current transaction are complete, and all bound data for the transaction are in a state called ready-to-commit from where they may be placed in either the initial or the final state. In the second phase, the transaction is either committed or rolled back.

Any *rollback-initiating service primitive* may be used to trigger a rollback of the entire transaction tree before completion of the first phase.

A heuristic decision may be taken for any bound data that are in the ready-to-commit state.

The TPSP and/or the TPSUI may handle bound data; the Commit functional unit enables the TPSP and/or the TPSUI to manage the effects of commitment or rollback on their bound data.

14.2.1.2 Phase 1 of Static Commitment

Phase 1 of Static Commitment uses the following service primitives:

– TP-PREPARE request which allows a superior TPSUI to request a *subordinate transaction subtree* to enter the READY state. When the *subordinate subtree* is ready-to-commit, it is indicated to the superior TPSUI by a TP-READY indication (unless the superior TPSUI has issued a TP-COMMIT request). The use of this service is optional.

– TP-PREPARE indication is used to indicate that the superior is requesting its *subordinate transaction subtree* to complete processing for the current transaction and place its bound data in the ready-to-commit state. This indication is issued when either a TP-PREPARE request or a TP-COMMIT request has been issued by the superior.

– TP-COMMIT request is used to indicate willingness to end the transaction, to indicate that the bound data handled by the TPSUI are in the ready-to-commit state, and to request all the subordinate nodes to enter the READY state. After this point, the TPSUI is no longer allowed to initiate rollback for this transaction and shall wait for a TP-COMMIT indication or a *rollback-initiating indication*.

TP-PREPARE request and TP-COMMIT request support two approaches to initiating commitment.

– If the TPSUI issues TP-COMMIT request, then it can no longer participate in the organization of the two-phase commit procedure.

– By issuing the optional TP-PREPARE request to a subordinate, the TPSUI may achieve the following:

  1) the *subordinate subtree* is requested to proceed with the first phase of the two-phase commit procedure while the TPSUI is still in the active phase of the transaction;

  2) by waiting for the corresponding TP-READY indication from the particular subordinate, the TPSUI may determine that the subordinate is ready-to-commit before it proceeds with the remainder of the commitment process;

  3) a subordinate is notified that no more messages will be sent to it, while the subordinate may still be allowed to send messages to its superior before it enters the commitment procedure.

14.2.1.3 Phase 2 of Static Commitment

Phase 2 of Static Commitment uses the following service primitives:

– TP-COMMIT indication is issued to inform a TPSUI that the transaction is in the process of being committed and to request the TPSUI to release its bound data in the final state;

– TP-DONE request is issued when the TPSUI has released its bound data in the final state;

– TP-COMMIT-COMPLETE indication is issued to a TPSUI to indicate that the prior transaction has been committed. If any dialogue exists with the Chained Transactions functional unit selected, the indication also indicates that the TPSUI is involved in the subsequent transaction.

14.2.1.4 Rollback

Rollback uses the following service primitives:

– a *rollback-initiating request* is issued by a TPSUI to request that the transaction be rolled back. More than one TPSUI in the tree may issue a *rollback-initiating request* for a given transaction;

– a rollback may also be initiated by the TPSP when it detects a condition which prevents the transaction from committing;

– a *rollback-initiating indication* is issued to inform a TPSUI that the transaction is in the process of being rolled back and to request the TPSUI to release its bound data in the initial state;

– TP-DONE request is issued when the TPSUI has released its bound data in the initial state;

– TP-ROLLBACK-COMPLETE indication is issued to a TPSUI to indicate that the prior transaction has been rolled back. If any dialogue exists with the Chained Transactions functional unit selected, the indication also indicates that the TPSUI is involved in the subsequent transaction.

14.2.1.5 Heuristic decision reporting

Heuristic decision reporting uses the following service primitives:

– TP-DONE request, with a Heuristic-Report parameter, is issued when the TPSUI has released its bound data in a state inconsistent with the final outcome of the transaction, or when a failure may prevent reporting of bound data inconsistency;

– TP-HEURISTIC-REPORT indication is issued to indicate to a TPSUI that a *subordinate subtree* has either released its bound data in an inconsistent state, or that a failure may prevent reporting of bound data inconsistency.

14.2.2 Overview of the Dynamic Commit functional unit

14.2.2.1 Introduction

If the Dynamic Commit functional unit is selected, then the dynamic commit procedures apply.

The dynamic commitment procedures are based on two-phase commitment with a number of the constraints imposed by the static commitment procedures removed.

A number of controls are provided which can influence how commitment can be initiated and how transaction trees can be constructed; these controls are either lifetime properties of dialogues set via the selection of functional units or set via the Superior-may-send-ready, Subordinate-may-send-ready, or Check-ready-directions parameters on the TP-BEGIN-DIALOGUE service, or in the case of the Check-ready-directions parameter, may be set for each use of the TP-BEGIN-TRANSACTION service.

14.2.2.2 Phase 1 of Dynamic Commitment

Phase 1 of Dynamic Commitment uses the following service primitives:

– TP-PREPARE request which allows a TPSUI to inform a neighbour that no more data will be sent to that neighbour as part of the transaction. The use of this service is optional;

– TP-PREPARE indication is used to indicate that the neighbour from whom it is received will send no more data to the recipient as part of the transaction. This indication may be issued when the neighbour has issued either TP-PREPARE request or a *transaction completion request*;

– TP-COMMIT request is used by a TPSUI to indicate willingness to end the transaction. This will occur when the TPSUI knows that no more data for the transaction will be sent or received from any neighbour, and that the bound data handled by the TPSUI are in the ready-to-commit state. After issuing TP-COMMIT request, the TPSUI is no longer allowed to initiate rollback for this transaction and shall wait for a TP-COMMIT indication or a *rollback-initiating indication*.

14.2.2.3 Phase 2 of Dynamic Commitment

Phase 2 of Dynamic Commitment uses the identical service primitives to phase 2 of Static Commitment.

14.2.3 Overview of the Unchecked Tree functional unit

The Unchecked Tree functional unit allows the suppression of TPSP checks on the construction of transaction trees when the Dynamic Commit functional unit is selected.

14.2.4 Overview of the Read-only functional unit

If the Read-only functional unit is selected on the *superior dialogue*, a TPSUI may request that it withdraw from participation in the transaction if it has completed all processing for the transaction and has not modified its bound data. In this case, bound data are by definition in their initial state and may be released by the TPSUI.

In the simplest case, the TPSUI will take no further part in the completion of the transaction and will not learn whether the transaction proceeded to commitment or rollback.

However while the TPSUI may not have modified bound data, the TPSP may have modified bound data; thus the TPSUI may be read-only but the node as a whole (including the TPSP) may not. In addition, if there is a *subordinate subtree*, bound data in the subtree may have been modified. In either case, the TPSUI will be informed of the result of the transaction and may need to act on such knowledge if, for example, completion reports are received from subordinates.

The TPSUI may also be informed of the outcome of the transaction if the outcome will affect the state of the TPSUI or any of its dialogues.

NOTE 1 – The execution of any deferred requests or indications depends on the outcome of the transaction.

The Read-only functional unit uses the following services:

– TP-READ-ONLY request is used by a TPSUI to indicate that the bound data handled by the TPSUI has not been modified during the current transaction and that the TPSUI has no preference as to whether the transaction should commit or rollback;

– TP-READ-ONLY indication is issued to a TPSUI which has not entered the *termination phase of a transaction* to indicate that a *subordinate subtree* is in the READ-ONLY state.

The outcome of the transaction is signalled to a TPSUI which issued a TP-READ-ONLY request by one of the following indications:

– TP-UNKNOWN indication when the outcome of the transaction is unknown at this node;

– TP-COMMIT indication if the outcome is known to be commitment;

– TP-ROLLBACK indication if the outcome is known to be rollback.

A TPSUI which has issued a TP-READ-ONLY request is aware of the termination of the transaction when one of the following service primitives is issued:

– TP-COMMIT-COMPLETE indication when the outcome of the transaction is known to be commitment;

– TP-ROLLBACK-COMPLETE indication when the outcome of the transaction is known to be rollback;

– TP-UNKNOWN-COMPLETE indication when the outcome of the transaction is unknown.

If any dialogue exists with the Chained Transactions functional unit selected, the indications above also indicate that the TPSUI is involved in the subsequent transaction.

By omitting the Read-only functional unit on the dialogue initiation request, the dialogue superior can dictate that a subordinate shall not release bound data before the termination of the transaction is confirmed.

NOTE 2 – Use of the TP-READ-ONLY service may result in a problem with the serializability of transactions. This may occur if the bound data is liable to be changed by any other activity and then accessed by another part of the transaction. TP-READ-ONLY should only be used if serializability can be guaranteed.

14.2.5 Overview of the Early-exit functional unit

The Early-exit functional unit allows a TPSUI to indicate that it is unable to contribute to the work of a transaction, its bound data has not been modified, and that the TPSUI has no preference as to whether the transaction commits or is rolled back.

The Early-exit functional unit uses the following services:

– TP-EARLY-EXIT request is used by a TPSUI to indicate that the TPSUI cannot contribute to the results of the transaction, bound data handled by the TPSUI has not been modified during the current transaction and that the TPSUI has no preference as to whether the transaction should commit or rollback;

– TP-EARLY-EXIT indication is issued to a TPSUI which has not entered the *termination phase of a transaction* to indicate that a *subordinate subtree* is in the EARLY-EXIT state.

NOTE – As regards the *superior dialogue*, there are few constraints on the issuing of a TP-EARLY-EXIT request; in particular it may be issued whether or not the TPSUI has control of the *superior dialogue* and whether or not there is a *handshake request or indication outstanding* or a *user error request or indication outstanding* on the *superior dialogue*.

The outcome of the transaction is signalled to a TPSUI which issued a TP-EARLY-EXIT request by one of the following indications:

– TP-UNKNOWN indication when the outcome of the transaction is unknown at this node;

– TP-ROLLBACK indication if the outcome is known to be rollback.

A TPSUI which has issued a TP-EARLY-EXIT request is aware of the termination of the transaction when one of the following service primitives is issued:

– TP-ROLLBACK-COMPLETE indication when the outcome of the transaction is known to be rollback;

– TP-UNKNOWN-COMPLETE indication when the outcome of the transaction is unknown.

If any dialogue exists with the Chained Transactions functional unit selected, the indications above also indicate that the TPSUI is involved in the subsequent transaction.

14.2.6 Overview of the Implicit Prepare functional unit

The Implicit Prepare functional unit shall only be selected when either or both of the Commit or One-phase Commit functional units are also selected.

The Implicit Prepare functional unit indicates that a *subordinate subtree* may proceed to enter the READY or READ-ONLY or ONE-PHASE state without having received a TP-PREPARE indication or a TP-ONE-PHASE indication on the *superior dialogue*.

NOTE – Typically the TPSUI will know that no more data will be sent to it and that the transaction is to be terminated because of equivalent semantics to TP-PREPARE indication being implied by the U-ASE data received.

14.2.7 Overview of the One-phase Commit functional unit

The One-phase Commit functional unit allows a node that did not modify any bound data to request a one-phase commit procedure.

Both static and dynamic one-phase commit procedures are supported.

If either the static and dynamic one-phase commit procedures are selected, then the TPSUI may not be informed of the outcome of the transaction if a failure occurs during commitment.

14.2.7.1 Static One-phase Commitment

The static one-phase commit procedures apply if the One-phase Commit functional unit is selected without the Commit functional unit.

NOTE – If there is *tree checking* at the root node and a static one-phase branch is established, then there is no other branch allowed at the node. This static one-phase branch is a *static one-phase exclusive branch* and every other branch will be an *exclusive branch* or a *two-phase expected branch*.

Static One-phase Commit uses the following services:

– TP-PREPARE request which allows a TPSUI to inform a neighbour that no more data will be sent to that neighbour as part of the transaction. The use of this service is optional;

– TP-PREPARE indication is used to indicate that the neighbour from whom it is received will send no more data to the recipient as part of the transaction. This indication may be issued to a subordinate when the superior has issued TP-PREPARE request. This indication may be issued to a superior when the subordinate has issued either TP-PREPARE request or a *transaction completion request*;

– TP-ONE-PHASE request is used by a TPSUI to indicate that the bound data handled by the TPSUI has not been modified during the current transaction, and that the TPSUI has no requirement for reliable reporting of the outcome; the TPSUI may only have one subordinate in the transaction tree on which neither a TP-READ-ONLY indication nor a TP-EARLY-EXIT indication has been received;

– TP-ONE-PHASE indication is issued to the single subordinate TPSUI which has not entered the *termination phase of a transaction* to indicate that the superior has requested one-phase commit.

The outcome of the transaction is signalled to a TPSUI which issued a TP-ONE-PHASE request by one of the following indications:

– TP-UNKNOWN indication when, due to some failure, the outcome of the transaction is unknown at this node;

– TP-COMMIT indication if the outcome is known to be commitment;

– TP-ROLLBACK indication if the outcome is known to be rollback.

A TPSUI which has issued a TP-ONE-PHASE request is aware of the termination of the transaction when one of the following service primitives is issued:

– TP-COMMIT-COMPLETE indication when the outcome of the transaction is known to be commitment;

– TP-ROLLBACK-COMPLETE indication when the outcome of the transaction is known to be rollback;

– TP-UNKNOWN-COMPLETE indication when, due to some failure, the outcome of the transaction is unknown.

If any dialogue exists with the Chained Transactions functional unit selected, the indications above also indicate that the TPSUI is involved in the subsequent transaction.

14.2.7.2 Dynamic One-phase Commitment

If the Commit, Dynamic Commit and One-phase Commit functional units are all selected, then the TPSUI may choose between full commitment and one-phase commitment dynamically according to the nature of the transaction.

Dynamic One-phase Commit uses the following services:

– TP-PREPARE request which allows a TPSUI to inform a neighbour that no more data will be sent to that neighbour as part of the transaction. The use of this service is optional;

– TP-PREPARE indication is used to indicate that the neighbour from whom it is received will send no more data to the recipient as part of the transaction. This indication may be issued when the neighbour has issued either TP-PREPARE request or a *transaction completion request*;

– TP-ONE-PHASE request is used by a TPSUI to indicate that the bound data handled by the TPSUI has not been modified during the current transaction, and that the TPSUI has no requirement for reliable reporting of the outcome;

– TP-ONE-PHASE indication is issued to a neighbouring TPSUI which has not entered the *termination phase of a transaction* to indicate that all nodes in the *transaction hinterland* of the neighbour have signalled read-only or one-phase or early-exit, and that the one-phase commit procedure are being used on the dialogue with the neighbour.

The outcome of the transaction is signalled to a TPSUI which issued a TP-ONE-PHASE request by one of the following indications:

– TP-UNKNOWN indication when, due to some failure, the outcome of the transaction is unknown at this node;

– TP-COMMIT indication if the outcome is known to be commitment;

– TP-ROLLBACK indication if the outcome is known to be rollback.

A TPSUI which has issued a TP-ONE-PHASE request is aware of the termination of the transaction when one of the following service primitives is issued:

– TP-COMMIT-COMPLETE indication when the outcome of the transaction is known to be commitment.

NOTE – This includes when the transaction is entirely read-only and/or one-phase.

– TP-ROLLBACK-COMPLETE indication when the outcome of the transaction is known to be rollback;

– TP-UNKNOWN-COMPLETE indication when, due to some failure, the outcome of the transaction is unknown.

If any dialogue exists with the Chained Transactions functional unit selected, the indications above also indicate that the TPSUI is involved in the subsequent transaction.

14.2.8 Overview of the Completion Diagnostics functional unit

The Completion Diagnostics functional unit allows a TPSUI to receive diagnostic information related to the completion of a transaction from subordinate nodes and/or to send similar information to its superior; diagnostics can be sent for a transaction which committed or for a transaction which rolled back.

For a transaction which rolled back, the information can indicate the severity of the rollback condition and whether a retry of the rolled back transaction might be worthwhile.

A TPSUI can provide information related to completion of a transaction via parameters to TP-DONE request. Subsequent TP-DONE requests which may become owed for the same transaction can carry the same or different values; a TPSUI may require to change the completion information as the result of further information received (e.g. such as completion reports from its own subordinates).

A TPSUI can be informed of the causes of rollback initiated by the local TPSP via parameters to TP-ROLLBACK indication.

A TPSUI can receive information related to the completion of a transaction from each subordinate via a TP-COMPLETION-REPORT indication, a TP-U-ABORT indication or a TP-P-ABORT indication. Completion reports may be lost in the event of failures.

14.2.9 Overview of the Heuristic Containment Required functional unit

The Heuristic Containment Required functional unit allows a TPSUI to require its subordinate to contain heuristic conditions; as a result the TPSUI will not receive heuristic reports from the subordinate.

14.3 Overview of the Chained Transactions functional unit

The Chained Transactions functional unit shall only be selected when the Commit functional unit or the One-phase Commit functional unit is selected.

The Chained Transactions functional unit and the Unchained Transactions functional unit are mutually exclusive for a dialogue.

When this functional unit is selected, the coordination level of the dialogue is always "commitment" or "one-phase commitment". There are no service primitives associated with this functional unit.

14.4 Overview of the Unchained Transactions functional unit

The Unchained Transactions functional unit shall only be selected when the Commit functional unit and/or the One-phase Commit functional unit is selected.

The Chained Transactions functional unit and the Unchained Transactions functional unit are mutually exclusive for a dialogue.

The superior determines when the coordination level of the dialogue is "commitment" or "one-phase commitment"; if the coordination level is "commitment" or "one-phase commitment" at the completion of a transaction, it changes to "none". At a given point in time, the two TPSUIs may be participants in the same transaction, in different transactions, or one or both TPSUIs may not be involved in a transaction.

A TPSUI may involve a subordinate in a transaction by means of the only service primitive in the Unchained Transactions functional unit: TP-BEGIN-TRANSACTION.

A TPSUI may also involve a subordinate in a transaction by means of the Begin-Transaction parameter of the TP-BEGIN-DIALOGUE service.

14.5 Begin Transaction service – TP-BEGIN-TRANSACTION

14.5.1 Purpose

This service is used by a TPSUI to include a subordinate TPSUI as a participant in the requestor's current transaction. If the requestor is not already a participant in a transaction, then a new transaction is initiated.

This service is associated with one particular dialogue.

14.5.2 Primitives and parameters

Table 14 lists the TP-BEGIN-TRANSACTION primitives and their parameter.

Table 14/X.861 – TP-BEGIN-TRANSACTION primitives and their parameter

TP-BEGIN-TRANSACTION		
Parameter	req	ind
Check-ready-directions	C	C(=)

Check-ready-directions is mandatory when the Dynamic Commit and Unchecked Tree functional units are selected on the dialogue, and is absent otherwise. It shall take one of the following values:

a) "true", when the TPSP shall check that the settings in the *subordinate's subtree* of the Superior-may-send-ready and Subordinate-may-send-ready parameters provided at dialogue establishment cannot lead to a deadlocked transaction tree;

b) "false", when the TPSUIs will take responsibility for ensuring that created transaction trees will be viable.

14.5.3 Sequence of primitives

The time sequence diagram of Figure 14 shows the TP-BEGIN-TRANSACTION sequence of primitives.

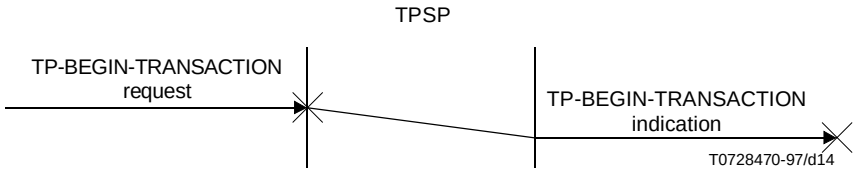

Figure 14/X.861 – TP-BEGIN-TRANSACTION sequence of primitives

14.5.4 TPSP constraints on TP-BEGIN-TRANSACTION request

The Unchained Transactions functional unit shall be selected.

The requestor shall be the superior.

The requestor shall have control of the dialogue.

The requestor shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The requestor shall not have a *user error indication outstanding*.

The coordination level of the dialogue shall be "none".

If there is a current transaction, it shall not be in the *termination phase*.

The requestor shall neither have a *dialogue termination request outstanding* nor a *dialogue termination indication outstanding*.

If the dialogue will have a coordination level of "one-phase commitment", then there shall be no superior transaction branch.

If this will be a *static one-phase exclusive branch*, then there shall not be a *two-phase expected branch*.

If this will be a *two-phase expected branch*, then there shall not be a *static one-phase exclusive branch*.

If this will be an *exclusive branch*, then there shall not be an existing *exclusive branch*.

If the Dynamic Commit functional unit is selected and *there is tree checking*, then the Check-ready-directions parameter shall be absent or set to "true".

14.5.5 Effects of a TP-BEGIN-TRANSACTION request

The coordination level of the dialogue is changed to:

a) "commitment" if the Commit functional unit is selected; or

b) "one-phase commitment" if the One-phase Commit functional unit is selected and the Commit functional unit is not selected.

The requestor becomes a participant in a new transaction if it is not already a participant in a transaction.

NOTE – If the subordinate is already involved in a transaction, the TP-BEGIN-TRANSACTION request may be rejected by a TP-P-ABORT indication with the Diagnostic parameter set to "begin-transaction-reject" or may be held until the subordinate is no longer involved in a transaction.

14.5.6 TPSP constraints on TP-BEGIN-TRANSACTION indication

The Unchained Transactions functional unit shall be selected.

The recipient shall be the subordinate.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue.

The recipient shall have neither a *handshake indication outstanding* nor a *dialogue termination indication outstanding*.

The coordination level of the dialogue shall be "none".

The recipient shall not be involved in a transaction.

14.5.7 Effects of a TP-BEGIN-TRANSACTION indication

The coordination level of the dialogue is changed to:

a) "commitment" if the Commit functional unit is selected; or

b) "one-phase commitment" if the One-phase Commit functional unit is selected and the Commit functional unit is not selected.

If the recipient has a *dialogue establishment indication outstanding*, the recipient becomes a participant in the same transaction as the requestor by issuing a TP-BEGIN-DIALOGUE response with the Result parameter set to "accepted".

If the recipient does not have a *dialogue establishment indication outstanding*, and can still reject the dialogue, the recipient becomes a participant in the same transaction as the requestor by manipulating bound data, or issuing any service primitives other than a TP-BEGIN-DIALOGUE response with the Result parameter set to "rejected(user)", otherwise the recipient becomes a participant in the same transaction as the requestor immediately.

14.5.8 Collisions

A TP-BEGIN-TRANSACTION indication is not issued to a TPSUI which has issued TP-END-DIALOGUE request with the Confirmation parameter set to "true"; instead a TP-P-ABORT indication is issued with the Diagnostic parameter set to "begin-transaction-end-dialogue-collision".

14.6 Deferred End Dialogue service – TP-DEFERRED-END-DIALOGUE

14.6.1 Purpose

This service is used by a TPSUI to end a dialogue with a subordinate TPSUI if the current transaction is committed, i.e. the termination of the specified dialogue takes place with the completion of commitment of the current transaction. In certain cases the dialogue may end if the transaction outcome is unknown, i.e. a TP-UNKNOWN-COMPLETE indication is issued.

This service is associated with one particular dialogue.

14.6.2 Primitives and parameters

Table 15 lists the TP-DEFERRED-END-DIALOGUE primitives.

Table 15/X.861 – TP-DEFERRED-END-DIALOGUE primitives

TP-DEFERRED-END-DIALOGUE		
No parameters	req	ind

14.6.3 Sequence of primitives

The time sequence diagram of Figure 15 shows the TP-DEFERRED-END-DIALOGUE sequence of primitives.

The corresponding TP-DEFERRED-END-DIALOGUE indication may be delayed and issued to the specified subordinate after indications or confirms resulting from subsequent requests or responses. However, the TP-DEFERRED-END-DIALOGUE indication shall be issued before any TP-PREPARE indication or a TP-READY indication or TP-ONE-PHASE indication.

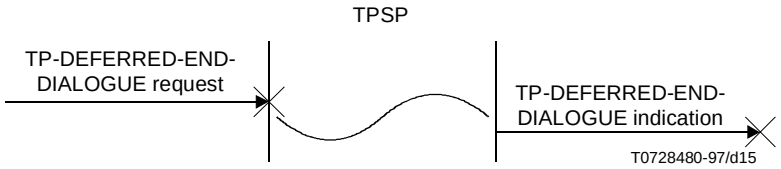

Figure 15/X.861 – TP-DEFERRED-END-DIALOGUE sequence of primitives

14.6.4 TPSP constraints on TP-DEFERRED-END-DIALOGUE request

The requestor shall be the superior.

The requestor shall have control of the dialogue.

The requestor shall neither have a *handshake request outstanding* nor a *handshake indication outstanding*.

The requestor shall not have a *user error indication outstanding*.

The *dialogue shall be coordinated*.

A TP-DEFERRED-END-DIALOGUE request shall not have been issued during the current transaction.

None of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The current transaction shall not be in the *termination phase*.

14.6.5 Effects of a TP-DEFERRED-END-DIALOGUE request

If no TP-EARLY-EXIT indication is issued for the dialogue, the dialogue terminates when either TP-COMMIT-COMPLETE indication or TP-UNKNOWN-COMPLETE indication is issued.

14.6.6 TPSP constraints on TP-DEFERRED-END-DIALOGUE indication

The recipient shall be the subordinate.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue.

The recipient shall not have a *handshake indication outstanding*.

The *dialogue shall be coordinated*.

A TP-DEFERRED-END-DIALOGUE indication shall not have been issued during the current transaction.

None of the following services shall have been issued during the current transaction:

– TP-PREPARE indication;

– TP-READY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The current transaction shall not be in the *termination phase*.

14.6.7 Effects of a TP-DEFERRED-END-DIALOGUE indication

If no TP-EARLY-EXIT request is issued, the dialogue terminates when either TP-COMMIT-COMPLETE indication or TP-UNKNOWN-COMPLETE indication is issued.

14.6.8 Collisions

A TP-DEFERRED-END-DIALOGUE indication is not issued to a TPSUI which has issued a TP-EARLY-EXIT request or any *rollback-initiating service primitives*.

A TP-DEFERRED-END-DIALOGUE indication is not issued to a TPSUI which has issued a *transaction completion request*; either:

– the transaction will be rolled back; a TP-ROLLBACK indication is issued unless a *rollback-initiating service primitive* has already been issued for the current transaction; the Diagnostic parameter will be set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected; or

– a TP-P-ABORT indication will be issued with the Diagnostic parameter set to "user-protocol-error" if the transaction cannot be rolled back.

14.7 Deferred Grant Control service – TP-DEFERRED-GRANT-CONTROL

14.7.1 Purpose

This service is used by a TPSUI to grant control of the dialogue to the specified subordinate if the current transaction is committed, or in certain cases if the transaction outcome is unknown, i.e. a TP-UNKNOWN-COMPLETE indication is issued.

This service is associated with one particular dialogue.

14.7.2 Primitives and parameters

Table 16 lists the TP-DEFERRED-GRANT-CONTROL primitives.

Table 16/X.861 – TP-DEFERRED-GRANT-CONTROL primitives

TP-DEFERRED-GRANT-CONTROL		
No parameters	req	ind

14.7.3 Sequence of primitives

The time sequence diagram of Figure 16 shows the TP-DEFERRED-GRANT-CONTROL sequence of primitives.

The corresponding TP-DEFERRED-GRANT-CONTROL indication may be delayed and issued to the specified subordinate after indications or confirms resulting from subsequent requests or responses. However, the TP-DEFERRED-GRANT-CONTROL indication shall be issued before any TP-PREPARE indication or TP-READY indication or TP-ONE-PHASE indication.

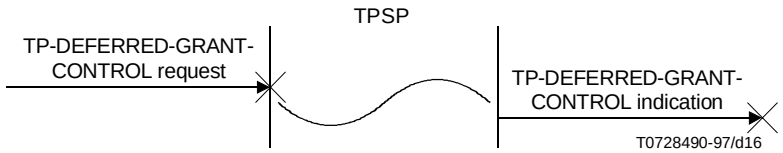

Figure 16/X.861 – TP-DEFERRED-GRANT-CONTROL sequence of primitives

14.7.4 TPSP constraints on TP-DEFERRED-GRANT-CONTROL request

The requestor shall be the superior.

The Polarized Control functional unit shall be selected.

The requestor shall have control of the dialogue.

The requestor shall not have a *handshake request outstanding*.

The requestor shall not have a *user error indication outstanding*.

The *dialogue shall be coordinated*.

A TP-DEFERRED-GRANT-CONTROL request shall not have been issued during the current transaction.

A TP-DEFERRED-END-DIALOGUE request shall not have been issued during the current transaction.

None of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The current transaction shall not be in the *termination phase*.

14.7.5 Effects of a TP-DEFERRED-GRANT-CONTROL request

If no TP-EARLY-EXIT indication is issued for the dialogue, the requestor will not have control of the specified dialogue when either TP-COMMIT-COMPLETE indication or TP-UNKNOWN-COMPLETE indication is issued.

14.7.6 TPSP constraints on TP-DEFERRED-GRANT-CONTROL indication

The recipient shall be the subordinate.

The Polarized Control functional unit shall be selected.

The recipient shall not have control of the dialogue.

The recipient shall not have a *handshake indication outstanding*.

The *dialogue shall be coordinated*.

A TP-DEFERRED-END-DIALOGUE indication shall not have been issued during the current transaction.

A TP-DEFERRED-GRANT-CONTROL indication shall not have been issued during the current transaction.

A TP-PREPARE indication or TP-READY indication or TP-ONE-PHASE indication shall not have been issued during the current transaction.

The current transaction shall not be in the *termination phase*.

14.7.7 Effects of a TP-DEFERRED-GRANT-CONTROL indication

If the current transaction is committed and no TP-EARLY-EXIT request is issued, the requestor will not have control of the specified dialogue when either TP-COMMIT-COMPLETE indication or TP-UNKNOWN-COMPLETE indication is issued.

14.7.8 Collisions

A TP-DEFERRED-GRANT-CONTROL indication is not issued to a TPSUI which has issued a TP-EARLY-EXIT request or any *rollback-initiating service primitive*.

14.8 TP-PREPARE request

14.8.1 Purpose

If the Commit functional unit is selected and the Dynamic Commit functional unit is not selected, this service primitive is issued by a superior TPSUI involved in a transaction to request a *subordinate subtree* to complete processing for the current transaction and place its bound data in the ready-to-commit state. In addition, the superior TPSUI issues this service to indicate that no more messages will be sent to the subordinate.

If either or both of the Dynamic Commit functional unit and One-phase Commit functional unit is selected, this service primitive is issued by a TPSUI involved in a transaction to indicate to the neighbour on the dialogue that no more messages will be sent to it by this TPSUI as part of the current transaction.

This service primitive is associated with one particular dialogue.

14.8.2 Primitives and parameters

Table 17 lists the TP-PREPARE request primitive and its parameter.

Table 17/X.861 –TP-PREPARE request and its PARAMETER

TP-PREPARE	
Parameter	req
Data-Permitted	C

Data-Permitted is a parameter that is present when the Polarized Control functional unit is selected and the requestor has control, and is absent otherwise. It shall take one of the following values:

a) "true" when the recipient is still allowed to issue TP-DATA requests to the requestor within the current transaction.

NOTE 1 – The TPSUI must take care that all such data has been received from the subordinate before requesting completion of the transaction.

b) "false" if no further TP-DATA requests are allowed to be issued towards the requestor.

NOTE 2 – With dynamic commitment, the Data-Permitted parameter can be supplied by either the superior or subordinate.

14.8.3 TPSP constraints on TP-PREPARE request

The requestor shall not have a *dialogue establishment indication outstanding*.

If neither the Dynamic Commit functional unit nor the One-phase Commit functional unit is selected, the requestor shall be the superior.

The requestor shall have control of the dialogue, or, either or both of the Dynamic Commit functional unit and One-phase Commit functional unit is selected and a TP-PREPARE indication has been received with the Data-Permitted parameter set to "true".

The requestor shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The requestor shall not have a *user error indication outstanding*.

The *dialogue shall be coordinated*.

If the requestor is the subordinate for a *coordinated dialogue* and the Implicit Prepare functional unit is not selected on the *superior dialogue*, then a TP-PREPARE indication or a TP-ONE-PHASE indication or a TP-READY indication shall have been issued on the *superior dialogue*.

None of the following services shall have been issued during the current transaction:

– TP-PREPARE request;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The current transaction shall not be in the *termination phase*.

14.8.4 Effects of issuing a TP-PREPARE request

A TP-PREPARE indication is issued to the specified neighbour unless the neighbour is in the *termination phase of the transaction*.

If the Data-Permitted parameter has the value "true" or the Shared Control functional unit is selected, then TP-DATA indications may be received, and if the Dynamic Commit functional unit is selected, a TP-PREPARE indication may be received.

14.8.5 Collisions

A TP-PREPARE indication is not issued to a TPSUI if a *transaction completion request* or a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued.

A TP-PREPARE indication is not issued to a TPSUI if there is a collision between the TP-PREPARE request and a TP-HANDSHAKE request or a TP-U-ERROR request; instead, a TP-ROLLBACK indication is issued; the Diagnostic parameter is set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

14.9 TP-PREPARE indication

14.9.1 Purpose

If the Commit functional unit is selected and the Dynamic Commit functional unit is not selected, this service primitive is issued to intermediate or leaf TPSUIs in a transaction tree to indicate that:

– all messages from the superior for the current transaction have been received;

– completion of the processing for the current transaction is requested;

– placement of the bound data handled by the TPSUI in the ready-to-commit state is requested.

If the Dynamic Commit functional unit is selected, this service primitive is issued to a TPSUI involved in a transaction to indicate that all messages from the neighbour on the dialogue for the current transaction have been received.

This service is associated with one particular dialogue.

14.9.2 Primitives and parameters

Table 18 lists the TP-PREPARE indication primitive and its parameter.

Table 18/X.861 – TP-PREPARE indication and its parameter

TP-PREPARE	
Parameter	ind
Data-Permitted	C

Data-Permitted is a parameter that is present when the Polarized Control functional unit is selected and the recipient does not have control. It shall take one of the following values:

a) "true" when the recipient is allowed to issue TP-DATA requests for the neighbouring TPSUI within the current transaction;

b) "false" if no further TP-DATA requests are allowed to be issued towards the neighbour.

This parameter is set to "true" only when the neighbouring TPSUI has issued a TP-PREPARE request with the Data-Permitted parameter set to "true".

14.9.3 TPSP constraints on TP-PREPARE indication

The recipient shall not have a *dialogue establishment request outstanding*.

If neither the Dynamic Commit functional unit nor the One-phase Commit functional unit is selected, the recipient shall be the subordinate.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue unless either or both of the Dynamic Commit functional unit and the One-phase Commit functional unit is selected and a TP-PREPARE request has been issued with the Data-Permitted parameter set to "true".

The recipient shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The recipient shall not have a *user error request outstanding*.

The *dialogue shall be coordinated*.

None of the following services shall have been issued for this dialogue during the current transaction:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The current transaction shall not be in the *termination phase*.

14.9.4 Effects of a TP-PREPARE indication

The TPSUI is aware that all messages from the neighbour for the current transaction have been received.

The TPSUI is requested by its neighbour to complete processing for the current transaction and place any bound data in the ready-to-commit state.

If the Polarized Control functional unit is selected and the Data-Permitted parameter is present and has the value "true", then TP-DATA requests may be issued, and if either or both of the Dynamic Commit functional unit or the One-phase Commit functional unit is selected, a TP-PREPARE request may be issued.

14.10 TP-READY indication

14.10.1 Purpose

This service primitive is issued to indicate that the *transaction hinterland* of the neighbouring node is in the READY state.

This service primitive is associated with one particular dialogue.

14.10.2 Primitives and parameters

Table 19 shows the TP-READY indication primitive.

Table 19/X.861 – TP-READY indication

TP-READY	
No parameter	ind

14.10.3 TPSP constraints on TP-READY indication

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall not have a *dialogue establishment indication outstanding*.

If the Dynamic Commit functional unit is not selected, the recipient shall be the superior.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue unless a TP-PREPARE request has been issued.

The recipient shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The recipient shall not have a *user error request outstanding*.

Ready can be received.

The coordination level of the dialogue shall be "commitment".

If the recipient is the *dialogue superior* and the Implicit Prepare functional unit is not selected, a TP-PREPARE request shall have been issued during the current transaction.

The current transaction shall not be in the *termination phase*.

All TPSUIs in the *transaction hinterland* of the neighbouring node shall have issued a *transaction completion request* or a TP-EARLY-EXIT request, and at least one node is in the READY state. This is a *transaction tree constraint*.

14.10.4 Effects of a TP-READY indication

The TPSUI is aware that processing of the current transaction in the *transaction hinterland* of the neighbour is complete and that bound data are in the ready-to-commit state.

14.11 TP-COMMIT request

14.11.1 Purpose

This service primitive is issued by a TPSUI to indicate that it has completed all processing for the current transaction and to request that the transaction be committed.

14.11.2 Primitives and parameters

Table 20 shows the TP-COMMIT request primitive.

Table 20/X.861 – TP-COMMIT request

TP-COMMIT	
No parameters	req

14.11.3 TPSUI conditions on a TP-COMMIT request

All bound data handled by the TPSUI shall have been placed in the ready-to-commit state.

All processing for the transaction by the TPSUI shall have been completed.

The release of the bound data handled by the TPSUI in the final state as part of the commitment processing shall preserve the ACID properties.

14.11.4 TPSP constraints on TP-COMMIT request

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall not have any *dialogue establishment requests outstanding* on any dialogues with a coordination level of "commitment".

The requestor shall have control of all *coordinated dialogues* on which none of the following services have been issued:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The requestor shall neither have any *handshake requests outstanding* nor *handshake indications outstanding* for any *coordinated dialogue*.

The requestor shall not have any *user error indication outstanding* on any *coordinated dialogue*.

There shall be no *subordinate dialogue* with a coordination level of "one-phase commitment".

One of the following conditions shall be true:

a) at least one dialogue shall have a coordination level of "commitment"; or

b) the node shall be a leaf node and the *superior dialogue* shall have a coordination level of "one-phase commitment"; or

NOTE – TP-ONE-PHASE request is forbidden if bound data exists, e.g. in case of the "server" or leaf; hence TP-COMMIT req/ind is required in the One-phase Commit functional unit even though the Commit functional unit is not selected.

c) a TP-BEGIN-DIALOGUE confirm, a TP-U-ABORT indication, or a TP-P-ABORT indication with the Rollback parameter set to "false" has been issued for at least one dialogue having a coordination level of "commitment" during the current transaction; or

d) a TP-READ-ONLY indication or a TP-EARLY-EXIT indication has been issued, for at least one dialogue having a coordination level of "commitment" during the current transaction.

If the requestor is the subordinate for a *coordinated dialogue* and the Implicit Prepare functional unit is not selected on the *superior dialogue*, then a TP-PREPARE indication or a TP-ONE-PHASE indication or a TP-READY indication shall have been issued on the *superior dialogue*.

The current transaction shall not be in the *termination phase*.

14.11.5 Effects of a TP-COMMIT request

A TP-PREPARE indication is issued to each direct subordinate TPSUI in the transaction tree if:

– no TP-PREPARE indication has already been issued to the subordinate; and

– the subordinate is not in the *termination phase of the transaction*; and

– the Implicit Prepare functional unit is not selected or the subordinate does not have control; and

– *ready can be received*.

If the Dynamic Commit functional unit is selected on the *superior dialogue*, a TP-PREPARE indication is issued to the superior TPSUI if:

– the *superior dialogue* has a coordination level of "commitment"; and

– a TP-PREPARE indication has not already been issued to the superior; and

– the superior is not in the *termination phase of the transaction*; and

– the superior does not have control; and

– *ready can be received*.

14.11.6 Collisions

A TP-PREPARE indication is not issued to a TPSUI if any *rollback-initiating service primitive* has been issued.

A TP-PREPARE indication is not issued to a TPSUI if a TP-EARLY-EXIT request has been issued; the transaction will be rolled back; a TP-ROLLBACK indication is issued; the Diagnostic parameter is set to "early-exit-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

A TP-PREPARE indication is not issued to a TPSUI if there is a collision of the TP-COMMIT request and a TP-HANDSHAKE request or a TP-U-ERROR request; instead, a TP-ROLLBACK indication is issued; the Diagnostic parameter is set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

After a TP-COMMIT request, none of the following service primitives will be issued for a *coordinated dialogue*:

– TP-REQUEST-CONTROL indication;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication; or

– TP-ONE-PHASE indication.

After a TP-COMMIT request, a TP-EARLY-EXIT indication is not issued; the transaction will be rolled back; a TP-ROLLBACK indication is issued; the Diagnostic parameter is set to "early-exit-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

14.12 TP-COMMIT indication

14.12.1 Purpose

This service primitive is issued to indicate that the outcome of the transaction is commitment and to order the TPSUI to release any bound data in the final state.

14.12.2 Primitives and parameters

Table 21 shows the TP-COMMIT indication primitive.

Table 21/X.861 – TP-COMMIT indication

TP-COMMIT	
No parameters	ind

14.12.3 TPSP constraints on TP-COMMIT indication

One of the following conditions shall be true:

a) at least one dialogue shall have a coordination level of "commitment"; or

b) the node shall be a leaf node and the *superior dialogue* shall have a coordination level of "one-phase commitment"; or

NOTE – TP-ONE-PHASE request is forbidden if bound data exists, e.g. in case of the "server" or leaf; hence the TP-COMMIT service is required in the One-phase Commit functional unit even though the Commit functional unit is not selected.

c) a TP-BEGIN-DIALOGUE confirm, a TP-U-ABORT indication, or a TP-P-ABORT indication with the Rollback parameter set to "false" has been issued for at least one dialogue having a coordination level of "commitment" during the current transaction; or

d) a TP-READ-ONLY indication or a TP-EARLY-EXIT indication has been issued, for at least one dialogue having a coordination level of "commitment" during the current transaction.

All bound data handled by the TPSP for the current transaction shall have been placed in the ready-to-commit state.

A *transaction completion request* shall have been issued.

All TPSUIs participating in the current transaction shall have issued a *transaction completion request* or a TP-EARLY-EXIT request. This is a *transaction tree constraint*.

No *rollback-initiating indication* shall have been issued or will be issued for TPSUIs participating in the current transaction. This is a *transaction tree constraint*.

14.12.4 Effects of a TP-COMMIT indication

The recipient is allowed to release its bound data in the final state.

The TPSUI owes a TP-DONE request.

In the case of a node crash, the TPSP may issue another TP-COMMIT indication on recovery.

14.13 TP-DONE request

14.13.1 Purpose

This service primitive is issued to indicate that any bound data handled by the TPSUI have been released and that any restructuring of the dialogue tree has been completed in reaction to rollback or abnormal dialogue termination.

If the Completion Diagnostics functional unit is selected on the *superior dialogue*, the TPSUI can supply information regarding the completion of the transaction; the information is reported to the superior unless the *superior dialogue* is aborted.

14.13.2 Primitives and parameters

Table 22 lists the TP-DONE request primitive and its parameters.

Table 22/X.861 – TP-DONE request and its parameters

TP-DONE	
Parameters	req
Heuristic-Report	U
Severity	C
Completion-Data	C

14.13.2.1 Heuristic-Report is an optional parameter which may be used to report any conditions related to heuristic decisions which may have been taken by the TPSUI. It shall take one of the following values:

a) "heuristic-mix": The bound data handled by the TPSUI are in a state which is inconsistent with the outcome of the transaction and the inconsistency cannot be corrected;

b) "heuristic-hazard": A failure occurred within the TPSUI which may prevent reporting of data inconsistency, and the TPSUI may not handle this situation.

This parameter shall only be present in the first TP-DONE request following a TP-COMMIT indication or *rollback-initiating service primitive;* it shall be absent if a TP-ONE-PHASE request or TP-READ-ONLY request or TP-EARLY-EXIT request was issued.

14.13.2.2 Severity is mandatory on the first TP-DONE request for a transaction when the TPSUI is a *transaction subordinate*, a TP-ROLLBACK request was issued, and the Completion Diagnostics functional unit is selected on the *superior dialogue*; the parameter is optional on the first TP-DONE request for a transaction when the TPSUI is a *transaction subordinate*, a *rollback-initiating service primitive* other than a TP-ROLLBACK request was issued, and the Completion Diagnostics functional unit is selected on the *superior dialogue*; on second and subsequent TP-DONE requests for a transaction, the parameter shall be present if and only if it was present on the first TP-DONE request for the transaction and a TP-COMPLETION-REPORT indication has been issued since the previous TP-DONE request; the parameter is absent otherwise. It shall take one of the following values:

a) permanent failure (general): All requests of this type will fail for the indefinite future;

b) permanent failure (specific): This request will fail but similar ones may work;

c) transient failure (general): Further requests of this type will probably work;

d) transient failure (specific): This request if attempted again now will probably work.

14.13.2.3 Completion-Data is optional on the first TP-DONE request for a transaction when the TPSUI is a *transaction subordinate*, the Completion Diagnostics functional unit is selected on the *superior dialogue* and one of the following conditions is true:

– a TP-COMMIT request was issued;

– a *rollback-initiating service primitive* was issued, and neither a TP-ONE-PHASE request nor a TP-READ-ONLY request was issued;

– a TP-ONE-PHASE request was issued, either a TP-COMMIT indication or a *rollback-initiating service primitive* was issued, and a TP-COMPLETION-REPORT was issued;

– a TP-READ-ONLY request was issued, either a TP-COMMIT indication or a *rollback-initiating service primitive* was issued, and a TP-COMPLETION-REPORT was issued.

On second and subsequent TP-DONE requests for a transaction, the parameter shall be present if and only if it was present on the first TP-DONE request for the transaction and a TP-COMPLETION-REPORT indication has been issued since the previous TP-DONE request; the parameter is absent otherwise. This parameter may be used to convey user-specific semantics associated with the completion of the transaction.

NOTE – The values of the Severity and Completion-Data parameters are not constrained by any previous values in a previous TP-DONE request.

14.13.3 TPSUI conditions on TP-DONE request

Before issuing the first TP-DONE request following a TP-COMMIT indication, all bound data handled by the requestor shall have been released in the final state, unless the Heuristic-Report parameter is used.

Before issuing the first TP-DONE request following a *rollback-initiating service primitive*, all bound data handled by the requestor shall have been released in the initial state, unless the Heuristic-Report parameter is used.

14.13.4 TPSP constraints on TP-DONE request

The requestor shall not have a *dialogue establishment indication outstanding*.

A TP-COMMIT indication, or a TP-UNKNOWN indication, or a *rollback-initiating service primitive* shall have been issued for the current transaction. At least one of the following conditions shall also be satisfied:

– A TP-DONE request has not been issued for the current transaction.

– A further TP-COMMIT indication or *rollback-initiating service indication* has been issued since the last TP-DONE request for the current transaction.

  NOTE – After a node crash, a repeated TP-COMMIT or TP-ROLLBACK indication can be received.

– A TP-P-ABORT indication, a TP-U-ABORT indication, a TP-COMPLETION-REPORT indication, or a TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" or "rejected(user)" has been issued for a *coordinated dialogue* during the *termination phase of the current transaction*, and a TP-DONE request has not been issued since the indication or the confirm.

If the Heuristic-Report parameter is used and the TPSUI is a subordinate for a dialogue with a coordination level of "commitment", a TP-PREPARE indication or a TP-READY indication or a TP-ONE-PHASE indication shall have been issued on the *superior dialogue* unless the Implicit Prepare functional unit is selected on the *superior dialogue*.

If the Heuristic-Report parameter is used, the request shall be the first TP-DONE request following a TP-COMMIT indication or a *rollback-initiating service primitive*.

14.13.5 Effects of a TP-DONE request

The TPSUI no longer owes a TP-DONE request.

If a TP-COMMIT indication was issued, a TP-COMMIT-COMPLETE indication is issued to the requestor after all TPSUIs whose heuristic reports will be reported to this node have issued a TP-DONE request.

If a TP-UNKNOWN indication was issued, a TP-UNKNOWN-COMPLETE indication is issued to the requestor.

If a *rollback-initiating service primitive* was issued, a TP-ROLLBACK-COMPLETE indication is issued to the requestor after all TPSUIs whose heuristic reports will be reported to this node have issued TP-DONE requests except any that are contained in a subtree of the transaction subtree whose root node has a *superior dialogue* that has been aborted.

The values of the Severity and Completion-Data parameters (if any) will be reported to the *transaction superior* if the dialogue still exists, unless there is a further TP-DONE request with these parameters when the values on the later TP-DONE request will be reported instead.

NOTE – A completion report will be delivered to the superior unless the *superior dialogue* or the TPSUI at the superior experiences a failure. If the TPSUI has not previously issued a TP-U-ABORT request on the *superior dialogue*, then a failure of the *superior dialogue* results in a TP-P-ABORT indication being issued to the TPSUI. However, if the TPSUI has issued a TP-U-ABORT request on the *superior dialogue*, then a failure of the *superior dialogue* will not result in a TP-P-ABORT indication being issued to the TPSUI.

14.14 TP-COMMIT-COMPLETE indication

14.14.1 Purpose

This primitive is issued by the TPSP to all TPSUIs which are engaged in a transaction to indicate that commitment (or one-phase commitment) is complete.

NOTE – In the case of one-phase commitment, a failure may prevent the actual result being determined for certain TPSUIs; in this case, a TP-UNKNOWN-COMPLETE indication would be issued.

14.14.2 Primitives and parameters

Table 23 shows the TP-COMMIT-COMPLETE indication primitive.

Table 23/X.861 – TP-COMMIT-COMPLETE indication

TP-COMMIT-COMPLETE	
No parameters	ind

14.14.3 TPSP constraints on TP-COMMIT-COMPLETE indication

The current transaction shall be in the *termination phase*.

All bound data of the node shall have been placed in the final state unless there is a local heuristic mix condition.

All bound data handled by *subordinate transaction subtrees* shall have been placed in the final state, with the possible exception of subordinates from which a TP-HEURISTIC-REPORT indication has been issued and subordinates whose heuristic reports will not be reported to this node. This is a *transaction tree constraint*.

A TP-COMMIT indication shall have been issued for the current transaction.

A TP-DONE request shall have been issued for the current transaction.

A TP-DONE request shall have been issued since the last TP-P-ABORT indication or TP-U-ABORT indication or TP-COMPLETION-REPORT indication issued for a *coordinated dialogue*.

14.14.4 Effects of a TP-COMMIT-COMPLETE indication

All dialogues for which a TP-DEFERRED-END-DIALOGUE request or a TP-DEFERRED-END-DIALOGUE indication was issued and a TP-EARLY-EXIT indication was not issued during the committed transaction are terminated.

The recipient has control of all *subordinate dialogues* except those for which it has issued a TP-DEFERRED-GRANT-CONTROL request and a TP-EARLY-EXIT indication was not issued during the committed transaction.

If the Polarized Control functional unit is selected on the *superior dialogue*, the recipient has control of the *superior dialogue* if a TP-DEFERRED-GRANT-CONTROL indication has been issued during the committed transaction; otherwise the recipient does not have control.

The coordination level of all dialogues on which the Unchained Transactions functional unit is selected is "none".

If the recipient has at least one *coordinated dialogue*, then the recipient is involved in a new transaction.

14.15 TP-ROLLBACK request

14.15.1 Purpose

This service primitive is issued by a TPSUI to terminate the transaction and release the bound data in the initial state.

14.15.2 Primitives and parameters

Table 24 shows the TP-ROLLBACK request primitive.

Table 24/X.861 – TP-ROLLBACK request

TP-ROLLBACK	
No parameters	req

14.15.3 TPSP constraints on TP-ROLLBACK request

The requestor shall not have a *dialogue establishment indication outstanding*.

One of the following conditions shall be true:

a) there shall be at least one *coordinated dialogue*; or

b) a TP-BEGIN-DIALOGUE confirm, a TP-U-ABORT indication, or a TP-P-ABORT indication with the Rollback parameter set to "false" has been issued for at least one *coordinated dialogue* during the current transaction; or

c) a TP-READ-ONLY indication or a TP-EARLY-EXIT indication has been issued for at least one dialogue during the current transaction.

The current transaction shall not be in the *termination phase*.

14.15.4 Effects of a TP-ROLLBACK request

The TPSUI owes a TP-DONE request.

A TP-ROLLBACK indication is issued to all TPSUIs involved in the current transaction, except in any of the circumstances following:

– a TPSUI which has issued any *rollback-initiating requests* or to which any *rollback-initiating indications* have been issued;

– a TPSUI which has issued a TP-READ-ONLY request or a TP-EARLY-EXIT request and a TP-UNKNOWN indication will be issued.

NOTE – A dialogue failure may mean that a TPSUI which issued a TP-ONE-PHASE request or TP-READ-ONLY request or TP-EARLY-EXIT request is no longer "involved in the current transaction"; in which case, a TP-UNKNOWN-COMPLETE indication will be issued.

14.15.5 Collisions

A TP-ROLLBACK indication is not issued if any *rollback-initiating service primitive* has been issued during the current transaction.

14.16 TP-ROLLBACK indication

14.16.1 Purpose

This service primitive is issued to indicate that the current transaction is being rolled back and to order release of bound data in the initial state.

The indication may carry information about a condition that has caused the transaction to be rolled back.

14.16.2 Primitives and parameters

Table 25 shows the TP-ROLLBACK indication primitive and its parameters.

Table 25/X.861 – TP-ROLLBACK indication and its parameters

TP-ROLLBACK	
Parameters	ind
Severity	C
Diagnostic	C

14.16.2.1 Severity is optional when the Diagnostic parameter is present, and is absent otherwise. This parameter is used to report the severity of the local condition that has caused the TPSP to invoke the rollback service. If present, it shall take one of the following values:

a) permanent failure (general): All requests of this type will fail for the indefinite future;

b) permanent failure (specific): This request will fail but similar ones may work;

c) transient failure (general): Further requests of this type will probably work;

d) transient failure (specific): This request if attempted again now will probably work.

14.16.2.2 Diagnostic is optional when the Completion Diagnostics functional unit is selected on any *coordinated dialogue*, and is absent otherwise. This parameter may be used by the TPSP to report the nature of the local condition that has caused the TPSP to invoke the rollback service. If present, it shall take one of the following values:

a) "user-data-transaction-completion-collision" when a user data request or a TP Service request has collided with a *transaction completion request*;

NOTE 1 – Such TP services could be one of TP-U-ERROR request, TP-HANDSHAKE request, TP-DEFERRED-END-DIALOGUE request.

b) "early-exit-transaction-completion-collision" when a TP-EARLY-EXIT request has collided with a *transaction completion request*;

c) "superior-rollback" when rollback has been initiated by the superior;

d) "subordinate-rollback" when rollback has been initiated by a subordinate; a later TP-COMPLETION-REPORT indication may provide further information;

e) "rollback-was-pending" when the transaction is being rolled back owing to an abort in the dialogue tree during commitment of the previous transaction; or

f) "local-rollback" when the local TPSP has invoked the rollback procedure and none of the above values is provided.

NOTE 2 – Diagnostics at one level are not propagated upwards by the TPSP but may be reflected in the Completion-Data parameter of TP-COMPLETION-REPORT indication.

NOTE 3 – A programming interface may return other diagnostic information which can be determined by the implementation; this is outside the scope of this Recommendation.

14.16.3 TPSP constraints on TP-ROLLBACK indication

At least one of the following conditions shall be true:

a) there is at least one *coordinated dialogue*; or

b) a TP-READ-ONLY indication or a TP-EARLY-EXIT indication has been issued for at least one dialogue during the current transaction; or

c) a TP-P-ABORT indication has been issued for a dialogue with the Chained Transactions functional unit selected, between a TP-COMMIT indication and the corresponding TP-COMMIT-COMPLETE indication and the deferred end dialogue service was not issued.

No *rollback-initiating service primitive* shall have been issued for the current transaction.

A TP-COMMIT indication or a TP-UNKNOWN indication shall not have been issued during the current transaction.

14.16.4 Effects of a TP-ROLLBACK indication

The recipient is allowed to release its bound data in the initial state.

The TPSUI owes a TP-DONE request.

In the case of a node crash, the TPSP may issue another TP-ROLLBACK indication on recovery.

14.17 TP-ROLLBACK-COMPLETE indication

14.17.1 Purpose

This service primitive is issued to all TPSUIs still involved in the transaction to indicate that rollback is complete.

NOTE – In the case of one-phase commitment, a failure may prevent the actual result being determined for certain TPSUIs; in this case, a TP-UNKNOWN-COMPLETE indication would be issued.

14.17.2 Primitives and parameters

Table 26 shows the TP-ROLLBACK-COMPLETE indication primitive.

Table 26/X.861 – TP-ROLLBACK-COMPLETE indication

TP-ROLLBACK-COMPLETE	
No parameters	ind

14.17.3 TPSP constraints on TP-ROLLBACK-COMPLETE indication

The recipient shall not have a *dialogue establishment request outstanding* for any *coordinated dialogue*.

The current transaction shall be in the *termination phase*.

A *rollback-initiating service primitive* shall have been issued for the current transaction.

All bound data of the node shall have been released in the initial state with the possible exception of a local heuristic mix condition.

All bound data handled by *subordinate transaction subtrees* shall have been placed in the initial state except for subordinates from which a TP-HEURISTIC-REPORT indication has been issued, and subordinates whose heuristic reports will not be reported to this node. This is a *transaction tree constraint*.

A TP-DONE request shall have been issued for the current transaction.

A TP-DONE request shall have been issued since the last TP-P-ABORT indication, TP-U-ABORT indication, TP-COMPLETION-REPORT indication, or TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" or "rejected(user)" issued for a *coordinated dialogue*.

14.17.4 Effects of a TP-ROLLBACK-COMPLETE indication

If the Polarized Control functional unit is selected for a *coordinated dialogue*, control of the dialogue is given to the TPSUI that had control at the beginning of the rolled back transaction.

NOTE – If the Unchained Transactions functional unit is selected for a dialogue, the superior has control.

The coordination level of any dialogue on which the Unchained Transactions functional unit is selected is "none".

If the recipient has at least one *coordinated dialogue*, then the recipient is involved in a new transaction.

14.18 Heuristic Reporting service – TP-HEURISTIC-REPORT indication

14.18.1 Purpose

This service indicates an actual or possible occurrence of a heuristic inconsistency within the *subordinate subtree*.

This service is associated with one particular dialogue which may have been previously terminated.

NOTE 1 – Heuristic reporting may not be reliable with coordination level of "one-phase commitment".

NOTE 2 – With the Implicit Prepare functional unit, heuristic decisions may be taken resulting from a prepare which is implicit in application semantics; the TPSP cannot detect such prepares and thus cannot warn of heuristic hazards when failures occur.

14.18.2 Primitives and parameters

Table 27 lists the TP-HEURISTIC-REPORT indication primitive and its parameter.

Table 27/X.861 – TP-HEURISTIC-REPORT indication and parameter

TP-HEURISTIC-REPORT	
Parameter	ind
Heuristic-Report	M

Heuristic-Report indicates the heuristic condition. It shall take one of the following values:

a) "heuristic-mix": The bound data of the *subordinate subtree* is in a state which is inconsistent with the outcome of the transaction and the inconsistency cannot be corrected;

b) "heuristic-hazard": A failure occurred which may prevent reporting of data inconsistency in the *subordinate subtree*.

14.18.3 TPSP constraints on a TP-HEURISTIC-REPORT indication

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall be the superior.

The *dialogue shall be coordinated*.

Unless the Implicit Prepare functional unit is selected, a TP-PREPARE request for the dialogue or a *transaction completion request* shall have been issued for the current transaction.

Unless the Implicit Prepare functional unit is selected, a TP-COMMIT indication or a *rollback-initiating service primitive* shall have been issued for the current transaction.

A TP-HEURISTIC-REPORT indication shall not have been issued since the last TP-COMMIT indication or *rollback-initiating service primitive*. TP-HEURISTIC-REPORT indications occur between the TP-COMMIT indication or the rollback-initiating service-primitive of their transaction and the next TP-COMMIT-COMPLETE indication or TP-ROLLBACK-COMPLETE indication.

The current transaction shall be in the *termination phase*.

NOTE 1 – The indication is not given if the TPSP is able to correct the effects of heuristic decisions by compensating actions.

NOTE 2 – A TP-HEURISTIC-REPORT indication may have the value "heuristic-hazard" during commitment, but never solely as a result of a dialogue abort.

14.18.4 Effects of a TP-HEURISTIC-REPORT indication

In the case of a node crash, the TPSP may issue another TP-HEURISTIC-REPORT indication on recovery.

14.19 TP-READ-ONLY request

14.19.1 Purpose

This service primitive is issued by an intermediate or leaf TPSUI to indicate that the bound data handled by the TPSUI has not been modified during the current transaction and that the TPSUI has no preference as to whether the transaction should commit or rollback.

14.19.2 Primitives and parameters

Table 28 shows the TP-READ-ONLY request primitive and its parameter.

Table 28/X.861 – TP-READ-ONLY request and its parameter

TP-READ-ONLY	
Parameter	req
Confirmation-Urgency	C

Confirmation-Urgency applies only if the Unchained Transactions functional unit has been selected on the *superior dialogue* and neither a TP-DEFERRED-END-DIALOGUE indication nor a TP-DEFERRED-GRANT-CONTROL indication has been issued nor a TP-DEFERRED-END-DIALOGUE request nor a TP-DEFERRED-GRANT-CONTROL request has been issued for any *subordinate dialogue*. It is provided by the requestor to specify the urgency with which the confirmation is required. This parameter shall take one of the following values:

a) "urgent" when the TPSUI requests minimal delay in receiving the confirm primitive;

b) "normal" when the TPSUI has no particular delay requirement in receiving the confirm primitive. In this case, the communication flow may be optimized by the TPSP.

14.19.3 TPSUI conditions on a TP-READ-ONLY request

No bound data handled by the TPSUI shall have been modified.

All processing for the transaction by the TPSUI shall have been completed.

14.19.4 TPSP constraints on TP-READ-ONLY request

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall not have any *dialogue establishment request outstanding* on any dialogue with a coordination level of "commitment".

The requestor shall have control of all *coordinated dialogues* on which none of the following services have been issued:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The requestor shall neither have any *handshake request outstanding* nor *handshake indication outstanding* for any *coordinated dialogue*.

The requestor shall not have any *user error indication outstanding* on any *coordinated dialogue*.

One of the following conditions shall be true:

a) at least one dialogue shall have a coordination level of "commitment"; or

b) the node shall be a leaf node and the *superior dialogue* shall have a coordination level of "one-phase commitment"; or

c) a TP-BEGIN-DIALOGUE confirm, a TP-U-ABORT indication, or a TP-P-ABORT indication with the Rollback parameter set to "false" has been issued for at least one dialogue having a coordination level of "commitment" during the current transaction.

If the requestor is the subordinate for a *coordinated dialogue* and the Implicit Prepare functional unit is not selected on the *superior dialogue*, then a TP-PREPARE indication or a TP-ONE-PHASE indication or a TP-READY indication shall have been issued on the *superior dialogue*.

The current transaction shall not be in the *termination phase*.

The TPSUI shall be an intermediate or leaf node in the transaction tree and the Read-only functional unit shall have been selected on the *superior dialogue*.

14.19.5 Effects of a TP-READ-ONLY request

The TPSUI may release its bound data.

A TP-PREPARE indication is issued to each direct subordinate TPSUI in the transaction tree if:

– no TP-PREPARE indication has already been issued to the subordinate; and

– the subordinate is not in the *termination phase of the transaction*; and

– the Implicit Prepare functional unit is not selected or the subordinate does not have control; and

– *ready can be received*.

A TP-READ-ONLY indication is issued to the superior TPSUI (if any) if:

– the bound data in the transaction subtree has not been modified in the current transaction; and

– the *superior dialogue* has a coordination level of "commitment"; and

– the superior is not in the *termination phase of the transaction*; and

– *ready can be sent*.

If the Dynamic Commit functional unit is selected on the *superior dialogue*, a TP-PREPARE indication is issued to the superior TPSUI if:

– the bound data in the transaction subtree has been modified in the current transaction; and

– the *superior dialogue* has a coordination level of "commitment"; and

– a TP-PREPARE indication has not already been issued to the superior; and

– the superior is not in the *termination phase of the transaction*; and

– the superior does not have control; and

– *ready can be received*.

14.19.6 Collisions

A TP-PREPARE indication is not issued to a TPSUI if a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued.

A TP-PREPARE indication is not issued to a TPSUI if there is a collision between the TP-READ-ONLY request and a *transaction completion request*.

A TP-READ-ONLY indication is not issued to a TPSUI if there is a collision of the TP-READ-ONLY request and a TP-HANDSHAKE request or a TP-U-ERROR request, instead a TP-ROLLBACK indication is issued unless a *rollback-initiating service primitive* has already been issued for the current transaction; the Diagnostic parameter of TP-ROLLBACK indication is set to "user-data-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

After a TP-READ-ONLY request, none of the following service primitives will be issued for a *coordinated dialogue*:

– TP-REQUEST-CONTROL indication;

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication; or

– TP-ONE-PHASE indication.

After a TP-READ-ONLY request, a TP-EARLY-EXIT indication is not issued; the transaction will be rolled back; a TP-ROLLBACK indication is issued with the Diagnostic parameter set to "early-exit-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

14.20 TP-READ-ONLY indication

14.20.1 Purpose

This service primitive is issued to indicate that the bound data in the *subordinate transaction subtree* has not been altered for the current transaction.

This service is associated with one particular dialogue.

14.20.2 Primitives and parameters

Table 29 shows the TP-READ-ONLY indication primitive.

Table 29/X.861 – TP-READ-ONLY indication

TP-READ-ONLY	
No parameters	ind

14.20.3 TPSP constraints on TP-READ-ONLY indication

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall be the superior.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue unless a TP-PREPARE request has been issued.

The recipient shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The recipient shall not have a *user error request outstanding*.

The *dialogue shall be coordinated*.

A TP-PREPARE request shall have been issued during the current transaction unless the Implicit Prepare functional unit was selected on the *subordinate dialogue*.

The current transaction shall not be in the *termination phase*.

All TPSUIs in the *subordinate transaction subtree* shall have issued a TP-READ-ONLY request or a TP-EARLY-EXIT request. This is a *transaction tree constraint*.

The Read-only functional unit shall have been selected on the *subordinate dialogue*.

14.20.4 Effects of a TP-READ-ONLY indication

The TPSUI is aware that processing of the current transaction in the *subordinate transaction subtree* is complete and that no changes have been made to bound data.

If the Unchained Transactions functional unit is selected and a TP-DEFERRED-END-DIALOGUE request was not issued and a TP-DEFERRED-GRANT-CONTROL request was not issued, the coordination level is "none".

14.21 TP-EARLY-EXIT request

14.21.1 Purpose

This service primitive is issued by a TPSUI to indicate that the bound data handled by the TPSUI and in the *subordinate subtree* have not been modified during the current transaction and that the TPSUI has no preference as to whether the transaction should commit or rollback.

Control of *coordinated dialogues* is not required in order to issue a TP-EARLY-EXIT request.

14.21.2 Primitives and parameters

Table 30 shows the TP-EARLY-EXIT request primitive and its parameters.

Table 30/X.861 – TP-EARLY-EXIT request and its parameters

TP-EARLY-EXIT	
Parameters	req
Severity	C
User-Data	C

14.21.2.1 Severity is used to report the severity of the condition that has caused the TPSUI to invoke the early-exit service. The parameter is mandatory when the Completion Diagnostics functional unit is selected on the *superior dialogue*; the parameter is absent otherwise. It shall take one of the following values:

a) permanent failure (general): All requests of this type will fail for the indefinite future;

b) permanent failure (specific): This request will fail but similar ones may work;

c) transient failure (general): Further requests of this type will probably work;

d) transient failure (specific): This request if attempted again now will probably work.

14.21.2.2 User-Data is optional when the Completion Diagnostics functional unit is selected on the *superior dialogue*, and is absent otherwise. This parameter may be used to convey user-specific semantics associated with the early-exit of the transaction.

14.21.3 TPSUI conditions on a TP-EARLY-EXIT request

No bound data handled by the TPSUI shall have been modified.

All processing for the transaction by the TPSUI shall have been completed.

14.21.4 TPSP constraints on TP-EARLY-EXIT request

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall not have any *dialogue establishment request outstanding* on any dialogue with a coordination level of "commitment".

The requestor is a subordinate for a *coordinated dialogue* and none of TP-READY indication or TP-ONE-PHASE indication have been issued on that dialogue.

The current transaction shall not be in the *termination phase*.

The Early-exit functional unit shall have been selected on the *superior dialogue*.

The requestor shall have received TP-EARLY-EXIT indication or TP-READ-ONLY indication on all *coordinated subordinate dialogues*.

14.21.5 Effects of a TP-EARLY-EXIT request

The TPSUI may release its bound data.

A TP-EARLY-EXIT indication is issued to the superior TPSUI if:

– there are no modified bound data in the transaction subtree; and

– the superior is not in the *termination phase of the transaction*.

A TP-ROLLBACK indication will be issued if the TPPM has bound data.

14.21.6 Collisions

A TP-EARLY-EXIT indication is not issued if there is a collision of the TP-EARLY-EXIT request and a *transaction completion request*; a TP-ROLLBACK indication will be issued with the Diagnostic parameter set to "early-exit-transaction-completion-collision".

14.22 TP-EARLY-EXIT indication

14.22.1 Purpose

This service primitive is issued to indicate that the bound data in the *subordinate transaction subtree* has not been altered for the current transaction; if the Unchained Transactions functional unit is selected, the *subordinate transaction subtree* is no longer part of the transaction tree.

This service is associated with one particular dialogue.

14.22.2 Primitives and parameters

Table 31 shows the TP-EARLY-EXIT indication primitive and its parameters.

Table 31/X.861 – TP-EARLY-EXIT indication and its parameters

TP-EARLY-EXIT	
Parameters	ind
Severity	C
User-Data	C

14.22.2.1 Severity is used to report the severity of the condition that has caused the subordinate TPSUI or the TPSP to invoke the early-exit service. The parameter is mandatory when the Completion Diagnostics functional unit is selected on the *subordinate dialogue*; the parameter is absent otherwise. When present, it shall take one of the following values:

a) permanent failure (general): All requests of this type will fail for the indefinite future;

b) permanent failure (specific): This request will fail but similar ones may work;

c) transient failure (general): Further requests of this type will probably work;

d) transient failure (specific): This request if attempted again now will probably work.

14.22.2.2 User-Data may be used to convey user-specific semantics associated with the early-exit of the transaction. The parameter is optional when the Completion Diagnostics functional unit is selected on the *subordinate dialogue*; the parameter is absent otherwise.

14.22.3 TPSP constraints on TP-EARLY-EXIT indication

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall be the dialogue superior.

The *dialogue shall be coordinated*.

The current transaction shall not be in the *termination phase*.

All TPSUIs in the *subordinate transaction subtree* shall have issued a TP-EARLY-EXIT request or a TP-READ-ONLY request. This is a *transaction tree constraint*.

The Early-exit functional unit shall have been selected on the *subordinate dialogue*.

None of the following services shall have been issued:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

14.22.4 Effects of TP-EARLY-EXIT indication

The TPSUI is aware that processing of the current transaction in the *subordinate transaction subtree* has been terminated, that the *subtree* is unable to contribute to the results of the transaction, and that no changes have been made to bound data.

If the Unchained Transactions functional unit is selected, the coordination level is "none".

14.23 TP-ONE-PHASE request

14.23.1 Purpose

This service primitive is issued by a TPSUI to indicate that it has completed all processing for the current transaction and to request that the transaction be committed.

This service primitive is issued to permit usage of the one-phase commit procedure. The TPSUI may not be informed of the outcome of the transaction in the event of failures during commitment.

14.23.2 Primitives and parameters

Table 32 lists the TP-ONE-PHASE request primitive.

Table 32/X.861 – TP-ONE-PHASE request

TP-ONE-PHASE	
No parameters	req

14.23.3 TPSUI conditions on TP-ONE-PHASE request

The TPSUI shall not have modified any bound data for the current transaction.

The TPSUI does not require reliable reporting of the outcome of the transaction.

14.23.4 TPSP constraints on TP-ONE-PHASE request

The requestor shall not have a *dialogue establishment indication outstanding*.

The requestor shall not have a *dialogue establishment request outstanding* on any *coordinated dialogue*.

The requestor shall have control of all *coordinated dialogues* for which none of the following services have been issued:

– TP-PREPARE indication;

– TP-READY indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

The requestor shall neither have any *handshake request outstanding* nor *handshake indication outstanding* for any *coordinated dialogue*.

The requestor shall not have any *user error indication outstanding* on any *coordinated dialogue*.

One of the following conditions shall be true:

a) at least one *dialogue shall be coordinated* and have the One-phase Commit functional unit selected; or

b) a TP-BEGIN-DIALOGUE confirm, a TP-U-ABORT indication, or a TP-P-ABORT indication with the Rollback parameter set to "false" has been issued for at least one *coordinated dialogue* which had the One-phase Commit functional unit selected during the current transaction; or

c) a TP-READ-ONLY indication or a TP-EARLY-EXIT indication has been issued for at least one dialogue with the One-phase Commit functional unit selected during the current transaction.

If the requestor is the subordinate for a *coordinated dialogue* and the Implicit Prepare functional unit is not selected on the *superior dialogue*, then a TP-PREPARE indication or a TP-ONE-PHASE indication or a TP-READY indication shall have been issued on the *superior dialogue*.

The current transaction shall not be in the *termination phase*.

14.23.5 Effects of a TP-ONE-PHASE request

A TP-ONE-PHASE indication is issued to one subordinate TPSUI in the transaction tree only if any other subordinates have signalled read-only or one-phase or early-exit and the superior has signalled one-phase commitment and that subordinate is not in the *termination phase of the transaction*.

A TP-PREPARE indication is issued to each direct subordinate TPSUI in the transaction tree if:

– a TP-ONE-PHASE indication will not be issued; and

– no TP-PREPARE indication has already been issued to the subordinate; and

– the subordinate is not in the *termination phase of the transaction*; and

– the Implicit Prepare functional unit is not selected or the subordinate does not have control; and

– *ready can be received*.

A TP-ONE-PHASE indication is issued to the superior TPSUI in the transaction tree only if the One-phase Commit and Dynamic Commit functional units are both selected on the *superior dialogue* and any other subordinates have signalled read-only or one-phase or early-exit and the *subordinate can send ready* and the superior is not in the *termination phase of the transaction*.

If the Dynamic Commit functional unit is selected on the *superior dialogue*, a TP-PREPARE indication is issued to the superior TPSUI if:

– the *superior dialogue* has a coordination level of "commitment"; and

– a TP-PREPARE indication has not already been issued to the superior; and

– the superior is not in the *termination phase of the transaction*; and

– the superior does not have control; and

– *ready can be received*.

14.23.6 Collisions

A TP-PREPARE indication is not issued to a TPSUI if a TP-EARLY-EXIT request or any *rollback-initiating service primitive* has been issued.

A TP-PREPARE indication is not issued to a TPSUI if there is a collision between the TP-ONE-PHASE request and a *transaction completion request*.

A TP-ONE-PHASE indication is not issued to a TPSUI if there is a collision of the TP-ONE-PHASE request and a TP-HANDSHAKE request or a TP-U-ERROR request, instead a TP-ROLLBACK indication is issued unless a *rollback-initiating service primitive* has already been issued for the current transaction; the Diagnostic parameter of TP-ROLLBACK indication is set to "early-exit-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

After a TP-ONE-PHASE request, none of the following service primitives will be issued for *coordinated dialogues*:

– TP-REQUEST-CONTROL indication;

– TP-PREPARE indication;

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-READY indication.

After a TP-ONE-PHASE request, a TP-EARLY-EXIT indication is not issued; the transaction will be rolled back; a TP-ROLLBACK indication is issued with the Diagnostic parameter set to "early-exit-transaction-completion-collision" if there is a *coordinated dialogue* with the Completion Diagnostics functional unit selected.

14.24 TP-ONE-PHASE indication

14.24.1 Purpose

This service primitive is issued to a TPSUI in a transaction tree to indicate that:

– all messages from the neighbour for the current transaction have been received;

– completion of the processing for the current transaction is requested;

– placement of the bound data handled by the TPSUI in the ready-to-commit state is requested;

– no bound data have been modified in the neighbour's *transaction hinterland*;

– no node in the neighbour's *transaction hinterland* requires reliable reporting of the outcome of the transaction.

This service is associated with one particular dialogue.

14.24.2 Primitives and parameters

Table 33 lists the TP-ONE-PHASE indication primitive.

Table 33/X.861 – TP-ONE-PHASE indication

TP-ONE-PHASE	
No parameters	ind

14.24.3 TPSP constraints on TP-ONE-PHASE indication

The recipient shall not have a *dialogue establishment request outstanding*.

The recipient shall not have a *dialogue establishment indication outstanding*.

If the Dynamic Commit functional unit is selected, *ready can be received*.

If the Dynamic Commit functional unit is not selected, the recipient shall be the subordinate.

If the Polarized Control functional unit is selected, the recipient shall not have control of the dialogue unless a TP-PREPARE request has been issued.

The recipient shall have neither a *handshake request outstanding* nor a *handshake indication outstanding*.

The recipient shall not have a *user error request outstanding*.

The *dialogue shall be coordinated*.

The One-phase Commit functional unit shall be selected for the dialogue.

A TP-ONE-PHASE indication or a TP-READY indication or a TP-READ-ONLY indication or a TP-EARLY-EXIT indication shall not have been issued during the current transaction.

The current transaction shall not be in the *termination phase*.

All TPSUIs in the *transaction hinterland* shall have issued one of the following services:

– a TP-READ-ONLY request;

– a TP-EARLY-EXIT request; or

– a TP-ONE-PHASE request.

This is a *transaction tree constraint*.

14.24.4 Effects of a TP-ONE-PHASE indication

The TPSUI is aware that processing of the current transaction in the *transaction hinterland* of the neighbour is complete and that no changes have been made to bound data, and that one-phase commit procedures will be used on the dialogue with the neighbour and within the *transaction hinterland*.

14.25 TP-UNKNOWN indication

14.25.1 Purpose

This primitive is issued by the TPSP to a TPSUI which has issued a TP-ONE-PHASE request or a TP-READ-ONLY request or a TP-EARLY-EXIT request to indicate that the outcome of the transaction is unknown.

14.25.2 Primitives and parameters

Table 34 lists the TP-UNKNOWN indication primitive.

Table 34/X.861 – TP-UNKNOWN indication

TP-UNKNOWN	
No parameters	ind

14.25.3 TPSP constraints on TP-UNKNOWN indication

The current transaction shall be in the *termination phase*.

A TP-ONE-PHASE request or a TP-READ-ONLY request or a TP-EARLY-EXIT request shall have been issued for the current transaction.

A TP-COMMIT indication or a TP-UNKNOWN indication or a *rollback-initiating service primitive* shall not have been issued for the current transaction.

14.25.4 Effects of a TP-UNKNOWN indication

The TPSUI owes a TP-DONE request.

Neither a TP-COMPLETION-REPORT indication nor a TP-HEURISTIC-REPORT indication may be received for the current transaction.

14.26 TP-UNKNOWN-COMPLETE indication

14.26.1 Purpose

This primitive is issued by the TPSP to a TPSUI which is engaged in a transaction to indicate that the transaction is terminated and the TPPM is not able to indicate the outcome of the transaction.

14.26.2 Primitives and parameters

Table 35 shows the TP-UNKNOWN-COMPLETE indication primitive.

Table 35/X.861 – TP-UNKNOWN-COMPLETE indication

TP-UNKNOWN-COMPLETE	
No parameters	ind

14.26.3 TPSP constraints on TP-UNKNOWN-COMPLETE indication

The current transaction shall be in the *termination phase*.

All bound data of the node shall have been placed in the initial state.

A TP-UNKNOWN indication shall have been issued for the current transaction.

A TP-DONE request shall have been issued for the current transaction.

A TP-DONE request shall have been issued since the last TP-P-ABORT indication, TP-U-ABORT indication, or TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" or "rejected(user)" issued for a *coordinated dialogue*.

14.26.4 Effects of a TP-UNKNOWN-COMPLETE indication

NOTE – If a TP-DEFERRED-END-DIALOGUE or TP-DEFERRED-GRANT-CONTROL service has been issued, then either TP-COMMIT indication or TP-ROLLBACK indication would be issued if the transaction outcome is known. If the transaction outcome is unknown, deferred actions are implemented for all dialogues except that if a TP-EARLY-EXIT request was issued, deferred actions are not implemented on the *superior dialogue*, and if a TP-EARLY-EXIT indication is issued, deferred actions are not implemented on a *subordinate dialogue*.

If a TP-EARLY-EXIT request was not issued and a TP-DEFERRED-END-DIALOGUE indication was issued during the transaction, the *superior dialogue* is terminated.

All *subordinate dialogues* for which a TP-DEFERRED-END-DIALOGUE request was issued and a TP-EARLY-EXIT indication was not issued during the transaction are terminated.

The recipient has control of all *subordinate dialogues* except in the cases following:

– if a TP-DEFERRED-GRANT-CONTROL request was issued and a TP-EARLY-EXIT indication was not issued during the transaction, then the recipient does not have control;

– if the Polarized Control and Chained Transactions functional units are selected, and a TP-EARLY-EXIT indication was issued during the transaction, then control of the dialogue is given to the TPSUI that had control at the beginning of the transaction.

If the Polarized Control functional unit is selected on the *superior dialogue*, then the recipient does not have control of the *superior dialogue* except in the cases following:

– if a TP-DEFERRED-GRANT-CONTROL indication has been issued and a TP-EARLY-EXIT request was not issued during the transaction, then the recipient has control;

– if the Chained Transactions functional unit is selected, and a TP-EARLY-EXIT request was issued during the transaction, then control of the dialogue is given to the TPSUI that had control at the beginning of the transaction.

The coordination level of all dialogues on which the Unchained Transactions functional unit is selected is "none".

If the recipient has at least one *coordinated dialogue*, then the recipient is involved in a new transaction.

14.27 TP-COMPLETION-REPORT indication

14.27.1 Purpose

This service primitive is issued for each *coordinated subordinate dialogue* for which diagnostic information related to completion of the transaction is available, including the cause and severity of rollback. The parameter values are those provided by the subordinate TPSUI or TPSP.

The service is associated with one particular dialogue.

14.27.2 Primitives and parameters

Table 36 shows the TP-COMPLETION-REPORT indication primitive and its parameters.

Table 36/X.861 – TP-COMPLETION-REPORT indication and its parameters

TP-COMPLETION-REPORT	
Parameters	ind
Severity	C
Completion-Data	C
Diagnostic	C

14.27.2.1 Severity is a parameter which is used to report the severity of the condition that has caused the subordinate TPSUI or the TPSP to invoke the rollback service. It is mandatory when the transaction has rolled back and absent otherwise. When present, it shall take one of the following values:

a) permanent failure (general): All requests of this type will fail for the indefinite future;

b) permanent failure (specific): This request will fail but similar ones may work;

c) transient failure (general): Further requests of this type will probably work;

d) transient failure (specific): This request if attempted again now will probably work.

14.27.2.2 Completion-Data is an optional parameter which may be used to convey user-specific semantics associated with the completion of the transaction.

14.27.2.3 Diagnostic is a parameter which may be used by the TPSP to report the nature of the condition that has caused the TPSP to invoke the rollback service. It is mandatory when the transaction has rolled back and is absent otherwise. If present, it shall take one of the following values:

a) "user-rollback" when a subordinate TPSUI has invoked the rollback procedure; the Completion-Data parameter may give further information;

b) "user-data-transaction-completion-collision" when a user data request or a TP Service request has collided with a *transaction completion request*.

NOTE 1 – Such TP services could be one of TP-U-ERROR request, TP-HANDSHAKE request, TP-DEFERRED-END-DIALOGUE request.

c) "early-exit-transaction-completion-collision" when a TP-EARLY-EXIT request has collided with a *transaction completion request*;

d) "other-provider-rollback" when the TPSP has invoked the rollback procedure.

NOTE 2 – A programming interface may return other diagnostic information which can be determined by the implementation; this is outside the scope of ISO/IEC 10026.

14.27.3 TPSP constraints on TP-COMPLETION-REPORT indication

The recipient shall not have a *dialogue establishment request outstanding*, unless the Diagnostic parameter is present with value "other-provider-rollback".

The recipient shall be the superior.

The *dialogue shall be coordinated*.

The Completion Diagnostics functional unit shall have been selected.

A TP-COMMIT indication or any *rollback-initiating service primitive* shall have been issued.

None of the following services shall have been issued for the dialogue during the current transaction:

– TP-READ-ONLY indication;

– TP-ONE-PHASE indication; or

– TP-EARLY-EXIT indication.

At most one TP-COMPLETION-REPORT indication may occur per dialogue per transaction.

14.27.4 Effects of a TP-COMPLETION-REPORT indication

The TPSUI owes a TP-DONE request.

A further TP-COMPLETION-REPORT indication may not be issued for this dialogue for this transaction.

Annex A

Service state table

A.1 Overview

The service state table describes the allowed sequence of service events for a given TP service boundary (at a given node, between a TPSUI and the TPSP).

A separate state is maintained for each dialogue of the node. In addition, a state is maintained for the node itself represented by node-related variables.

Service primitives which relate to a single dialogue affect only the state for that particular dialogue. Some service primitives affect all coordinated dialogues and the node state. The issuance of such service primitives are valid only if they are valid for all the dialogues they affect and the state of the node.

The state table specifies predicates which must be satisfied in order for individual service primitives to be valid in a given dialogue or node state. These predicates are based on the values of variables.

The state table also specifies actions to be performed. These actions involve setting variables to specified values. The referenced variables are of two types. One type is private to a particular dialogue and the other type is global among all dialogues of a TPSUI.

The overall state of the service boundary (for a given node) consists of the state of each dialogue of the TPSUI, together with the node state and with the associated values of the private and global variables.

When a new dialogue is created, a new dialogue context is also created, with its state set to the Idle state and all variables private to the dialogue are initialized to "false". If this newly created dialogue is the first one for that TPSUI, then the node context is also created and all the global variables are initialized to "false" as well.

When a dialogue is terminated, the dialogue context ceases to exist.

When all the dialogues associated with a node are terminated and there is no current transaction at the node, then the node context ceases to exist.

Exceptionally some TP services related to transaction completion are valid when there are no *coordinated dialogues* remaining; for this case a "node contingency table" is provided and it determines the actions to be implemented.

NOTE – This node contingency table is only required to correctly model the case where there are *subordinate dialogues* with the Read-only or Early-exit functional unit selected, together with the Unchained Transactions functional unit.

A.2 Dialogue states

1) Idle state.

2) The TPSUI has control of the dialogue.

3) The TPSUI does not have control of the dialogue.

   This state is used only if the Polarized Control functional unit is selected.

4) The TPSUI which does not have control of the dialogue has issued a TP-U-ERROR request.

   This state is used only if the Polarized Control functional unit is selected.

5) The TPSUI which has control of the dialogue has received a TP-U-ERROR indication.

   This state is used only if the Polarized Control functional unit is selected.

6) The TPSUI has issued a TP-HANDSHAKE request.

   This state is used only if the Handshake functional unit is selected.

7) The TPSUI has received a TP-HANDSHAKE indication.

This state is used only if the Handshake functional unit is selected.

8) The TPSUI has received a TP-HANDSHAKE indication and a TP-HANDSHAKE request is outstanding.

This state is used only if both the Handshake and Shared Control functional units are selected.

9) The TPSUI has received a TP-END-DIALOGUE (Confirmation = "true") indication and a TP-HANDSHAKE request is outstanding.

This state is used only for a dialogue with a coordination level of "none" and both the Shared Control and the Handshake functional units are selected.

10) The TPSUI has received a TP-HANDSHAKE indication and a TP-END-DIALOGUE (Confirmation = "true") request is outstanding.

This state is used only for a dialogue with a coordination level of "none" and both the Shared Control and the Handshake functional units are selected.

11) The TPSUI has issued a TP-END-DIALOGUE (Confirmation = "true") request.

This state is used only for a dialogue with a coordination level of "none".

12) The TPSUI has received a TP-END-DIALOGUE (Confirmation = "true") indication.

This state is used only for a dialogue with a coordination level of "none".

13) The TPSUI has issued a TP-HANDSHAKE-AND-GRANT-CONTROL request.

This state is used only if the Handshake and Polarized Control functional units are selected.

14) The TPSUI has received a TP-HANDSHAKE-AND-GRANT-CONTROL indication.

This state is used only if the Handshake and Polarized Control functional units are selected.

15) The TPSUI has issued a TP-PREPARE (Data-Permitted = "false") request.

This state is used only for a *coordinated dialogue* and the Polarized Control functional unit is selected; unless the Dynamic Commit functional unit is selected, the dialogue must be *a subordinate dialogue*.

16) The TPSUI has issued a TP-PREPARE request, with either the Data-Permitted parameter set to "true" or the Shared Control functional unit is selected on this dialogue.

This state is used only for a *coordinated dialogue*; unless the Dynamic Commit functional unit is selected, the dialogue must be *a subordinate dialogue*.

16.1) The TPSUI has issued a TP-PREPARE request and received a TP-PREPARE indication.

This state is used only for a dialogue with a coordination level of "commitment" and with the Dynamic Commit functional unit selected.

17) The TPSUI has received a TP-READY indication, or a TP-READ-ONLY indication, or a TP-EARLY-EXIT indication, or a TP-ONE-PHASE indication.

This state is used for a dialogue with a coordination level of "commitment"; in this case, unless the Dynamic Commit functional unit is selected, the dialogue must be *a subordinate dialogue*.

This state is also used for a *superior dialogue* with a coordination level of "one-phase commitment".

18) The TPSUI has received a TP-PREPARE (Data-Permitted = "false") indication.

This state is used only for a *coordinated dialogue* and the Polarized Control functional unit is selected; unless the Dynamic Commit functional unit is selected, the dialogue must be *a superior dialogue*.

19) The TPSUI has received a TP-PREPARE indication, with either the Data-Permitted parameter set to "true" or the Shared Control functional unit is selected on this dialogue.

This state is used only for a *coordinated dialogue*; unless the Dynamic Commit functional unit is selected, the dialogue must be *a superior dialogue*.

20) The TPSUI has issued a TP-COMMIT request, or a TP-READ-ONLY request, or a TP-EARLY-EXIT request, or a TP-ONE-PHASE request.

In the case of TP-COMMIT request or TP-READ-ONLY request, or a TP-EARLY-EXIT request, this state is used only for a dialogue with a coordination level of "commitment", or for a *superior dialogue* with a coordination level of "one-phase commitment" and either the TPSUI is a leaf in the transaction tree or there is at least one *subordinate dialogue* with a coordination level of "commitment".

In the case of TP-ONE-PHASE request, this state is only used for a *coordinated dialogue*.

21) The TPSUI has received a TP-COMMIT indication and is placing its bound data in the final state, or the TPSUI has received a TP-UNKNOWN indication, or the TPSUI has received one or more of the following indications after having issued a TP-DONE request: TP-COMPLETION-REPORT indication, TP-P-ABORT indication or TP-U-ABORT indication.

This state is only used for a dialogue with a coordination level of "commitment", or for a *superior dialogue* with a coordination level of "one-phase commitment" and either the TPSUI is a leaf in the transaction tree or there is at least one *subordinate dialogue* with a coordination level of "commitment".

22) The transaction outcome is either commitment or unknown, the TPSUI has discharged its responsibilities with respect to bound data and does not owe a TP-DONE request.

This state is used only for a dialogue with a coordination level of "commitment", or for a *superior dialogue* with a coordination level of "one-phase commitment" and either the TPSUI is a leaf in the transaction tree or there is at least one *subordinate dialogue* with a coordination level of "commitment".

23) A *rollback-initiating service primitive* has been issued, or the TPSUI has received one or more of the following indications or confirms after having issued TP-DONE request: TP-P-ABORT indication, TP-U-ABORT indication, a TP-COMPLETION-REPORT indication, or a TP-BEGIN-DIALOGUE confirm with the Result parameter set to "rejected(provider)" or "rejected(user)".

This state is used only for a dialogue with a coordination level of "commitment" or "one-phase commitment".

24) The transaction outcome is known to be rollback, the TPSUI has discharged its responsibilities with respect to bound data and does not owe a TP-DONE request.

This state is used only for a dialogue with a coordination level of "commitment" or "one-phase commitment".

25) The TPSUI has received one of the following service primitive with Rollback = "false" for a dialogue with a coordination level of "commitment" or "one-phase commitment" before the *termination phase of the transaction*:

– TP-BEGIN-DIALOGUE [rejected(provider) or rejected(user)] confirm;

– TP-P-ABORT indication; or

– TP-U-ABORT indication.

This state is used only for a terminated dialogue with a coordination level of "commitment" or "one-phase commitment" when the context of this "zombie" dialogue will be involved in the termination of a transaction.

A.3 Variables

The variables are Boolean. When a new dialogue context is created, all the variables private to this dialogue are initialized to "false" and the dialogue is in the Idle State, then the transition corresponding to the initiating service primitive (TP-BEGIN-DIALOGUE request or TP-BEGIN-DIALOGUE indication) is fired.

A.3.1 Dialogue-related variables

The name of the variables related to a single dialogue starts with a "D"; they are the following:

– **D2exp** (**2**-phase **EXP**ected branch): This variable, when set to "true", indicates that this is a *two-phase expected branch*.

– **Da** (**A**ccepted by TPSUI): This variable, when set to "true", indicates that the TPSUI has already received an indication from its partner TPSUI and the TP-BEGIN-DIALOGUE confirm shall no longer occur. This variable is meaningful only for a dialogue with a subordinate.

– **Danu** (**A**tomic-action **N**ot **U**sed): This variable, when set to "true", indicates that a TP-U-ABORT indication has been issued after a TP-COMMIT indication and before a TP-COMMIT-COMPLETE indication, for a dialogue with the Chained Transactions functional unit selected, and for which no TP-DEFERRED-END-DIALOGUE service has been issued.

– **Dar** (**A**fter **R**ollback): This variable, when set to "true", indicates that a TP-ROLLBACK-COMPLETE indication has been issued and no indication or confirm have been received from a subordinate.

  This variable is used to detect if a TP-U-ABORT indication with Rollback = "false" is valid after a TP-ROLLBACK-COMPLETE indication for a *subordinate dialogue* for which the Chained Transactions functional unit is selected.

– **Db** (a**B**orted): This variable, when set to "true", indicates that the dialogue is being aborted.

– **Dc** (**C**ontrol): This variable, when set to "true", indicates that the TPSUI had control of the dialogue at the beginning of a transaction.

  This variable is used to determine which TPSUI acquires control of the dialogue if rollback has occurred.

– **Dcd** (**C**ompletion **D**iagnostics): This variable, when set to "true", indicates that the Completion Diagnostics functional unit is selected.

– **Dco** (**CO**mmit): This variable, when set to "true", indicates that the Commit functional unit is selected.

– **Dcr** (TP-BEGIN-DIALOGUE **C**onfirmation **R**equired): This variable, when set to "true", indicates that a TP-BEGIN-DIALOGUE (Confirmation = "always") request or indication has been issued, and that a response or confirm has not been issued.

– **Dcra** (**C**ompletion **R**eport **A**llowed): This variable, when set to "true", indicates that a TP-COMPLETION-REPORT indication may be received.

  This variable is used to detect if a TP-COMPLETION-REPORT indication is valid for a given dialogue. It is set to "true" if the Completion Diagnostics functional unit is selected when a TP-COMMIT indication or any *rollback-initiating service primitive* is issued. It is set to "false" once a transaction ends or a report is issued for the dialogue.

– **Dcrr** (**C**an **R**eceive **R**eady): This variable, when set to "true", indicates that *ready can be received* on the dialogue.

  This variable has no meaning for a dialogue where the Commit functional unit is not selected.

– **Dcsr** (**C**an **S**end **R**eady): This variable, when set to "true", indicates that *ready can be sent* on the dialogue.

  This variable has no meaning for a dialogue where the Commit functional unit is not selected.

– **Ddyn** (**DYN**amic): This variable, when set to "true", indicates that the Dynamic Commit functional unit is selected.

– **De** (deferred **E**nd dialogue): This variable, when set to "true", indicates that the TPSUI has issued TP-DEFERRED-END-DIALOGUE request or received TP-DEFERRED-END-DIALOGUE indication, i.e. the dialogue will be terminated if the provider supported transaction commits or if the outcome is unknown.

– **Dee** (**E**arly-**E**xit FU): This variable, when set to "true", indicates that the Early-exit functional unit is selected.

– **Deei** (**E**arly-**E**xit **I**ndication): This variable, when set to "true", indicates that a TP-EARLY-EXIT indication has been issued.

  This variable is used to determine what should happen to control at the end of a transaction; it is set to "false" once the transaction ends.

– **Dex** (**EX**clusive branch): This variable, when set to "true", indicates that this is an *exclusive branch*.

– **Dg** (deferred **G**rant control): This variable, when set to "true", indicates whether the TPSUI has issued TP-DEFERRED-GRANT-CONTROL request or received TP-DEFERRED-GRANT-CONTROL indication, i.e. control of the dialogue will be transferred if the provider supported transaction commits or if the outcome is unknown.

– **Dh** (**H**andshake): This variable, when set to "true", indicates that the Handshake functional unit is selected.

– **Dhr** (**H**euristic **R**eport): This variable, when set to "true", indicates that a TP-HEURISTIC-REPORT indication may be received.

This variable is used to detect if a TP-HEURISTIC-REPORT indication is valid for a given dialogue.

– **Dhcr** (**H**euristic **C**ontainment **R**equired FU): This variable, when set to "true", indicates that the Heuristic Containment Required functional unit is selected.

– **Dip** (**I**mplicit **P**repare): This variable, when set to "true", indicates that the Implicit Prepare functional unit is selected.

– **Dl** (coordination **L**evel): This variable reflects the coordination level of the dialogue.

This value, when set to "true", indicates that the *dialogue is coordinated*, i.e. the coordination level is "commitment" or "one-phase commitment"; this value, when set to "false", indicates that the coordination level is set to "none".

In order to distinguish between coordination levels of "commitment" and "one-phase commitment", then an additional variable must be tested as appropriate, e.g. Ddyn, Dco.

– **Dop** (**O**ne-**P**hase Commit FU): This variable, when set to "true", indicates that the One-phase Commit functional unit is selected.

– **Dopi** (**O**ne-**P**hase **I**ndication): This variable, when set to "true", indicates that a TP-ONE-PHASE indication has been issued.

This variable is set to "false" once the transaction ends.

– **Dores** (**O**ne-phase, **R**ead-only, **E**arly-exit **S**elected): This variable, when set to "true", indicates that the One-phase functional unit and/or the Read-only functional unit and/or the Early-exit functional unit are selected.

– **Dres** (**R**ead-only, **E**arly-exit **S**elected): This variable, when set to "true", indicates that the Read-only functional unit and/or the Early-exit functional unit are selected.

– **Dro** (**R**ead-**O**nly FU): This variable, when set to "true", indicates that the Read-only functional unit is selected.

– **Droi** (**R**ead-**O**nly **I**ndication): This variable, when set to "true", indicates that a TP-READ-ONLY indication has been issued.

This variable is set to "false" once the transaction ends.

– **Dsh** (**SH**ared Control): This variable, when set to "true" indicates that the Shared Control functional unit is selected.

– **Dsopex** (**S**tatic **O**ne-**P**hase **EX**clusive branch): This variable, when set to "true", indicates that this is a *static one-phase exclusive branch*.

– **Dsup** (**SUP**erior dialogue): This variable, when set to "true", indicates that the dialogue has been established by a TP-BEGIN-DIALOGUE indication.

– **Dtrb** (**T**o be **R**olled **B**ack): This variable, when set to "true", indicates that a TP-P-ABORT indication has been issued after a TP-COMMIT indication and before a TP-COMMIT-COMPLETE indication, for a dialogue which has the Chained Transactions functional unit selected, and for which no TP-DEFERRED-END-DIALOGUE service has been issued.

– **Du** (**U**nchained): This variable, when set to "true", indicates that the Unchained Transactions functional unit is selected. This variable, when set to "false", indicates that the Chained Transactions functional unit is selected.

The variable is only relevant when the Commit functional unit and/or One-phase Commit functional unit is selected.

– **Dx** (e**X**tension of transaction tree): This variable, when set to "true", indicates that a TP-BEGIN-TRANSACTION request has been issued, and that if either a TP-U-ABORT request is issued by the TPSUI or a TP-P-ABORT indication is triggered by the TPSP, before the TP-BEGIN-TRANSACTION indication is issued, the TP-U/P-ABORT indication shall be received with the Rollback parameter set to "false".

  This variable is set to "true" when a TP-BEGIN-TRANSACTION request is issued. It is set to "false" on a TP-ROLLBACK-COMPLETE indication, a TP-READY indication, a TP-COMMIT indication, a TP-HANDSHAKE confirm, and a TP-HANDSHAKE-AND-GRANT-CONTROL confirm.

A.3.2 Node-related variables

The name of the variables related to all dialogues of a TPSUI starts with an "N" (standing for "Node"); they are the following:

– **N2exp** (**2**-phase **EXP**ected branch): This variable, when set to "true", indicates that a *two-phase expected branch* exists.

– **Ncora** (TP-**C**OMMIT **R**equest **A**llowed): This variable, when set to "true", indicates that a TP-COMMIT request may be issued for the current transaction.

  Ncora is set to "true" if at the establishment of a transaction branch, the Commit functional unit is selected on the dialogue or the One-phase Commit functional unit is selected on the *superior dialogue*; it remains set if the branch leaves the transaction as the result of a read-only signal or early-exit signal; it is set to "false" when the transaction ends.

– **Ndial** (**DIAL**ogue established): This variable, when set to "true", indicates that a TP-BEGIN-DIALOGUE indication has been issued, i.e. the node is an intermediate or a leaf.

– **Nee** (**E**arly-**E**xit FU): This variable, when set to "true", indicates that the Early-exit functional unit is selected on the dialogue to the superior.

– **Neer** (**E**arly-**E**xit **R**equest): This variable, when set to "true", indicates that a TP-EARLY-EXIT request has been issued.

– **Neera** (**E**arly-**E**xit **R**equest **A**llowed): This variable, when set to "true", indicates that a TP-EARLY-EXIT request is allowed.

  This variable is used to determine whether one of the conditions for issuing a TP-EARLY-EXIT request has been fulfilled, i.e. the Early-exit functional unit is selected on the *coordinated superior dialogue* and either the TPSUI is a leaf in the transaction tree, or a TP-EARLY-EXIT indication or a TP-READ-ONLY indication has been received from every *transaction subordinate*.

– **Nex** (**EX**clusive branch): This variable, when set to "true", indicates that an *exclusive branch* exists.

– **Nfa** (**F**ailure action **A**llowed):This variable, when set to "true", indicates that the TPSUI is allowed to issue a TP-U-ABORT request on a dialogue having a coordination level of "commitment" during the *termination phase of a transaction*. This variable is set to "true" when a TP-P-ABORT indication, TP-U-ABORT indication, or TP-BEGIN-DIALOGUE [rejected(provider) or rejected(user)] confirm has been received on a dialogue whose coordination level is "commitment" and set to "false" upon issuance of TP-DONE request.

– **Nip** (**I**mplicit **P**repare): This variable, when set to "true", indicates that the Implicit Prepare functional unit is selected on the dialogue to the superior.

– **Nopr** (**O**ne-**P**hase **R**equest): This variable, when set to "true", indicates that a TP-ONE-PHASE request has been issued.

– **Nopra** (**O**ne-**P**hase **R**equest **A**llowed): This variable, when set to "true", indicates that a TP-ONE-PHASE request may be issued for the current transaction.

  Nopra is set to "true" if at the establishment of a transaction branch, the One-phase functional unit is selected on the dialogue; it remains set if the branch leaves the transaction as the result of a read-only signal or early-exit signal; it is set to "false" when the transaction ends.

– **Npa** (**P**repare **A**llowed): This variable, when set to "true", indicates that the TPSUI may issue TP-PREPARE request or TP-COMMIT request or a TP-READ-ONLY request or a TP-ONE-PHASE request and, also, that a TP-DONE request may be issued with the Heuristic-Report parameter.

  When the TPSUI is the root of a transaction tree or the Implicit Prepare functional unit is selected on the superior branch, Npa is set to "true" at the beginning of a transaction, otherwise Npa is set to "true" on receipt of a TP-PREPARE indication or a TP-EARLY-EXIT indication or a TP-ONE-PHASE indication on the superior branch.

– **Nr** (**R**oot of transaction tree): This variable, when set to "true", indicates that the node is the root of a transaction tree.

– **Nrn** (TP-BEGIN-DIALOGUE **R**eject **N**o longer allowed): This variable, when set to "true", indicates that the TPSUI has issued a request or response on any dialogue, and the *superior dialogue* can no longer be rejected.

  A TPSUI can no longer issue TP-BEGIN-DIALOGUE response when this variable is set to "true".

– **Nro** (**R**ead-**O**nly Functional Unit): This variable, when set to "true", indicates that the Read-only functional unit is selected on the *superior dialogue*. This variable is meaningful only for an intermediate or leaf node.

– **Nror** (**R**ead-**O**nly **R**equest): This variable, when set to "true", indicates that a TP-READ-ONLY request has been issued.

– **Nsopex** (**S**tatic **O**ne-**P**hase **EX**clusive branch): This variable, when set to "true", indicates that a *static one-phase exclusive branch* exists.

– **Nsubop** (**SUB**ordinate branch exists with "**O**ne-**P**hase Commitment"): This variable, when set to "true", indicates that a *subordinate branch* exists with coordination level "one-phase commitment".

– **Nsupco** (**SUPerior** branch exists with "**Co**mmitment"): This variable, when set to "true", indicates that a *superior branch* exists with coordination level "commitment".

– **Nsupest** (**SUP**erior dialogue **EST**ablished): This variable, when set to "true", indicates that a TP-BEGIN-DIALOGUE (Confirmation = "negative") indication has been received, or a TP-BEGIN-DIALOGUE (Result = "accepted") response has been issued.

– **Nsupop** (**SUP**erior branch exists with "**O**ne-**P**hase Commitment"): This variable, when set to "true", indicates that a superior branch exists with coordination level "one-phase commitment".

– **Nt** (**T**ermination): This variable, when set to "true", indicates that the provider-supported transaction is being terminated.

– **Ntch** (**T**ree **CH**ecking): This variable, when set to "true", indicates that there is *tree checking* at this node.

  When the node enters a transaction tree, the variable is set to "true" either upon a local decision to check tree extension rules or after being ordered by the superior node in the transaction tree to check the tree extension rules. The variable is set (from "true") to "false" after completion of the current transaction. Ntch may also be set to true as the result of a local decision by the TPSP.

– **Nunki** (**UNK**nown **I**ndication): This variable, when set to "true", indicates that a TP-UNKNOWN indication has been issued.

A.4 Actions

NOTE 1 – The action labels are case sensitive, i.e. [a] and [A] are distinct actions.

[a] Action related to unconditional termination of a dialogue

  The dialogue context ceases to exist. Service primitives can no longer be issued on this dialogue.

  Since the status of the node may have changed (from root node of a transaction tree to node not belonging to any transaction tree):

  – if Dsup = "true", then set Nip = "false" and Nro = "false" and Nee = "false";

  – if Dl is "true" and there are no other dialogues with Dl set to "true", set Nr to "false" and Npa to "false".

[b] Action related to any request or response which may be the first one issued on a dialogue:

  – if Dsup is "true", set Nrn to "true".

[c] Action related to TP-BEGIN-DIALOGUE response and confirm with Result set to "accepted":

– set Dcr to "false";

– if Dsup is "true", set Nrn to "true" and Nsupest to "true", otherwise, set Da to "true".

[d] Action related to TP-DONE request

If this is the last dialogue to be processed for the request, then:

– set Nfa to "false";

– set Npa to "false".

[e] Action related to a TP-P-ABORT indication or TP-U-ABORT request issued after a TP-COMMIT indication for a dialogue which has the Chained Transactions functional unit selected and for which no TP-DEFERRED-END-DIALOGUE has been issued:

– set Dtrb to "true".

[f] Action related to a rollback-initiating service primitive:

– set Nt to "true";

– set Nfa to "true";

– if Dip is set to "true" and Dhcr is set to "false", then set Dhr to "true";

– if Dcd is set to "true", then set Dcra to "true".

[g] Action related to TP-DEFERRED-END-DIALOGUE request and indication:

– set De to "true".

[h] Action related to TP-DEFERRED-GRANT-CONTROL request and indication:

– set Dg to "true".

[i] Action related to TP-COMMIT-COMPLETE indication or TP-UNKNOWN-COMPLETE indication:

– set Danu to "false".

[j] Action related to TP-U-ABORT indication issued after a TP-COMMIT indication for a dialogue which has the Chained Transactions functional unit selected and for which no TP-DEFERRED-END-DIALOGUE has been issued:

– set Danu to "true".

[k] Action related to TP-ROLLBACK-COMPLETE indication when the dialogue is not being terminated:

– set Da to "true";

– set Dg to "false";

– set De to "false";

– set Dcra to "false";

– set Dx to "false";

– if Du is "true", then set Dl to "false" and D2exp to "false" and Dex to "false" and Dsopex to "false";

– if Du is "false" and Dsup is "false", then set Dar to "true".

If this is the last dialogue to be processed for the indication, then:

– set Nr to "false";

– set Npa to "false".

NOTE 2 – These variables will be set correctly in action [z] if the node is the root of a new transaction.

[l] Action related to the reception of TP-U-ABORT indication, TP-P-ABORT indication, TP-BEGIN-DIALOGUE (Result = "rejected(provider)" or "rejected(user)") and (Rollback = "false") confirm:

– if Dl is set to "true", set Nfa to "true".

[m] Action related to a superior issuing a TP-COMMIT request or a TP-READ-ONLY request or a TP-ONE-PHASE request or a TP-PREPARE request:

– if Dhcr is "false", then set Dhr to "true".

[n] Action related to TP-COMMIT-COMPLETE indication, or TP-UNKNOWN-COMPLETE indication, or TP-ROLLBACK-COMPLETE indication or dialogue termination without rollback after a TP-PREPARE request has been issued:

– set Dhr to "false";

– set Dcra to "false".

[o] Action related to a superior receiving an indication or confirm

The subordinate TPSUI has issued some service request or response. The dialogue can no longer be rejected:

– set Da to "true";

– set Dar to "false".

[p] Action related to termination of a dialogue

The dialogue has been terminated. Service requests, invoked by the partner TPSUI, for which an indication has not been delivered yet, are suppressed:

– set Db to "true";

– set Dx to "false".

[q] Action related to TP-BEGIN-TRANSACTION request:

– set Dc to "true";

– set Dl to "true";

– if the TPSUI has no superior or has a coordination level of "none" on the *superior dialogue*, then set Nr to "true" and set Npa to "true";

– if Nr = "false" and Nip = "true", then set Npa to "true";

– set Dx to "true";

– if Dco = "false", then set Nsubop to "true";

– if Dl = "true" and Dco = "true", then set Ncora to "true";

– if Dl = "true" and Dop = "true", then set Nopra to "true";

– set Neera to "false";

– if Dl = "true" and Dcrr = "true" and either Dores = "false" or Ntch = "true", then set D2exp to "true";

– if D2exp = "true", then set N2exp to "true";

– if Dl = "true" and Dcsr = "true" and Dcrr = "false" and either Dres = "false" or Ntch = "true", then set Dex to "true";

– if Dl = "true" and Dco = "false" and either Dres = "false" or Ntch = "true", then set Dsopex to "true" and Dex to "true" and Nsopex to "true";

– if Dex = "true", then set Nex to "true".

[r] Action related to TP-HEURISTIC-REPORT indication:

– set Dhr to "false".

[s] Action related to TP-COMMIT-COMPLETE indication or TP-UNKNOWN-COMPLETE indication when the dialogue is not being terminated:

– if the Polarized Control functional unit is selected, and Dsup = Dg, then set Dc to "true", otherwise set Dc to "false";

– set Dg to "false";

– set De to "false";

– set Dcra to "false";

– if Du is set to "true", then set Dl to "false" and D2exp to "false" and Dex to "false" and Dsopex to "false";

– set Dopi to "false";

– set Droi to "false";

– set Deei to "false".

If this is the last dialogue to be processed for the indication, then:

– set Nr to "false";

– set Npa to "false".

[t] Action related to a superior following issuance of TP-READY indication, TP-COMMIT indication, TP-READ-ONLY indication, TP-EARLY-EXIT indication, TP-ONE-PHASE indication, TP-HANDSHAKE confirm, and TP HANDSHAKE-AND-GRANT-CONTROL confirm

The *subordinate dialogue* can no longer be rejected, and an indication or a confirm has been issued since any TP-ROLLBACK-COMPLETE indication:

– set Da to "true";

– set Dar to "false";

– set Dx to "false".

[u] Action related to TP-BEGIN-TRANSACTION indication:

– set Dc to "false";

– set Dl to "true";

– if Dl = "true" and Dco = "true", then set Nsupco to "true";

– if Dl = "true" and Dco = "false", then set Nsupop to "true";

– if Dl = "true" and Dco = "true", then set Ncora to "true";

– if Dl = "true" and Dop = "true" and Ddyn = "false", then set Ncora to "true";

– if Dl = "true" and Dop = "true" and Ddyn = "true", then set Nopra to "true";

– if Dl = "true" and (Dcsr = "true" and Dcrr = "true" and Check-ready-directions is absent or set to "true") or (Dcsr = "true" and Dcrr = "false"), then set Ntch to "true";

– if Dee ="true", then set Neera to "true";

– if Dl = "true" and Dcsr = "true" and either Dcrr = "false" or Ntch = "true", then set Dex to "true" and Nex = "true".

[v] Action related to TP-BEGIN-DIALOGUE request:

– set Dc to "true";

– if Ndial is "true", set Nrn to "true";

– set Dsh to "true" if the Shared Control functional unit is selected;

– set Dh to "true" if the Handshake functional unit is selected;

– set Dco to "true" if the Commit functional unit is selected;

– set Dop to "true" if the One-phase Commit functional unit is selected;

– set Dro to "true" if the Read-only functional unit is selected;

– set Dee to "true" if the Early-exit functional unit is selected;

– if Dro = "true" or Dee = "true", then set Dres to "true";

– if Dres = "true" or Dop = "true", then set Dores to "true";

– set Ddyn to "true" if the Dynamic Commit functional unit is selected;

– set Dip to "true" if the Implicit Prepare functional unit is selected;

– set Dcd to "true" if the Completion Diagnostics functional unit is selected;

– set Dhcr to "true" if the Heuristic Containment Required functional unit is selected;

– set Du to "true" if the Unchained Transactions functional unit is selected;

– set Dl to "true" if either the Commit functional unit or the One-phase Commit functional unit is selected and if the Chained Transactions functional unit is also selected;

– set Dl to "true" if either or both of the Commit functional unit and the One-phase Commit functional unit is selected, and if the Unchained Transactions functional unit is also selected, Begin-Transaction = "true";

– if Dl is set to "true" and no other dialogues with Dl set to "true" exist, set Nr to "true" and set Npa to "true";

– if Confirmation = "always", then set Dcr to "true";

– set Dcrr to "true" if the Commit functional unit is selected and the Dynamic Commit functional unit is not selected, or the Dynamic Commit functional unit is selected and Subordinate-may-send-ready is set to "true";

– set Dcsr to "true" if the Dynamic Commit functional unit is selected and Superior-may-send-ready is set to "true";

– if Dl = "true" and Dco = "false", then set Nsubop to "true";

– if Dl = "true" and Dop = "true", then set Nopra to "true";

– if Dl = "true" and Dco = "true", then set Ncora to "true";

– if Dl = "true", then set Neera to "false";

– if Dl = "true" and Dcrr = "true" and either Dores = "false" or Ntch = "true", then set D2exp to "true";

– if D2exp = "true", then set N2exp to "true";

– if Dl = "true" and Dcsr = "true" and Dcrr = "false" and either Dres = "false" or Ntch = "true", then set Dex to "true";

– if Dl = "true" and Dco = "false" and either Dres = "false" or Ntch = "true", then set Dsopex to "true" and Dex to "true" and Nsopex to "true";

– if Dex = "true", then set Nex to "true".

[w] Action related to TP-BEGIN-DIALOGUE indication:

– set Dsup to "true";

– set Ndial to "true";

– set Dsh to "true" if the Shared Control functional unit is selected;

– set Dh to "true" if the Handshake functional unit is selected;

– set Dco to "true" if the Commit functional unit is selected;

– set Dop to "true" if the One-phase Commit functional unit is selected;

– set Dro to "true" and Nro to "true" if the Read-only functional unit is selected;

– set Dee to "true" and Nee to "true" if the Early-exit functional unit is selected;

– if Dro = "true" or Dee = "true", then set Dres to "true";

– if Dres = "true" or Dop = "true", then set Dores to "true";

– set Ddyn to "true" if the Dynamic Commit functional unit is selected;

– set Dip to "true" and Nip to "true" if the Implicit Prepare functional unit is selected;

– set Dcd to "true" if the Completion Diagnostics functional unit is selected;

– set Dhcr to "true" if the Heuristic Containment Required functional unit is selected;

– set Du to "true" if the Unchained Transactions functional unit is selected;

– set Dl to "true" if either the Commit functional unit or the One-phase Commit functional unit is selected and if the Chained Transactions functional unit is also selected;

– set Dl to "true" if either or both of the Commit functional unit and the One-phase Commit functional unit is selected, and if the Unchained Transactions functional unit is also selected, Begin-Transaction = "true";

– if Dl = "true" and Nip = "true", then set Npa to "true";

– if Confirmation = "always", then set Dcr to "true", otherwise set Nsupest to "true";

– set Dcsr to "true" if the Commit functional unit is selected and the Dynamic Commit functional unit is not selected, or the Dynamic Commit functional unit is selected and Subordinate-may-send-ready is set to "true";

– set Dcrr to "true" if the Dynamic Commit functional unit is selected and Superior-may-send-ready is set to "true";

– if Dl = "true" and Dco = "true", then set Nsupco to "true";

– if Dl = "true" and Dco = "false", then set Nsupop to "true";

– if Dl = "true" and Dco = "true", then set Ncora to "true";

– if Dl = "true" and Dop = "true" and Ddyn = "false", then set Ncora to "true";

– if Dl = "true" and Dop = "true" and Ddyn = "true", then set Nopra to "true";

– if Dl = "true" and (Dcsr = "true" and Dcrr = "true" and Check-ready-directions is absent or set to "true") or (Dcsr = "true" and Dcrr = "false"), then set Ntch to "true";

– if Dee = "true" and Dl = "true", then set Neera to "true";

– if Dl = "true" and Dcsr = "true" and either Dcrr = "false" or Ntch = "true", then set Dex to "true" and Nex to "true".

[x] Action related to TP-PREPARE indication, TP-READY indication, or TP-ONE-PHASE indication on the *superior dialogue*:

– if Dsup is "true", then set Npa to "true".

[y] Action related to TP-COMMIT request:

– set Nt to "true".

[z] Action related to TP-COMMIT-COMPLETE indication or TP-ROLLBACK-COMPLETE indication or TP-UNKNOWN-COMPLETE indication

If this is the last dialogue to be processed for the indication, then:

– set Nt to "false";

– set Nsubop to "false";

– set Nsupop to "false";

– set Nsupco to "false";

– set Ntch to "false";

– set Nopr to "false";

– set Nror to "false";

– set Nopra to "false";

– set Neer to "false";

– set Nunki to "false";

– set Nsopex to "false";

– set Nex to "false";

– set N2exp to "false";

– if there exists at least one dialogue with Dl set to "true", then:

  – if on the dialogue where Dsup is "true", Dl is "false", then set Nr to "true" and set Npa to "true";

  – if Nr = "false" and Nip = "true", then set Npa to "true";

  – if there exists a *subordinate dialogue* with Dl = "true" and Dco = "false", then set Nsubop to "true";

  – if there exists a *superior dialogue* where Dl is "true" and Dco is "true", then set Nsupco to "true";

  – if there exists a *superior dialogue* where Dl is "true" and Dco is "false", then set Nsupop to "true";

  – if there exists a dialogue where Dl = "true" and Dco = "true", then set Ncora to "true";

  – if there exists a *superior dialogue* where Dl = "true" and Dop = "true" and Ddyn = "false", then set Ncora to "true";

– if there exists a *subordinate dialogue* where Dl = "true" and Dop = "true" and Ddyn = "false", then set Nopra to "true";

– if there exists a dialogue where Dl = "true" and Dop = "true" and Ddyn = "true", then set Nopra to "true";

– if there exists a dialogue where D2exp = "true", then set N2exp to "true";

– if there exists a dialogue where Dex = "true", then set Nex to "true";

– if there exists a dialogue where Dsopex = "true", then set Nsopex to "true";

– if Nr = "false" and on the *superior dialogue* (Dcsr = "true" and Dcrr = "true" and Check-ready-directions is absent or set to "true") or (Dcsr = "true" and Dcrr = "false"), then set Ntch to "true".

[A] Action related to TP-READ-ONLY request:

– set Nt to "true";

– set Nror to "true".

[B] Action related to TP-READ-ONLY indication:

– set Droi to "true";

– if either of a TP-EARLY-EXIT indication or a TP-READ-ONLY indication has been received for all *coordinated subordinate dialogues* and Nee is "true" and Nr is "false", then set Neera to "true";

– if Du = "true" and of = "false" and Dg = "false", then set Dl to "false";

– if Du = "true" and D2exp = "true", then set D2exp to "false" and N2exp to "false";

– if Du = "true" and Dex = "true", then set Dex to "false" and Nex to "false";

– if Du = "true" and Dsopex = "true", then set Dsopex to "false" and Nsopex to "false";

– if there exists a dialogue where D2exp = "true", then set N2exp to "true";

– if there exists a dialogue where Dex = "true", then set Nex to "true";

– if there exists a dialogue where Dsopex = "true", then set Nsopex to "true".

[C] Action related to TP-ONE-PHASE request:

– set Nt to "true";

– set Nopr to "true".

[D] Action related to TP-ONE-PHASE indication:

– set Dopi to "true";

– if Du = "true" and D2exp = "true", then set D2exp to "false" and N2exp to "false";

– if Du = "true" and Dex = "true", then set Dex to "false" and Nex to "false";

– if there exists a dialogue where D2exp = "true", then set N2exp to "true";

– if there exists a dialogue where Dex = "true", then set Nex to "true".

[E] Action related to TP-COMPLETION-REPORT indication:

– set Dcra to "false".

[F] Action related to TP-COMMIT indication for a *subordinate dialogue*:

– if Dcd = "true", then set Dcra to "true".

[G] Action related to TP-EARLY-EXIT request:

– set Nt to "true";

– set Neer to "true".

[H] Action related to TP-EARLY-EXIT indication:

– set Deei to "true";

– if either of a TP-EARLY-EXIT indication or a TP-READ-ONLY indication has been received for all *coordinated subordinate dialogues* and Nee = "true" and Nr = "false", then set Neera to "true";

– if Du = "true", set Dl to "false";

– if Du = "true" and D2exp = "true", then set D2exp to "false" and N2exp to "false";

– if Du = "true" and Dex = "true", then set Dex to "false" and Nex to "false";

– if Du = "true" and Dsopex = "true", then set Dsopex to "false" and Nsopex to "false";

– if there exists a dialogue where D2exp = "true", then set N2exp to "true";

– if there exists a dialogue where Dex = "true", then set Nex to "true";

– if there exists a dialogue where Dsopex = "true", then set Nsopex to "true".

[J] Action related to TP-UNKNOWN indication:

– set Nunki to "true";

– set Dhr to "false".

[K] Action related to TP-UNKNOWN-COMPLETE indication:

– set Nunki to "false".

[L] Action related to TP-READY indication on the *superior dialogue*:

– if Du = "true" and Dex = "true", then set Dex to "false" and Nex to "false";

– if there exists a dialogue where Dex = "true", then set Nex to "true".

A.5 Node Crash

The TPSUI is aware of a node crash through local means. If the node is in the active or read-only or early-exit or one-phase state, the state tables will not be reinstantiated and the bound data under the control of the TPSUI will be placed into the initial state through local means. If the node is in the ready or decided (commit) state, the state tables will be reinstantiated in state 20 for all dialogues having a coordination level of commitment, with the following predicates set to true: Db, Dhr, Dl, Dsup (for the *superior dialogue*; false for all other dialogues), Npa, Nt.

A.6 Keys

In the state table:

– each column (except the left one) represents a state;

– each row represents a service primitive (optionally with attributes).

Below the number of each state, the predicates that are always true in this state are listed. The predicates do not have to be tested within the state table. This information is given to help the reader.

The intersection of a row and a column represents all transitions that may be fired in the given state for the given service primitives. This intersection contains zero, one or more subcells that represent each a transition.

Example 1:

	2
TP-DEFERRED-END-DIALOGUE req	^Dsup ^Dl ^De 2 [g]

The left column lists service primitives which are the input events. When the service primitive is followed by an "*" the event shall be applied to all *coordinated dialogues*; if there is not at least one *coordinated dialogue*, then and only then, is the event applied to the node contingency table.

In the upper part of the subcell, some variables may be listed. The transition is fired only if all the variables listed in the cell have the required value; in the example above, Dl shall be "true" and both Dsup and De shall be "false" (^Dsup means NOT Dsup) in order for the transition to fire.

In the middle of the subcell a number indicates the resulting state; in the example, the resulting state is 2.

If the resulting state is followed by an "*", then this is the resulting state of all *coordinated dialogues*; in this case the actions are performed only once.

At the lower part of the cell, the action(s) to be performed are listed in square brackets. Zero, one, or more actions may be performed when a transition is fired. These actions shall be performed left to right.

Example 2:

	11 ^Dl
TP-HANDSHAKE indication	Dsup Dh Dsh 10
	^Dsup Dsh 10 [o][p]

For a given state and a given service primitive, if more than one subcell appears, the upper one shall be considered first. At most one transition shall be fired for each intersection. If no transition fires, it is treated as a blank intersection. In the example above, if Dsup is "false" and Dsh is "true", the transition in the bottom subcell fires, the actions [o] and [p] are executed and the resulting state is 10.

A.7 Blank intersections

For a given state and a given service primitive, if no transition applies, i.e. the relevant intersection is blank, then the issuance of the service primitive is illegal. If the service primitive is a request or a response, then any effect is a local matter (typically, a local warning will be issued).

NOTE – See ITU-T Rec. X.862 | ISO/IEC 10026-3 for information on why blank intersections involving indications and confirms shall not occur.

A.8 Node Contingency Table

This table is utilized if and only if the following conditions are true:

– there is a transaction at the TPSUI;

– there are no longer any *coordinated dialogues*.

NOTE – This can only occur if the node was a root node in the transaction and a TP-READ-ONLY indication or a TP-EARLY-EXIT indication has been issued for all dialogues that were involved in the transaction.

A.8.1 Node Contingency Table related variables

A number of variables are used exclusively in the Node Contingency Table. The name of these variables starts with an "X" to distinguish them from Node-related variables; they are the following:

– **Xcia** (**C**ompletion **I**ndication **A**llowed): This variable, when set to "true", indicates that the *transaction is in the termination phase*, a TP-DONE request is not owed, and that a TP-COMMIT-COMPLETE indication or a TP-ROLLBACK-COMPLETION indication may be issued.

– **Xcri** (**C**ompletion **R**equest **I**ssued): This variable, when set to "true", indicates that a TP-COMMIT request or a TP-ONE-PHASE request has been issued by a TPSUI with no *coordinated dialogues*, and that a TP-COMMIT indication or a TP-ROLLBACK indication may be issued.

– **Xdo** (**D**one **O**wed): This variable, when set to "true", indicates that a TP-DONE request is owed from a TPSUI with no *coordinated dialogues* before a TP-COMMIT-COMPLETE indication or a TP-ROLLBACK-COMPLETE indication can be issued.

– **Xrbck** (**R**oll**B**a**CK**): This variable, when set to "true", indicates that the current transaction is being rolled back.

A.8.2 Node Contingency Table

If one of the services below is invoked when there is no *coordinated dialogue* remaining, then the "Predicates" are checked to determine whether the service is valid. Then the "Values to set" actions are implemented.

Service	Predicates	Values to set
TP-COMMIT req	Nr, ^Nt, Npa, Ncora	Nt to "true", Xcri to "true".
TP-COMMIT ind	Nr, Nt, Xcri, ^Xdo, ^Xrbck	Xdo to "true".
TP-DONE req	Nr, Nt, Xdo	Nfa to "false", Npa to "true" ; Xdo to "false" ; Xcia to "true".
TP-COMMIT-COMPLETE ind	Nr, Nt, Xcri, Xcia	Nfa to "false", Npa to "false"; Xdo to "false"; Xcia to "true".
TP-ROLLBACK req	Nr, ^Nt	Xdo to "true", Xrbck to "true".
TP-ROLLBACK ind	Nr, (^Nt or Xcri)	Xdo to "true", Xrbck to "true", Xcri to "false".
TP-ROLLBACK-COMPLETE ind	Nr, Nt, ^Xdo, Xrbck, Xcia	Nr to "false", Nt to "false", Xrbck to "false", Npa to "false".
TP-ONE-PHASE req	Nr, ^Nt, Npa, Nopra	Nt to "true", Xcri to "true".

A.9 Service State Table

Table A.1 contains the OSI TP Service State Table.

Table A.1/X.861 (sheet 1 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-DATA req		Dsup ^Dcr 2 [b]					Dsup ^Dcr Dsh 7 [b]			
		^Dsup 2					^Dsup Dsh 7			
TP-DATA ind		Dsup Dsh 2	Dsup 3			Dsup Dsh 6				
		^Dsup ^Dcr Dsh 2 [o]	^Dsup ^Dcr 3 [o]			^Dsup ^Dcr Dsh 6 [o]				
TP-BEGIN-DIALOGUE req ((Commit FU not selected and One-phase Commit FU not selected) or (Unchained Transactions and Begin-Transaction = "false")) *ie coord level = "none"*	^Ndial 2 [v]									
	Nsupest 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU not selected) and ((Chained Transactions) or (Unchained Transactions and Begin-Transaction = "true") and (neither Read-only or Early-exit selected)) *ie coord level = "commitment" (static)* *and will be two-phase expected branch*	^Ndial ^Nt ^Nsopex 2 [v]									
	Nsupest ^Nt ^Nsopex 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU not selected) and ((Chained Transactions) or (Unchained Transactions and Begin-Transaction = "true") and (Read-only and/or Early-exit selected)) *ie coord level = "commitment" (static)* *and may be two-phase expected branch* *depending on tree checking*	^Ndial ^Nt Ntch ^Nsopex 2 [v]									
	Nsupest ^Nt Ntch ^Nsopex 2 [v]									
	^Ndial ^Nt ^Ntch 2 [v]									
	Nsupest ^Nt ^Ntch 2 [v]									

Table A.1/X.861 (sheet 2 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
									Dsup ^Dcr 19 [b] ^Dsup 19						
Dsup Dsh 11 ^Dsup Dsh 11 [o]					Dsup 16 ^Dsup ^Dcr 16 [o]										

Table A.1/X.861 (sheet 3 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-BEGIN-DIALOGUE req ((Commit FU not selected) and (One-phase Commit FU selected and ((Chained Transactions) or (Unchained Transactions and Begin-Transaction = "true") and (Read-only and/or Early-exit selected))	^Ndial ^Nt Ntch ^N2exp ^Nex 2 [v]									
	Nsupest ^Nt ^Nsupco ^Nsupop Ntch ^N2exp ^Nex 2 [v]									
	^Ndial ^Nt ^Ntch 2 [v]									
	Nsupest ^Nt ^Nsupco ^Nsupop ^Ntch 2 [v]									
ie coord level = "one-phase commitment" *and may be static one-phase exclusive* *branch depending on whether tree checking*										
TP-BEGIN-DIALOGUE req ((Commit FU not selected) and (One-phase Commit FU selected and ((Chained Transactions) or (Unchained Transactions and Begin-Transaction = "true") and (neither Read-only or Early-exit selected))	^Ndial ^Nt ^N2exp ^Nex 2 [v]									
	Nsupest ^Nt ^Nsupco ^Nsupop ^N2exp ^Nex 2 [v]									
ie coord level = "one-phase commitment" *and will be static one-phase exclusive branch*										
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (none of Read-only or Early-exit or One-phase selected) and (Subordinate-may-send-ready = "true") and (Check-ready-directions = "true" or absent) *ie coord level = "commitment" (dynamic)* *and two-phase expected, not exclusive*	^Ndial ^Nt ^Nsopex 2 [v]									
	Nsupest ^Nt ^Nsopex 2 [v]									

Table A.1/X.861 (sheet 4 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 5 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (none of Read-only or Early-exit or One-phase selected) and (Subordinate-may-send-ready = "true") and (Check-ready-directions = "false") *ie coord level = "commitment" (dynamic)* *and two-phase expected, not exclusive*	^Ndial ^Nt ^Ntch ^Nsopex 2 [v] Nsupest ^Nt ^Ntch ^Nsopex 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (Read-only and/or Early-exit and/or One-phase selected) and (Subordinate-may-send-ready = "true") and (Check-ready-directions = "true" or absent)) *ie coord level = "commitment" (dynamic)* *and may be two-phase expected, not exclusive* *depending on whether tree checking*	^Ndial ^Nt Ntch ^Nsopex 2 [v] Nsupest ^Nt Ntch ^Nsopex 2 [v] ^Ndial ^Nt ^Ntch 2 [v] Nsupest ^Nt ^Ntch 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (Read-only and/or Early-exit and/or One-phase selected) and (Subordinate-may-send-ready = "true") and (Check-ready-directions = "false")) *ie coord level = "commitment" (dynamic)* *and not two-phase expected, not exclusive*	^Ndial ^Nt ^Ntch 2 [v] Nsupest ^Nt ^Ntch 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (none of Read-only or Early-exit selected) & (Subordinate-may-send-ready = "false") and (Check-ready-directions = "true" or absent) *ie coord level = "commitment" (dynamic)* *and not two-phase expected, but exclusive*	^Ndial ^Nt ^Nex 2 [v] Nsupest ^Nt ^Nex 2 [v]									

Table A.1/X.861 (sheet 6 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 7 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (none of Read-only or Early-exit selected)	^Ndial ^Nt ^Ntch ^Nex 2 [v]									
(Subordinate-may-send-ready = "false") and (Check-ready-directions = "false") *ie coord level = "commitment" (dynamic)* *and not two-phase expected, but exclusive*	Nsupest ^Nt ^Ntch ^Nex 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and (Read-only and/or Early-exit selected)	^Ndial ^Nt Ntch ^Nex 2 [v]									
(Subordinate-may-send-ready = "false") and (Check-ready-directions = "true" or absent))	Nsupest ^Nt Ntch ^Nex 2 [v]									
	^Ndial ^Nt ^Ntch 2 [v]									
ie coord level = "commitment" (dynamic) *and not two-phase expected, may be exclusive* *depending on whether tree checking*	Nsupest ^Nt ^Ntch 2 [v]									
TP-BEGIN-DIALOGUE req ((Commit FU selected and Dynamic Commit FU selected) and (Unchained Transactions and Begin-Transaction = "true") and	^Ndial ^Nt ^Ntch 2 [v]									
Read-only and/or Early-exit selected) (Subordinate-may-send-ready = "false") and (Check-ready-directions = "false")) *ie coord level = "commitment" (dynamic)* *and not two-phase expected, not exclusive*	Nsupest ^Nt ^Ntch 2 [v]									
TP-BEGIN-DIALOGUE ind (Polarized Control selected)	^Ndial 3 [w]									
TP-BEGIN-DIALOGUE ind (Shared Control selected)	^Ndial 2 [w]									
TP-BEGIN-DIALOGUE rsp (Result = "accepted")		Dsup Dcr 2 [c]	Dsup Dcr 3 [c]		Dsup Dcr 5 [c]		Dsup Dcr 7 [c]			

Table A.1/X.861 (sheet 8 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
			Dsup Dcr 14 [c]					Dsup Dcr 18 [c]	Dsup Dcr 19 [c]				Dsup Dcr ^Db 23 [c]		

Table A.1/X.861 (sheet 9 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates **Event**			^Dsh	^Dsh	^Dsh	Dh	Dh	Dsh Dh	Dsh Dh ^Dl	Dsh Dh ^Dl
TP-BEGIN-DIALOGUE rsp (Result = "rejected")		^Nrn Dsup 1 [a]	^Nrn Dsup 1 [a]		^Nrn Dsup 1 [a]		^Nrn Dsup 1 [a]			
TP-BEGIN-DIALOGUE cnf (Result = "accepted")		^Dsup Dcr 2 [c]	^Dsup Dcr 3 [c]	^Dsup Dcr 4 [c]		^Dsup Dcr 6 [c]				
TP-BEGIN-DIALOGUE cnf (Result = "rejected" and Rollback = "false")		^Dsup ^Dl ^Da 1 [a]	^Dsup ^Dl ^Da 1 [a]	^Dsup ^Dl ^Da 1 [a]		^Dsup ^Dl ^Da 1 [a]				
		^Dsup Dl ^Da 25 [p]	^Dsup Dl ^Da 25 [p]	^Dsup Dl ^Da 25 [p]		^Dsup Dl ^Da 25 [p]				
TP-BEGIN-DIALOGUE cnf (Result = "rejected" and Rollback = "true")										
TP-END-DIALOGUE req (Confirmation = "true")		^Dl ^Dcr 11 [b]								
TP-END-DIALOGUE req (Confirmation = "false")		^Dl ^Dcr 1 [a]								
TP-END-DIALOGUE ind (Confirmation = "true")		^Dl ^Dcr Dsh 12 [o]	^Dl ^Dcr 12 [o]	^Dl ^Dcr 2 [o]		^Dl ^Dcr Dsh 9 [o]				
TP-END-DIALOGUE ind (Confirmation = "false")		^Dl ^Dcr Dsh 1 [a]	^Dl ^Dcr 1 [a]	^Dl ^Dcr 1 [a]		^Dl ^Dcr Dsh 1 [a]				
TP-END-DIALOGUE rsp										
TP-END-DIALOGUE cnf										
TP-U-ERROR req		Dsup ^Dcr 2 [b]	Dsup ^Dcr 4 [b]			Dsh 6	Dsup ^Dcr 2 [b]	6	6	11
		^Dsup 2	^Dsup 4				^Dsup 2			

Table A.1/X.861 (sheet 10 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
	^Nrn Dsup 1 [a]		^Nrn Dsup 1 [a]				^Nrn Dsup 1 [a]	^Nrn Dsup 1 [a]	^Nrn Dsup 1 [a]				^Nrn Dsup ^Db 1 [a]		
		^Dsup Dcr 13 [c]		^Dsup Dcr 15 [c]	^Dsup Dcr 16 [c]								^Dsup Dcr ^Db 23 [c]	^Dsup Dcr ^Db 24 [c]	
^Dsup ^Da 1 [a]		^Dsup ^Dl ^Da 1 [a] ^Dsup Dl ^Da 25 [p]		^Dsup ^Da 25 [p][n]	^Dsup ^Da 25 [p][n]								^Dsup ^Db ^Da 23 [p][l]	^Dsup ^Db ^Da 23 * [p][l]	
										^Dsup ^Da ^Db 23 * [p][f]					
Dsh 1 [a]															
	1 [a]														
1 [a]															
	Dsup 2 [b] ^Dsup 2		Dsup ^Dcr 2 [b] ^Dsup 2												

Table A.1/X.861 (sheet 11 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-U-ERROR ind		Dsup ^Dsh 5	Dsup 3			Dsup ^Dsh 3				
		Dsup Dsh 2				Dsup Dsh 2				
		^Dsup ^Dcr ^Dsh 5 [o]	^Dsup ^Dcr 3 [o]			^Dsup ^Dcr ^Dsh 3 [o]				
		^Dsup ^Dcr Dsh 2 [o]				^Dsup ^Dcr Dsh 2 [o]	7	7	12	7
TP-U-ABORT req		Dsup ^Dcr ^Dl 1 [a]	Dsup ^Dcr ^Dl 1 [a]	^Dl 1 [a]	Dsup ^Dcr ^Dl 1 [a]	^Dl 1 [a]	Dsup ^Dcr ^Dl 1 [a]	^Dl 1 [a]	1 [a]	1 [a]
		Dsup ^Dcr Dl 23 * [p][f]	Dsup ^Dcr Dl 23 * [p][f]	Dl 23 * [p][f]	Dsup ^Dcr 23 * [p][f]	Dl 23 * [p][f]	Dsup ^Dcr 23 * [p][f]	Dl 23 * [p][f]		
		^Dsup ^Dl 1 [a]	^Dsup ^Dl 1 [a]		^Dsup ^Dl 1 [a]		^Dsup ^Dl 1 [a]			
		^Dsup Dl 23 * [p][f]	^Dsup Dl 23 * [p][f]		^Dsup Dl 23 * [p][f]		^Dsup Dl 23 * [p][f]			
TP-U-ABORT ind (Rollback = "false")		Dsup ^Dl 1 [a]	Dsup ^Dl 1 [a]	Dsup ^Dl 1 [a]		Dsup ^Dl 1 [a]				
		^Dsup ^Dcr ^Dl 1 [a]	^Dsup ^Dcr ^Dl 1 [a]	^Dsup ^Dcr ^Dl 1 [a]	^Dl 1 [a]	^Dsup ^Dcr ^Dl 1 [a]	^Dl 1 [a]	^Dl 1 [a]	1 [a]	1 [a]
		Dx ^Dcr 25 [p]	Dx ^Dcr 25 [p]	Dx ^Dcr 25 [p]	Dx 25 [p]	Dx ^Dcr 25 [p]	Dx 25 [p]	Dx 25 [p]		
		Dar ^Dcr 25 [p]	Dar ^Dcr 25 [p]	Dar ^Dcr 25 [p]		Dar ^Dcr 25 [p]				

Table A.1/X.861 (sheet 12 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
Dsup ^Dsh 3		Dsup 3													
Dsup Dsh 2															
^Dsup ^Dsh 3 [o]		^Dsup ^Dcr 3 [o]													
^Dsup Dsh 2 [o]															
1 [a]	1 [a]	^Dl 1 [a]	Dsup ^Dcr ^Dl 1 [a]								Nfa ^Db ^Du ^De 21 [p][e]				
		Dl 23 * [p][f]	Dsup ^Dcr Dl 23 * [p][f]	23 * [p][f]	23 * [p][f]	23 * [p][f]	23 * [p][f]	Dsup ^Dcr 23 * [p][f]	Dsup ^Dcr 23 * [p][f]	Nfa ^Db 20 [p]	Nfa ^Db Du ^De 21 [p]		^Db 23 [p]		
			^Dsup ^Dl 1 [a]								Nfa ^Db De 21 [p]				
			^Dsup Dl 23 * [p][f]					^Dsup 23 * [p][f]	^Dsup 23 * [p][f]						
1 [a]	1 [a]	Dsup ^Dl 1 [a]								Dsup Nroq 20 [p][l]	^Db ^Du ^De 21 [p][l][j]	^Db ^Du ^De 21 * [p][l][j]		Dsup ^Db 23 * [p][l]	
		^Dsup ^Dcr ^Dl 1 [a]	^Dl 1 [a]				^Dsup Droi 1 [a]			Dsup Neer 20 [p][l]	^Db Du ^De 21 [p][l]	^Db Du ^De 21 * [p][l]		^Dsup ^Dcr ^Db 23 * [p][l]	
		Dx ^Dcr 25 [p]	Dx 25 [p]	Dx ^Dcr 25 [p][n]	Dx ^Dcr 25 [p][n]		^Dsup Deei 1 [a]				^Db De 21 [p][l]	^Db De 21 * [p][l]	^Db 23 [p]		
		Dar ^Dcr 25 [p]													

Table A.1/X.861 (sheet 13 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-U-ABORT ind (Rollback = "true")		Dsup Dl 23 * [p][f]	Dsup Dl 23 * [p][f]	Dsup Dl 23 * [p][f]	Dl 23 * [p][f]	Dsup Dl 23 * [p][f]	Dl 23 * [p][f]	Dl 23 * [p][f]		
		^Dsup ^Dcr Dl 23 * [p][f]	^Dsup ^Dcr Dl 23 * [p][f]	^Dsup ^Dcr Dl 23 * [p][f]		^Dsup ^Dcr Dl 23 * [p][f]				
TP-P-ABORT ind (Rollback = "false")		^Dl 1 [a]	^Dl 1 [a]	^Dl 1 [a]	^Dl 1 [a]	^Dl 1 [a]	^Dl 1 [a]	^Dl 1 [a]	1 [a]	1 [a]
		Dsup Dcr Dl 1 [a]	Dsup Dcr Dl 1 [a]		Dsup Dcr Dl 1 [a]		Dsup Dcr Dl 1 [a]			
		Dx 25 [p]	Dx 25 [p]	Dx 25 [p]	Dx 25 [p]	Dx 25 [p]	Dx 25 [p]	Dx 25 [p]		
TP-P-ABORT ind (Rollback = "true" Diagnostic ^ = "begin-tr.-reject")		Dsup ^Dcr Dl 23 * [p][f]	Dsup ^Dcr Dl 23 * [p][f]	Dl 23 * [p][f]	Dsup ^Dcr Dl 23 * [p][f]	Dl 23 * [p][f]	Dsup ^Dcr Dl 23 * [p][f]	Dl 23 * [p][f]		
		^Dsup Dl 23 * [p][f]	^Dsup Dl 23 * [p][f]		^Dsup Dl 23 * [p][f]		^Dsup Dl 23 * [p][f]			
TP-P-ABORT ind (Rollback = "true" Diagnostic = "begin-tr.-reject")										
TP-GRANT-CONTROL req		Dsup ^Dcr ^Dsh 3 [b]			Dsup ^Dcr 3 [b]					
		^Dsup ^Dsh 3			^Dsup 3					
TP-GRANT-CONTROL ind			Dsup 2	Dsup 2						
			^Dsup ^Dcr 2 [o]	^Dsup ^Dcr 2 [o]						

Table A.1/X.861 (sheet 14 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
		Dsup Dl 23 * [p][f]		Dsup 23 * [p][f]	Dsup 23 * [p][f]										
		^Dsup ^Dcr Dl 23 * [p][f]	^Dcr 23 * [p][f]	^Dsup ^Dcr 23 * [p][f]	^Dsup ^Dcr 23 * [p][f]	23 * [p][f]	^Droi ^Deei 23 * [p][f]	23 * [p][f]	23 * [p][f]	^Db 23 * [p][f]					
1 [a]	1 [a]	^Dl 1 [a]	^Dl 1 [a]				^Dsup Droi 1 [a]				^Db ^Du ^De 21 [p][l][e]	^Db ^Du ^De 21 * [p][l][e]	^Db Dsup Dcr 1 [a]		
			Dsup Dcr Dl 1 [a]				^Dsup Deei 1 [a]				^Db Du ^De 21 [p][l]	^Db Du ^De 21 * [p][l]	^Db Dsup ^Dcr 23 [p]		
		Dx 25 [p]	Dx 25 [p]	Dx 25 [p][n]	Dx 25 [p][n]					^Db 20 [p][l]	^Db De 21 [p][l]	^Db De 21 * [p][l]	^Db ^Dsup 23 [p]	^Db 23 * [p][l]	
		Dl 23 * [p][f]	Dsup ^Dcr Dl 23 * [p][f]	23 * [p][f]	23 * [p][f]	23 * [p][f]	^Droi ^Deei 23 * [p][f]	23 * [p][f]	23 * [p][f]	^Db 23 * [p][f]					
			^Dsup Dl 23 * [p][f]												
										^Db Dx 23 * [p][f]					

Table A.1/X.861 (sheet 15 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates **Event**			^Dsh	^Dsh	^Dsh	Dh	Dh	Dsh Dh	Dsh Dh ^Dl	Dsh Dh ^Dl
TP-REQUEST-CONTROL req			Dsup ^Dcr 3 [b] ^Dsup 3				Dsup ^Dcr ^Dsh 7 [b] ^Dsup ^Dsh 7			
TP-REQUEST-CONTROL ind		Dsup ^Dsh 2 ^Dsup ^Dcr ^Dsh 2 [o]				Dsup ^Dsh 6 ^Dsup ^Dcr ^Dsh 6 [o]				
TP-HANDSHAKE req		Dsup ^Dcr Dh 6 [b] ^Dsup Dh 6					Dsup ^Dcr Dsh 8 [b] ^Dsup Dsh 8			
TP-HANDSHAKE ind		Dsup Dh Dsh 7 ^Dsup ^Dcr Dh Dsh 7 [o]	Dsup Dh 7 ^Dsup ^Dcr Dh 7 [o]	Dsup Dh 2 ^Dsup ^Dcr Dh 2 [o]		Dsup Dsh 8 ^Dsup ^Dcr Dsh 8 [o]				
TP-HANDSHAKE rsp							Dsup ^Dcr ^Dsh 3 [b] Dsup ^Dcr Dsh 2 [b] ^Dsup ^Dsh 3 ^Dsup Dsh 2	6		11
TP-HANDSHAKE cnf						Dsup 2 ^Dsup ^Dcr 2 [t]		7	12	

Table A.1/X.861 (sheet 16 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
Dsup Dh Dsh 10															
^Dsup ^Dcr Dh Dsh 10 [o]															

Table A.1/X.861 (sheet 17 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates **Event**			^Dsh	^Dsh	^Dsh	Dh	Dh	Dsh Dh	Dsh Dh ^Dl	Dsh Dh ^Dl
TP-HDSKE&GRANT-CTL req		Dsup ^Dcr Dh ^Dsh 13 [b] ^Dsup Dh ^Dsh 13								
TP-HDSKE&GRANT-CTL ind			Dsup Dh 14 ^Dsup ^Dcr Dh 14 [o]	Dsup Dh 2 ^Dsup ^Dcr Dh 2 [o]						
TP-HDSKE&GRANT-CTL rsp										
TP-HDSKE&GRANT-CTL cnf										

Table A.1/X.861 (sheet 18 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
			Dsup ^Dcr 2 [b] ^Dsup 2												
		Dsup 3 ^Dsup ^Dcr 3 [t]													

Table A.1/X.861 (sheet 19 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-BEGIN-TRANSACTION req (Check-ready-directions absent) or (Check-ready-directions = "true")		^Nt ^Dsup Du ^Dl Dco ^Ddyn ^Dres ^Nsopex 2 [q]								
		^Nt ^Dsup Du ^Dl Dco ^Ddyn Ntch ^Nsopex 2 [q]								
		^Nt ^Dsup Du ^Dl Dco ^Ddyn ^Ntch Dres 2 [q]								
		^Nt ^Dsup Du ^Dl Dop ^Dco ^Dres ^Nsupco ^Nsupop ^N2exp ^Nex 2 [q]								
		^Nt ^Dsup Du ^Dl Dop ^Dco Ntch ^Nsupco ^Nsupop ^N2exp ^Nex 2 [q]								

Table A.1/X.861 (sheet 20 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 21 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
		^Nt ^Dsup Du ^Dl Dop ^Dco ^Nsupco ^Nsupop ^Ntch Dres 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn Dcrr ^Dores ^Nsopex 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn Dcrr Dores Ntch ^Nsopex 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn Dcsr ^Dcrr ^Dres ^Nex 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn Dcsr ^Dcrr Ntch ^Nex 2 [q]								

Table A.1/X.861 (sheet 22 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 23 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
		^Nt ^Dsup Du ^Dl Ddyn ^Ntch Dcrr Dores 2 [g]								
		^Nt ^Dsup Du ^Dl Ddyn ^Ntch ^Dcrr Dres 2 [g]								

Table A.1/X.861 (sheet 24 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 25 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-BEGIN-TRANSACTION req (Check-ready-directions = "false")		^Nt ^Dsup Du ^Dl Ddyn Dcrr ^Dores ^Ntch ^Nsopex 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn Dcsr ^Dcrr ^Dres ^Ntch ^Nex 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn ^Ntch Dcrr Dores 2 [q]								
		^Nt ^Dsup Du ^Dl Ddyn ^Ntch ^Dcrr Dres 2 [q]								
TP-BEGIN-TRANSACTION ind		^Nr Dsup Du ^Dl Dsh 2 [u]	^Nr Dsup Du ^Dl 3 [u]	^Nr Dsup Du ^Dl 4 [u]		^Nr Dsup Du ^Dl Dsh 6 [u]				
TP-DEFERRED-END-DIALOGUE req		^Dsup Dl ^De 2 [g]								

Table A.1/X.861 (sheet 26 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 27 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates **Event**			^Dsh	^Dsh	^Dsh	Dh	Dh	Dsh Dh	Dsh Dh ^Dl	Dsh Dh ^Dl
TP-DEFERRED-END-DIALOGUE ind		Dsup Dl ^De Dsh 2 [g]	Dsup Dl ^De 3 [g]	Dsup Dl ^De 4 [g]		Dsup Dl ^De Dsh 6 [g]				
TP-DEFERRED-GRANT-CONTROL req		^Dsup Dl ^Dsh ^De ^Dg 2 [h]								
TP-DEFERRED-GRANT-CONTROL ind			Dsup Dl ^De ^Dg 3 [h]	Dsup Dl ^De ^Dg 4 [h]						

Table A.1/X.861 (sheet 28 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl

Table A.1/X.861 (sheet 29 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-PREPARE req (Data-Permitted = "false") or absent		Npa ^Dsup Dl ^Dsh 15 [m]								
		Npa ^Dsup Dl Dsh 16 [m]								
		Dsup ^Dcr Npa Dl ^Dco ^Dsh 15 [b]								
		Dsup ^Dcr Npa Dl ^Dco Dsh 16 [b]								
		Dsup ^Dcr Npa Dl Ddyn ^Dsh 15 [b]								
		Dsup ^Dcr Npa Dl Ddyn Dsh 16 [b]								

Table A.1/X.861 (sheet 30 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
									Npa ^Dsup 16.1 [m]						
									Dsup ^Dcr ^Dco 16.1 [b]						
									Dsup ^Dcr Ddyn 16.1 [b]						

Table A.1/X.861 (sheet 31 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates **Event**			^Dsh	^Dsh	^Dsh	Dh	Dh	Dsh Dh	Dsh Dh ^Dl	Dsh Dh ^Dl
TP-PREPARE req (Data-Permitted = "true")		Npa ^Dsup Dl ^Dsh 16 [m]								
		Dsup ^Dcr Npa Dl ^Dco ^Dsh 16 [b]								
		Dsup ^Dcr Npa Dl Ddyn ^Dsh 16 [b]								
TP-PREPARE ind (Data-Permitted = "false") or absent		Dsup Dl Dsh 19 [x]	Dsup Dl 18 [x]							
		^Dsup ^Dcr Dl ^Dco Dsh 19 [t]	^Dsup ^Dcr Dl ^Dco 18 [t]							
		^Dsup ^Dcr Dl Ddyn Dsh 19 [t]	^Dsup ^Dcr Dl Ddyn 18 [t]							
TP-PREPARE ind (Data-Permitted = "true")			Dsup Dl 19 [x]							
			^Dsup ^Dcr Dl ^Dco 19 [t]							
			^Dsup ^Dcr Dl Ddyn 19 [t]							

Table A.1/X.861 (sheet 32 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
					Dsup 16.1 [x] ^Dsup ^Dcr ^Dco 16.1 [t] ^Dsup ^Dcr Ddyn 16.1 [t]										

Table A.1/X.861 (sheet 33 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-READY ind		^Dsup ^Dcr Dl Dco Dcrr Dip Dsh 17 [t]	^Dsup ^Dcr Dl Dco Dcrr Dip 17 [t]							
		Dsup ^Dcr Dl Ddyn Dcrr Dsh 17 [x][L]	Dsup ^Dcr Dl Ddyn Dcrr 17 [x][L]							
TP-COMMIT req *		Nr Dl Dco ^Dcr 20 [m][y]								
		Npa Dsup Dl ^Dcr 20 [b][y]								
		^Nr Npa ^Dsup Dl Dco ^Dcr 20 [m][y]								
TP-COMMIT ind *										
TP-DONE req * (without Heuristic-Report)										

Table A.1/X.861 (sheet 34 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
				^Dsup ^Dcr Dcrr 17 [t]	^Dsup ^Dcr Dcrr 17 [t]	^Dsup ^Dcr Dcrr 17 [t]		^Dsup Ddyn Dcrr 17 [t]	^Dsup Ddyn Dcrr 17 [t]						
				Dsup Ddyn Dcrr 17 [L]	Dsup Ddyn Dcrr 17 [L]	Dsup Dcrr 17 [L]		Dsup ^Dcr Ddyn Dcrr 17 [L]	Dsup ^Dcr Ddyn Dcrr 17 [L]						
							Dsup Nsupop ^Dcr 20 [b][y]								Ncora ^Dtrb 20 [y]
				Dsup Ddyn 20 [y]	Dsup Ddyn 20 [y]		Dsup Ddyn ^Dcr 20 [b][y]	Dsup ^Dcr 20 [b][y]	Dsup ^Dcr 20 [b][y]						
				^Dsup Dco ^Dcr 20 [y]	^Dsup Dco ^Dcr 20 [y]	20 [y]	^Dsup 20 [y]	^Dsup Ddyn 20 [y]	^Dsup Ddyn 20 [y]						
										Dsup ^Neer 21					
										^Dsup ^Neer 21 [t][F]					
											22 [d]		Nfa Dsup ^Dcr 24 [b][d]		
													Nfa ^Dsup 24 [d]		

Table A.1/X.861 (sheet 35 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-DONE req * (with Heuristic-Report)										

Table A.1/X.861 (sheet 36 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
											Npa ^Nror ^Nopr ^Neer 22 [d]		Nfa Npa Dsup ^Dcr ^Nror ^Nopr ^Neer 24 [b][d]		
													Nfa Npa ^Dsup ^Nror ^Nopr ^Neer 24 [d]		

Table A.1/X.861 (sheet 37 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-COMMIT-COMPLETE ind *										

Table A.1/X.861 (sheet 38 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
												^Nfa ^Nunki De 1 [a][n][z]			
												^Nfa ^Nunki Dtrb 25 [n][z]			
												^Nfa ^Nunki Danu 25 [i][n][z]			
												^Nfa ^Nunki Db 1 [a][n][z]			
												^Nfa ^Nunki Dsh 2 [n][s][z]			
												^Nfa ^Nunki Dsup ^Dsh ^Dg 3 [n][s][z]			
												^Nfa ^Nunki Dsup ^Dsh Dg 2 [n][s][z]			
												^Nfa ^Nunki ^Dsup ^Dsh ^Dg 2 [n][s][z]			
												^Nfa ^Nunki ^Deei ^Dsup ^Dsh Dg 3 [n][s][z]			

Table A.1/X.861 (sheet 39 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-ROLLBACK req *		Dsup ^Dcr Dl 23 [b][f]	Dsup ^Dcr Dl 23 [b][f]	Dl 23 [f]	Dsup ^Dcr Dl 23 [b][f]	Dl 23 [f]	Dsup ^Dcr Dl 23 [b][f]	Dl 23 [f]		
		^Dsup Dl 23 [f]	^Dsup Dl 23 [f]		^Dsup Dl 23 [f]		^Dsup Dl 23 [f]			
TP-ROLLBACK ind *		Dl 23 [f]	Dl 23 [f]	Dl 23 [f]	Dl 23 [f]	Dl 23 [f]	Dl 23 [f]	Dl 23 [f]		
TP-ROLLBACK-COMPLETE ind *										
TP-HEURISTIC-REPORT ind										
TP-READ-ONLY req *		Npa Dsup ^Dcr Dl Nro 20 [b][A]								
		^Nr Npa ^Dsup ^Dcr Dl Dco Nro 20 [m][A]								

Table A.1/X.861 (sheet 40 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
		Dl 23 [f]	Dsup ^Dcr Dl 23 [b][f] ^Dsup Dl 23 [f]	Dsup Ddyn 23 [f] ^Dsup 23 [f]	Dsup Ddyn 23 [f] ^Dsup 23 [f]	23 [f]	Dsup Ddyn 23 [b][f] ^Dsup 23 [f]	Dsup ^Dcr 23 [b][f] ^Dsup Ddyn 23 [f]	Dsup ^Dcr 23 [b][f] ^Dsup Ddyn 23 [f]						^Dtrb 23 [f]
		Dl 23 [f]	Dl 23 [f]	23 [f]	23 [f]	23 [f]	23 [f]	23 [f]	23 [f]	23 [f]					Dtrb 23 [f]
														^Nfa Db 1 [a][n][z] ^Nfa Dsh 2 [k][n][z] ^Nfa ^Dsh Dc 2 [k][n][z] ^Nfa ^Dsh ^Dc 3 [k][n][z]	
											Dhr ^Dhcr ^Nunki 21 [r]	Dhr ^Dhcr ^Nunki 22 [r]	^Dcr Dhr ^Dhcr ^Nunki 23 [o][r]	^Dcr Dhr ^Dhcr ^Nunki 24 [o][r]	
				Dsup Nro 20 [A] ^Nr ^Dsup ^Dcr Dco Nro 20 [m][A]	Dsup Nro 20 [A] ^Nr ^Dsup ^Dcr Dco Nro 20 [m][A]	Dsup Nro 20 [A] ^Nr ^Dsup Dco Nro 20 [m][A]	Dsup ^Dcr Nro 20 [b][A] ^Nr ^Dsup Dco Nro 20 [m][A]	Dsup ^Dcr Nro 20 [b][A] ^Nr ^Dsup Dco Nro 20 [m][A]	Dsup ^Dcr Nro 20 [b][A] ^Nr ^Dsup Dco Nro 20 [m][A]						^Dtrb Nro 20 [A]	

Table A.1/X.861 (sheet 41 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates **Event**			^Dsh	^Dsh	^Dsh	Dh	Dh	Dsh Dh	Dsh Dh ^Dl	Dsh Dh ^Dl
TP-READ-ONLY ind		^Dsup ^Dcr Dl Dip Dro Dsh 17 [t][B]	^Dsup ^Dcr Dl Dip Dro 17 [t][B]							
TP-EARLY-EXIT req *		Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]		
TP-EARLY-EXIT ind		^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]		
TP-ONE-PHASE req *		Nr Dl ^Dcr Nopra 20 [m][C] Npa Dsup Dl ^Dcr Nopra 20 [b][C] ^Nr Npa ^Dsup Dl Dco ^Dcr Nopra 20 [m][C]								

Table A.1/X.861 (sheet 42 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
				^Dsup ^Dcr Dro 17 [t][B]	^Dsup ^Dcr Dro 17 [t][B]	^Dsup Dro 17 [t][B]		^Dsup Ddyn Dro 17 [B]	^Dsup Ddyn Dro 17 [B]						
		Dsup ^Dcr Dl Neera 20 [b][G]	Dsup ^Dcr Dl Neera 20 [b][G]					Dsup ^Dcr Neera 17 [b][G]	Dsup ^Dcr Neera 17 [b][G]						
							^Nr ^Dsup Neera 17 [b][G]								
		^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dl Dee 20 [t][H]	^Dsup ^Dcr Dee 20 [t][H]	^Dsup ^Dcr Dee 20 [t][H]										
				Dsup Ddyn Nopra 20 [C]	Dsup Ddyn Nopra 20 [C]	Dsup Ddyn Nopra 20 [C]	Dsup Nsupop ^Dcr Nopra 20 [b][C]								^Dtrb Nopra 20 [C]
							Dsup Ddyn ^Dcr Nopra 20 [b][C]	Dsup Ddyn ^Dcr Nopra 20 [b][C]	Dsup Ddyn ^Dcr Nopra 20 [b][C]						
				^Dsup Ddyn ^Dcr Nopra 20 [m][C]	^Dsup Ddyn ^Dcr Nopra 20 [m][C]	^Dsup ^Dcr Nopra 20 [m][C]	^Dsup Ddyn Nopra 20 [C]	^Dsup Ddyn Nopra 20 [C]	^Dsup Ddyn Nopra 20 [C]						

Table A.1/X.861 (sheet 43 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-ONE-PHASE ind		Dsup Dl Dop ^Ddyn Dsh 17 [x][D]	Dsup Dl Dop ^Ddyn 17 [x][D]							
		Dsup Dl Dop Ddyn Dcrr Dsh 17 [x][D]	Dsup Dl Dop Ddyn Dcrr 17 [x][D]							
		^Dsup ^Dcr Dl Dop Ddyn Dcrr Dip Dsh 17 [t][D]	^Dsup ^Dcr Dl Dop Ddyn Dcrr Dip 17 [t][D]							
TP-UNKNOWN ind *										

Table A.1/X.861 (sheet 44 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
				Dsup Dop Ddyn Dcrr 17 [D]	Dsup Dop Ddyn Dcrr 17 [D]	Dsup Dop Dcrr 17 [D]		Dsup ^Dcr Dop Ddyn Dcrr 17 [D]	Dsup ^Dcr Dop Ddyn Dcrr 17 [D]						
				^Dsup ^Dcr Dop Ddyn Dcrr 17 [t][D]	^Dsup ^Dcr Dop Ddyn Dcrr 17 [t][D]	^Dsup Dop Dcrr 17 [t][D]		^Dsup Dop Ddyn Dcrr 17 [t][D]	^Dsup Dop Ddyn Dcrr 17 [t][D]						
										Nror 21 [J]					
										Neer 21 [J]					
										Nopr 21 [J]					

Table A.1/X.861 (sheet 45 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-UNKNOWN-COMPLETE ind *										

Table A.1/X.861 (sheet 46 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
												^Nfa Dsup ^Neer Nunki De 1 [a][n][z]			
												^Dsup ^Deei Nunki De 1 [a][n][z]			
												^Nfa Nunki Dtrb 25 [n][z]			
												^Nfa Nunki Danu 25 [i][n][z]			
												^Nfa Nunki Db 1 [a][n][z]			
												^Nfa Nunki Dsh 2 [s][z]			
												^Nfa Nunki Dsup ^Dsh ^Dg 3 [s][z]			
												^Nfa Nunki Dsup ^Dsh Dg ^Neer 2 [s][z]			
												^Nfa Nunki ^Dsup ^Dsh ^Dg 2 [s][z]			

Table A.1/X.861 (sheet 47 of 48) – Service State Table

State	1	2	3	4	5	6	7	8	9	10
Predicates			^Dsh	^Dsh	^Dsh			Dsh	Dsh	Dsh
						Dh	Dh	Dh	Dh	Dh
Event									^Dl	^Dl
TP-COMPLETION-REPORT ind (with Severity and Diagnostic^ = "other-provider-rollback")										
TP-COMPLETION-REPORT ind (with Severity and Diagnostic = "other-provider-rollback")										
TP-COMPLETION-REPORT ind (without Severity)										

Table A.1/X.861 (sheet 48 of 48) – Service State Table

11	12	13	14	15	16	16.1	17	18	19	20	21	22	23	24	25
^Dl	^Dl	^Dsh Dh	^Dsh Dh	^Dsh Dl	Dl	Ddyn Dl	Dl	^Dsh Dl	Dl	Dl	Dl	Dl	Dl	Dl	Dl
												^Nfa ^Deei Nunki ^Dsup ^Dsh Dg 3 [s][z]			
													^Dsup ^Dcr Dcd ^Droi ^Dopi ^Deei 23 [E]	^Dsup ^Dcr Dcd ^Droi ^Dopi ^Deei 23 [E]	
													^Dsup Dcd ^Droi ^Dopi ^Deei 23 [E]	^Dsup Dcd ^Droi ^Dopi ^Deei 23 [E]	
											^Dsup ^Dcr Dcd ^Nunki ^Droi ^Dopi ^Deei 21 [E]	^Dsup ^Dcr Dcd ^Nunki ^Droi ^Dopi ^Deei 21 [E]			

End of Table A.1

Annex B

Summary of changes to the third edition

This third edition applies changes from Amendment 1 to ISO/IEC 10026-2 (Edition 2); the amendment covers commitment optimizations – it includes amendments to the Model, Service Definition, and Protocol Specification.

Functional content of Amendment 1

The following facilities are contained in Amendment 1.

a) **Dynamic two-phase commit**: This is introduced as an alternative to the (static) two-phase commit procedures of the original standard. With this approach, and under explicit constraints, either the initiator or the acceptor of a transaction branch can "signal ready" or "order commitment". On the dialogue and transaction levels, restrictions may be imposed for signalling ready and for requiring explicit prepares. This allows the existing "static" two-phase commitment mechanism of ISO/IEC 10026:1992.

b) **Implied prepare**: The "superior-oriented" prepare (TP-PREPARE service) is now optional. A TPSUI can now "signal-ready" based on some implicit semantic from its peer that indicates that the peer will not send any more information that would affect its bound data.

c) **Read-only**: This optional service can be used by a TPSUI to indicate that all its processing is complete, its bound data has not been modified, and the TPSUI has no preference as to whether the transaction commits or is rolled back.

d) **Early-exit**: This optional service can be used by a TPSUI to indicate that it is unable to contribute to the work of the transaction, its bound data has not been modified, and the TPSUI has no preference as to whether the transaction commits or is rolled back.

e) **One-phase commit (dynamic and static)**: Static one-phase commitment is provided for a TPSUI that has no superior in the transaction tree and has no bound data. Static one-phase commitment may be supported by very simple systems.

   Dynamic one-phase commitment can only be used if all transaction branches except one signal read-only or early-exit or one-phase.

   Neither static nor dynamic one-phase commitment require logging since recovery is not relevant.

f) **Cancel**: TP uses the optional CCR service (C-CANCEL) that allows accelerated and non-confirmed rollback as an optional facility. The C-CANCEL service has the same effect as a C-ROLLBACK but without any of its restrictions for sending. A C-CANCEL is followed up with a real C-ROLLBACK at a later time to terminate the transaction branch (no impact on Service Definition).

g) **Recovery Context Handle (RCH) on dialogue initiation**: In ISO/IEC 10026-3 (Edition 2), the RCH is only specified during association establishment. An optional facility allows it to be specified on a dialogue basis (no impact on Service Definition).

h) **Diagnostics on completion**: Diagnostics are now optionally supported on commitment and rollback related services.

i) **Heuristic reports containment**: The sending of heuristic reports can now be suppressed at the dialogue level.

Amendment 1 has introduced some global service constraints, i.e. service constraints at one node which depend on actions at another node. The application is expected to ensure that these constraints are met. These constraints are necessary to avoid collisions of application data with a ready signal, which could otherwise occur using implied prepare with shared control. If the application is incapable or unwilling to "police" these constraints, it should only use polarized control with implied prepares.

ITU-T RECOMMENDATIONS SERIES

Series A	Organization of the work of the ITU-T
Series B	Means of expression: definitions, symbols, classification
Series C	General telecommunication statistics
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	TMN and network maintenance: international transmission systems, telephone circuits, telegraphy, facsimile and leased circuits
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	**Data networks and open system communication**
Series Y	Global information infrastructure
Series Z	Programming languages