INTERNATIONAL STANDARD

ISO/IEC 9805

First edition
1990-11-15

AMENDMENT 2
1992-12-15

Information technology – Open Systems Interconnection – Protocol specification for the Commitment, Concurrency and Recovery service element

AMENDMENT 2: Session mapping changes

Technologies de l'information – Interconnexion de systèmes ouverts – Spécification du protocole pour l'élément de service d'engagement, de concurrence et de reprise

AMENDEMENT 2: Modification de la mise en correspondance avec la session

Reference number
ISO/IEC 9805:1990/Amd. 2:1992 (E)

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Amendment 2 to International Standard ISO/IEC 9805:1990 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*.

Annex C of this International Standard is for information only.

International Organization for Standardization
Case postale 56 • CH-1211 Genève 20 • Switzerland
Printed in Switzerland

Information technology – Open Systems Interconnection – Protocol specification for the Commitment, Concurrency and Recovery service element

Amendment 2: Session mapping changes

Introduction
{NO CHANGE}

1 Scope
{ADD THE FOLLOWING PARAGRAPH TO THE END OF THE CLAUSE}

This International Standard specifies the following protocol versions:

a) Protocol Version 1 which does not make use of the Session Data Separation functional unit.

b) Protocol Version 2 which makes use of the Session Data Separation functional unit to protect data not belonging to the CCR atomic action.

2 Normative references
{ADD THE FOLLOWING REFERENCES TO THIS CLAUSE}

ISO 8822:1988/Amd.5:-[1], *Information processing systems - Open Systems Interconnection - Basic connection oriented presentation service definition, Amendment 5: Additional Session Synchronization Functionality to the Presentation Service User.*

ISO 8326:1987/Amd.4:-[1], *Information processing systems - Open Systems Interconnection - Basic connection oriented session services definition, Amendment 4: Additional Resynchronization Functionality.*

ISO/IEC 9804:1990/Amd.2:-[1], *Information technology - Open Systems Interconnection - Service specification for the Commitment, Concurrency and Recovery service element, Amendment 2: Session Mapping Changes.*

3 Definitions
{NO CHANGE}

4 Symbols and abbreviations

4.1 Data units
{NO CHANGE}

4.2 Types of application-protocol-data-units
{ADD THE FOLLOWING TO THE LIST}

C-INITIALIZE-RI (when using CCR Protocol Version 2)
C-INITIALIZE-RC (when using CCR Protocol Version 2)

---

[1] To be published.

4.3 Other abbreviations
{NO CHANGE}

5 Conventions
{NO CHANGE}

6 Overview of the CCR protocol

6.1 Service Support
{NO CHANGE}

6.2 Constraints on ACSE services
{CHANGE PARAGRAPH 6.2.2 TO READ AS FOLLOWS.}

6.2.2 When establishing the association, the following Presentation and Session Requirements must be specified on the A-ASSOCIATE service:

presentation kernel functional unit

session kernel functional unit

session typed data functional unit

session major synchronize functional unit (when using CCR Protocol Version 1)

session minor synchronize functional unit

session resynchronize functional unit

session data separation functional unit

{ADD A NEW CLAUSE 6.2.4, AS FOLLOWS}

6.2.4 If CCR Protocol Version 2 is being used, the ACSE User information on an A-ASSOCIATE request shall contain the C-INITIALIZE-RI APDU. In addition, the ACSE User information on an A-ASSOCIATE response must contain the C-INITIALIZE-RC APDU.

6.3 Use of the presentation service
{CHANGE PARAGRAPH 6.3.1 TO READ AS FOLLOWS.}

6.3.1 CCR uses the following presentation (ISO 8822) services:

P-DATA
P-TYPED-DATA
P-SYNC-MAJOR (when using CCR Protocol Version 1)
P-SYNC-MINOR
P-RESYNCHRONIZE(restart) (when using CCR Protocol Version 1)
P-RESYNCHRONIZE(abandon) (when using CCR Protocol Version 2)

6.4 Relationship to the session-service and the transport-service
{CHANGE THE FIRST SENTENCE OF CLAUSE 6.4.3 TO READ.}

"If CCR Protocol Version 1 is being used and the Transport-expedited service is used by the session layer, the CCR service-user:"

{CHANGE THE NOTE UNDER CLAUSE 6.4.3 TO READ AS FOLLOWS}

NOTE - With CCR Protocol Version 1, the use of the session resynchronization service for C-ROLLBACK is liable to cause purging of user data outside the atomic action. If the Transport-expedited service is used by session and the above restrictions are not followed, the C-BEGIN can be purged and user-data preceding it. This will not occur if CCR Protocol Version 2 is used.

6.5 Operation of the CCRPM
{NO CHANGE}

{ADD A NEW CLAUSE 6.6, AS FOLLOWS}

6.6 Rules of Extensibility for CCR Protocol Version 2

For the C-INITIALIZE-RI APDU, a receiving CCRPM shall

a) ignore any undefined element;

b) where named bits are used, treat any bit as insignificant when no name is assigned to it.

7 Elements of procedures
{ADD "i)" TO THE PROCEDURE LIST AS FOLLOWS}

i) initialization, if CCR Protocol Version 2 is being used

7.1 Begin branch procedure

7.1.1 Purpose
{NO CHANGE}

7.1.2 APDUs used
{NO CHANGE}

7.1.3 Prerequisite requirements
{NO CHANGE}

7.1.4 Procedure operation
{NO CHANGE}

7.1.5 Use of the C-BEGIN-RI APDU fields
{ADD THE FOLLOWING TEXT BETWEEN THE FIRST AND SECOND PARAGRAPH}

If CCR Protocol Version 2 is being used, then the CCRPM shall represent the "Atomic Action Identifier - Master's Name" parameter of the C-BEGIN request in the abstract syntax by using either the "name" form or the "sender" value of the "side" form of the

"masters-name" field. The latter form may only be used if the Master's Name is the AE-title of the requestor, as passed on the A-ASSOCIATE service used to establish the supporting association.

{ADD THE FOLLOWING TEXT BETWEEN THE THIRD AND FOURTH PARAGRAPHS}

If CCR Protocol Version 2 is being used and if the "masters-name" field in the "atomic-action-identifier" field is the "sender" value of the "side" form, the "Atomic Action Identifier - Master's Name" parameter of the C-BEGIN indication shall be the requestor's AE-title that was passed in the A-ASSOCIATE service used to establish the supporting association.

7.1.6 Use of the C-BEGIN-RC APDU field
{NO CHANGE}

7.1.7 Collisions
{CHANGE THE PARENTHETICAL PHRASE IN THE NOTE TO READ AS FOLLOWS.}

(except when issued with C-ROLLBACK or, when using CCR Protocol Version 1, when issued with C- COMMIT)

7.2 Prepare subordinate procedure
{NO CHANGE}

7.3 Offer commitment procedure
{NO CHANGE}

7.4 Order commitment

7.4.1 Purpose
{NO CHANGE}

7.4.2 APDUs used
{NO CHANGE}

7.4.3 Prerequisite requirements
{CHANGE THE SECOND SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

The requestor shall also either be the owner of the session major/activity token, if using CCR Protocol Version 1, or the synchronize-minor token, if using CCR Protocol Version 2.

7.4.4 Order commitment procedure
{NO CHANGE}

7.4.4.1 C-COMMIT request primitive
{REPLACE THE SECOND SENTENCE BY THE FOLLOWING TEXT}

If CCR Protocol Version 1 is being used, it issues a P-SYNC-MAJOR request primitive with the APDU as a data value of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-SYNC-MINOR request primitive with the APDU as a data value of the primitive's User Data parameter.

7.4.4.2 C-COMMIT-RI APDU
{REPLACE THE FIRST SENTENCE BY THE FOLLOWING TEXT}

If CCR Protocol Version 1 is being used, the accepting CCRPM receives a C-COMMIT-RI APDU from its peer as user data on a P-SYNC-MAJOR indication primitive. If CCR Protocol Version 2 is being used, the accepting CCRPM receives a C-COMMIT-RI APDU from its peer as user data on a P-SYNC-MINOR indication primitive.

7.4.4.3 C-COMMIT response primitive
{REPLACE THE SECOND SENTENCE BY THE FOLLOWING TEXT}

If CCR Protocol Version 1 is being used, it issues a P-SYNC-MAJOR response primitive with the APDU as a data value of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-SYNC-MINOR response primitive with the APDU as a data value of the primitive's User Data parameter.

7.4.4.4 C-COMMIT-RC APDU
{REPLACE THE FIRST SENTENCE BY THE FOLLOWING TEXT}

If CCR Protocol Version 1 is being used, the requesting CCRPM forms a C-COMMIT-RC APDU from its peer as user data on a P-SYNC-MAJOR indication primitive. If CCR Protocol Version 2 is being used, the requesting CCRPM receives a C-COMMIT-RC APDU from its peer as user data on a P-SYNC-MINOR indication primitive.

7.4.5 Use of the C-COMMIT-RI APDU fields
{NO CHANGE}

7.4.6 Use of the C-COMMIT-RC APDU field
{NO CHANGE}

7.4.7 Collision
{NO CHANGE}

7.5 Rollback

7.5.1 Purpose
{NO CHANGE}

7.5.2 APDUs used
{NO CHANGE}

7.5.3 Prerequisite requirements
{NO CHANGE}

7.5.4 Rollback procedure
{NO CHANGE TO THE INTRODUCTORY TEXT UNDER 7.5.4}}

7.5.4.1 C-ROLLBACK request primitive
{CHANGE SECOND SENTENCE OF FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, it issues a P-RESYNCHRONIZE(restart) request primitive with the APDU as a data value of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-RESYNCHRONIZE(abandon)

request primitive with the APDU as a data value of the primitive's User Data parameter.

7.5.4.2 C-ROLLBACK-RI APDU
{CHANGE FIRST SENTENCE OF FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, the accepting CCRPM receives a C-ROLLBACK-RI APDU from its peer as user data on a P-RESYNCHRONIZE(restart) indication primitive. If CCR Protocol Version 2 is being used, the accepting CCRPM receives a C-ROLLBACK-RI APDU from its peer as user data on a P-RESYNCHRONIZE(abandon) indication primitive.

7.5.4.3 C-ROLLBACK response primitive
{CHANGE SECOND SENTENCE OF FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, it issues a P-RESYNCHRONIZE(restart) response primitive with the APDU as a data value of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-RESYNCHRONIZE(abandon) response primitive with the APDU as a data value of the primitive's User Data parameter.

7.5.4.4 C-ROLLBACK-RC APDU
{CHANGE FIRST SENTENCE OF FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, the requesting CCRPM receives a C-ROLLBACK-RC APDU from its peer as user data on a P-RESYNCHRONIZE(restart) confirm primitive. If CCR Protocol Version 2 is being used, the requesting CCRPM receives a C-ROLLBACK-RC APDU from its peer as user data on a P-RESYNCHRONIZE(abandon) confirm primitive.

7.5.5 Use of the C-ROLLBACK-RI APDU fields
{NO CHANGE}

7.5.6 Use of the C-ROLLBACK-RC APDU fields
{NO CHANGE}

7.5.7 Disruptive effect
{CHANGE THE FIRST SENTENCE TO READ AS FOLLOWS.}

Because the C-ROLLBACK service is mapped on the P-RESYNCHRONIZE service, CCR APDUs other than a C-ROLLBACK-RI from the association-initiator are discarded (by the underlying session service-provider).

7.5.8 Collision with a C-ROLLBACK-RI APDU
{NO CHANGE}

7.6 Branch recovery procedure

7.6.1 Purpose
{NO CHANGE}

7.6.2 APDUs used
{NO CHANGE}

7.6.3 Prerequisite requirements
{NO CHANGE}

7.6.4 Branch recovery procedure

{NO CHANGE}

7.6.5 Use of the C-RECOVER-RI APDU fields

{ADD THE FOLLOWING TWO PARAGRAPHS AT THE END OF THE CLAUSE}

For the requesting CCRPM (Version 2 only): If the Atomic Action Identifier or Branch Identifier parameters of the C-RECOVER request contain the AE-title of the requestor, as passed on the A-ASSOCIATE service used to establish the supporting association, the CCRPM shall represent this in the abstract syntax by using either the "name" form or the "sender" value of the "side" form of the corresponding APDU field. Similarly, if the parameters contain the AE-title of the acceptor passed on the A-ASSOCIATE service used to establish the supporting association, the CCRPM shall represent this in the abstract syntax by using either the "name" form or the "receiver" value of the "side" form of the corresponding APDU field.

For the accepting CCRPM (Version 2 only): If the "masters-name" field in the "atomic-action-identifier" or the "superiors-name" field in the "branch-identifier" is the "sender" value of the "side" form, the corresponding parameter value shall be the C-RECOVER requestor's AE-title passed on the A-ASSOCIATE service used to establish the supporting association. Similarly, if the "receiver" value of the "side" form is used, the corresponding parameter shall be the C-RECOVER acceptor's AE-title passed in the A-ASSOCIATE service used to establish the supporting association.

7.6.6 Use of the C-RECOVER-RC APDU fields

{ADD THE FOLLOWING TWO PARAGRAPHS AND NOTE AT THE END OF THE CLAUSE}

For the accepting CCRPM (Version 2 only): If the Atomic Action Identifier or Branch Identifier parameters of the C-RECOVER response contain the AE-title of the acceptor, as passed on the A-ASSOCIATE service used to establish the supporting association, the CCRPM shall represent this in the abstract syntax by using either the "name" form or the "sender" value of the "side" form of the corresponding APDU field. Similarly, if the parameters contain the AE-title of the requestor, as passed on the A-ASSOCIATE service used to establish the supporting association, the CCRPM shall represent this in the abstract syntax by using either the "name" form or the "receiver" value of the "side" form of the corresponding APDU field.

For the requesting CCRPM (Version 2 only): If the "masters-name" field in the "atomic-action-identifier" or the "superiors-name" field in the "branch-identifier" is the "sender" value of the "side" form, the corresponding parameter value shall be the C-RECOVER acceptor's AE-title passed on the A-ASSOCIATE service used to establish the supporting association. Similarly, if the "receiver" value of the "side" form is used, the corresponding parameter shall be the C-RECOVER requestor's AE-title passed in the A-ASSOCIATE service used to establish the supporting association.

Note -- The "sender" and "receiver" values identify the peer's by their roles in the transmission of a particular APDU, not the procedure. Thus, a value of "sender" on a C-RECOVER-RI corresponds to a value of "receiver" on the replying C-RECOVER-RC.

7.7 Order commitment and begin branch procedure

7.7.1 Purpose
{NO CHANGE}

7.7.2 APDUs used
{NO CHANGE}

7.7.3 Prerequisite requirements
{NO CHANGE}

7.7.4 Procedure operation
{NO CHANGE TO THE INTRODUCTORY TEXT UNDER 7.7.4}

7.7.4.1 C-COMMIT request primitive + C-BEGIN request primitive
{CHANGE THE SECOND SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, it issues a P-SYNC-MAJOR request primitive with the APDUs as data values of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-SYNC-MINOR request primitive with the APDUs as data values of the primitive's User Data parameter.

7.7.4.2 C-COMMIT-RI APDU + C-BEGIN_RI APDU
{CHANGE THE FIRST SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, the accepting CCRPM receives a C-COMMIT-RI and a C-BEGIN-RI APDU from its peer as user data on a P-SYNC-MAJOR indication primitive. If CCR Protocol Version 2 is being used, the accepting CCRPM receives a C-COMMIT-RI and a C-BEGIN-RI APDU from its peer as user data on a P-SYNC-MINOR indication primitive.

7.7.4.3 C-COMMIT response primitive
{CHANGE THE SECOND SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, it issues a P-SYNC-MAJOR response primitive with the APDU as a data value of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-SYNC-MINOR response primitive with the APDU as a data value of the primitive's User Data parameter.

7.7.4.4 C-COMMIT-RC APDU
{CHANGE THE FIRST SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, the accepting CCRPM receives a C-COMMIT-RC APDU from its peer as user data on a P-SYNC-MAJOR confirm primitive. If CCR Protocol Version 2 is being used, the accepting CCRPM receives a C-COMMIT-RC APDU from its peer as user data on a P-SYNC-MINOR confirm primitive.

7.7.5 Use of the C-COMMIT-RI APDU and C-BEGIN-RI APDU fields
{NO CHANGE}

7.7.6 Use of the C-COMMIT-RC APDU field
{NO CHANGE}

7.7.7 Collisions
{NO CHANGE}

7.8 Rollback and begin branch procedure

7.8.1 Purpose
{NO CHANGE}

7.8.2 APDUs used
{NO CHANGE}

7.8.3 Prerequisite requirements
{NO CHANGE}

7.8.4 Procedure operation
{NO CHANGE TO THE INTRODUCTORY TEXT UNDER 7.8.4}

7.8.4.1 C-ROLLBACK request primitive + C-BEGIN request primitive
{CHANGE THE SECOND SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, it issues a P-RESYNCHRONIZE (restart) request primitive with the APDUs as data values of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-RESYNCHRONIZE (abandon) request primitive with the APDUs as data values of the primitive's User Data parameter.

7.8.4.2 C-ROLLBACK-RI APDU + C-BEGIN-RI APDU
{CHANGE THE FIRST SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, the accepting CCRPM receives a C-ROLLBACK-RI and a C-BEGIN-RI APDU from its peer as user data on a P-RESYNCHRONIZE(restart) indication primitive. If CCR Protocol Version 2 is being used, the accepting CCRPM receives a C-ROLLBACK-RI and a C-BEGIN-RI APDU from its peer as user data on a P-RESYNCHRONIZE(abandon) indication primitive.

7.8.4.3 C-ROLLBACK response primitive
{CHANGE THE SECOND SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, it issues a P-RESYNCHRONIZE (restart) response primitive with the APDU as a data value of the primitive's User Data parameter. If CCR Protocol Version 2 is being used, it issues a P-RESYNCHRONIZE (abandon) response primitive with the APDU as a data value of the primitive's User Data parameter.

7.8.4.4 C-ROLLBACK-RC APDU
{CHANGE THE FIRST SENTENCE OF THE FIRST PARAGRAPH TO READ AS FOLLOWS.}

If CCR Protocol Version 1 is being used, the accepting CCRPM receives a C-ROLLBACK-RC APDU from its peer as user data on a P-RESYNCHRONIZE(restart)

confirm primitive. If CCR Protocol Version 2 is being used, the accepting CCRPM receives a C-ROLLBACK-RC APDU from its peer as user data on a P-RESYNCHRONIZE(abandon) confirm primitive.

7.8.5 Use of the C-ROLLBACK-RI APDU and C-BEGIN-RI APDU fields
{NO CHANGE}

7.8.6 Use of the C-ROLLBACK-RC APDU field
{NO CHANGE}

7.8.7 Disruptive effects
{NO CHANGE}

7.8.8 Collisions
{CHANGE THE FIRST SENTENCE OF 7.8.8.2 TO READ AS FOLLOWS.}

Resolution of the collision of C-ROLLBACK-RI APDU + C-BEGIN-RI APDU with C-ROLLBACK-RI APDU [both mapped to either P-RESYNCHRONIZE(restart), in the case of CCR Protocol Version 1, or P-RESYNCHRONIZE(abandon), in the case of CCR Protocol Version 2] depends on which side was the association-initiator.

{ADD A NEW CLAUSE 7.9 AS FOLLOWS}

7.9 Association establishment procedure

7.9.1 Purpose
This procedure is used to negotiate the CCR version to be used between two CCR protocol machines over the association. This procedure is only used by CCRPMs that support CCR Protocol Version 2.

7.9.2 APDUs used

This procedure uses the following CCR APDUs:

C-INITIALIZE-RI
C-INITIALIZE-RC

The structure of these APDUs is shown in figure 1.

```
C-INITIALIZE-RI ::= [11] SEQUENCE {
                         version-number [0] BIT STRING
                                                {version1(0), version2(1)}
                                                DEFAULT version2
                         }

C-INITIALIZE-RC ::= [12] SEQUENCE {
                         version-number [0] BIT STRING
                                                {version1(0), version2(1)}
                                                DEFAULT version2
                         }
```

Figure 1

The C-INITIALIZE-RI APDU field is listed in table x. The C-INITIALIZE-RC APDU field is listed in table y.

Table 1 - C-INITIALIZE-RI fields

Field name	Presence	Source	Sink
version-number	M	CCRPM	CCRPM

The version-number parameter on the C-INITIALIZE-RI is used to indicate which versions of CCR are being proposed for use on this association.

Table 2 - C-INITIALIZE-RC fields

Field name	Presence	Source	Sink
version-number	M	CCRPM	CCRPM

The version-number parameter on the C-INITIALIZE-RC has two possible uses. If the association is accepted, the parameter shall indicate the version of CCR which will be used on this association. If the association is rejected, the parameter shall indicate which versions of CCR are available for use on a future association.

7.9.3 Procedure operation

The procedure is performed concurrently with the A-ASSOCIATE procedure (see ACSE) when the association will be used for CCR. The procedure is driven by the following events:

a) A-ASSOCIATE request primitive from the requestor;

b) a C-INITIALIZE-RI APDU received by the accepting CCRPM;

c) A-ASSOCIATE response primitive from the acceptor; and

d) a C-INITIALIZE-RC APDU received by the CCRPM of the requestor.

When interworking with CCR Protocol Version 1, the last event does not occur and procedure is completed by:

e) A-ASSOCIATE confirm primitive with no CCR APDU in the User information.

7.9.3.1 A-ASSOCIATE request primitive

When an A-ASSOCIATE request is made and the Application Context contains the CCR ASE, then the CCRPM generates a C-INITIALIZE-RI APDU, which is carried in the user information parameter of the A-ASSOCIATE request.

7.9.3.2 C-INITIALIZE-RI APDU

When this APDU is received, the CCRPM selects one of the proposed versions to use on this association. The selected version must be one of the versions valid for this CCRPM and must appear on the list of proposed versions in the version-number parameter of this APDU. If there is no version that satisfies this condition, then the association must be rejected. If this APDU is not received in the User information of the A-ASSOCIATE indication, then the peer CCRPM supports CCR Protocol Version 1.

7.9.3.3 A-ASSOCIATE response primitive

When an A-ASSOCIATE response is made and the Application Context contains the CCR ASE and the C-INITIALIZE-RI APDU was received, then the CCRPM generates a C-INITIALIZE-RC APDU, which is carried in the User information parameter of the A-ASSOCIATION request.

7.9.3.4 C-INITIALIZE-RC APDU

If the association was accepted, the CCRPM must use the version of CCR that was indicated by the version-number parameter on this APDU. If the association was rejected, the CCRPM may note the available CCR versions from the version-number parameter for future reference.

7.9.3.5 A-ASSOCIATE confirm with no CCR APDU

If no CCR APDU was received in the User information of the A-ASSOCIATE confirm primitive, then the peer CCRPM supports CCR Protocol Version 1. The CCRPM shall use CCR Protocol Version 1.

8 CCRPM State Table

{NO CHANGE TO THE INTRODUCTORY TEXT UNDER CLAUSE 8}

8.1 General

{NO CHANGE}

8.2 Incoming events

{NO CHANGE}

8.3 Outgoing events

{NO CHANGE}

8.4 Specific actions

{NO CHANGE}

8.5 Predicates

{CHANGE ITEM "b)" TO READ AS FOLLOWS}

b) the possession of the major/activity sync token (for CCR Protocol Version 1);

8.6 Enablements

{NO CHANGE}

8.7 Variables

{NO CHANGE}

8.8 Notation

{NO CHANGE}

8.9 Conventions

{NO CHANGE}

8.10 Actions to be taken by the CCRPM

{NO CHANGE}

8.11 Changes to atomic action data

{CHANGE TABLE 25, PREDICATE "p1" TO READ AS FOLLOWS.}

p1 Atomic action data for the superior of the current branch is accessible in stable storage. If CCR Protocol Version 1 is being used, the major/activity token is in the possession of the requestor. If CCR Protocol Version 2 is being used, the minor sync token is in the possession of the requestor.

{CHANGE THE C-BEGIN-RC/A7 CELL IN STATE TABLE 28 FROM BLANK TO CONTAINING "A7"}

{CHANGE THE TITLE OF CLAUSE 9 TO READ AS FOLLOWS}

9 Mapping to the presentation service in CCR Protocol Version 1

{ADD A NEW CLAUSE 10 . THIS CLAUSE CONTAINS ESSENTIALLY THE SAME TEXT AS CLAUSE 9, BUT WITH THE SEVERAL CHANGES WHERE THE CLAUSES ARE THE SAME IS MARKED. TABLE 33 IS THE SAME AS TABLE 32 WITH THE FOLLOWING CHANGES. RENUMBER SUCCEEDING CLAUSES AND TABLES ACCORDINGLY.}

10 Mapping to the presentation service in CCR Protocol Version 2

{SAME TEXT AS CLAUSE 9 INTRODUCTORY TEXT}

{MAKE THE FOLLOWING CHANGES INDICATED BELOW TO ROWS 6-14 IN TABLE 33.}

CCR primitive	CCR APDUs	Presentation Primitive
C-ROLLBACK req/ind	C-ROLLBACK-RI	P-RESYNC(restart,abandon) req/ind
C-ROLLBACK rsp/cnf	C-ROLLBACK-RC	P-RESYNC(restart,abandon) rsp/cnf
C-ROLLBACK req/ind + C-BEGIN req/ind	C-ROLLBACK-RI followed by C-BEGIN-RI	P-RESYNC(restart,abandon) req/ind
C-ROLLBACK rsp/cnf + C-BEGIN rsp/cnf	C-ROLLBACK-RC followed by C-BEGIN-RC	P-RESYNC(restart,abandon) rsp/cnf
(see 7.8.8.1)	C-ROLLBACK-RC followed by C-BEGIN-RI	P-RESYNC(restart,abandon) rsp/cnf
C-COMMIT req/ind	C-COMMIT-RI	P-SYNC-~~MAJOR~~MINOR req/ind
C-COMMIT rsp/cnf	C-COMMIT-RC	P-SYNC-~~MAJOR~~MINOR rsp/cnf
C-COMMIT req/ind + C-BEGIN req/ind	C-COMMIT-RI followed by C-BEGIN-RI	P-SYNC-~~MAJOR~~MINOR req/ind
C-COMMIT rsp/cnf + C-BEGIN rsp/cnf	C-COMMIT-RC followed by C-BEGIN-RC	P-SYNC-~~MAJOR~~MINOR rsp/cnf

10.1 Begin branch

{SAME TEXT AS 9.1 INTRODUCTORY TEXT}

10.1.1 Use of the P-SYNC-MINOR req/ind parameters

10.1.1.1 Type
{SAME TEXT AS 9.1.1.1}

10.1.1.2 Synchronization Point Serial Number: The use of this value is not determined by this standard.

10.1.1.3 User data:
{SAME TEXT AS 9.1.1.3}

10.1.1.4 Data Separation: This parameter is set to TRUE by the CCRPM on the request primitive.

10.1.2 Use of the P-SYNC-MINOR rsp/cnf parameters

10.1.2.1 Synchronization Point Serial Number: This value is identical to that on the preceding P-SYNC-MINOR indication that carried the C-BEGIN-RI.

10.1.2.2 User Data
{SAME TEXT AS 9.1.2.2}

10.2 Prepare subordinate
{SAME TEXT AS 9.2}

10.3 Offer commitment
{SAME TEXT AS 9.3}

10.4 Order commitment

The order commitment procedure uses the P-SYNC-MINOR service.

10.4.1 Use of the P-SYNC-MINOR req/ind parameters

10.4.1.1 Synchronization Point Serial Number: The use of this value is not determined by this standard.

10.4.1.2 User Data
{SAME TEXT AS 9.4.1.2}

10.4.1.3 Data Separation: This parameter is set to TRUE by the CCRPM on the request primitive.

10.4.2 Use of the P-SYNC-MINOR rsp/cnf parameters

10.4.2.1 Synchronization Point Serial Number: This value is identical to that on the preceding P-SYNC-MINOR indication that carried the C-COMMIT-RI APDU.

10.4.2.2 User Data: The User Data parameter is used to carry the C-COMMIT-RC APDU. User Data (if any) on the C-COMMIT response primitive is included in the C-COMMIT-RC APDU and is expressed using one or more presentation context specified by the requestor on the C-COMMIT response primitive.

10.5 Rollback

The rollback procedure uses the P-RESYNCHRONIZE(abandon) service.

10.5.1 Use of the P-RESYNCHRONIZE req/ind parameters

10.5.1.1 Resynchronize type: This parameter is set to the value of "abandon" .

10.5.1.2 Synchronization Point Serial Number: The use of this value is not determined by this standard..

10.5.1.3 Tokens
{SAME TEXT AS 9.5.1.3}

10.5.1.4 User data
{SAME TEXT AS 9.5.1.4}

10.5.2 Use of the P-RESYCHRONIZE rsp/cnf parameters

10.5.2.1 Synchronization Point Serial Number: The use of this value is not determined by this standard.

10.5.2.2 User Data
{SAME TEXT AS 9.5.2.2}

10.6 Branch recovery
{SAME TEXT AS 9.6}

10.7 Order commitment and begin branch procedure

The order commitment and begin branch procedure uses the P-SYNC-MINOR service.

10.7.1 Use of the P-SYNC-MINOR req/ind parameters

10.7.1.1 Synchronization point serial number
{SAME TEXT AS 9.7.1.1[2)]}

10.7.1.2 User Data
{SAME TEXT AS 9.7.1.2[2)]}

10.7.1.3 Data Separation: This parameter is set to TRUE by the CCRPM on the request primitive.

10.7.2 Use of the P-SYNC-MINOR rsp/cnf parameters
{SAME TEXT AS 9.7.2 EXCEPT FOR THE FOLLOWING CHANGE}

{CHANGE "P-SYNC-MAJOR" *IN THE FIRST SENTENCE TO* "P-SYNC-MINOR"*}*

[2)] From Technical Corrigenda 2 arising from Defect Report 9805/003.

10.8 Rollback and begin branch procedure

The rollback and begin branch procedure uses the P-RESYNCH (abandon) service.

10.8.1 Use of the P-RESYNCHRONIZE req/ind parameters

10.8.1.1Resynchronization type: This parameter is set to the value "abandon" .

10.8.1.2 Synchronization Point Serial Number
{SAME TEXT AS 9.8.1.2[2)]}

10.8.1.3 Tokens
{SAME TEXT AS 9.8.1.3[2)]}

10.8.1.4 User Data
{SAME TEXT AS 9.8.1.4[2)]}

10.8.2 Use of the P-RESYNCHRONIZE rsp/cnf parameter

10.8.2.1 Synchronization Point Serial Number: The use of this value is not covered in this International Standard.

10.8.2.2 User Data
{SAME TEXT AS 9.8.2.2}

{END OF NEW CLAUSE 10}

10 Concatenations and mappings
{CHANGE ROW 2 IN TABLE 33 TO READ AS FOLLOWS.}

2	C-COMMIT	P-SYNC-MAJOR, if CCR Protocol Version 1 P-SYNC-MINOR, if CCR Protocol Version 2

11 Precedence
{NO CHANGE}

12 Conformance
{NO CHANGE}

ANNEX A

{normative}

Definition of CCR datatypes

{CHANGE THE TITLE OF A.2 TO THE FOLLOWING}

A.2 Definitions for CCR Protocol Version 1

{ADD THE FOLLOWING NEW CLAUSE}

A.3 Definitions for CCR Protocol Version 2

{THIS CLAUSE CONTAINS THE SAME TEXT AS CLAUSE A.2, WITH THE FOLLOWING CHANGES}

{CHANGE "recovery-state" *TO THE FOLLOWING: "recovery-state" APPEARS IN THE* "C-RECOVER-RI" *AND THE* "C-RECOVER-RC" *APDUS}*

```
recovery-state          [2] ENUMERATED {commit (0), ready (1), done (2),
                                        unknown (3), retry-later (5)}
```

{CHANGE THE "ATOMIC-ACTION-IDENTIFIER" *PDU TO THE FOLLOWING}*

```
ATOMIC-ACTION-IDENTIFIER    ::=     SEQUENCE
    {masters-name           CHOICE {
                                name        [0]     EXPLICIT AE-Title,
                                side        [1]     ENUMERATED
                                                    {sender     (0),
                                                     receiver   (1) } }

    atomic-action-suffix    CHOICE {
                                form1       [2]     OCTET STRING,
                                form2       [3]     INTEGER}
    }
```

{CHANGE THE "BRANCH-IDENTIFIER" *PDU TO THE FOLLOWING}*

```
BRANCH-IDENTIFIER       ::=     SEQUENCE
    {superiors-name     CHOICE {
                                name        [0]     EXPLICIT AE-Title,
                                side        [1]     ENUMERATED
                                                    {sender     (0),
                                                     receiver   (1) } }

    branch-suffix       CHOICE {
                                form1       [2]     OCTET STRING,
                                form2       [3]     INTEGER}
    }
```

{REMOVE THE "BRANCH-SUFFIX" *PDU}*