



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

**UIT-T**

SECTEUR DE LA NORMALISATION  
DES TÉLÉCOMMUNICATIONS  
DE L'UIT

**X.860**

(12/97)

SÉRIE X: RÉSEAUX POUR DONNÉES ET  
COMMUNICATION ENTRE SYSTÈMES OUVERTS

Applications OSI – Traitement transactionnel

---

**Interconnexion des systèmes ouverts –  
Traitement transactionnel réparti: Modèle**

Recommandation UIT-T X.860

(Antérieurement Recommandation du CCITT)

---

RECOMMANDATIONS UIT-T DE LA SÉRIE X  
**RÉSEAUX POUR DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS**

|  |                    |
|--|--------------------|
| <b>RÉSEAUX PUBLICS POUR DONNÉES</b>                  |                    |
| Services et fonctionnalités                          | X.1–X.19           |
| Interfaces   | X.20–X.49          |
| Transmission, signalisation et commutation           | X.50–X.89          |
| Aspects réseau                                       | X.90–X.149         |
| Maintenance  | X.150–X.179        |
| Dispositions administratives                         | X.180–X.199        |
| <b>INTERCONNEXION DES SYSTÈMES OUVERTS</b>           |                    |
| Modèle et notation                                   | X.200–X.209        |
| Définitions des services                             | X.210–X.219        |
| Spécifications des protocoles en mode connexion      | X.220–X.229        |
| Spécifications des protocoles en mode sans connexion | X.230–X.239        |
| Formulaires PICS                                     | X.240–X.259        |
| Identification des protocoles                        | X.260–X.269        |
| Protocoles de sécurité                               | X.270–X.279        |
| Objets gérés de couche                               | X.280–X.289        |
| Tests de conformité                                  | X.290–X.299        |
| <b>INTERFONCTIONNEMENT DES RÉSEAUX</b>               |                    |
| Généralités  | X.300–X.349        |
| Systèmes de transmission de données par satellite    | X.350–X.399        |
| <b>SYSTÈMES DE MESSAGERIE</b>                        | X.400–X.499        |
| <b>ANNUAIRE</b>                                      | X.500–X.599        |
| <b>RÉSEAUTAGE OSI ET ASPECTS SYSTÈMES</b>            |                    |
| Réseautage   | X.600–X.629        |
| Efficacité   | X.630–X.639        |
| Qualité de service                                   | X.640–X.649        |
| Dénomination, adressage et enregistrement            | X.650–X.679        |
| Notation de syntaxe abstraite numéro un (ASN.1)      | X.680–X.699        |
| <b>GESTION OSI</b>                                   |                    |
| Cadre général et architecture de la gestion-systèmes | X.700–X.709        |
| Service et protocole de communication de gestion     | X.710–X.719        |
| Structure de l'information de gestion                | X.720–X.729        |
| Fonctions de gestion et fonctions ODMA               | X.730–X.799        |
| <b>SÉCURITÉ</b>                                      | X.800–X.849        |
| <b>APPLICATIONS OSI</b>                              |                    |
| Engagement, concomitance et rétablissement           | X.850–X.859        |
| <b>Traitement transactionnel</b>                     | <b>X.860–X.879</b> |
| Opérations distantes                                 | X.880–X.899        |
| <b>TRAITEMENT RÉPARTI OUVERT</b>                     | X.900–X.999        |

*Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.*

## RECOMMANDATION UIT-T 860

### INTERCONNEXION DES SYSTÈMES OUVERTS – TRAITEMENT TRANSACTIONNEL RÉPARTI: MODÈLE

#### Résumé

La présente Recommandation fournit une introduction générale traitant des concepts et du modèle du traitement réparti (TP *transaction processing*) et définit les besoins auxquels doit satisfaire le traitement TP.

#### Source

La Recommandation UIT-T X.860, révisée par la Commission d'études 7 de l'UIT-T (1997-2000), a été approuvée le 12 décembre 1997 selon la procédure définie dans la Résolution n° 1 de la CMNT.

## AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

La Conférence mondiale de normalisation des télécommunications (CMNT), qui se réunit tous les quatre ans, détermine les thèmes d'études à traiter par les Commissions d'études de l'UIT-T lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution n° 1 de la CMNT.

Dans certains secteurs de la technologie de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

## NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

## DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

© UIT 1998

Droits de reproduction réservés. Aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'UIT.

## TABLE DES MATIÈRES

|          | <i>Page</i>  |
|----------|--|
| 1        | Domaine d'application..... 1   |
| 2        | Références normatives ..... 1  |
| 2.1      | Recommandations   Normes internationales identiques..... 1   |
| 2.2      | Paires de Recommandations   Normes internationales équivalentes par leur contenu technique..... 2                    |
| 2.3      | Autres références ..... 2  |
| 3        | Définitions..... 2   |
| 3.1      | Termes définis dans d'autres Recommandations..... 2  |
| 3.2      | Termes définis dans la présente Recommandation ..... 4   |
| 4        | Abréviations ..... 8   |
| 5        | Conventions ..... 8  |
| 6        | Définition des besoins ..... 9   |
| 6.1      | Introduction ..... 9   |
| 6.2      | Besoins de l'utilisateur ..... 9   |
| 6.3      | Besoins concernant la modélisation..... 10   |
| 6.4      | Besoins concernant le service et le protocole OSI TP ..... 10  |
| 7        | Concepts du traitement transactionnel réparti ..... 10   |
| 7.1      | Transaction ..... 10   |
| 7.2      | Transaction répartie ..... 10  |
| 7.3      | Données de transaction et niveau de coordination..... 11   |
| 7.4      | Relations d'arbre ..... 11   |
| 7.5      | Dialogue ..... 12  |
| 7.6      | Arbre de dialogue ..... 12   |
| 7.7      | Branches d'une transaction ..... 12  |
| 7.8      | Arbre de transaction..... 13   |
| 7.9      | Canaux..... 13   |
| 7.10     | Prise de contact..... 13   |
| 7.11     | Arrière-plan ..... 14  |
| 8        | Modèle du service OSI TP ..... 14  |
| 8.1      | Nature du service OSI TP ..... 14  |
| 8.2      | Règles s'appliquant aux arbres de dialogue ..... 15  |
| 8.3      | Règles s'appliquant aux arbres de transaction..... 17  |
| 8.4      | Dénomination ..... 18  |
| 8.5      | Transfert de données..... 19   |
| 8.6      | Coordination des ressources ..... 20   |
| 8.7      | Rétablissement..... 25   |
| 8.8      | Gestion de la simultanéité et blocage mutuel..... 32   |
| 8.9      | Sécurité..... 32   |
| Annexe A | – Relation entre le modèle OSI TP et la structure de la couche Application ..... 32                                  |
| A.1      | Introduction ..... 32  |
| A.2      | Processus d'application dans le traitement OSI TP ..... 32   |
| A.3      | Entités d'application dans le traitement OSI TP ..... 32   |
| A.4      | Frontières du service OSI TP..... 33   |
| Annexe B | – Présentation didactique de la gestion de la simultanéité et du blocage mutuel pour le traitement OSI TP ..... 34   |
| Annexe C | – Informations didactiques concernant le protocole de retour-arrière présumé avec engagement en deux phases ..... 35 |

|  | <i>Page</i> |
|--|-------------|
| Annexe D – Combinaison d'optimisations d'engagement.....                                     | 36          |
| D.1 Engagement dynamique avec commande polarisée.....  | 36          |
| D.2 Préparation implicite non sélectionnée et signal "prêt" autorisé dans les deux sens..... | 36          |
| D.3 Préparation implicite.....   | 37          |
| Annexe E – Résumés des modifications apportées par la deuxième édition .....                 | 38          |

## Introduction

La présente Recommandation fait partie d'une série de normes élaborées en vue de faciliter l'interconnexion de systèmes informatiques. Elle est liée à d'autres Recommandations et Normes internationales appartenant à l'ensemble défini par le modèle de référence pour l'interconnexion des systèmes ouverts (voir Rec. UIT-T X.200 | ISO/CEI 7498-1). Le modèle de référence subdivise le domaine de la normalisation de l'interconnexion en une série de couches de spécifications dont chacune est de taille maîtrisable.

L'objectif de l'interconnexion des systèmes ouverts est de permettre, moyennant un minimum d'accords techniques en dehors des normes d'interconnexion, d'interconnecter des systèmes informatiques:

- a) de constructeurs différents;
- b) gérés de façon différente;
- c) de niveaux de complexité différents; et
- d) de technologies différentes.

Les Recommandations de la série X.860 de l'UIT-T et l'ISO/CEI 10026 définissent un modèle de traitement transactionnel OSI (modèle OSI TP) et un service de traitement transactionnel OSI (service OSI TP) et spécifient un protocole de traitement transactionnel OSI (protocole OSI TP) disponible au sein de la couche Application du modèle de référence OSI.

Le service OSI TP est un service de couche Application. Il s'intéresse à des informations pouvant être mises en relation sous la forme de transactions réparties qui font intervenir deux systèmes ouverts ou plus.

La présente Recommandation fournit un nombre de fonctionnalités suffisant pour la prise en charge du traitement transactionnel et définit un cadre général de coordination de ressources OSI TP multiples situées au sein de systèmes ouverts distincts.

La présente Recommandation ne spécifie pas l'interface avec les ressources locales ou les fonctionnalités d'accès qui sont fournies au sein du système local. Ces sujets pourront néanmoins faire l'objet de futures extensions à la présente Recommandation.





# INTERCONNEXION DES SYSTÈMES OUVERTS – TRAITEMENT TRANSACTIONNEL RÉPARTI: MODÈLE<sup>1)</sup>

(révisée en 1997)

## 1 Domaine d'application

La présente Recommandation:

- a) fournit une introduction générale aux concepts et aux mécanismes du traitement réparti;
- b) définit un modèle de traitement réparti;
- c) définit les besoins auxquels doit satisfaire le service OSI TP; et
- d) tient compte du besoin de coexistence avec d'autres éléments du service d'application, tels que l'accès à des bases de données distante (RDA, *remote data base access*), l'élément de service d'opérations distantes (ROSE, *remote operations service element*) et les applications qui ne sont pas basés sur l'élément ROSE.

La présente Recommandation fournit un nombre de dispositions suffisant pour permettre la spécification des services et des protocoles de communication en mode transaction qui satisfont aux propriétés d'atomicité, de cohérence, d'isolation et de durabilité (propriétés ACID, *atomicity, consistency, isolation and durability*) telles qu'elles sont définies dans la Rec. UIT-T X.851 | ISO/CEI 9804.

La présente Recommandation ne spécifie pas de forme particulière d'implémentation ou de produit et n'impose aucune contrainte de réalisation pour les entités et interfaces au sein d'un système informatique.

## 2 Références normatives

La présente Recommandation se réfère à certaines dispositions des Recommandations UIT-T et textes suivants qui de ce fait en sont partie intégrante. Les versions indiquées étaient en vigueur au moment de la publication de la présente Recommandation. Toute Recommandation ou tout texte étant sujet à révision, les utilisateurs de la présente Recommandation sont invités à se reporter, si possible, aux versions les plus récentes des références normatives suivantes. La liste des Recommandations de l'UIT-T en vigueur est régulièrement publiée.

### 2.1 Recommandations | Normes internationales identiques

- Recommandation UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: le modèle de référence de base.*
- Recommandation UIT-T X.210 (1993) | ISO/CEI 10731:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: conventions pour la définition des services de l'interconnexion de systèmes ouverts.*
- Recommandation UIT-T X.215 (1995) | ISO/CEI 8326:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service de session.*
- Recommandation UIT-T X.216 (1994) | ISO/CEI 8822:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service de présentation.*
- Recommandation UIT-T X.217 (1995) | ISO/CEI 8649:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition de service applicable à l'élément de service de contrôle d'association.*
- Recommandation UIT-T X.501 (1997) | ISO/CEI 9594-2:1997, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: les modèles.*
- Recommandation UIT-T X.650 (1996) | ISO/CEI 7498-3:1997, *Technologies de l'information – Interconnexion des systèmes ouverts (OSI) – Modèle de référence de base: Dénomination et adressage.*

<sup>1)</sup> La Recommandation UIT-T X.860 et l'ISO/CEI 10026-1 "Technologies de l'information – Interconnexion des systèmes ouverts – Traitement réparti – Partie 1: Le modèle" ont été élaborées en étroite collaboration et sont alignées sur le plan technique.

- Recommandation UIT-T X.851 (1997) | ISO/CEI 9804:1994, *Technologies de l'information – Interconnexion des systèmes ouverts (OSI) – Définition du service pour l'élément de service d'engagement, de concomitance et de rétablissement.*
- Recommandation UIT-T X.863 (1994) | ISO/CEI 10026-4:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Traitement transactionnel réparti – formulaire de déclaration de conformité d'une instance de protocole.*

## 2.2 Paires de Recommandations | Normes internationales équivalentes par leur contenu technique

- Recommandation X.800 du CCITT (1991), *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT.*  
ISO/CEI 7498-2:1989, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Partie 2: Architecture de sécurité.*
- Recommandation UIT-T X.861 (1997), *Interconnexion des systèmes ouverts – Traitement transactionnel réparti: définition du service.*  
ISO/CEI 10026-2:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Traitement transactionnel réparti – Partie 2: Service OSI TP.*
- Recommandation UIT-T X.862 (1997), *Interconnexion des systèmes ouverts – Traitement transactionnel réparti: spécification du protocole.*  
ISO/CEI 10026-3:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Traitement transactionnel réparti – Partie 3: Spécification du protocole.*

## 2.3 Autres références

- ISO/CEI 9545:1994, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Structure de la couche application.*  
NOTE – La présente Recommandation utilise la terminologie et les techniques de modélisation de la première édition (1989) de la structure de couche Application (ISO/CEI 9545:1989).

## 3 Définitions

Les définitions suivantes s'appliquent aux fins de la présente Recommandation:

### 3.1 Termes définis dans d'autres Recommandations

**3.1.1** La présente Recommandation utilise les termes suivants définis dans la Rec. UIT-T X.200 | ISO/CEI 7498-1:

- a) entité d'application;
- b) processus d'application;
- c) unité de données de protocole d'application;
- d) concaténation;
- e) système ouvert;
- f) service de présentation;
- g) point d'accès au service de présentation;
- h) unité de données du service de présentation;
- i) système ouvert réel;
- j) séparation.

**3.1.2** La présente Recommandation utilise les termes suivants définis dans la Rec. X.800 du CCITT | ISO/CEI 7498-2:

- a) contrôle d'accès;
- b) audit (vérification);
- c) authentification;
- d) confidentialité;
- e) intégrité;
- f) non-répudiation.

**3.1.3** La présente Recommandation utilise les termes suivants définis dans la Rec. X.650 | ISO/CEI 7498-3:

- a) identificateur d'invocation de processus d'application;
- b) titre de processus d'application;
- c) identificateur d'invocation de l'entité d'application;
- d) qualificateur de l'entité d'application;
- e) titre de l'entité d'application.

**3.1.4** La présente Recommandation utilise le terme suivant défini dans la Rec. UIT-T X.215 | ISO/CEI 8326:

- qualité de service.

**3.1.5** La présente Recommandation utilise les termes suivants définis dans la Rec. UIT-T X.210 | ISO/CEI 10731:

- a) demande;
- b) indication;
- c) réponse;
- d) confirmation;
- e) primitive de service; primitive;
- f) fournisseur du service;
- g) utilisateur du service.

**3.1.6** La présente Recommandation utilise les termes suivants définis dans l'ISO/CEI 9545:

- a) association d'application; association;
- b) contexte d'application;
- c) nom de contexte d'application;
- d) invocation d'entité d'application;
- e) invocation de processus d'application;
- f) élément de service d'application;
- g) élément de service de commande d'association;
- h) fonction de commande d'association multiple;
- i) fonction de commande d'association unique;
- j) objet d'association unique.

**3.1.7** La présente Recommandation utilise les termes suivants définis dans la Rec. UIT-T X.501 | ISO/CEI 9594-2:

- a) arbre d'information de l'annuaire;
- b) entrée d'annuaire; entrée;
- c) nom distinctif;
- d) classes d'objets;
- e) nom distinctif relatif.

**3.1.8** La présente Recommandation utilise les termes suivants définis dans la Rec. UIT-T X.851 | ISO/CEI 9804:

- a) données d'action atomique;
- b) atomicité;
- c) données liées;
- d) cohérence;
- e) durabilité;
- f) état final;
- g) décision heuristique;
- h) état initial;
- i) isolation.

## 3.2 Termes définis dans la présente Recommandation

La présente Recommandation définit les termes suivants.

**3.2.1 transaction répartie prise en charge par l'application:** transaction dans laquelle l'utilisateur du service OSI TP est responsable de l'entretien des propriétés ACID.

**3.2.2 succession chaînée:** succession de branches contiguës liées d'une transaction (prises en charge par le fournisseur) dans le même dialogue, visant à réaliser un but commun.

**3.2.3 machine de protocole de canal (CPM, *channel protocol machine*):** partie de l'invocation AEI impliquée dans le traitement OSI TP qui établit les canaux de traitement de transaction et qui met fin à leur existence.

**3.2.4 canal; canal de traitement de transaction:** relation concernant une association entre deux invocations d'entité d'application qui facilitent l'activité de rétablissement d'un fournisseur de service de traitement (TPSP). Les canaux ne sont pas visibles pour les invocations TPSUI.

**3.2.5 maître de l'engagement:** voisin à destination duquel un nœud a émis un signal "prêt".

NOTE – L'esclave de l'engagement sera le supérieur du dialogue en cas de procédure d'engagement statique.

**3.2.6 esclave de l'engagement:** voisin en provenance duquel a été reçu un signal "prêt".

NOTE 1 – Les procédures CCR utilisent le terme "subordonné de l'engagement"; le traitement OSI TP utilise le terme "esclave de l'engagement" pour éviter une confusion avec le subordonné du dialogue.

NOTE 2 – Le maître de l'engagement sera un subordonné du dialogue dans le cas de procédures d'engagement statique.

NOTE 3 – Les termes maître de l'engagement et esclave de l'engagement ne s'appliquent pas lorsqu'un signal "lecture seule", un signal "sortie prématurée" ou un signal "une phase" est émis.

**3.2.7 engagement; engagement d'une transaction:** achèvement d'une transaction avec la libération des données de transaction dans l'état final.

NOTE 1 – L'engagement nécessite une procédure d'engagement en deux phases si des données liées sont impliquées; un engagement en une phase peut être utilisé si des données liées ne sont pas impliquées; prière de se référer au 8.6.1 en ce qui concerne les procédures d'engagement en deux phases et au 8.6.4 en ce qui concerne les procédures d'engagement en une phase.

NOTE 2 – Les termes "engagement" et "retour-arrière" ont un domaine d'application différent de celui qui est défini dans la Rec. UIT-T X.851 | ISO/CEI 9804. La présente Recommandation traite de l'engagement et du retour-arrière d'une transaction complète, alors que la Rec. UIT-T X.851 | ISO/CEI 9804 fait référence à l'engagement et au retour-arrière d'une seule branche d'action atomique.

**3.2.8 coordinateur de l'engagement:** machine protocolaire TPPM qui arbitre le résultat final de la transaction.

NOTE – Le coordinateur de l'engagement sera situé à la racine de l'arbre de transaction dans le cas d'une procédure d'engagement statique en deux phases. Il sera situé au niveau d'une feuille ou d'un nœud intermédiaire dans le cas d'une procédure d'engagement statique en une phase. La position du coordinateur de l'engagement peut être prédéterminé, ou déterminé de manière dynamique dans le cas d'une procédure d'engagement dynamique en deux phases.

**3.2.9 arrière-plan de l'engagement:** l'arrière-plan de l'engagement actuel d'un nœud se constitue de l'ensemble de nœuds de l'arbre de transaction comprenant:

- a) les nœuds voisins en provenance desquels ont été des signaux "prêt" ont été reçus; et
- b) les arrière-plans de ces nœuds voisins et ainsi de suite d'une manière récursive.

NOTE 1 – L'arrière-plan de l'engagement ne contient pas ceux des nœuds qui ont émis des signaux "lecture seule", "une phase" ou "sortie prématurée".

NOTE 2 – L'arrière-plan de l'engagement d'un nœud sera identique au sous-arbre de transaction pour un nœud dans le cas d'un engagement statique en deux phases qui n'utilise aucun des signaux "lecture seule", "une phase" ou "sortie prématurée".

**3.2.10 ordre d'engagement:** déclaration par laquelle un nœud indique, à un voisin qui a émis un signal "prêt", que la transaction sera engagée.

**3.2.11 commande:** permission accordée à une invocation TPSUI pour une communication avec son partenaire pour un dialogue donné.

**3.2.12 niveau de coordination:** accord entre deux invocations TPSUI au sujet du mécanisme qui sera utilisé pour garantir les quatre propriétés d'une transaction; le niveau de coordination peut avoir l'une des valeurs "engagement", "engagement en une phase" ou "néant".

**3.2.13 dialogue coordonné:** dialogue dont le niveau de coordination actuel est "engagement" ou "engagement en une phase".

NOTE – Un dialogue prenant en charge des branches de transaction chaînées est toujours coordonné et un dialogue prenant en charge des branches de transaction non chaînées n'est coordonné que lorsqu'il prend en charge une branche de transaction.

**3.2.14 dialogue:** relation entre deux invocations TPSUI qui communiquent entre elles. L'instigateur du dialogue est le supérieur et le destinataire le subordonné.

**3.2.15 arbre de dialogue:** arbre constitué d'invocations TPSUI qui représentent les entités en dialogue et leurs relations.

**3.2.16 transaction répartie:** transaction dont les parties peuvent être exécutées dans plus d'un système ouvert.

**3.2.17 procédures d'engagement dynamique:** procédures d'engagement en deux phases qui ne sont pas soumises aux contraintes des procédures de l'engagement statique; compte tenu d'une commande optionnelle, le coordinateur d'engagement peut être un nœud prédéterminé de l'arbre de transaction (éventuellement différent de la racine), ou peut être déterminé d'une manière dynamique.

**3.2.18 signal de sortie prématurée:** déclaration faite par un nœud à un supérieur indiquant que ce nœud et son sous-arbre se retirent de la transaction parce qu'ils ne peuvent pas apporter de contribution au traitement de la transaction; les conditions nécessaires sont que les données liées du nœud n'ont pas été modifiées par la transaction, que des signaux "lecture seule" ou "sortie prématurée" ont été reçus de tous les nœuds subordonnés éventuels de l'arbre de transaction et qu'un compte rendu du résultat de la transaction n'est pas nécessaire.

**3.2.19 risque heuristique:** situation qui se produit lorsque les données liées du sous-arbre du subordonné sont dans un état inconnu après une défaillance de la communication avec ce subordonné.

**3.2.20 mélange heuristique:** situation qui se produit lorsque les données liées de la transaction sont dans un état incohérent après une ou plusieurs décisions heuristiques.

**3.2.21 intermédiaire:** entité d'un arbre qui possède un supérieur et un ou plusieurs subordonnés.

**3.2.22 feuille:** entité d'un arbre qui possède un supérieur et aucun subordonné.

**3.2.23 ressource locale:** ressource qui réside dans le même système ouvert réel que le demandeur de la ressource, ou ressource qui est gérée par une entité résidant dans le même système ouvert réel que le demandeur de la ressource.

**3.2.24 enregistrement de journalisation d'engagement:** enregistrement fait dans le journal de rétablissement, qui traduit une décision d'engagement de la transaction.

**3.2.25 enregistrement de journalisation de dommage:** enregistrement fait dans le journal de rétablissement, qui traduit une incohérence de l'état actuel de données liées dans le sous-arbre.

**3.2.26 enregistrement de journalisation heuristique:** enregistrement fait dans le journal de rétablissement, qui traduit une décision heuristique du nœud.

**3.2.27 enregistrement de journalisation "prêt":** enregistrement fait dans le journal de rétablissement, qui contient les contenant les informations nécessaires à un rétablissement et indique que les données liées de ce nœud sont prêtes pour un rétablissement et – dans le cas où il existe plus d'un voisin dans l'arbre de transaction – que l'un des signaux "prêt", "une phase", "lecture seule" ou "sortie prématurée" ont été reçus en provenance de la totalité de ces voisins sauf un.

**3.2.28 données à longue durée de vie:** données qui font l'objet d'un accès et d'une manipulation par l'invocation TPSUI au sein du domaine de validité d'une transaction prise en charge par un fournisseur ou au sein d'une transaction prise en charge par une application, mais pour lesquelles l'invocation TPSUI prend la responsabilité du rétablissement en cas de défaillance.

NOTE – Les "données à longue durée de vie" ne sont pas des "données liées" et réciproquement.

**3.2.29 voisin:** entité d'un arbre qui a une relation directe avec une autre entité.

NOTE – Il en résulte qu'un subordonné et son supérieur sont voisins.

**3.2.30 nœud:** invocation TPSUI ainsi que sa machine TPPM.

**3.2.31 écroulement de nœud:** défaillance du nœud (c'est-à-dire de la machine TPPM ou de l'invocation TPSUI) ou de l'environnement local prenant en charge le nœud entraînant l'interruption des dialogues et la perte éventuelle de toutes les données qui ne sont pas stockées dans des mémoires sûres.

**3.2.32 signal "une phase":** déclaration par laquelle un nœud indique à un voisin qu'il ne possède pas de données liées (au sens strict défini par les procédures CCR) et que des signaux "lecture seule" ou "sortie prématurée" ont été reçus de tous les autres voisins éventuels dans l'arbre de transaction.

**3.2.33 mode de commande polarisé:** mode de communication au cours d'un dialogue dans lequel une seule invocation TPSUI impliquée dans le dialogue est autorisée à effectuer la commande à un instant donné.

**3.2.34 machine protocolaire (PM, *protocol machine*):** terme générique désignant soit une machine protocolaire de traitement de transaction, soit une machine protocolaire de canal.

**3.2.35 transaction répartie prise en charge par le fournisseur:** transaction pour laquelle le fournisseur du service OSI TP est responsable du maintien des propriétés ACID.

**3.2.36 signal "lecture seule":** déclaration par laquelle un nœud indique à un supérieur que ses données liées n'ont pas été modifiées par la transaction, que des signaux "lecture seule" ou "sortie prématurée" ont été reçus en provenance de tous les nœuds subordonnés éventuels dans l'arbre de transaction et qu'un compte rendu concernant le résultat de la transaction n'est pas nécessaire.

**3.2.37 signal "prêt":** déclaration par laquelle un nœud indique à un voisin qu'un enregistrement de journalisation "prêt" a été stocké. Dans le cas où il existe plusieurs voisins, le destinataire du signal est celui qui n'a pas envoyé de signal "prêt", de signal "une phase", de signal "lecture seule" ou de signal "sortie prématurée" lorsque l'enregistrement de journalisation "prêt" a été stocké.

NOTE – Il en résulte que le signal "prêt" exclut le signal "une phase", le signal "lecture seule" ou le signal "sortie prématurée".

**3.2.38 état "prêt à l'engagement":** état des données liées dans lequel, jusqu'à la fin de la transaction par engagement ou retour-arrière, les données liées peuvent être libérées dans leur état initial ou final.

**3.2.39 rétablissement:** action, effectuée après une défaillance, visant à éliminer les conséquences non souhaitables.

**3.2.40 journal de rétablissement:** stockage dans une mémoire sûre, utilisé pour enregistrer des données et des informations d'états à des fins de redémarrage et de rétablissement.

**3.2.41 ressource distante:** ressources situées sur un système ouvert réel distinct du système ouvert réel qui en fait la demande.

**3.2.42 ressource:** données et capacités de traitement nécessaires à une invocation TPSUI pour réaliser la partie de la transaction dont elle est responsable.

**3.2.43 retour-arrière; retour-arrière de transaction:** achèvement d'une transaction avec rétablissement des données liées dans leur état initial.

NOTE – Les termes "engagement" et "retour-arrière" ont un domaine d'application différent de celui qui est défini dans la Rec. UIT-T X.851 | ISO/CEI 9804. La présente Recommandation et ISO/CEI 10026 traitent de l'engagement et du retour-arrière de la totalité d'une transaction, alors que la Rec. UIT-T X.851 | ISO/CEI 9804 traite de l'engagement et du retour-arrière d'une seule branche d'une action atomique.

**3.2.44 racine:** entité unique dans un arbre qui n'a aucun supérieur et un ou plusieurs subordonnés.

**3.2.45 mémoire sûre:** mémoire fiable non volatile dans laquelle les informations emmagasinées continuent à exister après tout type de défaillance pouvant faire l'objet d'un rétablissement au sein d'un système ouvert réel.

**3.2.46 mode de commande partagé:** mode de communication au cours d'un dialogue dans lequel les invocations TPSUI impliquées dans le dialogue exercent toutes deux la commande.

**3.2.47 procédures d'engagement statique:** procédure d'engagement en deux phases avec la contrainte que la décision d'engagement est prise par la racine de l'arbre de transaction et propagée au sein de cet arbre.

NOTE – Cette procédure est équivalente aux procédures d'engagement des Rec. UIT de la série X.860 (1992) | ISO/CEI 10026:1992 et ISO/CEI 10026:1995.

**3.2.48 subordonné:** entité qui accepte une relation (proposée par un supérieur).

**3.2.49 sous-arbre subordonné:** sous-arbre d'un nœud subordonné.

**3.2.50 sous-arbre:** sous-ensemble d'un arbre. Le sous-arbre d'un nœud donné contient:

- a) le nœud lui-même, appelé nœud racine du sous-arbre; et
- b) les sous-arbres de tout nœud subordonné du nœud racine, et ainsi de suite de façon récursive.

Un nœud feuille constitue son propre sous-arbre.

**3.2.51 supérieur:** entité initialisant une relation.

**3.2.52 transaction:** ensemble d'opérations liées caractérisées par quatre propriétés: l'atomicité, la cohérence, l'isolation et la durabilité. Une transaction est identifiée d'une manière non ambiguë par un identificateur de transaction.

NOTE – Le terme "transaction" sera utilisé, par souci de concision, comme synonyme du terme "transaction répartie prise en charge par le fournisseur" à partir du 7.8.

**3.2.53 branche de transaction:** partie d'une transaction répartie, qui est exécutée par un couple d'invocations TPSUI participant à un dialogue.

NOTE – Le terme "branche de transaction" sera utilisé, par souci de concision, comme synonyme du terme "branche de transaction répartie prise en charge par le fournisseur" à partir du 7.8.

**3.2.54 identificateur de branche de transaction:** identificateur non ambigu pour une branche particulière d'une transaction donnée.

**3.2.55 données de transaction:** données qui font l'objet d'un accès et d'une manipulation par l'invocation TPSUI au sein du domaine de validité d'une transaction prise en charge par un fournisseur ou d'une transaction prise en charge par une application; les données de transaction sont soit des "données liées", soit des "données à longue durée de vie".

**3.2.56 arrière-plan de transaction:** l'arrière-plan du nœud B, vu du nœud A, se constitue du nœud B et de l'arrière-plan de transaction (vu du nœud B) de tous les voisins de B, à l'exception du nœud A, qui participent ou ont participé à la transaction actuelle sur une branche de transaction avec le nœud B.

NOTE – Les nœuds qui ne participent plus à la transaction parce qu'ils ont émis des signaux "lecture seule" ou "sortie prématurée" continuent à faire partie de l'arrière-plan de cette dernière jusqu'à la sa fin.

**3.2.57 identificateur de transaction:** identificateur globalement non ambigu d'une transaction donnée.

**3.2.58 journalisation de transaction:** enregistrement des informations d'état et des données d'un nœud dans un journal de rétablissement.

**3.2.59 élément de service d'application de traitement transactionnel (TPASE, *transaction processing application*):** partie de la machine protocolaire de traitement transactionnel (TPPM) qui gère le protocole OSI TP pour une seule association d'application.

**3.2.60 machine protocolaire de traitement transactionnel (TPPM, *transaction processing protocol machine*):** fournisseur du service OSI TP pour une invocation TPSUI et une seule. Une machine TPPM traite le protocole TP pour toutes les associations utilisées pour l'activité de son invocation TPSUI.

**3.2.61 fournisseur du service de traitement transactionnel (TPSP, *transaction processing service provider*):** fournisseur du service OSI TP pour toutes les invocations TPSUI impliquées dans un arbre de dialogue donné. Le fournisseur TPSP traite plusieurs invocations de processus d'application (API) et constitue une vue conceptuelle globale du service OSI TP.

**3.2.62 utilisateur du service de traitement transactionnel (TPSU, *transaction processing service user*):** utilisateur du service OSI TP; ce terme fait référence à un ensemble spécifique de capacités de traitement au sein d'un processus d'application.

**3.2.63 invocation d'utilisateur TPSU (TPSUI, *transaction processing service user invocation*):** instance particulière d'utilisateur TPSU qui exécute des fonctions pour une occasion spécifique de traitement de l'information.

**3.2.64 titre TPSU:** nom non ambigu qui identifie un utilisateur TPSU donné au sein du domaine de validité du processus d'application auquel appartient cet utilisateur TPSU. Le titre TPSU contient d'une manière implicite le type des capacités de traitement de l'utilisateur TPSU.

**3.2.65 rétablissement de transaction:** action effectuée après une défaillance, visant à mettre toutes les données liées de cette transaction dans un état cohérent.

**3.2.66 arbre de transaction:** arbre dont les nœuds sont les entités et les branches sont les relations entre ces entités.

**3.2.67 arbre:** ensemble d'entités liées disposées d'une manière hiérarchique et connectée par des relations.

**3.2.68 succession non chaînée:** succession - dans un même dialogue - de branches non contiguës (prises en charge par le fournisseur) d'une transaction, visant un but commun.

**3.2.69 dialogue non coordonné:** dialogue dont le niveau de coordination actuel est "néant".

**3.2.70 élément ASE utilisateur:** élément ASE propre à l'application.

## 4 Abréviations

Les abréviations suivantes s'appliquent aux fins de la présente Recommandation:

|       |  |
|-------|--|
| ACID  | atomicité, cohérence, isolation et durabilité ( <i>atomicity, consistency, isolation, and durability</i> )                       |
| ACSE  | élément de service de contrôle d'association ( <i>association control service element</i> )                                      |
| AE    | entité d'application ( <i>application-entity</i> )   |
| AEI   | invocation d'entité d'application ( <i>application-entity invocation</i> )   |
| ALS   | structure de la couche Application ( <i>application layer structure</i> )  |
| AP    | processus d'application ( <i>application-process</i> )   |
| APDU  | unité de données de protocole d'application ( <i>application-protocol-data-unit</i> )  |
| API   | invocation de processus d'application ( <i>application-process invocation</i> )  |
| ASE   | élément de service d'application ( <i>application service element</i> )  |
| CCR   | engagement, concurrence et rétablissement ( <i>commitment, concurrency, and recovery</i> )                                       |
| CPM   | machine protocolaire de canaux ( <i>channel protocol machine</i> )   |
| MACF  | fonction de contrôle d'associations multiples ( <i>multiple association control function</i> )                                   |
| OSI   | interconnexion des systèmes ouverts ( <i>open systems interconnection</i> )  |
| OSIE  | environnement d'interconnexion des systèmes ouverts ( <i>open systems interconnection environment</i> )                          |
| PICS  | déclaration de conformité d'implémentation de protocole ( <i>protocol implementation conformance statement</i> )                 |
| PM    | machine protocolaire (machine TPPM ou machine CPM) ( <i>protocol machine</i> )   |
| PSAP  | point d'accès au service de présentation ( <i>presentation service access point</i> )  |
| PSDU  | unité de données du service de présentation ( <i>presentation-service-data-unit</i> )  |
| RDA   | accès à une base de données distante ( <i>remote database access</i> )   |
| ROSE  | élément de service d'opérations distantes ( <i>remote operations service element</i> )   |
| SACF  | fonction de contrôle d'association unique ( <i>single association control function</i> )   |
| SAO   | objet d'association unique ( <i>single association object</i> )  |
| TP    | traitement transactionnel ( <i>transaction processing</i> )  |
| TPASE | élément de service d'application pour le traitement transactionnel ( <i>transaction processing application service element</i> ) |
| TPPM  | machine protocolaire de traitement transactionnel ( <i>transaction processing protocol machine</i> )                             |
| TPSP  | fournisseur du service de traitement transactionnel ( <i>transaction processing service provider</i> )                           |
| TPSU  | utilisateur du service de traitement transactionnel ( <i>transaction processing service user</i> )                               |
| TPSUI | invocation d'utilisateur du service de traitement transactionnel ( <i>transaction processing service user invocation</i> )       |
| U-ASE | élément de service d'application utilisateur ( <i>user-application service element</i> )   |

## 5 Conventions

La présente Recommandation utilise les conventions définies dans la Rec. UIT-T X.210 | ISO/CEI 10731, telles qu'elles s'appliquent au service OSI TP.



## 6 Définition des besoins

### 6.1 Introduction

Le présent sous-paragraphe résume les besoins du traitement OSI TP. Il contient des besoins qui sont pris en compte par la présente Recommandation, ainsi que des besoins qui ne sont pas pris en compte et qui nécessitent un complément d'étude. Ces besoins supplémentaires sont candidats pour une normalisation ultérieure.

### 6.2 Besoins de l'utilisateur

Afin de satisfaire les besoins de l'utilisateur, les normes de traitement OSI TP:

- a) définissent des procédures qui prennent en charge des transactions réparties, conformément à l'analyse faite au 7.2. Ces procédures:
  - 1) permettent l'organisation d'une transaction répartie sous la forme d'un arbre de transaction;
  - 2) assurent une coordination entre participants multiples (qui contient un engagement avec participants multiples), y compris pour les ressources locales;
  - 3) permettent de rétablir, après l'apparition d'une défaillance, un état cohérent pour l'état/le contexte d'une transaction répartie ainsi que pour les données liées;
  - 4) permettent de détecter la défaillance d'une transaction répartie en vue de réaliser les propriétés ACID;
  - 5) permettent de redémarrer une transaction répartie après la réussite du rétablissement de l'état; et
  - 6) indiquent le statut d'achèvement d'une transaction;
- b) permettent de délimiter une succession de transactions liées logiquement;
- c) permettent de regrouper des utilisateurs TPSU à l'intérieur d'un processus d'application;
- d) répondent à un ou plusieurs des besoins de sécurité suivants:

NOTE – La fourniture de la sécurité fera l'objet d'une normalisation ultérieure sous la forme d'un amendement.

- 1) contrôle d'accès: il doit être possible de mettre en œuvre des stratégies multiples de contrôle d'accès. Les types de contrôle d'accès qui seront fournis au minimum sont décrits dans la Rec. X.800 du CCITT | ISO/CEI 7498-2 (contrôle d'accès imposé par l'administration et avec sélection dynamique, contrôle d'accès sur la base de règles et sur la base de l'identité);
  - 2) degré de finesse du contrôle d'accès: il devra être possible de classer par groupes des objets OSI TP, afin de simplifier la spécification du contrôle d'accès et de permettre la distribution de la base de données d'autorisation. Une telle classification doit être faite à des fins d'optimisation et ne doit pas se substituer à un audit détaillé;
  - 3) authentification entre:
    - i) des invocations TPSUI correspondantes;
    - ii) des machines TPPM;
    - iii) des invocations d'entité d'application; et
    - iv) des utilisateurs TPSU et des machines TPPM. On considère toutefois que ce point constitue un problème local.
  - 4) non-répudiation: prévient le reniement de la participation à une transaction ou à un dialogue donné;
  - 5) confidentialité: prévient la réception non autorisée de tout ou partie de l'information échangée dans un arbre de dialogue;
  - 6) intégrité: détecte les modifications non autorisées de tout ou partie de l'information échangée au sein d'un arbre de dialogue; et
  - 7) audit: enregistre les événements de sécurité significatifs qui se manifestent au sein d'un arbre de dialogue;
- e) permettent des essais de conformité du protocole défini par la Rec. UIT-T X.862 | ISO/CEI 10026-3 et délimitent clairement les prescriptions de conformité statique (au moyen du formulaire PICS défini dans la Rec. UIT-T X.863 | ISO/CEI 10026-4).

### 6.3 Besoins concernant la modélisation

Le modèle OSI TP fournit un modèle de traitement réparti de transaction et les mécanismes de communication nécessaires à sa réalisation. Ces mécanismes sont conformes à l'architecture OSI définie dans la Rec. UIT-T X.200 | ISO/CEI 7498-1 et dans l'ISO/CEI 9545; ils prennent en compte les points suivants:

- a) définition de mécanismes permettant de subdiviser en transactions les interactions entre les processus d'application de deux systèmes ouverts ou plus. Ces mécanismes répondent en particulier aux besoins suivants:
  - 1) indication du statut d'achèvement d'une transaction;
  - 2) prise en charge de transactions qui n'ont pas besoin de la totalité des mécanismes d'engagement réparti pour garantir les propriétés ACID, lorsque l'application en est responsable; et
  - 3) souplesse afin de pouvoir faire le choix de la méthode de transfert de données en fonction de la sémantique de la transaction;
- b) spécification de mécanismes d'utilisation des services de la couche Présentation;
- c) niveau acceptable pour les performances et l'efficacité des procédures; et
- d) procédures couvrant une large gamme de besoins (transactions courtes ou longues, simples ou complexes).

NOTE – Certaines de ces procédures sont candidates à une normalisation ultérieure.

### 6.4 Besoins concernant le service et le protocole OSI TP

Le service et le protocole OSI TP répondent aux besoins suivants:

- a) souplesse de traitement de situations de charge variable;
- b) exploitation efficace dans des situations de forte charge, de charge faible ou de charge en rafales;
- c) traitement efficace d'unités APDU courtes;
- d) temps de réponse acceptable pour les utilisateurs;
- e) résistance aux défaillances, y compris les moyens nécessaires pour le rétablissement et le redémarrage du traitement après que les défaillances aient été corrigées ou contournées;
- f) utilisation optimale des ressources; et
- g) minimisation de la dépendance de la commande des ressources locales vis-à-vis des communications.

Afin de satisfaire ces besoins, le protocole OSI TP:

- a) optimise l'utilisation du service de la couche Présentation;
- b) minimise la charge de communication nécessaire pour traiter chaque transaction. En particulier, le protocole OSI TP limite le nombre d'échanges aller-retour requis par les protocoles de communication de manière à ne pas dépasser le nombre d'échanges requis par la sémantique de l'application;
- c) optimise les opérations pour les besoins du traitement de transaction de volume important; et
- d) optimise les opérations pour les besoins du cas normal plutôt que pour les besoins de cas exceptionnels.

## 7 Concepts du traitement transactionnel réparti

### 7.1 Transaction

Une transaction est un ensemble d'opérations liées, caractérisées par les quatre propriétés d'atomicité, de cohérence, d'isolation et de durabilité.

### 7.2 Transaction répartie

Une transaction qui fait intervenir plus d'un système ouvert est appelée transaction répartie.

Une transaction répartie se compose d'au moins autant de parties qu'il y a de systèmes ouverts impliqués dans son exécution. Une partie de la transaction répartie dépend, au sein de chaque système, d'une entité appelée utilisateur du service TP (TPSU, *transaction processing service user*).

L'utilisateur TPSU est l'utilisateur du service OSI TP; il fait référence à un ensemble spécifique de capacités de traitement au sein d'un processus d'application. Le nombre d'utilisateurs TPSU dans un processus d'application donné peut être zéro, un ou plus.

NOTE – Un utilisateur TPSU peut être réparti à son tour au sein d'un processus d'application. La présente Recommandation n'exclut pas une telle extension mais ne la traite pas, car la répartition au sein d'un système est en-dehors du domaine d'application de l'interconnexion OSI.

Une invocation d'utilisateur TPSU (TPSUI, *transaction processing service user invocation*) modélise – du point de vue de l'environnement OSIE – une instance donnée d'utilisateur TPSU, située au sein d'une invocation de processus d'application et qui exécute des fonctions pour une occasion spécifique de traitement de l'information.

Le maintien des quatre propriétés des transactions nécessite une coordination entre les invocations TPSUI qui exécutent une transaction répartie. Une telle coordination nécessite à son tour une communication entre ces invocations.

### 7.3 Données de transaction et niveau de coordination

Une invocation TPSUI manipule des données qui se trouvent dans le domaine de validité d'une transaction et place ces données dans leur état final ou dans leur état initial, selon que la transaction procède à un engagement ou à un retour-arrière. Ces données sont appelées données de transaction.

Le mécanisme utilisé pour coordonner le résultat d'une transaction est déterminé par le niveau de coordination. Les trois niveaux de coordinations suivants sont pris en charge pour l'invocation TPSUI:

- a) "engagement", lorsque le fournisseur TPSP est responsable de la délimitation des transactions et du compte rendu du résultat de la transaction, y compris dans les cas d'apparition d'une défaillance lors de la fin de la transaction; le fournisseur TPSP utilise un mécanisme d'engagement en deux phases pour prendre en charge ce niveau de coordination:
- b) "engagement en une phase", lorsque le fournisseur TPSP est responsable de la délimitation des transactions et du compte rendu du résultat de la transaction, sauf dans le cas d'apparition d'une défaillance lors de la fin de la transaction; l'invocation TPSUI est alors responsable de la détermination du résultat et de toute mesure de rétablissement par des moyens autres que ceux fournis par le traitement de transaction; ou
- c) "néant", lorsque l'invocation TPSUI est responsable de la délimitation des transactions et de toute action de rétablissement éventuellement nécessaire.

L'invocation TPSUI manipule des données de transaction lors d'une transaction. Les données de transaction qui sont protégées par l'utilisation du niveau de coordination "engagement" sont appelées données liées (comme défini dans la Rec. UIT-T X.851 | ISO/CEI 9804). Les données de transaction qui sont protégées par l'application sont appelées "données à longue durée de vie". Le Tableau 1 indique les combinaisons autorisées pour les données de transaction et les niveaux de coordination.

**Tableau 1/X.860 – Combinaisons autorisées pour les données de transaction et les niveaux de coordination**

| Données de transaction        | Niveau de coordination |                         |       |
|-------------------------------|------------------------|-------------------------|-------|
|                               | Engagement             | Engagement en une phase | Néant |
| données liées                 | OUI                    | NON                     | NON   |
| données à longue durée de vie | OUI                    | OUI                     | OUI   |

NOTE – Les mécanismes éventuellement nécessaires pour le maintien des propriétés ACID des données à longue durée de vie sont en dehors du domaine d'application de la présente Recommandation.

### 7.4 Relations d'arbre

Dans la présente Recommandation, un arbre se constitue d'un ensemble d'entités liées ordonnées d'une manière hiérarchique et connectées par des relations. Deux entités liées par une relation sont des voisins. Une relation individuelle définit des rôles pour les deux voisins:

- le supérieur de la relation est l'entité qui a initialisé la relation; et
- le subordonné de la relation est l'entité qui l'a acceptée.

Toute entité ne peut avoir qu'un seul supérieur. Une entité appartenant déjà à un arbre ne peut pas se joindre à un autre arbre. Il en résulte qu'un arbre ne contient pas de cycles.

## 7.5 Dialogue

Les invocations TPSUI communiquent entre elles au moyen de relations d'homologue à homologue. Une relation de ce type entre deux invocations TPSUI est appelée dialogue.

Au cours d'un dialogue, les invocations TPSUI communiquent pour:

- a) transférer des données;
- b) notifier des erreurs;
- c) démarrer ou terminer une transaction;
- d) terminer leur dialogue de manière normale ou abrupte; et
- e) effectuer des activités de prise de contact.

La commande du dialogue peut être faite de deux manières:

- a) mode de commande polarisé, dans lequel une seule invocation TPSUI exerce la commande du dialogue à un instant donné; et
- b) mode de commande partagé, dans lequel les deux invocations TPSUI exercent simultanément la commande du dialogue.

Dans le mode de commande polarisé, une invocation TPSUI doit disposer de la commande du dialogue pour lancer une demande autre que l'une des suivantes:

- a) notification d'erreur;
- b) retour-arrière de transaction;
- c) sortie prématurée d'une transaction;
- d) terminaison abrupte du dialogue;
- e) demande de la commande.

## 7.6 Arbre de dialogue

Un arbre de dialogue est un arbre dont les invocations TPSUI constituent les entités et les dialogues constituent les relations entre les entités. Le but d'un arbre de dialogue est de prendre en charge une succession d'une ou de plusieurs transactions.

L'invocation TPSUI qui établit le dialogue dans l'arbre est appelée supérieur direct de l'invocation TPSUI avec laquelle le dialogue est établi; cette dernière est appelée subordonné direct de l'invocation TPSUI supérieure adjacente.

L'invocation TPSUI qui n'a pas de supérieur dans l'arbre de dialogue est appelée invocation TPSUI racine. Une invocation TPSUI qui n'a pas de subordonné est appelée invocation TPSUI feuille. Une invocation TPSUI qui possède à la fois un supérieur et au moins un subordonné est appelée invocation TPSUI intermédiaire.

## 7.7 Branches d'une transaction

Le fournisseur TPSP fournit aux invocations TPSUI, lorsque cela est nécessaire, un service d'engagement utilisable pour un dialogue donné. L'utilisation éventuelle de ce service par les invocations TPSUI pour le dialogue est déterminée par l'une des valeurs suivantes de niveau de coordination:

- a) "engagement" lorsque les invocations TPSUI utilisent le service d'engagement en deux phases;
- b) "engagement en une phase" lorsque les invocations TPSUI utilisent le service d'engagement en une phase; ou
- c) "néant" lorsque les invocations TPSUI n'utilisent pas de service d'engagement.

La partie d'une transaction répartie qui est traitée par un couple d'utilisateurs TPSUI partageant un dialogue est appelée branche de transaction.

Il existe deux types de base pour une branche de transaction, définis en fonction du partage de responsabilité entre le fournisseur TPSP et les invocations TPSUI:

- a) branches d'une transaction prises en charge par l'application: il s'agit de branches d'une transaction traitant un dialogue avec un niveau de coordination "néant".

Pour de telles branches de transaction, l'invocation TPSUI est responsable du maintien des propriétés ACID, du rétablissement et de la délimitation des branches de la transaction.

Le fournisseur TPSP offre uniquement l'accès au transfert de données, les notifications d'erreur et les services de commande de dialogue; il n'a pas connaissance du début et de la fin de l'existence de ces branches.

- b) branches d'une transaction prises en charge par le fournisseur: il s'agit de branches d'une transaction traitant un dialogue avec un niveau de coordination "engagement" ou "engagement en une phase".

Pour des branches prises en charge par le fournisseur avec un niveau de coordination égal à "engagement", le fournisseur TPSP est responsable de la coordination du maintien des propriétés ACID (et utilise par conséquent des identificateurs de transaction globalement non ambigus, des engagements, etc.), du rétablissement et de la délimitation des branches de la transaction; il est également en charge de la fourniture de l'accès aux autres services.

Pour des branches prises en charge par le fournisseur avec un niveau de coordination égal à "engagement en une phase", l'invocation TPSUI supérieure déclare qu'il n'y aura pas de données liées et ne demande pas de compte-rendu fiable du résultat de la transaction. Le fournisseur TPSP est responsable de la délimitation des branches de la transaction et de la fourniture à l'invocation TPSUI supérieure du compte-rendu du résultat de la transaction en l'absence de défaillances.

Le terme "branche d'une transaction prise en charge par le fournisseur" sera remplacé dans le texte qui suit par le terme "branche de transaction", par souci de concision. Le terme "branche d'une transaction prise en charge par l'application" sera utilisé d'une manière explicite en cas de nécessité.

## 7.8 Arbre de transaction

Un arbre de transaction contient des nœuds (invocations TPSUI avec leurs machines TPPM), qui constituent les entités et des branches de transaction, qui constituent les relations entre les entités. Un arbre de transaction a pour but de prendre en charge une transaction unique.

Un arbre de transaction est créé à partir d'un arbre de dialogue existant. Les nœuds de l'arbre de transaction sont ceux de l'arbre de dialogue. L'invocation TPSUI qui démarre la branche de transaction dans l'arbre de transaction est appelée supérieur direct de l'invocation TPSUI avec laquelle la branche de transaction est établie; cette dernière est appelée subordonné direct de l'invocation TPSUI supérieure adjacente.

L'invocation TPSUI qui n'a pas de supérieur dans l'arbre de transaction est appelée TPSUI racine. Une invocation TPSUI qui n'a pas de subordonné est appelée invocation TPSUI feuille. Une invocation TPSUI qui possède à la fois un supérieur et au moins un subordonné est appelée invocation TPSUI intermédiaire.

Lorsqu'une décision d'engagement est prise dans un arbre de transaction, le fournisseur TPSP garantit que tous les services liés au transfert de données entre invocations TPSUI, aux notifications d'erreur et aux opérations de prise de contact ont été effectués avec succès pour toutes les branches de transaction pour lesquelles le mode de commande avec polarisation a été sélectionné.

## 7.9 Canaux

Les invocations d'entité d'application ont besoin de communiquer entre elles pendant un rétablissement sans faire intervenir d'invocation TPSUI. Les canaux répondent à ce besoin. Un canal est modélisé sous la forme d'une relation sur une association; il a pour but le rétablissement d'une ou de plusieurs branches de transaction.

Un canal est établi entre deux invocations d'entité d'application au moyen d'une association existante ou établie à cet effet. Les canaux sont créés et supprimés par une machine protocolaire de canal (CPM, *channel protocol machine*). Les machines CPM situées dans deux systèmes homologues peuvent établir un ou plusieurs canaux entre elles à des fins de rétablissement.

Un canal donné possède les propriétés suivantes:

- a) il n'est pas directement visible pour les invocations TPSUI. Il n'existe donc pas de primitives OSI TP se référant aux canaux dans le service OSI TP; et
- b) une machine CPM attribue un canal à une machine TPPM à des fins de rétablissement.

Les canaux sont modélisés, à des fins de rétablissement, comme étant utilisés pour rétablir une seule branche de transaction à la fois.

## 7.10 Prise de contact

Les invocations TPSUI peuvent avoir besoin de synchroniser leurs activités en vue d'atteindre un point de traitement mutuellement agréé. Les informations sémantiques d'un tel point dépendent de l'application.

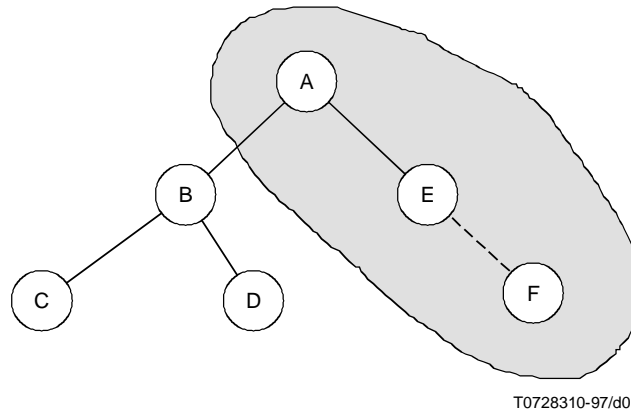
Le fournisseur TPSP offre aux invocations TPSUI, lorsque cela est nécessaire, un service de prise de contact disponible pendant la durée du dialogue; ce service constitue un outil de structuration de l'application qui est indépendant du mode de commande des dialogues.

## 7.11 Arrière-plan

### 7.11.1 Arrière-plan de transaction

Un arbre de transaction est construit comme indiqué au 7.8. Un arrière-plan de transaction est une région de l'arbre de transaction vue de la perspective d'un nœud donné dans une direction donnée.

La Figure 1 représente un arbre de transaction dont le nœud F a quitté la transaction avant sa fin (il a, par exemple, émis un signal "lecture seule" ou "sortie prématurée" à destination du nœud E); l'arrière-plan de transaction du nœud A vu du nœud B se constitue des nœuds A, E et F.



T0728310-97/d01

Figure 1/X.860 – Arrière-plan de transaction du nœud A, vu du nœud B

### 7.11.2 Arrière-plan d'engagement

L'arrière-plan d'engagement d'un nœud se constitue de l'ensemble des nœuds voisins de l'arbre de transaction en provenance desquels des signaux "prêt" ont été reçus, ainsi que, d'une manière récursive, de l'arrière-plan d'engagement de ces nœuds. Si, dans l'exemple précédent:

- 1) le nœud F émet un signal "lecture seule" à destination du nœud E;
- 2) le nœud E émet un signal "prêt" à destination du nœud A; et
- 3) le nœud A émet un signal "prêt" à destination du nœud B;

l'arrière-plan d'engagement du nœud B se constitue alors des nœuds A et E, mais non du nœud F.

NOTE – L'arrière-plan d'engagement d'un nœud sera identique au sous-arbre de transaction de ce nœud dans le cas de procédures d'engagement statique en deux phases qui n'utilisent pas les signaux "lecture seule", "sortie prématurée" ou "engagement en une phase".

## 8 Modèle du service OSI TP

### 8.1 Nature du service OSI TP

Le terme service OSI TP désigne le service qui est offert par le fournisseur TPSP et qui est utilisé par les invocations TPSUI.

Les fonctions suivantes sont associées au service OSI TP:

- a) établissement, maintien et terminaison du dialogue entre deux invocations TPSUI. Le service OSI TP:
  - 1) permet la sélection d'un utilisateur TPSU au sein d'un ensemble d'utilisateurs TPSU. Le titre d'utilisateur TPSU est utilisé à cette fin;
  - 2) garantit la compatibilité des attributs demandés par l'invocation TPSUI appelante avec les attributs de l'invocation TPSUI appelée. Si tel est le cas, le dialogue est établi entre une nouvelle invocation de l'utilisateur TPSU appelé et l'invocation TPSUI appelante;

NOTE – Du point de vue de l'environnement OSIE, le terme "nouvelle invocation" signifie une invocation d'utilisateur TPSU qui ne figure pas actuellement dans l'environnement OSIE. Le fait que la nouvelle invocation corresponde, dans un système ouvert réel, à une nouvelle instance d'utilisateur TPSU ou à une instance ancienne qui est réutilisée est un problème local.

- 3) fournit aux deux invocations TPSUI les moyens d'interaction et d'accès à des ressources distantes, en les impliquant éventuellement dans une transaction.
- b) coordination globale et fiable des ressources en fonction du niveau de coordination choisi, dans le but de terminer une transaction avec ou sans succès. Ceci permet aux ressources d'atteindre un état cohérent, sauf peut-être en cas de décisions heuristiques. Les propriétés ACID s'appliquent à la totalité de la transaction, en particulier aux ressources locales et distantes.

La coordination des ressources peut être effectuée entièrement par le fournisseur TPSP, ou être partagée entre le fournisseur TPSP et l'invocation TPSUI, ce qui permet la commande et la gestion des ressources locales par le fournisseur TPSP, l'invocation TPSUI ou par les deux. Dans le cas de coordination partagée, l'invocation TPSUI regroupe tout ou partie des informations concernées dans ses ressources locales et commande l'engagement ou le retour-arrière ultérieur de ces ressources en fonction de la décision du fournisseur TPSP.

Le service OSI TP:

- 1) englobe les fonctions nécessaires à la coordination de toutes les ressources distantes en vue de garantir l'application des propriétés ACID. Le fournisseur TPSP est responsable de la coordination de l'engagement ou du retour-arrière correct de la totalité de l'ensemble de ressources distantes lorsque la transaction se termine;
- 2) fournit la capacité d'inclure des ressources locales dans la terminaison de la transaction. Le fournisseur TPSP peut, compte tenu son partage de responsabilités avec les invocations TPSUI, se comporter d'une des manières suivantes:
  - i) inclure les ressources locales avec les ressources distantes dans la terminaison de la transaction, ou
  - ii) fournir aux invocations TPSUI toutes les informations dont elles ont besoin pour inclure correctement les (autres) ressources locales, de sorte que les propriétés ACID puissent s'appliquer aux ressources.

Le fournisseur TPSP garantit, par l'exécution du protocole approprié, que toutes les ressources respectent les propriétés ACID. Le fournisseur TPSP contient en particulier des mécanismes de rétablissement adéquats permettant de rétablir un état cohérent pour toutes les ressources après une défaillance et de reprendre ensuite, le traitement de la transaction dans un état cohérent, une fois que possible.

## **8.2 Règles s'appliquant aux arbres de dialogue**

### **8.2.1 Croissance des arbres de dialogue**

Une invocation TPSUI peut activer d'autres invocations TPSUI distantes pour exécuter des parties d'une transaction répartie; le système ouvert distant appelle pour ce faire une nouvelle invocation TPSUI et établit ensuite un dialogue avec elle (voir également 8.2.3 et 8.4.1). Ceci permet d'ajouter un nouveau dialogue à l'arbre de dialogue..

NOTE – Le terme "nouvelle invocation" désigne, du point de vue de l'environnement OSIE, une invocation d'utilisateur TPSU qui ne figure pas actuellement dans cet environnement. Le fait que la nouvelle invocation soit mappée dans un système ouvert réel à une nouvelle instance d'utilisateur TPSU ou à une instance existante qui est réutilisée est un problème local.

Les attributs du dialogue qui indiquent le type de traitement de transaction à effectuer sont spécifiés au moment de l'établissement du dialogue. Ils déterminent le sous-ensemble de fonctionnalités de communication à sélectionner pour ce dialogue. Ils peuvent comprendre:

- a) le mode de commande polarisé ou le mode de commande partagé;
- b) le service de prise de contact; et
- c) le service d'engagement en deux phases ou le service d'engagement en une phase.

Un dialogue non coordonné (avec un niveau de coordination initial "néant") peut être ajouté à l'arbre de dialogue à tout moment. Un dialogue coordonné (avec un niveau de coordination "engagement") ne peut être ajouté qu'au moment où il est permis de démarrer une transaction ou d'ajouter une branche de transaction à la transaction en cours.

Une invocation TPSUI peut établir des dialogues avec une ou plusieurs invocations TPSUI subordonnées. Deux invocations TPSUI ne partagent toutefois qu'un seul dialogue au plus. La communication peut avoir lieu pour certains dialogues ou simultanément pour tous les dialogues d'une invocation TPSUI. Tous les dialogues d'une invocation TPSUI appartiennent au même arbre de dialogue.

### **8.2.2 Elagage des arbres de dialogue**

Deux invocations TPSUI mettent fin à leur dialogue lorsqu'elles n'ont plus besoin de communiquer entre elles. Elles peuvent terminer le dialogue à tout moment, à condition de garantir la conservation des quatre propriétés ACID.

La terminaison normale d'un dialogue ne peut avoir lieu que si, et seulement si, aucune branche de transaction n'est en cours pour ce dialogue. La terminaison d'un dialogue est possible:

- a) lorsque le niveau de coordination est "néant"; ou
- b) lorsque la branche de transaction en cours est terminée et que la branche suivante n'a pas encore démarré.

La terminaison d'un dialogue peut également survenir après une défaillance de la communication ou après un écroulement de nœud. Dans ce cas, la branche de transaction correspondante peut également se terminer avec le dialogue.

Il n'est pas nécessaire de mettre fin aux dialogues dans le sous-arbre de l'invocation TPSUI subordonnée lorsque le dialogue entre deux invocations TPSUI se termine. Il est ainsi possible de créer un nouvel arbre de dialogue qui fait partie d'un arbre de dialogue déjà établi. Ce nouvel arbre de dialogue est indépendant de l'arbre de dialogue duquel il est issu. Le nœud intermédiaire qui a terminé le dialogue avec son supérieur devient alors la racine du nouvel arbre de dialogue.

La structure des arbres de dialogue se modifie au fur et à mesure que des dialogues sont établis et se terminent.

### **8.2.3 Prise en charge des arbres de dialogue**

Un dialogue entre deux invocations TPSUI est pris en charge par une seule association d'application.

Il existe une correspondance biunivoque permanente entre un dialogue donné et l'association d'application impliquée. Il est toutefois nécessaire de faire une distinction entre la durée de vie d'un dialogue et la durée de vie d'une association d'application car cette dernière peut s'étendre sur la durée de vie d'un ou de plusieurs dialogues.

Le service OSI TP n'impose aucune contrainte pour l'établissement ou l'existence des associations d'application. En particulier, aucune contrainte n'est imposée à la structure d'arbre ou aux autres topologies de structure entre les invocations d'entité d'application. On considère donc que ces structures constituent un graphe de systèmes ouverts interconnectés.

Une association d'application doit être établie pour prendre en charge un dialogue:

- a) entre les invocations d'entité d'application prenant en charge les besoins de communication des utilisateurs TPSU concernés par le dialogue demandé;
- b) avec un contexte d'application qui répond aux besoins de communication des utilisateurs TPSU concernés par le dialogue demandé;
- c) avec une prise en charge de services de présentation et de session compatibles avec les besoins du dialogue demandé; et
- d) avec une qualité de service compatible avec les besoins du dialogue demandé.

### **8.2.4 Initiative des activités et structure d'arbre**

Les rôles de nœud supérieur et de nœud subordonné d'un dialogue de traitement de transaction ou d'une branche de transaction sont asymétriques du point de vue du protocole de traitement de transaction. Cette asymétrie correspond à l'hypothèse de base de ce modèle qui attribue habituellement au nœud supérieur le rôle d'instigateur des activités, tandis que le subordonné contribue à ces activités en réagissant à des demandes reçues du supérieur. Un subordonné joue d'une manière récursive le rôle de supérieur vis-à-vis de ses propres subordonnés.

Une application peut nécessiter, dans certains cas, le transfert de l'initiative et de la responsabilité finale de l'exécution ultérieure d'une tâche d'application d'un nœud vers un nœud voisin, c'est-à-dire d'un client vers un serveur. Le nœud qui souhaite abandonner l'initiative peut être celui qui a créé au départ la tâche qui est en cours de traitement dans la transaction concernée, mais ce nœud peut soit vouloir revenir à une position d'observateur (devenir un subordonné au sens du traitement de transaction), soit vouloir se déconnecter complètement de la transaction (au moins pour l'instant) avec l'intention d'en examiner ultérieurement l'aboutissement et les résultats détaillés indépendamment de l'exécution en cours.

L'opération par laquelle un nœud racine d'un arbre de transaction et un nœud subordonné voisin peuvent échanger leurs rôles présente de grandes difficultés, car des parties importantes des échanges de protocole doivent être réorientées. Il en résulte qu'un nœud doit pouvoir initialiser spontanément l'établissement d'un arbre de dialogue et l'établissement ultérieur d'un arbre de transaction tout en jouant dès le départ un rôle de subordonné dans l'arbre. Il peut alors définir l'essence d'une tâche d'application et transférer au nœud racine la prise en charge de son exécution.

Un dialogue de traitement de transaction dont l'établissement est demandé par un système distant se voit offrir une association adéquate déjà existante, qui doit être utilisée à cette fin par le système qui établit le dialogue.



Le protocole de traitement de transaction fait l'hypothèse que l'entité demandeuse, qui représente un contexte spécifique de traitement d'information, est un nœud (de traitement de transaction). Le dialogue entrant a besoin de trouver un nœud subordonné qui peut être, compte tenu d'une décision locale, le nœud demandeur mentionné ou un nœud nouvellement créé (ce dernier reprend alors le contexte de traitement d'information donné).

### **8.3 Règles s'appliquant aux arbres de transaction**

#### **8.3.1 Croissance des arbres de transaction**

Une nouvelle branche de transaction ne peut être ajoutée à un arbre de transaction qu'avant le démarrage des procédures de terminaison de la transaction (prière de se référer au 8.6).

Il existe deux manières pour faire croître un arbre de transaction:

- a) une nouvelle branche de transaction est ajoutée à l'arbre de transaction (tel qu'il est vu par le fournisseur TPSP) en établissant un nouveau dialogue coordonné (c'est-à-dire avec un niveau de coordination "engagement" ou "engagement en une phase"); et
- b) lorsqu'il est permis de modifier le niveau de coordination (voir également les "successions non chaînées" au 8.3.3), une nouvelle branche de transaction est ajoutée à l'arbre de transaction quand le niveau de coordination passe de "néant" à "engagement" ou à "engagement en une phase". Le niveau de coordination ne peut être modifié que par un nœud supérieur de l'arbre de dialogue.

#### **8.3.2 Durée de vie des arbres de transaction**

Un arbre de dialogue n'existe que pendant la durée d'une transaction unique.

Lorsqu'il est permis de modifier le niveau de coordination (voir aussi les "successions non chaînées" au 8.3.3), celui-ci peut uniquement passer à la valeur "néant" au moment de l'achèvement d'une branche de transaction.

La croissance et la terminaison d'un arbre de transaction ne sont pas des actions instantanées, car elles nécessitent toutes deux un certain nombre d'échanges élémentaires qui se propagent à travers l'arbre de transaction.

#### **8.3.3 Prise en charge des arbres de transaction**

L'existence d'un dialogue établi entre deux invocations TPSUI est une condition préalable à l'établissement d'une branche de transaction entre ces deux invocations TPSUI.

Il existe normalement une correspondance biunivoque entre un dialogue et une branche de transaction lorsque le niveau de coordination possède la valeur "engagement" ou "engagement en une phase". Le fournisseur TPSP a connaissance de la relation entre les dialogues d'un arbre de dialogue et les branches de transaction de l'arbre, ou des arbres de transaction correspondants; il coordonne leurs opérations combinées pour assurer, par exemple, la cohérence des informations sémantiques d'engagement pour l'ensemble des systèmes ouverts impliqués dans une transaction.

La racine d'un arbre de transaction ne correspond pas nécessairement à la racine de l'arbre de dialogue correspondant. Il existe une correspondance biunivoque entre les nœuds de l'arbre de transaction et les nœuds de l'arbre de dialogue qui le prend en charge, dans les limites de l'arbre de transaction et compte tenu des relations entre supérieur et subordonné. L'arbre de transaction et l'arbre de dialogue correspondant ont la même orientation.

Un dialogue avec un niveau de coordination "néant" ne prend en charge aucune branche de transaction de l'arbre de transaction.

Il est possible d'utiliser le même arbre de dialogue pour prendre en charge une succession de transactions distinctes. La relation entre les dialogues de l'arbre de dialogue persiste entre transactions successives. Une succession d'une ou de plusieurs branches de transaction peut avoir lieu dans les limites d'un dialogue donné. Deux types de successions sont possibles:

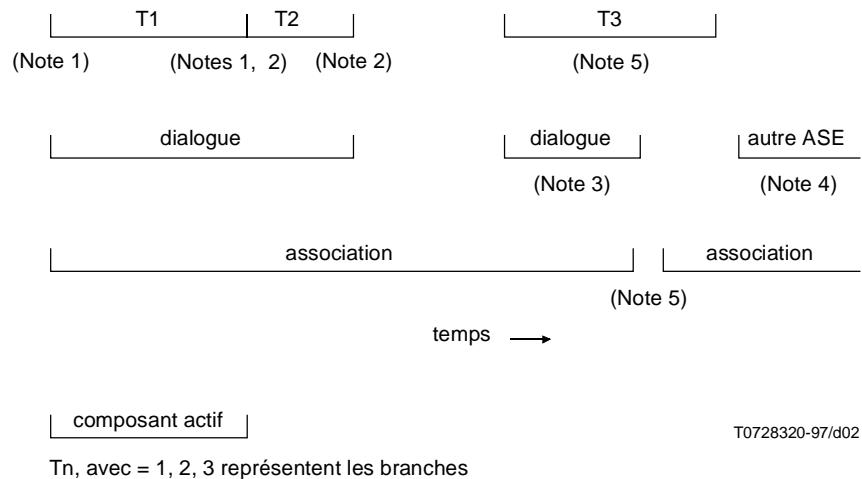
- a) successions chaînées: il s'agit de successions ininterrompues de branches de transaction d'un même dialogue qui s'exécutent avec le même niveau de coordination ("engagement" ou "engagement en une phase"). Chaque branche de transaction est lancée directement par la machine TPPM supérieure; et
- b) successions non chaînées: il s'agit de successions de branches de transaction d'un même dialogue pour lesquelles le niveau de coordination du dialogue repasse par la valeur "néant" entre chaque branche de transaction. Le niveau de coordination peut prendre l'une des valeurs "néant", "engagement" ou "engagement en une phase" au moment de l'établissement du dialogue, ainsi qu'au moment de sa terminaison. Chaque branche de transaction est lancée par l'invocation TPSUI supérieure.

S'il n'y a pas de transaction en cours dans une partie d'un arbre de dialogue (c'est-à-dire que les dialogues ont un niveau de coordination "néant"), une invocation TPSUI située au sein de cette partie peut alors lancer une nouvelle transaction. Cela peut conduire à l'existence simultanée de zéro, d'un ou de plusieurs arbres de transaction pour le même arbre de dialogue.

Les arbres de transaction sont disjoints à tout instant. La séparation entre deux arbres de transaction sera garantie par l'existence d'au moins un dialogue avec un niveau de coordination "néant".

Une fois le dialogue terminé entre un nœud intermédiaire et son supérieur, le nœud intermédiaire devient alors la racine d'un nouvel arbre de dialogue et peut éventuellement devenir la racine d'un arbre de transaction.

La Figure 2 indique les correspondances dans le temps entre les branches de transaction, les dialogues et les associations. Cet ensemble de correspondances est représenté pour deux systèmes ouverts adjacents.



NOTE 1 – Le début d'une branche de transaction se présente soit au début d'un dialogue, soit pendant la durée d'un dialogue.

NOTE 2 – La fin d'un dialogue implique la fin de la branche de transaction en cours. La fin d'une branche de transaction se présente soit au cours d'un dialogue, soit à la fin d'un dialogue.

NOTE 3 – Un dialogue peut succéder à un autre dialogue dans les limites de la même association d'application.

NOTE 4 – Un nouvel élément ASE peut utiliser l'association d'application lorsque le dialogue se termine.

NOTE 5 – Le dialogue se termine immédiatement en cas d'échec de l'association d'application. Toutefois, si l'application se trouve dans l'état PRÊT ou DÉCIDE (engagé), le rétablissement de la transaction s'effectuera alors sur une autre association et la transaction continuera à exister jusqu'à ce que le rétablissement se soit terminé.

**Figure 2/X.860 – Branches de transaction, dialogues et associations d'application**

## 8.4 Dénomination

Le traitement OSI TP nécessite, en plus des fonctionnalités de dénomination déjà définies par la Rec. UIT-T X.650 | ISO/CEI 7498-3 pour l'interconnexion des systèmes ouverts, l'emploi de titres d'utilisateur TPSU ainsi que d'identificateurs de transaction et de branches de transaction. Les définitions de ces titres et identificateurs sont données au paragraphe 3.

### 8.4.1 Titre d'utilisateur TPSU

Le titre d'utilisateur TPSU est utilisé pendant l'établissement du dialogue pour sélectionner, au sein d'un processus d'application donné, un utilisateur TPSU avec lequel le dialogue doit être établi. Le dialogue est alors établi entre l'invocation TPSUI appelante et une invocation TPSUI de l'utilisateur TPSU spécifié par le titre TPSU. Le dialogue est établi sur une association d'application (déjà existante ou nouvellement établie) entre deux invocations d'entité d'application prenant en charge les invocations TPSUI respectives.

Le titre qui désigne l'utilisateur TPSU cible pour l'établissement du dialogue indique également les capacités de traitement d'utilisateur TPSU.

Si le dialogue est établi sur une association d'application déjà existante, le titre TPSU peut alors être utilisé pour déduire l'information nécessaire à la machine TPPM appelante afin de choisir une association d'application convenable parmi les associations d'application disponibles. Un exemple d'une telle information est le contexte d'application.

Le titre TPSU peut être utilisé pour fournir à la machine TPPM appelante l'information nécessaire à l'établissement de l'association requise dans le cas où une association déjà existante n'est pas disponible.

Le titre TPSU doit être non ambigu dans le domaine d'un processus d'application.

#### **8.4.2 Identificateur de transaction**

Un identificateur de transaction désigne sans ambiguïté une transaction donnée dans l'environnement OSIE. L'identificateur de transaction se compose:

- a) du titre de l'entité d'application qui prend en charge le nœud racine de la transaction; et
- b) du suffixe de transaction dont la valeur est non ambiguë dans le domaine de validité de l'entité d'application qui prend en charge le nœud racine de la transaction. Le suffixe de transaction peut être, par exemple, un nombre entier qui est incrémenté d'une unité au moment de l'établissement de toute nouvelle transaction instanciée.

NOTE – L'identificateur de transaction doit également être globalement non ambigu dans le temps, compte tenu des besoins de rétablissement et d'audit commercial.

#### **8.4.3 Identificateur de branche de transaction**

Un identificateur de branche de transaction désigne sans ambiguïté une branche de transaction dans le domaine de validité d'une transaction donnée. L'identificateur de branche de transaction se compose:

- a) du titre de l'entité d'application prenant en charge le nœud supérieur de la branche de transaction; et
- b) du suffixe de branche de transaction dont la valeur est non ambiguë dans le domaine de validité de l'entité d'application prenant en charge le nœud supérieur de la branche de transaction.

### **8.5 Transfert de données**

#### **8.5.1 Besoins et objectifs**

Le service OSI TP peut transférer des données en vue des satisfaire aux besoins d'échange de données entre les invocations TPSUI qui interviennent dans une transaction répartie, avec les objectifs suivants:

- a) le service OSI TP permet à l'invocation TPSUI d'acheminer les données en utilisant ses propres règles sémantiques;
- b) le transfert de données concerne toujours un dialogue unique;
- c) l'invocation TPSUI est libre d'organiser le style de son échange d'informations sémantiques en utilisant à cette fin un ou plusieurs éléments spécifiques ASE utilisateur. Ses échanges sémantiques peuvent notamment se baser sur plusieurs types de règles; et
- d) la définition des éléments ASE utilisateur peut être la même lorsque ces éléments sont utilisés avec ou sans traitement OSI TP.

#### **8.5.2 Coordination du transfert de données**

Les éléments ASE utilisateur génèrent des unités APDU de transfert de données qui sont mappées avec les services sous-jacents sous la coordination de la fonction SACF.

La machine TPPM traite le protocole de gestion de dialogue, mais ne génère pas directement les unités APDU de transfert de données.

Elle détermine toutefois l'ordre de succession dans le temps de l'utilisation de l'association d'application sous-jacente pour le traitement transactionnel.

Ainsi, le transfert de données du traitement OSI TP:

- a) ne peut se faire que dans les limites d'un dialogue;
- b) est soumis aux modes de commande. En particulier, dans le mode de commande polarisé, les données ne peuvent être émises que si l'invocation TPSUI possède la commande du dialogue. Le choix du mode de commande dépend des besoins spécifiques formulés par les éléments ASE utilisateur; et
- c) dépend des états de la machine TPPM.

La machine TPPM garantit la coordination entre le transfert de données et les phases de l'engagement pendant la terminaison de la transaction.

## **8.6 Coordination des ressources**

### **8.6.1 Engagement en deux phases**

La phase de terminaison d'une transaction répartie est lancée à la suite d'une demande émise par une ou plusieurs invocations TPSUI appartenant à l'arbre de transaction. Le fournisseur TPSP coordonne la phase d'engagement des invocations TPSUI au sein de cet arbre, afin de garantir la libération des données liées de la transaction dans un état cohérent. La coordination de la phase de terminaison s'effectue en deux étapes:

- a) phase 1 de l'engagement; et
- b) phase 2 de l'engagement.

Dans la phase 1 de l'engagement, les nœuds tentent de faire passer toutes les données de l'arbre de transaction dans l'état "prêt pour l'engagement". Les données se trouvent dans l'état "prêt pour l'engagement" si elles peuvent être libérées dans leur état initial ou final tant que la transaction ne s'est pas terminée par un engagement ou par un retour-arrière. La phase 2 débute une fois que toutes les données de l'arbre de transaction sont dans l'état "prêt pour l'engagement"; la transaction fait l'objet d'un retour-arrière si cet état ne peut pas être atteint.

Les transactions peuvent être engagées dans la phase 2 de l'engagement pour l'arbre de transaction si:

- a) toutes les données liées se trouvent dans l'état "prêt pour l'engagement"; et
- b) il n'existe d'opération en cours qui modifie l'arbre de transaction, par exemple en établissant de nouvelles branches de transaction ou en modifiant la communication entre les nœuds.

Si la transaction peut être engagée, le fournisseur TPSP propage alors la décision d'engagement dans l'arbre de transaction et coordonne l'achèvement de la transaction.

L'engagement est terminé une fois que ces deux étapes se sont achevées. Une nouvelle transaction peut alors éventuellement commencer.

NOTE – Ces procédures peuvent être modifiées lorsque des procédures "lecture seule", "sortie prématurée" ou "engagement en une phase" sont utilisées; prière de se référer aux 8.6.2, 8.6.3 et 8.6.4.

#### **8.6.1.1 Procédures d'engagement statique**

Dans la phase 1, chacun des nœuds reçoit de son supérieur l'information de démarrage de la phase de terminaison, indiquant en particulier qu'il ne recevra plus de données de son supérieur et qu'il doit faire passer ses données liées dans l'état "prêt pour l'engagement".

Si le nœud accepte de donner suite, il tente alors de faire passer les données liées de son sous-arbre dans l'état "prêt pour l'engagement". Il tente de faire passer ses propres données locales liées dans cet état et informe d'une manière récursive ses subordonnés pour les ressources distantes.

Une fois que les données liées de son sous-arbre se trouvent dans l'état "prêt pour l'engagement", il informera alors son supérieur éventuel en émettant un signal "prêt" et attendra le résultat final de la transaction; ce processus se répète vers le haut d'une manière récursive; la phase 2 démarre lorsque le nœud concerné n'a pas de supérieur dans l'arbre de transaction. Un nœud qui émet un signal "prêt" à destination de son supérieur devient un esclave dans le processus d'engagement et le supérieur devient son maître d'engagement. Un nœud qui reçoit des messages "prêt" d'un ensemble de ses subordonnés (y compris leurs sous-arbres) se voit attribuer cet ensemble comme arrière-plan d'engagement.

Le mode "prêt" d'un nœud peut être invalidé si des données liées sont touchées par la réception de données d'application, après que l'un des signaux "prêt", "lecture seule" ou "une phase" a été émis. Il est nécessaire d'éviter de telles collisions de données prêtes, qui ne sont possible que pour certaines combinaisons d'unités fonctionnelles; les applications utilisant ces unités fonctionnelles doivent s'assurer que de telles collisions ne se produisent pas. Ce type de collision n'est possible pour une branche que si le mode de commande polarisé n'est pas sélectionné.

NOTE – La Rec. UIT-T X.862 | ISO/CEI 10026-3 contient des scénarios qui illustrent de telles collisions.

Un nœud démarre le retour-arrière de la transaction s'il n'est pas en mesure de faire passer les données dans l'état "prêt pour l'engagement".

Une fois que la décision d'engagement est prise pour la phase 2, tout nœud reçoit de son supérieur l'instruction de libérer dans l'état final les données liées de son sous-arbre. Le nœud engage ses données locales et donne, pour les ressources distantes, des instructions d'engagement à ses subordonnés; le processus se poursuit d'une manière récursive.

Le nœud informe son supérieur lorsque toutes les données liées de son sous-arbre ont été libérées dans l'état final; le processus se poursuit d'une manière récursive, après quoi la transaction se termine.

### 8.6.1.2 Préparation implicite

Il est possible de permettre uniquement au nœud racine de démarrer la phase de terminaison, auquel cas le signal de début de la phase de terminaison (signal de préparation) est véhiculé par le fournisseur TPSP du nœud supérieur vers le nœud subordonnée.

Une variante consiste à faire véhiculer ce signal d'une manière implicite par la sémantique de l'application, auquel cas on parle de préparation implicite. Le signal peut être véhiculé dans un message d'application donné, il peut être déduit à partir d'une succession de messages ayant fait l'objet d'un accord, ou il peut être indiqué d'une manière implicite par le début de la transaction.

NOTE – L'utilisation de la préparation implicite peut permettre d'atteindre plus rapidement l'état "prêt" dans un sous-arbre de transaction et de réduire ainsi le temps total d'exécution de la transaction.

Le passage du signal de préparation implicite n'est pas visible pour le fournisseur TPSP. Il en résulte que, du point de vue de ce fournisseur, il n'est pas possible de faire la distinction entre la signalisation implicite et le fait d'accorder à tout nœud le droit d'initialiser la phase de terminaison. Si le mécanisme de préparation implicite est utilisé entre un supérieur et un subordonné, ce dernier peut alors à tout instant entrer dans la phase 1 de l'engagement et tenter de faire passer son sous-arbre dans l'état "prêt".

Un nœud intermédiaire qui n'utilise pas le mécanisme de préparation implicite avec son supérieur, mais qui l'utilise avec un ou plusieurs subordonnés ne peut pas entrer dans la phase 1 de l'engagement tant qu'il n'en a pas reçu la permission dans un signal issu du supérieur. Le fournisseur TPSP ne dispose toutefois d'aucune mesure lui permettant d'intervenir dans le démarrage de l'engagement par les subordonnés.

NOTE – Une telle intervention pour l'ensemble d'un sous-arbre conduirait à des échanges de protocole extrêmement complexes.

### 8.6.1.3 Procédures d'engagement dynamique

Dans la phase 1, tout nœud peut appliquer localement la procédure de terminaison s'il dispose de la permission d'initialiser la terminaison (parce qu'il est racine, ou parce que la préparation implicite est utilisée), dans la mesure où il peut s'assurer qu'aucune donnée ne sera plus reçue d'un voisin quelconque.

Si le nœud accepte de traiter l'engagement, il tente alors de faire passer ses données liées locales dans l'état "prêt pour l'engagement". Le nœud lance le retour-arrière de la transaction si cette tentative échoue.

Une fois que les données liées se trouvent dans l'état "prêt pour l'engagement" et que des signaux "prêt", "lecture seule", "sortie prématurée" ou "une phase" ont été reçus de tous les voisins sauf un, le nœud émet alors un signal "prêt" à destination du seul voisin qui ne lui n'a pas envoyé de signal "prêt" et attend le résultat final de la transaction; le processus se poursuit d'une manière récursive. Les nœuds qui lui envoient les signaux "prêt" sont les esclaves de l'engagement; le nœud auquel il envoie éventuellement un signal "prêt" est le maître de l'engagement.

Si un nœud utilisant les procédures d'engagement dynamique émet un signal "prêt" à destination d'un subordonné sans que le signal de préparation ait été émis au préalable, le signal "prêt" indique alors au subordonné le démarrage de la phase de terminaison.

NOTE 1 – Il se peut que le subordonné soit déjà au courant si le mécanisme de préparation implicite est utilisé.

L'arrière-plan d'engagement d'un nœud à un instant donné se constitue de l'ensemble de nœuds de l'arbre de transaction défini comme suit:

- a) nœuds voisins en provenance desquels des signaux "prêt" ont été reçus; et
- b) arrière-plans de ces nœuds et ainsi de suite d'une manière récursive.

Le mode "prêt" d'un nœud peut être invalidé si des données liées sont touchées par la réception de données d'application après qu'un signal "prêt" a été émis. Il est nécessaire d'éviter de telles collisions de "données prêtes", qui ne sont possible que pour certaines combinaisons d'unités fonctionnelles; les applications utilisant ces unités fonctionnelles doivent s'assurer que de telles collisions ne se produisent pas. Ce type de collision n'est possible sur une branche que si le mode de commande polarisé n'est pas sélectionné.

NOTE 2 – La Norme ISO/CEI 10026-3 contient des scénarios qui illustrent ces collisions.

Une fois que toutes les données liées d'un nœud se trouvent dans l'état "prêt pour l'engagement" et que des signaux "prêt", "lecture seule", "sortie prématurée" ou "une phase" ont été reçus de tous les voisins, le nœud entrera alors dans la phase 2 de l'engagement et la décision d'engagement peut alors être prise. Un nœud peut recevoir un signal "prêt" d'un nœud auquel il a envoyé un tel signal (collision de signaux "prêt"). Le protocole de traitement transactionnel utilise dans ce cas un mécanisme de prise de décision pour déterminer quel est le nœud qui prendra la décision d'engagement et deviendra ainsi le coordinateur de l'engagement.

NOTE 3 – Une variante consisterait à permettre aux nœuds qui ont reçu et émis des signaux "prêt" de devenir tous deux des coordinateurs de l'engagement et de propager l'engagement. Ceci introduit toutefois une complexité supplémentaire dans les machines de protocole qui doivent traiter le cas d'échec de l'écriture de l'enregistrement de journal d'engagement dans l'un des nœuds ou dans les deux; un mécanisme de prise de décision a été adopté dans un but de simplification.

Une fois que la décision d'engagement a été prise, dans la phase 2, chaque nœud reçoit de son maître d'engagement l'instruction de libérer dans l'état final les données de son arrière-plan d'engagement. Le nœud procède alors à l'engagement de ces données locales. Il ordonne à ses esclaves d'engagement d'en faire de même et le processus se poursuit d'une manière récursive.

Le nœud informe son maître d'engagement une fois que toutes les données liées de son arrière-plan d'engagement ont été libérées dans leur état final. Le processus se poursuit d'une manière récursive, après quoi la transaction se termine.

Il existe des mécanismes qui permettent de déterminer la direction de propagation des signaux "prêt". Il est possible de créer des arbres de transaction pour lesquels il est impossible de procéder à un engagement. De tels arbres se trouvent dans une situation de blocage mutuel. Des contrôles sont effectués afin de garantir que de tels arbres ne peuvent pas être créés, mais il existe un risque que des précautions excessives soient prises et interdisent certaines structures d'arbre alors que celles-ci sont parfaitement viables. Le traitement transactionnel fournit une fonctionnalité optionnelle permettant de mettre hors service certains de ces contrôles, auquel cas l'application est responsable de garantir qu'elle ne construit pas d'arbres conduisant à un blocage mutuel.

### **8.6.2 Lecture seule**

Un nœud qui a traité toutes les demandes de son supérieur et qui n'a procédé à aucune modification de données de transaction par rapport à leur état initial (données liées ou données à longue durée de vie) peut tenter de se retirer des procédures d'engagement en deux phases, compte tenu du fait que ses données de transaction ne seront affectées ni par un engagement, ni par un rétablissement de la transaction. Si un tel nœud reçoit des signaux "lecture seule" ou "sortie prématurée" de tous ses subordonnés, il peut alors à son tour émettre un signal "lecture seule" vers son supérieur, après quoi il ne participe plus aux procédures d'engagement. Un tel nœud n'aura pas connaissance du résultat de la transaction (engagement ou rétablissement), excepté dans le cas de rétablissement si l'unité fonctionnelle de transactions chaînées est sélectionnée pour le dialogue ou si une action différée le concernant est en attente. Un nœud qui émet un signal "lecture seule" n'a aucune obligation d'écriture d'enregistrement de journalisation dans une mémoire sûre.

Dans le cas de transactions non chaînées, un nœud qui émet un signal "lecture seule" cesse d'appartenir à l'arbre de transaction à partir du moment où il a reçu une indication d'achèvement. Il est libre de lancer d'autres actions, même si la transaction à laquelle il participait est toujours en cours. Le supérieur d'un tel nœud peut initialiser pour ce nœud une nouvelle branche de transaction concernant la même transaction.

NOTE – Un nœud qui émet un signal "lecture seule" n'est pas un esclave d'engagement et ne fait pas partie de l'arrière-plan d'engagement.

### **8.6.3 Sortie prématurée**

Un nœud qui ne peut pas apporter de contribution au traitement d'une transaction et qui a reçu des signaux "lecture seule" ou "sortie prématurée" de tous ses subordonnés éventuels de l'arbre de transaction peut se retirer de la transaction en émettant un signal "sortie prématurée" à destination de son supérieur. Les autres conditions d'émission du signal de sortie prématurée sont que les données de transaction de ce nœud n'ont pas été modifiées par la transaction et qu'un compte rendu de résultat de la transaction n'est pas nécessaire. Le fournisseur TPSP rejettera toute demande ultérieure du supérieur concernant la branche de transaction concernée.

NOTE 1 – Un nœud peut, par exemple, déterminer à partir d'une demande reçue de son supérieur qu'il n'a pas accès aux données nécessaires pour satisfaire la demande.

Dans le cas de transactions non chaînées, un nœud qui émet un signal "sortie prématurée" cesse d'appartenir à l'arbre de transaction à partir du moment où il a reçu une indication d'achèvement. Il est libre de lancer d'autres actions, même si la transaction à laquelle il participait est toujours en cours. Le supérieur d'un tel nœud peut initialiser pour ce nœud une nouvelle branche de transaction concernant la même transaction.

NOTE 2 – Un nœud qui émet un signal "sortie prématurée" n'est pas un esclave d'engagement et ne fait pas partie de l'arrière-plan d'engagement.

### **8.6.4 Engagement en une phase**

Un nœud qui n'a pas de données liées peut utiliser les procédures d'engagement en une phase s'il désire être informé du résultat de la transaction. Un tel nœud peut émettre un signal "une phase" vers son dernier voisin restant s'il a reçu des signaux "une phase", "lecture seule" ou "sortie prématurée" de tous ses voisins sauf un. Le nœud recevra une notification du résultat de la transaction s'il a émis un signal "une phase", sauf si une défaillance de dialogue ou de nœud entre le coordinateur de l'engagement et lui-même rendent impossible la transmission de la décision. Un nœud qui émet un signal "une phase" n'a aucune obligation d'écriture d'enregistrement de journalisation dans une mémoire sûre.

NOTE 1 – Un nœud utilisant un engagement en une phase peut posséder des données de transaction (c'est-à-dire des données qui sont manipulées au cours de la transaction) qui ne sont pas des données liées au sens strict de la définition des procédures CCR; ces données à longue durée de vie peuvent être libérées dans un état mis à jour à la fin de la transaction, mais cette mise à jour ne sera pas coordonnée avec le résultat de la transaction dans le cas de certaines défaillances (les données peuvent, par exemple, être libérées dans un état mis à jour alors qu'un retour-arrière est effectué pour la transaction). Il est possible dans ce cas de tenter une coordination par de moyens autres que le protocole de traitement transactionnel.

NOTE 2 – Un nœud qui émet un signal "une phase" n'est pas un esclave d'engagement et ne fait pas partie de l'arrière-plan d'engagement.

### **8.6.5 Retour-arrière**

Le retour-arrière d'une transaction peut être lancé par n'importe quel nœud de l'arbre de transactions qui n'a pas antérieurement émis un signal "prêt", un signal "une phase", un signal "lecture seule" ou un signal "sortie prématurée". Le retour-arrière replace les données liées de la transaction dans leur état initial.

Le fait de lancer une procédure de retour-arrière ne conduit pas en soi à la fin au dialogue sous-jacent. Une invocation TPSUI peut interrompre le dialogue si elle désire y mettre fin. Un retour-arrière est effectué pour la transaction si cette dernière est interrompue avant le début de ses procédures de terminaison.

Il est possible (mais non nécessaire) de lancer une nouvelle transaction après l'achèvement d'une procédure de retour-arrière.

### **8.6.6 Décisions heuristiques**

Lorsqu'un nœud subordonné est entré dans la phase 1 de l'engagement, il peut décider de libérer dans leur état final ou dans leur état initial tout ou partie des données liées qui se trouvent dans l'état "prêt", même s'il n'a pas été informé par son supérieur du résultat final de la transaction. Une telle décision est appelée décision heuristique.

Les décisions heuristiques peuvent être prises par des nœuds individuels après une défaillance de la communication, ou dans certaines situations locales propres au système. Le choix de la prise d'une ou de plusieurs décisions heuristiques et du contenu des décisions à prendre est un problème local. Une décision heuristique prise par un nœud n'est pas propagée vers les autres nœuds du domaine de validité du traitement OSI TP.

Un nœud qui a pris une décision heuristique a l'obligation d'enregistrer cette décision dans une mémoire sûre dans un enregistrement de journalisation heuristique. Cet enregistrement est effacé et la terminaison normale de la transaction se poursuit s'il est établi que l'état des données liées du nœud et le résultat de la transaction sont cohérents.

### **8.6.7 Détection d'une incohérence heuristique**

Un nœud qui a pris une décision heuristique détermine l'existence d'une incohérence heuristique si l'état de ses données liées n'est pas cohérent avec le résultat de la transaction. Le nœud peut procéder à cette détermination dès qu'il a été informé par son supérieur direct du résultat de la transaction. Il y a violation des propriétés ACID si l'état des données liées du nœud n'est pas cohérent avec le résultat de la transaction. Cette situation constitue un mélange heuristique.

Une situation de risque heuristique se produit lorsqu'un nœud n'est pas en mesure de déterminer l'état exact des données liées pour ses nœuds subordonnés dans leurs sous-arbres. Ceci peut se produire après une perte de communication avec un ou plusieurs subordonnés. Le compte rendu de l'état des données liées du sous-arbre ne peut pas être fait au supérieur direct si le résultat final de la transaction est un retour-arrière, en raison de la procédure de retour-arrière présumé (se référer au 8.7.2 et à l'Annexe C). Cette situation constitue un risque heuristique, car l'état des données liées à l'intérieur du sous-arbre peut être un mélange heuristique.

Une situation de risque heuristique se produit également lorsqu'une invocation TPSUI n'est pas en mesure de déterminer si l'état des données liées locales est cohérent avec le résultat de la transaction. Ceci peut se produire après une perte locale de communication.

### **8.6.8 Comptes rendus**

Un nœud peut appliquer une stratégie d'endiguement heuristique de manière à confiner et corriger une incohérence heuristique au sein du sous-arbre local où elle se manifeste. Un tel nœud ne rendra pas compte à son supérieur d'un dommage heuristique, mais se comportera par ailleurs comme décrit.

Tout nœud prend connaissance des informations concernant l'état des données liées à l'intérieur de son sous-arbre, à moins qu'un ou plusieurs nœuds ne disposent déjà de cette information; il en résulte que le nœud racine prendra connaissance des informations relatives à l'état des données liées dans la totalité de l'arbre de transaction.

Toute machine TPPM du fournisseur TPSP collecte les comptes rendus d'état des données liées de son sous-arbre. Ces comptes rendus proviennent des sources suivantes:

- a) comparaison de l'état des données liées du nœud avec le résultat final de la transaction; et
- b) compte rendu fourni par tout subordonné au sujet de l'état des données liées dans son sous-arbre.

Si le nœud détermine que l'état des données liées dans son sous-arbre est cohérent avec le résultat final de la transaction, la machine TPPM informe alors son supérieur que toutes les données liées dans son sous-arbre se trouvent dans un état cohérent.

Si le nœud détermine que l'état des données liées dans son sous-arbre n'est pas cohérent avec le résultat final de la transaction, sans qu'il soit capable de remédier à cette incohérence, alors la machine TPPM:

- a) mémorise dans un enregistrement de journalisation de dommage, la connaissance de l'incohérence des données liées dans le sous-arbre du nœud, au fur et à mesure de la réception des comptes rendus. Prière de se référer au tableau 2 pour ce qui est des valeurs résultantes de l'enregistrement de journalisation de dommage selon l'incohérence signalée;
- b) rend compte de l'incohérence à son invocation TPSUI;
- c) rend compte au nœud supérieur éventuel lorsqu'un compte rendu d'état complet des données liées dans son sous-arbre est disponible; et
- d) rend compte de l'incohérence à une entité locale, par exemple à un opérateur du système.

**Tableau 2/X.860 – Mise à jour de l'enregistrement de journalisation de dommage**

| Etat précédent de l'enregistrement de journalisation de dommage | Incohérence faisant l'objet d'un compte rendu |                     |                     |
|---|---|---------------------|---------------------|
|   | Aucune incohérence                            | Risque heuristique  | Mélange heuristique |
| aucun enregistrement de journalisation de dommage               | aucun compte rendu                            | risque heuristique  | mélange heuristique |
| risque heuristique  | risque heuristique                            | risque heuristique  | mélange heuristique |
| mélange heuristique   | mélange heuristique                           | mélange heuristique | mélange heuristique |

L'enregistrement de journalisation de dommage est conservé après la propagation du résultat final de la transaction, tant que la confirmation assurant que le supérieur a reçu le rapport approprié n'a pas été obtenue.

NOTE 1 – Cela n'implique pas que l'information concernant l'incohérence ne puisse pas être conservée jusqu'à la réparation du dommage.

NOTE 2 – Le mécanisme par lequel le nœud subordonné s'assure que le supérieur a connaissance du dommage heuristique est en-dehors du domaine d'application du traitement OSI TP.

NOTE 3 – Les comptes rendus heuristiques peuvent être supprimés d'une manière optionnelle, la description précédente traite le cas où ils ne sont pas supprimés.

NOTE 4 – Les comptes rendus heuristiques ne sont pas transmis de manière fiable à des nœuds utilisant les procédures d'engagement en une phase.

La collecte d'informations concernant l'état de données liées et la transmission vers le supérieur peuvent introduire des retards dans l'achèvement de la transaction. Ces retards n'existent pas si le dialogue utilise les fonctionnalités "endiguement heuristique exigé". Un nœud peut toutefois appliquer un endiguement heuristique même si les fonctionnalités précédentes ne sont pas utilisées; il émettra alors ses messages de traitement transactionnel comme s'il effectuait des comptes rendus, mais ces comptes rendus seront toujours vides ou absents.

Lorsqu'un nœud utilise les fonctionnalités de préparation implicite pour le dialogue avec son supérieur, mais que ce dernier n'utilise pas la préparation implicite ou l'engagement dynamique avec son propre supérieur, le nœud du niveau le plus bas a la possibilité de prendre une décision heuristique avant que le nœud du niveau le plus élevé ait donné au nœud intermédiaire la permission d'entrer dans la phase 1 de l'engagement. Le nœud intermédiaire doit appliquer dans ce cas l'endiguement heuristique afin d'éviter l'envoi d'un compte rendu heuristique au nœud du niveau le plus élevé.



## 8.7 Rétablissement

### 8.7.1 Types de défaillances

#### 8.7.1.1 Introduction

Le Tableau 3 identifie les causes potentielles de défaillances, les types de défaillance qui peuvent se produire pendant une transaction et les actions devant être prises pour remettre la transaction dans un état pouvant être géré.

**Tableau 3/X.860 – Types de défaillances**

| Raisons possibles   | Type de défaillance                                       | Action prise par la machine TPPM  |
|---|---|---|
| erreur d'application  | récupérable localement                                    | aucune  |
| interruption de la transaction ou du dialogue   | récupérable avant que le nœud ne soit prêt à l'engagement | retour-arrière  |
| interruption du dialogue  | récupérable après que le nœud est prêt à l'engagement     | procédures de rétablissement  |
| écroulement du nœud<br>défaillance de la machine TPPM ou de l'invocation d'entité d'application | données d'action atomique indisponibles                   | récupération de données d'action atomique, interruption du dialogue, éventuellement interruption de l'association et procédures de rétablissement |
| défaillance du moyen de stockage  | destruction de données d'action atomique                  | en dehors du domaine d'application de la présente Recommandation, de X.861 et X.862   |

#### 8.7.1.2 Défaillances récupérables localement

Lorsqu'une défaillance se produit, il se peut que l'invocation TPSUI soit en mesure de procéder à une récupération par ses propres moyens, de sorte que la transaction peut poursuivre son engagement. L'incident n'a pas de répercussions extérieures si l'invocation TPSUI ou la machine TPPM effectue le rétablissement (exception faite de quelques retards possibles). Ce cas est un problème local.

#### 8.7.1.3 Défaillances récupérables avant que le nœud soit prêt à l'engagement

Toute défaillance se produisant avant que le nœud soit prêt à l'engagement conduit à une procédure de retour-arrière. Ce type de défaillance peut avoir l'une des causes suivantes:

- a) interruption de la transaction pour l'une des raisons suivantes:
  - 1) incapacité pour l'invocation TPSUI d'agir sur la transaction en cours entraînant une demande explicite de retour-arrière;
  - 2) situation de blocage mutuel réparti, dans laquelle une transaction fait partie d'un cycle d'attente avec d'autres transactions;
  - 3) défaillance du moyen de stockage rendant impossible l'accès à la valeur courante des données liées, alors que ces données sont disponibles dans leur état initial; ou
  - 4) défaillance du moyen de stockage conduisant à la destruction des données liées, mais avec un état de la transaction connu et nécessitant une intervention locale pour reconstruire les données liées; ou
- b) interruption du dialogue pour l'une des raisons suivantes:
  - 1) défaillance de l'association d'application prenant en charge le dialogue, qui peut résulter d'une défaillance de l'élément ACSE, du service de présentation ou d'un autre service de prise en charge (par exemple le service de session);
  - 2) erreur de protocole d'application de l'un des protocoles suivants utilisés pour le dialogue:
    - i) protocole d'élément ASE utilisateur;
    - ii) protocole de traitement OSI TP; ou
    - iii) protocole CCR;

- 3) défaillance d'un élément ASE utilisateur;
- 4) défaillance de l'invocation TPSUI et/ou de la machine TPPM les rendant incapables de poursuivre la communication sur le dialogue (écroulement de nœud).

#### **8.7.1.4 Défaillances récupérables après que le nœud est prêt à l'engagement**

Les défaillances récupérables qui se produisent après que le nœud est prêt à l'engagement sont celles qui causent des interruptions du dialogue. Une procédure de rétablissement est lancée après une défaillance pour mettre fin à la transaction. Prière de se référer au 8.7.1.3 b) pour les raisons possibles de l'interruption du dialogue.

#### **8.7.1.5 Données d'action atomique indisponibles**

Ce type de défaillance se produit lorsque la copie de travail des données d'action atomique pour la transaction courante devient indisponible après la défaillance d'un système ouvert. La copie de travail des données d'action atomique peut devenir indisponible (c'est-à-dire être perdue), par exemple, après une défaillance de machine TPPM, une défaillance d'invocation d'entité d'application ou un écroulement de nœud.

Le journal de rétablissement doit être lu pour rétablir la copie de travail des données d'action atomique pour la transaction. Tous les dialogues et/ou les associations sous-jacentes qui appartiennent à la transaction en cours sont interrompus.

#### **8.7.1.6 Destruction de données d'action atomique**

Les données d'action atomique ont été perdues en raison d'une défaillance éventuelle du moyen de stockage. La procédure de rétablissement est en dehors du domaine d'application de la présente Recommandation.

### **8.7.2 Prise en charge du rétablissement de transactions**

Le fournisseur TPSP offre les fonctionnalités nécessaires au rétablissement après une défaillance de la communication ou d'un écroulement de nœud. Le rétablissement d'une transaction signifie que toutes les données liées qui ont été impliquées dans la transaction seront restaurées dans leur état final ou dans leur état initial après une défaillance. Le fournisseur TPSP est responsable de la réinstallation de toutes les ressources dans un état cohérent, c'est-à-dire l'état final ou l'état initial.

Le fournisseur TPSP effectue le rétablissement de la transaction dans les limites de l'arbre de transaction. Les invocations TPSUI sont responsables du rétablissement à l'extérieur de cet arbre.

La prise en compte du rétablissement des transactions nécessite que les étapes clés du déroulement des branches (données d'action atomique) aient été convenablement journalisées dans tout système ouvert impliqué dans l'arbre de transaction. Le protocole d'engagement en deux phases pour le retour-arrière présumé est utilisé à cette fin.

Les informations suivantes ne doivent pas être perdues et doivent donc être sauvegardées dans le journal de rétablissement:

- 1) données d'action atomique: ces données ne sont pas concernées par les procédures d'engagement et de retour-arrière, mais elles sont utilisées pendant le processus de rétablissement; et
- 2) données liées: il s'agit des données sur lesquelles agit la transaction. Ces données sont soumises aux procédures d'engagement et de retour-arrière; elles sont invisibles de l'extérieur de la transaction pendant son exécution et ne sont disponibles pour toute transaction qu'après la fin de la transaction courante;

### **8.7.3 Etats des nœuds**

La transaction peut se trouver dans un état actif ou dans un processus de terminaison lorsqu'une défaillance se produit. Dans ce dernier cas, le déroulement temporel est un facteur important du mécanisme de rétablissement. La terminaison d'une transaction n'est pas une action instantanée, étant donné que plusieurs étapes et échanges sont nécessaires dans l'ensemble de l'arbre de transaction entre le moment où une invocation TPSUI demande la terminaison et le moment où cette invocation est informée de l'exécution de sa demande.

Les branches d'une transaction peuvent se trouver dans différents états lorsqu'une défaillance se produit pendant la terminaison d'une transaction. Le rétablissement peut par conséquent nécessiter différents types d'action, selon les états des nœuds de l'arbre de transaction.

Le nœud peut se trouver dans l'un des états suivants:

- a) **ACTIF**: le traitement de la transaction est en cours. Le nœud peut choisir d'ordonner le retour-arrière de la transaction et de libérer les données liées dans leur état initial sous sa responsabilité et sans mettre leur cohérence en danger.

Selon l'approche présumée de retour-arrière, le nœud n'est pas obligé de consigner dans le journal de rétablissement la création d'une branche transactionnelle quelconque.

- b) **PRÊT**: deux cas sont possibles:

- 1) le nœud est en mesure de placer les données dans leur état final (engagement) ou initial (retour-arrière) sous sa propre responsabilité. Le nœud a reçu des signaux "prêt", "lecture seule", "sortie prématurée" ou "une phase" de tous ses voisins sauf un, c'est-à-dire qu'il existe un et un seul nœud adjacent dont il n'a pas reçu de tel signal. Il en résulte que toutes les données liées de ce nœud et de tous les nœuds de son arrière-plan d'engagement se trouvent dans l'état "prêt".

Le nœud stockera un enregistrement de journalisation de l'état "prêt" dans le journal de rétablissement avant d'émettre un signal "prêt" pour la totalité de l'arrière-plan d'engagement (indiquant que l'engagement peut être effectué) à destination du nœud duquel il n'a pas reçu de signaux "prêt", "lecture seule", "sortie prématurée" ou "une phase". L'enregistrement de journalisation contient l'identificateur de transaction, le vote de l'état prêt, la liste des nœuds adjacents ayant émis le signal "prêt" et l'identification du nœud auquel le signal "prêt" a été envoyé; ce dernier sera le maître de l'engagement. Le nœud passe dans l'état PRÊT à partir du moment où l'enregistrement de journalisation "prêt" a été stocké; ou

- 2) le nœud est passé précédemment dans l'état "prêt", il reçoit un signal "prêt" du nœud à destination duquel il avait émis un signal "prêt" et détermine qu'il sera l'esclave d'engagement de ce nœud.

- c) **LECTURE-SEULE**: les données de transaction du nœud n'ont pas été modifiées pendant la transaction en cours et des signaux "lecture seule" ou "sortie prématurée" ont été reçus de tous les subordonnés. Le nœud n'a pas besoin d'être informé du résultat de la transaction.

Aucune journalisation n'est nécessaire lorsqu'un nœud passe dans l'état LECTURE-SEULE

- d) **SORTIE-PRÉCOCE**: le nœud a quitté la transaction avant la phase de terminaison; les données de transaction n'ont pas été modifiées pendant la transaction en cours et des signaux "lecture seule" ou "sortie prématurée" ont été reçus de tous les subordonnés. Le nœud n'a pas besoin d'être informé du résultat de la transaction.

Aucune journalisation n'est nécessaire lorsqu'un nœud passe dans l'état SORTIE-PRÉCOCE.

- e) **UNE-PHASE**: le nœud n'a pas fait d'accès à des données liées pendant la transaction en cours et des signaux "lecture seule", "sortie prématurée" ou "une phase" ont été reçus de tous les subordonnés. Le nœud doit être informé du résultat de la transaction si aucune défaillance ne survient, mais un rétablissement n'est pas nécessaire pour ce nœud.

Aucune journalisation n'est nécessaire lorsqu'un nœud passe dans l'état UNE-PHASE.

- f) **DÉCIDE (engagement)**: un nœud passe dans l'état DÉCIDE (engagement) lorsqu'il est en mesure de faire passer sous sa propre responsabilité les données liées dans l'état final et que l'une des deux situations suivantes se présente:

- 1) tous les voisins adjacents ont émis des signaux "prêt", "lecture seule", "sortie prématurée" ou "une phase" et le nœud a déterminé qu'il est le coordinateur de l'engagement; ou
- 2) le nœud est passé précédemment dans l'état PRÊT et a reçu entre temps un ordre d'engagement en provenance de son maître d'engagement.

Si le nœud a déterminé qu'il est le coordinateur de l'engagement et décide d'engager la transaction, il consignera alors un enregistrement de journalisation d'engagement avant de propager la décision vers les esclaves d'engagement. Après avoir décidé d'engager la transaction, et si le dialogue existe toujours, le nœud propagera également la décision d'engagement vers tout voisin dont il a reçu un signal "une phase". L'enregistrement de journalisation d'engagement comportera l'identificateur de transaction, la décision d'engagement et la liste des nœuds adjacents ayant signalé qu'ils étaient prêts. Le nœud passe dans l'état DÉCIDE (engagement) à partir du moment où il a stocké l'enregistrement de journalisation d'engagement. Un tel nœud qui stocke un enregistrement de journalisation d'engagement sans avoir reçu d'ordre d'engagement est appelé coordinateur de l'engagement pour la transaction.

Une collision de signaux "prêt" peut avoir lieu, par exemple, lorsque deux nœuds voisins émettent l'un vers l'autre des signaux "prêt" qui entrent en collision sur le dialogue. Dans un tel cas, un mécanisme de décision (décrit dans la Rec. UIT-T X.862 | ISO/CEI 10026-3 dans les signaux de collision) conduit à ignorer l'un des signaux "prêt" d'une manière cohérente pour les deux nœuds, après quoi les procédures normales s'appliquent.

NOTE 1 – Ce traitement de la collision de signaux "prêt" a pour objectif de limiter la complexité du protocole en renvoyant aussi rapidement que possible l'état qui est atteint normalement en l'absence de collision. Ceci est fait en renonçant à l'avantage que présente la diminution du temps moyen nécessaire pour informer les nœuds au sujet de la décision d'engagement lorsque les deux nœuds deviennent des sources autonomes de décisions d'engagement.

Le nœud passe immédiatement dans l'état DÉCIDE (engagement) s'il se trouvait déjà dans l'état PRÊT au moment où il reçoit un ordre d'engagement. Le nœud propage la décision vers ses esclaves d'engagement. Si le dialogue existe toujours, le nœud propagera également la décision d'engagement vers tout voisin dont il a reçu un signal "une phase". Il peut, d'une manière optionnelle, stocker un enregistrement de journalisation d'engagement dans le journal de rétablissement. Cet enregistrement contient l'identificateur de transaction, la décision d'engagement et la liste des esclaves d'engagement. Un tel nœud, qui n'est pas coordinateur de l'engagement, n'a pas l'obligation de stocker la décision d'engagement dans le journal de rétablissement; il peut toutefois accélérer la procédure de rétablissement s'il procède de la sorte.

Dans tous les cas, une fois que l'engagement s'est achevé au niveau du nœud (les données liées sont libérées) et que tous les nœuds adjacents qui avaient émis le signal "prêt" ont rendu compte de l'achèvement ou de la journalisation de leur engagement, le nœud peut supprimer l'enregistrement de journalisation "prêt" ou l'enregistrement de journalisation "engagé". S'il n'est pas le coordinateur de l'engagement, il rend ensuite compte au maître d'engagement de l'achèvement de l'engagement.

Si le nœud n'était pas le coordinateur de l'engagement, mais stocke l'enregistrement optionnel de journal d'engagement, il peut également d'une manière optionnelle rendre compte de l'achèvement de l'engagement une fois qu'il a effectué cette opération (prière toutefois de se référer également au compte rendu heuristique décrit ci-dessous).

Un nœud qui a émis un ordre d'engagement à destination d'un voisin peut, d'une manière optionnelle, supprimer l'information concernant ce voisin dans les enregistrements de journal d'engagement ou les enregistrements de journalisation "prêt" une fois que le voisin a rendu compte de l'achèvement de l'engagement.

g) **DÉCIDE (retour-arrière)**: le nœud a pris la décision de procéder à un retour-arrière de la transaction ou a reçu un ordre de retour-arrière pendant qu'il se trouvait dans l'état ACTIF, ou pendant qu'il se trouvait dans l'un des états PRÊT, LECTURE-SEULE ou UNE-PHASE et qu'il a soit:

- 1) reçu un ordre de retour-arrière d'un voisin à qui il avait envoyé l'un des signaux "prêt", "lecture seule" ou "une phase"; ou
- 2) déterminé qu'il était coordinateur de l'engagement et qu'il a décidé de procéder au retour-arrière de la transaction.

La décision de retour-arrière ne nécessite en elle-même aucune journalisation. Si un nœud se trouvait dans l'état PRÊT et qu'il a reçu un ordre de retour-arrière ou s'il a déterminé qu'il était coordinateur de l'engagement, il peut alors supprimer son enregistrement de journalisation "prêt".

h) **DÉCIDE (inconnu)**: le nœud qui se trouvait dans l'un des états LECTURE-SEULE ou SORTIE-PRÉCOCE a reçu une indication qu'il ne sera pas informé du résultat de la transaction.

Aucune journalisation n'est nécessaire lorsqu'un nœud passe dans l'état DÉCIDE (inconnu).

L'état d'un nœud détermine si une décision heuristique est possible. Une décision heuristique n'est possible que si les données se trouvent dans l'état "prêt pour l'engagement". Un nœud dans l'état ACTIF peut avoir certaines de ces données qui se trouvent déjà dans l'état "prêt pour l'engagement", sous condition qu'il a la permission de démarrer la phase 1 de l'engagement, comme décrit au 8.6.1.2. Si un nœud se trouve dans l'état PRÊT, ses données liées se trouvent alors dans l'état "prêt pour l'engagement".

Un nœud peut prendre des décisions heuristiques s'il se trouve dans l'état ACTIF et qu'il a la permission de démarrer la phase 1 de l'engagement ou d'entrer dans l'état PRÊT. Le nœud ne change pas d'état dans l'un ou l'autre cas.

L'état des données liées au moment de l'achèvement d'une transaction affectée par des décisions heuristiques peut, en fonction de contraintes choisies par le supérieur, faire l'objet d'un compte rendu destiné au supérieur et éventuellement à la racine de la transaction. Si une décision heuristique est prise (c'est-à-dire si un nœud libère les données dans leur état final ou initial avant d'avoir reçu le résultat final de la transaction), le nœud stocke alors un enregistrement de journalisation heuristique dans le journal de rétablissement avant de libérer les données. L'enregistrement de journalisation heuristique contient l'identificateur de transaction et la décision.

NOTE 2 – Le compte rendu heuristique est toujours propagé vers la racine de l'arbre de transaction et peut donc être transmis dans le même sens que les ordres d'engagement ou en sens inverse.

Un enregistrement de journalisation de dommage est stocké par un nœud qui a pris une décision heuristique et qui n'est pas en mesure de réparer le dommage dans le cas où le résultat final de la transaction diffère de sa décision heuristique.

Les deux alinéas suivants s'appliquent si un compte rendu heuristique est exigé et que le résultat final de la transaction a été un engagement.

Si le nœud a reçu l'ordre d'engagement de son supérieur, il n'effectuera pas de compte rendu d'achèvement tant qu'il n'a pas été informé de l'état des données liées de son sous-arbre de transaction.

Si le nœud a reçu l'ordre d'engagement d'un subordonné, ce dernier informera alors le nœud au sujet de l'état des données liées dans le sous-arbre du subordonné, une fois que cet état est connu. Le supérieur attendra ce compte rendu s'il n'a pas été reçu au moment où le supérieur est par ailleurs prêt à rendre compte de l'achèvement de l'engagement; ceci garantit que l'état des données liées sera déterminé d'une manière fiable.

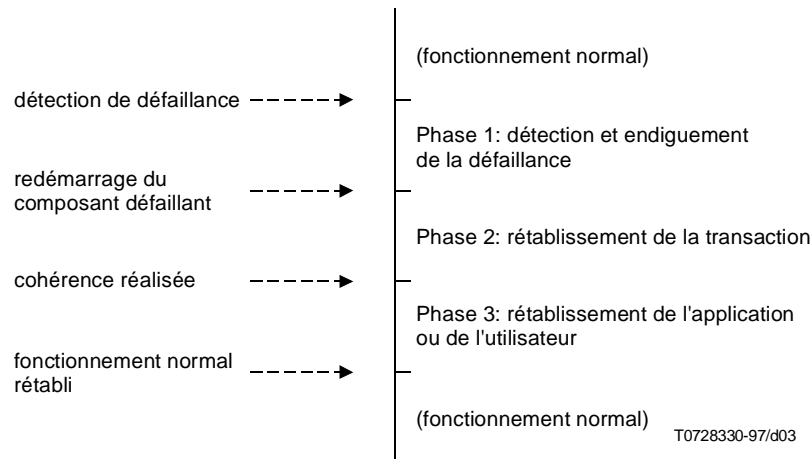
#### 8.7.4 Phases de rétablissement

##### 8.7.4.1 Aperçu général

Le rétablissement du traitement OSI TP peut être subdivisé en trois étapes distinctes appelées phases de rétablissement. Ces étapes sont les suivantes:

- a) détection et endiguement des défaillances;
- b) rétablissement de la transaction; et
- c) rétablissement du dialogue ou de l'utilisateur.

L'objectif de chaque phase est indépendant de l'état du nœud, tandis que le type de l'action de rétablissement prise dans chaque phase dépend de l'état du nœud. La Figure 3 illustre la succession des phases de rétablissement.



**Figure 3/X.860 – Phases du rétablissement**

La phase 1 débute après la détection d'une défaillance, ce qui démarre le rétablissement. La communication sur certaines branches de l'arbre de transaction est impossible du point de vue du fournisseur TPSP. Cette phase tente limiter le coût de la défaillance, c'est-à-dire de minimiser le temps pendant lequel des ressources rares sont immobilisées d'une manière improductive.

La phase 2 est lancée pour rétablir ou redémarrer les composants défaillants de la transaction. Lorsque ces composants ont été redémarrés, des activités sont alors lancées pour déterminer si les données liées sont dans un état cohérent et pour les rétablir éventuellement dans un état cohérent. Le type de l'activité de rétablissement lancée à ce stade dépend de l'état du nœud.

La phase 3 peut être entamée uniquement après:

- a) la réussite de la phase 2 du rétablissement; ou
- b) une défaillance de la communication qui se produit lorsque que le nœud se trouve dans l'état ACTIF. Le retour-arrière présumé a pour effet de garantir la cohérence de l'état des données liées.

Le traitement OSI TP ne fournit aucune fonctionnalité spécifique pour la phase 3 du rétablissement qui s'effectue sous la responsabilité de l'application ou de l'utilisateur.

### 8.7.4.2 Phase 1: détection et endiguement de la défaillance

Cette phase est entamée après des types de défaillances indiqués dans le Tableau 3 ci-dessus.

Pour tous les types de défaillances, à l'exception de l'indisponibilité des données d'action atomique, l'état actuel du nœud détermine l'action de rétablissement éventuelle à effectuer.

L'état du nœud est rétabli à partir des enregistrements de journalisation si des données d'action atomique deviennent indisponibles; l'état du nœud détermine dans ce cas l'action de rétablissement éventuelle à effectuer.

L'état d'un nœud qui a subi un écroulement est restauré comme suit:

- a) le nœud est rétabli dans l'état PRÊT si un enregistrement de journalisation "prêt" est disponible pour cette transaction; ou
- b) le nœud est rétabli dans l'état DÉCIDE (engagement) si un enregistrement de journal d'engagement est disponible pour cette transaction. Dans ce cas, le résultat de la transaction est un engagement; ou
- c) la transaction est oubliée si aucun enregistrement n'a été trouvé. Le Tableau 4 résume la restauration de l'état du nœud après que des données d'action atomique sont devenues indisponibles.

NOTE – La présence d'un enregistrement de journalisation heuristique n'affecte pas la restauration de l'état du nœud.

**Tableau 4/X.860 – Restauration de l'état d'un nœud après une indisponibilité de données atomiques**

| Type d'enregistrement de journalisation   | Aucun               | Prêt | Engagement          |
|---|---------------------|------|---------------------|
| état du nœud  | transaction oubliée | PRÊT | DÉCIDE (engagement) |
| NOTE – L'expression "transaction oubliée" ne désigne pas un état; elle concerne des actions prises par la machine CPM lorsque les informations se rapportant à la transaction cessent d'exister, à l'exception des enregistrements éventuels de journalisation heuristique ou de journalisation de dommage. |                     |      |                     |

Les actions de rétablissement sont les suivantes:

- a) état ACTIF: le nœud ramène ses données liées à leur état initial et transmet le retour-arrière à tous les nœuds avec lesquels il est éventuellement en communication.  
  
Le nœud oublie la transaction une fois que le retour-arrière est terminé et que la phase 1 du rétablissement s'achève. Le rétablissement du traitement transactionnel est terminé.
- b) état PRÊT: le nœud a pris éventuellement une décision heuristique avant que la défaillance survienne. Aucune action particulière n'est effectuée dans ce cas, car un enregistrement de journalisation heuristique a déjà été stocké.  
  
En variante, il se peut que le nœud ait pris une décision heuristique, auquel cas il procède à l'écriture d'un enregistrement de journalisation heuristique.  
  
Le rétablissement passe à la phase 2.
- c) état LECTURE-SEULE ou SORTIE-PRÉCOCE: le nœud ne change pas d'état si le dialogue existe toujours avec le supérieur. Dans le cas contraire, le nœud informe de la fin de la transaction tous ses subordonnés avec lesquels il est en communication; il oublie la transaction et met fin à la phase 1 du rétablissement, après quoi le rétablissement du traitement transactionnel se termine.
- d) état UNE-PHASE: le nœud reste dans l'état UNE-PHASE si le dialogue existe toujours avec le voisin à destination duquel le signal "une phase" a été émis. Dans le cas contraire, le nœud informe tous ses voisins avec lesquels il est en communication que le résultat de la transaction est inconnu; il oublie la transaction et met fin à la phase 1 du rétablissement, après quoi le rétablissement du traitement transactionnel se termine.
- e) état DÉCIDE (engagement) ou DÉCIDE (retour-arrière): le nœud se comporte de façon normale, comme indiqué au 8.6, pour la partie de l'arbre de transaction qui n'est pas affectée par la défaillance.

La phase 1 du rétablissement s'achève pour la partie de l'arbre de transaction affectée et le rétablissement passe à la phase 2.

### 8.7.4.3 Phase 2: rétablissement de la transaction

Cette phase débute après l'achèvement de la phase 1 du rétablissement. La phase 2 du rétablissement est entamée lorsque la communication avec un nœud adjacent a été interrompue et que le résultat final de la transaction doit être communiqué. Le nœud cherche à restaurer la communication à des fins de rétablissement uniquement dans deux situations suivantes:

- a) avec le nœud supérieur, si le nœud se trouve dans l'état PRÊT, ou
- b) avec un nœud subordonné, si le nœud se trouve dans l'état DÉCIDE (engagement) et que la communication avec le subordonné a été interrompue avant la fin du compte rendu d'état des données liées du sous-arbre du subordonné.

Le rétablissement de la communication utilise un canal.

L'état actuel du nœud détermine l'action de rétablissement éventuelle.

Les actions de rétablissement sont les suivantes:

- a) état ACTIF: le nœud n'entame jamais la phase 2 du rétablissement tant qu'il se trouve dans l'état ACTIF car la procédure de retour-arrière présumé le dispense de communiquer le résultat de la transaction.
- b) état PRÊT: aucune action de rétablissement n'est nécessaire si la communication avec un esclave de l'engagement a échoué. Le nœud procède comme suit si la communication a échoué avec le maître de l'engagement:

- 1) rétablir la communication avec le nœud supérieur; et
- 2) interroger le nœud supérieur au sujet du résultat de la transaction. Le nœud procède comme suit lorsqu'il reçoit la réponse du maître de l'engagement qui véhicule le résultat final de la transaction:;
  - i) si le maître de l'engagement indique qu'il ne connaît pas la transaction, le nœud se conforme alors à la procédure de retour-arrière présumé en passant dans l'état DÉCIDE (retour-arrière) et en libérant ses données liées dans leur état initial, à moins qu'une décision heuristique n'ait été prise. Le nœud propage également, le cas échéant, le retour-arrière vers tous les esclaves d'engagement avec lesquels il communique.

Le nœud oublie la transaction et la phase 2 du rétablissement s'achève. Le rétablissement du traitement transactionnel est terminé.

- ii) si le maître de l'engagement répond en indiquant que le résultat de la transaction est un engagement, le nœud passe alors dans l'état DÉCIDE (engagement) et libère ses données liées dans leur état final, à moins qu'une décision heuristique n'ait été prise. Le nœud propage également l'engagement vers tous ses subordonnés avec lesquels il communique éventuellement. Le nœud se comporte comme indiqué ci-dessous dans l'alinéa c) pour les nœuds subordonnés avec lesquels la communication a été interrompue.

Le nœud oublie la transaction et la phase 2 du rétablissement s'achève une fois que le nœud a rendu compte à son supérieur de l'état des données liées dans son sous-arbre. Le rétablissement du traitement transactionnel se termine.

- c) état DÉCIDE (engagé): aucune action de rétablissement n'est nécessaire lorsque la communication avec le maître de l'engagement a échoué. Le nœud procède alors comme suit si la communication a échoué avec un esclave de l'engagement:
  - 1) rétablir la communication avec l'esclave de l'engagement; et
  - 2) propager l'engagement vers cet esclave.

La phase 2 du rétablissement s'achève lorsque le nœud reçoit un compte rendu d'état des données liées dans le sous-arbre du subordonné. Le rétablissement du traitement transactionnel se termine; ou

- d) aucune action de rétablissement n'est nécessaire si le nœud a oublié la transaction. Le nœud passe néanmoins à la phase 2 du rétablissement à la demande d'un esclave d'engagement ou du maître éventuel de l'engagement.

Si la demande provient du nœud supérieur, le nœud lui rend alors compte de l'état des données liées dans son sous-arbre.

NOTE – Le nœud tient compte de la présence de tout enregistrement de journalisation de dommage lorsqu'il notifie l'état des données liées dans son sous-arbre.

Le nœud répond en indiquant que le résultat de la transaction est un retour-arrière si la demande provient d'un esclave de l'engagement,

## 8.8 Gestion de la simultanéité et blocage mutuel

Les mécanismes de gestion de la simultanéité mis en œuvre dans différents systèmes ouverts sortent du cadre des spécifications du traitement OSI TP. Compte tenu du fait que certains mécanismes de gestion de la simultanéité n'excluent pas l'apparition de blocages mutuels entre des ressources appartenant à plusieurs systèmes ouverts (blocages mutuels globaux), on admet que de tels blocages mutuels sont évités ou au moins détectés.

NOTE – Un des moyens mis en œuvre à cette fin consiste à utiliser des temporisations associées aux transactions réparties. Si une transaction donnée n'obtient pas le verrouillage demandé dans un intervalle de temps donné, elle fera alors l'objet d'un retour-arrière, car il existe une présomption de blocage mutuel.

## 8.9 Sécurité

NOTE – La fourniture de fonctions de sécurité fera l'objet d'une normalisation future sous la forme d'un amendement.

# Annexe A

## Relation entre le modèle OSI TP et la structure de la couche Application

### A.1 Introduction

La présente édition du traitement OSI TP utilise la terminologie et les mécanismes de modélisation de la structure de couche Application (voir ISO/CEI 9545).

La présente annexe décrit la structure fondamentale d'une invocation de processus d'application (API, *application-process invocation*) au sein d'une invocation d'entité d'application (AEI, *application-entity invocation*) qui prend en charge une invocation TPSUI.

### A.2 Processus d'application dans le traitement OSI TP

Les utilisateurs TPSU font partie de processus d'application. Un processus d'application donné peut contenir un ou plusieurs utilisateurs TPSU.

Un utilisateur TPSU utilise le service OSI TP pour réaliser une communication OSI TP. Il effectue deux tâches distinctes propres à l'application:

- a) le traitement visant à réaliser une partie de la transaction; et
- b) la communication permettant de dialoguer avec des utilisateurs TPSU partenaires.

Le service OSI TP est disponible au sein de l'invocation d'entité d'application.

Dans un processus d'application, les besoins de communication entre utilisateurs TPSU propres à l'application sont satisfaits par un ou plusieurs éléments ASE utilisateur au sein d'un ou plusieurs objets SAO.

Les informations sémantiques de choix des utilisateurs TPSU au sein d'un processus d'application sont véhiculées par le protocole OSI TP en utilisant l'adressage de la couche Application avec des titres et des identificateurs spécifiques définis dans la présente Recommandation.

### A.3 Entités d'application dans le traitement OSI TP

#### A.3.1 Invocations d'entités d'application dans le traitement OSI TP

Une invocation d'entité d'application qui communique et qui est impliquée dans le traitement OSI TP contient un ou plusieurs objets SAO. Il en résulte que l'invocation TPSUI peut utiliser simultanément plusieurs objets SAO. Cela ne signifie toutefois pas qu'il existe des relations entre plusieurs objets SAO et plusieurs invocations TPSUI. En effet, un objet SAO donné ne peut être utilisé que par une seule invocation TPSUI à un instant donné.

#### A.3.2 Fonction MACF du traitement OSI TP

Une entité d'application qui prend en charge le traitement OSI TP comprend toujours une fonction MACF.



La fonction MACF est, au sein du traitement OSI TP, le composant de la machine TPPM qui coordonne les interactions entre les associations multiples dans une invocation d'entité d'application afin de fournir le service OSI TP. Ses fonctions sont les suivantes:

- a) allocation d'associations d'application, à des fins d'utilisation et de réutilisation par les dialogues ou les canaux;
- b) établissement et terminaison du lien entre une invocation TPSUI et un ou plusieurs objets SAO appropriés; et
- c) coordination des activités entre les multiples associations d'application, en vue de garantir les propriétés ACID des transactions. La fonction MACF doit en particulier:
  - 1) provoquer la génération du protocole pour une ou plusieurs associations individuelles, comme nécessaire pour la prise en charge du service OSI TP;
  - 2) avoir journalisé dans une mémoire sûre les données d'action atomique nécessaires pour la prise en charge du rétablissement, ainsi que les informations nécessaires au rétablissement des données liées de l'utilisateur associées à la transaction; et
  - 3) coordonner, après une défaillance d'application ou de communication, les mécanismes nécessaires pour rétablir chaque objet SAO au sein de l'invocation d'entité d'application pour le compte de l'invocation TPSUI.

### **A.3.3 Objets SAO dans le traitement OSI TP**

Tout objet SAO comprend un ou plusieurs éléments ASE utilisateur avec association unique qui prennent en charge la communication propre à l'application entre les invocations TPSUI, ainsi que les caractéristiques d'association unique de la machine TPPM.

Dans le contexte du traitement OSI TP, les procédures d'engagement, de simultanéité et de rétablissement (CCR, *commitment, concurrency, and recovery*) ne sont présentes que si un niveau de coordination "engagement" ou "engagement en une phase" est nécessaire. Les procédures CCR sont invoquées uniquement par la machine TPPM lorsqu'elles sont présentes.

Dans le domaine d'un dialogue, l'association d'application qui prend en charge le dialogue entre deux invocations TPSUI est partagée par l'entité TPASE, l'entité ACSE, un ou plusieurs éléments ASE utilisateur et éventuellement l'ensemble CCR. Le contenu sémantique d'un élément ASE utilisateur ne peut être échangé que lorsque la machine TPPM se trouve dans certains états précis. L'environnement OSI TP impose des contraintes aux éléments ASE utilisateur, de sorte que les règles du traitement OSI TP sont respectées aussi bien au niveau du service OSI TP qu'au niveau du protocole OSI TP.

La fonction SACF de tout objet SAO modélise les fonctions suivantes:

- a) coordination nécessaire pour des interactions entre les caractéristiques de l'association unique de la machine TPPM et les autres éléments ASE contenus dans l'objet SAO, telle qu'elle est spécifiée par la définition du contexte d'application de l'association;
- b) coordination de l'utilisation du service de présentation par les composants individuels de l'objet SAO; et
- c) concaténation et séparation des unités APDU, selon le cas.

Une définition de contexte d'application décrit la manière dont les composants normalisés sont incorporés dans l'environnement de traitement transactionnel.

### **A.3.4 Élément ASE utilisateur**

Les besoins spécifiques de communication de l'application pour un utilisateur TPSU au sein d'un processus d'application peuvent être satisfaits par un ou plusieurs objets SAO appartenant à la machine TPPM.

## **A.4 Frontières du service OSI TP**

La fonctionnalité globale du service OSI TP est fournie par le biais de la fonction MACF (par exemple, pour la traduction du service d'engagement (se référer à la Rec. UIT-T X.861 | ISO/CEI 10026-2) vers des primitives individuelles du service CCR dans des objets SAO multiples).

Le service OSI TP permet à l'invocation TPSUI d'effectuer l'incorporation des ressources locales dans l'engagement, ainsi que la coordination du service OSI TP avec d'autres services de la couche Application.

Le fournisseur TPSP contient une machine TPPM pour toute instance TPSUI et une machine CPM pour toute invocation d'entité d'application. Toute machine TPPM contient:

- a) une fonctionnalité MACF pour le traitement OSI TP; et
- b) pour toute association, un objet SAO comprenant:
  - 1) une fonctionnalité SACF pour le traitement OSI TP;

- 2) un élément ACSE;
- 3) un élément TPASE;
- 4) un ou plusieurs éléments ASE utilisateur; et
- 5) le service CCR si le traitement de l'engagement est requis.

Une machine CPM comprend:

- a) une fonctionnalité MACF pour le traitement OSI TP; et
- b) pour toute association, un objet SAO comprenant:
  - 1) une fonctionnalité SACF pour le traitement OSI TP;
  - 2) un élément ACSE;
  - 3) un élément TPASE; et
  - 4) le service CCR.

## **Annexe B**

### **Présentation didactique de la gestion de la simultanéité et du blocage mutuel pour le traitement OSI TP**

Une transaction répartie est définie sous la forme d'une unité de travail atomique qui réussit ou échoue d'un seul tenant. Il est toutefois souhaitable que des transactions puissent s'exécuter simultanément au sein d'un système, ce qui nécessite la fourniture de mécanismes de gestion de simultanéité pour l'accès aux ressources partagées.

Il est important, dans un système réparti, que les divers gestionnaires de ressources locales participant à la transaction puissent prendre en charge des niveaux compatibles de gestion de simultanéité (c'est-à-dire fournissant tous des procédures de "mises à jour autorisées avec lectures susceptibles d'être répétées").

Les mécanismes assurant la compatibilité de gestion de la commande de simultanéité sortent du cadre du traitement OSI TP (ils peuvent, par exemple, être traités par l'application).

Le verrouillage à des fins de gestion de la simultanéité pose le problème bien connu du blocage mutuel. Ceci n'est pas un problème de l'environnement OSI si un seul système est impliqué: ce système peut choisir de mettre en œuvre un mécanisme local quelconque pour résoudre le problème du blocage mutuel. Une situation de blocage mutuel peut par contre se produire dans l'environnement OSI entre des transactions qui tentent d'accéder à des ressources situées sur un certain nombre de systèmes ouverts.

Il existe essentiellement deux démarches de gestion du blocage mutuel pouvant convenir à des systèmes OSI TP, à savoir la détection du blocage mutuel et la prévention du blocage mutuel.

Dans la première démarche (détection de blocage mutuel), le système attend qu'un blocage mutuel se produise. Les algorithmes de détection utilisent en général des graphes d'attente. Un graphe d'attente est un graphe orienté indiquant quelles sont les transactions qui sont en attente d'autres transactions. Dans l'environnement du traitement transactionnel réparti, les graphes d'attente locaux doivent être combinés pour produire un graphe d'attente global.

L'inconvénient principal de la détection de blocage mutuel réside dans la charge supplémentaire résultant de la gestion des graphes d'attente et de la détection de cycles dans ces graphes.

La prévention des blocages mutuels évite l'apparition de tels blocages par l'interruption des transactions lorsque se présentent des situations susceptibles de conduire à un blocage mutuel. Les transactions se déroulent normalement tant que les ressources demandées sont disponibles. Si une ressource demandée n'est pas disponible, il est alors nécessaire d'interrompre soit la transaction en cours d'exécution, soit la transaction qui a verrouillé la ressource, le critère de choix dépendant du schéma de prévention utilisé.

Un des procédés utilisables est l'algorithme de temporisation de blocage mutuel, qui associe à toute transaction une temporisation maximale d'attente d'un verrouillage de ressource.

Supposons qu'une transaction T1 nécessite le verrouillage d'une ressource R1. L'algorithme de temporisation de blocage mutuel est alors mis en œuvre de la manière suivante:

### **temporisation de blocage mutuel**

si la transaction T1 obtient le verrouillage de la ressource R1 avant la fin de la temporisation de blocage mutuel, elle poursuit alors son traitement;

la transaction T1 est interrompue si elle n'obtient pas le verrouillage de la ressource R1 avant la fin de la temporisation.

Il existe au moins deux raisons qui peuvent conduire à préférer une prévention du blocage mutuel dans le cas de certains systèmes. Il est possible, en premier lieu, que les coûts supplémentaires de détection (lignes de code système, etc.) ne soient pas justifiés. Une autre éventualité est le risque que la détection crée des goulets d'étranglement (tels que des blocages de trafic), conduisant à un encombrement excessif pendant le temps nécessaire à la détection et à la résolution de la situation du blocage mutuel.

Il existe un certain nombre d'algorithmes permettant de détecter et d'éviter un blocage mutuel. Certains d'entre eux peuvent être utilisés dans des environnements répartis. Ces algorithmes sont de deux types:

- a) précis: il s'agit d'algorithmes qui détectent tous les blocages mutuels réels et eux seuls;
- b) imprécis: il s'agit d'algorithmes qui détectent tous les blocages mutuels réels, mais qui signalent parfois une situation de blocage mutuel dans un cas de fonctionnement normal.

Il n'existe pas à l'heure actuelle d'algorithme "précis" de gestion du blocage mutuel qui soit bien compris et qui puisse fonctionner avec une bonne efficacité de communication dans des environnements hétérogènes. On fait donc l'hypothèse que la détection des blocages mutuels locaux est effectuée par des temporisations, c'est-à-dire par un algorithme "imprécis".

## **Annexe C**

### **Informations didactiques concernant le protocole de retour-arrière présumé avec engagement en deux phases**

Une transaction répartie est caractérisée par l'existence d'un certain nombre d'acteurs qui participent aux activités d'engagement de la transaction. Ces activités doivent s'effectuer d'une manière ordonnée et ont été regroupées de ce fait en deux phases.

Au cours de la phase I, les participants enregistrent dans une mémoire sûre les modifications apportées aux ressources protégées, afin de garantir les propriétés "A", "C" et "D" des propriétés ACID. La phase II démarre en enregistrant l'engagement si tous les participants ont réussi. Toutes les ressources utilisées par la transaction sont libérées pendant la phase II, ce qui permet de garantir la propriété "I" des propriétés ACID.

Un coordinateur de l'engagement est responsable de déterminer si la phase I s'est achevée avec succès. Pour ce faire, le coordinateur de l'engagement collecte les votes "prêt" en provenance des participants. La transaction se termine comme ENGAGÉE si un consensus PRÊT est atteint.

Le coordinateur de l'engagement est responsable, pendant le rétablissement d'une défaillance utilisant la procédure de retour-arrière présumé, de la fourniture à ses voisins du résultat final de la transaction uniquement si cette dernière est ENGAGÉE. Si un nœud a une transaction qui se trouve dans l'état PRÊT après une défaillance, il est responsable de l'interrogation du maître de l'engagement au sujet du résultat final de la transaction. Si le maître de l'engagement n'a pas connaissance de la transaction, le subordonné présume alors qu'elle doit subir un RETOUR-ARRIÈRE. La réponse du maître de l'engagement peut également être ENGAGEMENT ou une demande de renouvellement ultérieur de la demande. En l'absence de défaillance, le maître de l'engagement doit informer dans tous les cas ses esclaves d'engagement du résultat de la transaction.

Le protocole de retour-arrière présumé nécessite seulement qu'un coordinateur de l'engagement enregistre ses esclaves d'engagement lorsque ceux-ci sont dans l'état "prêt" et que la transaction peut être ENGAGÉE. Cela est fait au niveau du nœud du coordinateur de l'engagement dans l'enregistrement de journal d'engagement. Un nœud intermédiaire peut également, lorsque son supérieur lui ordonne l'engagement, stocker un enregistrement de journal d'engagement, mais il n'est pas obligé de le faire. La tenue d'un tel enregistrement constitue une charge supplémentaire pour chaque transaction. Elle accélère toutefois le rétablissement, dans la mesure où la décision finale peut se propager à partir du nœud dans le sous-arbre, même si la communication avec son supérieur est toujours coupée. Tous les nœuds, à l'exception du coordinateur de l'engagement, doivent enregistrer leur maître d'engagement (immédiat) dans un enregistrement de journalisation "prêt" lorsqu'ils ont reçu les votes "prêt" de tous les subordonnés et qu'ils sont prêts eux-mêmes et avant de transmettre leur vote "prêt" au supérieur.

## Annexe D

### Combinaison d'optimisations d'engagement

La présente annexe analyse diverses combinaisons d'optimisations liées à l'engagement (pouvant être sélectionnées au moyen d'unités fonctionnelles au sein du service et du protocole de traitement transactionnel) et en analyse les mérites relatifs.

#### D.1 Engagement dynamique avec commande polarisée

L'utilisation de l'unité fonctionnelle de commande polarisée pour un dialogue limite les degrés de liberté des invocations TPSUI, parce que la commande par une invocation TPSUI est nécessaire pour tout dialogue coordonné pour lequel l'unité fonctionnelle de commande polarisée a été sélectionnée, à moins qu'une primitive d'indication TP-PREPARE ou TP-READY ait été reçue sur le dialogue.

L'unité fonctionnelle de commande polarisée présente l'avantage de fournir un maintien de la discipline d'utilisation des dialogues par les invocations TPSUI, ce qui conduit à des cas de collision moins complexes; elle introduit toutefois des limitations strictes lorsqu'elle est utilisée avec la procédure d'engagement dynamique, ce qui peut conduire à préférer l'utilisation de la commande partagée pour des applications nécessitant l'engagement dynamique et la prévention des collisions par l'utilisation d'un protocole de maintien de la discipline (par exemple, l'appel de procédure à distance ou le modèle client-serveur).

Tant que l'avant-dernière procédure "prêt" n'est pas appliquée, un nœud qui émet une primitive de demande TP-COMMIT émettra également une unité PDU C-PREPARE-RI sur tous les dialogues pour lesquels une unité PDU C-PREPARE-RI ou une unité PDU C-READY-RI n'a pas encore été échangée. Cette procédure implique qu'une invocation TPSUI peut recevoir plusieurs primitives d'indication TP-PREPARE.

#### D.2 Préparation implicite non sélectionnée et signal "prêt" autorisé dans les deux sens

##### D.2.1 Dernier subordonné

Cette optimisation permet à un nœud se trouvant en cours de préparation d'émettre une unité PDU C-PREPARE-RI à destination de tous ses voisins sauf un, le dernier voisin, qui recevra une unité PDU C-READY-RI, lorsque le nœud en cours de préparation et son arrière-plan de transaction seront passés dans l'état "prêt". Le dernier nœud, qui est devenu un coordinateur d'engagement potentiel, peut choisir de devenir le coordinateur de l'engagement ou peut utiliser à son tour l'optimisation de dernier subordonné avec l'un de ses subordonnés.

Une collision peut toutefois survenir entre une unité PDU C-READY-RI et une unité PDU C-PREPARE-RI lors de la commande partagée et il est possible que cette collision ne soit pas détectée.

#### Avantages

L'optimisation de dernier subordonné présente l'avantage suivant: étant donné que celui-ci démarre en premier la libération de ses données sélectionnées et peut procéder en dernier au retour-arrière de la transaction, ceci permet de choisir un système qui impose des contraintes importantes aux données liées du dernier subordonné. Cette optimisation est utilisée en général entre une station de travail et un serveur, qui est alors le dernier subordonné.

## **Combinaison avec d'autres unités fonctionnelles**

L'unité fonctionnelle de commande polarisée impose certaines contraintes: un nœud doit disposer de la commande de tous ses dialogues pour devenir l'instigateur d'un engagement et le dernier subordonné ne doit pas être en charge de la commande du dialogue avec son supérieur.

Tout nœud peut devenir l'instigateur de l'engagement si l'unité fonctionnelle d'engagement dynamique est sélectionnée. Dans le cas contraire, l'instigateur de l'engagement sera la racine de l'arbre de transaction.

L'unité fonctionnelle "lecture seule" peut être sélectionnée sans contraintes pour tous les dialogues, auquel cas l'unité PDU C-READY-RI sera remplacée par une unité PDU C-NOCHANGE-RI, mais cette optimisation ne peut pas être utilisée vers le bas dans l'arbre de transaction.

L'unité fonctionnelle "une phase" peut être sélectionnée sans contraintes.

Les paramètres "le supérieur peut émettre le signal prêt" et "le subordonné peut émettre le signal prêt" de la primitive de demande TP-BEGIN-DIALOGUE peuvent être utilisés pour limiter l'utilisation de l'optimisation de dernier subordonné: si une invocation TPSUI veut interdire à l'un de ses subordonnés de devenir le dernier subordonné, elle interdit à sa machine TPPM d'émettre une unité PDU C-READY-RI à destination de ce dernier.

### **D.2.2 Dernier subordonné tardif**

L'optimisation de dernier subordonné tardif est une variante de l'optimisation de dernier subordonné; le nœud effectuant la préparation émet une unité PDU C-PREPARE-RI à destination de tous ses voisins sauf un, qui reçoit une unité PDU C-READY-RI.

L'avantage de l'optimisation de dernier subordonné tardif est d'éviter à l'instigateur de l'engagement d'avoir à choisir entre ses subordonnés (ce qui peut entraîner des complications dans des systèmes réels avec plusieurs voisins), mais le dernier subordonné tardif se trouve plus rapidement dans l'état "prêt" que dans le cas du dernier subordonné normal et perd plus tôt le droit d'utiliser un retour-arrière de la transaction pour libérer ses données liées. Il existe également une probabilité d'apparition de collisions entre signaux "prêt".

## **D.3 Préparation implicite**

### **D.3.1 Etat "prêt" spontané**

Cette combinaison permet un mécanisme qui est appelé état "prêt spontané" ou état "prêt non sollicité".

Tout nœud de l'arbre de transaction peut faire passer ses données liées dans l'état "prêt pour l'engagement". Une unité PDU C-READY-RI est émise à destination du dernier voisin lorsque toutes les données liées du nœud et celles de tous ses voisins sauf un (le sous-arbre du nœud si l'unité fonctionnelle d'engagement dynamique n'est pas sélectionnée) se trouvent dans l'état "prêt pour l'engagement" (il peut également s'agir d'une unité PDU C-NOCHANGE-RI si l'arrière-plan de transaction est un sous-arbre dont tous les nœuds se trouvent dans l'état "lecture seule").

La combinaison de cette unité fonctionnelle avec l'unité fonctionnelle "engagement dynamique" permet de construire un arbre de transaction dans lequel tout nœud lancera l'engagement.

### **Avantages**

Cette optimisation permet d'effectuer des transactions rapides en éliminant la vague de préparation sans les limitations de l'optimisation "préparation avec données autorisées" (si le nœud préparé n'est pas en mesure d'effectuer son traitement, la seule solution est alors le retour-arrière de la transaction). L'optimisation avec état "prêt" spontané présente toutefois l'inconvénient suivant: les données liées du nœud sont mises dans l'état "prêt" dès que la partie locale de la transaction s'est terminée, ce qui interdit un retour-arrière en cas de débordement de temporisation et crée un risque de libération des ressources sur décision heuristiques locales si la décision d'engagement est retardée (sauf si le nœud se trouve dans l'état "lecture seule").

## **Combinaison avec d'autres unités fonctionnelles**

L'unité fonctionnelle "commande polarisée" impose certaines contraintes: un nœud doit être responsable de la commande du dialogue pour lequel le signal "prêt" spontané ou le signal "lecture seule" spontané est émis.

L'unité fonctionnelle "lecture seule" peut être sélectionnée librement pour tous les dialogues, mais cette optimisation sera utilisée uniquement pour les branches de transaction qui ne se trouvent pas sur l'itinéraire entre le coordinateur de l'engagement et la racine de l'arbre de la transaction, parce qu'une PDU C-NOCHANGE-RI ne peut être émise que vers le haut de l'arbre de transaction.

L'unité fonctionnelle "engagement en une phase" peut conduire à la collision entre un signal "prêt" spontané et une unité PDU C-NOCHANGE-RI; dans un tel cas l'émetteur de l'unité PDU C-READY-RI et celui de l'unité PDU C-NOCHANGE-RI envoient tous deux une primitive d'indication TP-COMMIT.

Les paramètres "le supérieur peut émettre le signal prêt" et "le subordonné peut émettre le signal prêt" de la primitive de demande TP-BEGIN-DIALOGUE peuvent être utilisés pour limiter l'utilisation de l'optimisation de l'état "prêt" spontané: cet état ne peut être émis que par l'une des extrémités de la branche de transaction (un au moins des paramètres doit être positionné sur "Vrai").

### **D.3.2 Instigateur d'engagement autonome**

Cette optimisation permet à tout nœud, qui a sélectionné l'unité fonctionnelle "préparation implicite" pour le dialogue supérieur, de lancer l'engagement sans attendre de message implicite ou explicite en provenance de la racine.

Une telle initialisation de l'engagement par un nœud autre que la racine peut être limitée au sous-arbre du nœud ou peut être répercutée vers le nœud supérieur.

Un certain nombre de nœuds – la racine et tous les nœuds qui ont une préparation implicite dans leur dialogue supérieur – peuvent initialiser indépendamment l'engagement dans un arbre qui utilise la préparation implicite. Le service TP n'impose aucune contrainte directe pour l'initialisation de l'engagement par ces nœuds (la vérification de ces contraintes nécessiterait des échanges de protocole supplémentaires).

Bien que les nœuds ne soient soumis à aucune contrainte de la part du traitement transactionnel, les invocations TPSUI qui ont la permission d'initialiser l'engagement peuvent être organisées de manière à ce que leur totalité, sauf une, attende un signal de préparation issu d'un voisin quelconque avant d'initialiser l'engagement.

La manière dont l'engagement est initialisé n'affecte pas directement la direction de transmission des signaux "prêt" et n'affecte pas, en conséquence, l'emplacement du coordinateur de l'engagement. La manière dont l'engagement est initialisé n'affecte pas non plus la collecte des comptes rendus heuristiques et d'achèvement, qui se fait toujours en direction de la racine de l'arbre de transaction.

#### **Avantages**

Cette optimisation permet l'initialisation de l'engagement par un nœud autre que la racine. Elle permet également à des sous-arbres d'initialiser l'engagement en parallèle avec d'autres traitements effectués à des niveaux supérieurs de l'arbre.

Elle permet, si l'invocation TPSUI en fait le choix, d'initialiser l'engagement au niveau de tout nœud; le maintien de la discipline pour cette action n'est toutefois pas régi par le fournisseur TPSP.

#### **Combinaison avec d'autres unités fonctionnelles**

L'unité fonctionnelle "commande polarisée" introduit certaines contraintes, car l'émetteur du signal de préparation doit disposer de la commande.

Les paramètres "le supérieur peut émettre le signal prêt" et "le subordonné peut émettre le signal prêt" de la primitive de demande TP-BEGIN-DIALOGUE imposeront certaines contraintes à la direction de transmission des signaux "prêt" et à aux emplacements possibles du coordinateur de l'engagement d'une manière indépendante de l'initialisation de la coordination.

## **Annexe E**

### **Résumés des modifications apportées par la deuxième édition**

La deuxième édition applique les modifications contenues dans l'Amendement 1 de l'ISO/CEI 10026-1:1992; le présent amendement traite des optimisations de l'engagement et contient des amendements au modèle, à la définition de service et à la spécification de protocole.

#### **Contenu fonctionnel de l'Amendement 1**

L'Amendement 1 contient les fonctionnalités suivantes.

- a) **Engagement dynamique en deux phases:** cette fonctionnalité est introduite comme variante pour les procédures (statiques) d'engagement en deux phases de version 1992 de la norme. Dans cette démarche, compte tenu de contraintes explicites, l'instigateur et l'accepteur de branches de transaction peuvent l'un ou l'autre signaler l'état "prêt" ou donner l'ordre d'engagement. Il existe certaines restrictions, aux niveaux dialogue et transaction, pour la signalisation de l'état "prêt" et la demande explicite de préparation. Ceci permet d'utiliser la procédure statique existante d'engagement en deux phases de la Norme internationale ISO/CEI 10026:1992.

- b) **Préparation implicite:** la "préparation vers le supérieur" (service C-PREPARE) est devenue optionnelle. Une invocation TPSUI peut maintenant émettre un signal "prêt" sur la base d'un certain contenu sémantique implicite reçu de son homologue indiquant que ce dernier n'émettra plus d'informations susceptibles d'affecter ses données liées.
- c) **Lecture seule:** ce service optionnel peut être utilisé par un nœud pour indiquer à son supérieur que ses données liées n'ont pas été modifiées et que le nœud n'a aucune préférence pour l'engagement ou le retour-arrière de la transaction.
- d) **Sortie prématurée:** ce service optionnel peut être utilisé par un nœud pour indiquer à son supérieur que lui-même et son sous-arbre ne font aucune contribution à la transaction, que les données liées n'ont pas été modifiées et qu'il se retire de la transaction.
- e) **Engagement en une phase** (dynamique ou statique): l'engagement statique en une phase est fourni pour un nœud qui n'a pas de supérieur dans l'arbre de transaction et pas de données liées. L'engagement statique en une phase peut être pris en charge avec des moyens très simples.

L'engagement dynamique en une phase ne peut être utilisé que si toutes les branches de transaction, sauf une, ont émis l'un des signaux "lecture seule", "sortie prématurée" ou "une phase".

Ces deux engagements en une phase ne nécessitent pas de journalisation car un retour-arrière n'est pas pertinent.

- f) **Annulation:** le traitement transactionnel utilise le nouveau service CCR optionnel (C-CANCEL) qui fournit une fonctionnalité optionnelle de retour-arrière accéléré et non confirmé. Le service C-CANCEL a le même effet qu'un service C-ROLLBACK, mais sans aucune des limitations de ce dernier. Un service C-CANCEL est suivi ultérieurement par un service C-ROLLBACK réel en vue de terminer la branche de transaction.
- g) **Rétablissement de descripteur opaque de contexte** (RCH, *recovery context handle*) au moment de l'initialisation du dialogue: le rétablissement RCH est spécifié dans l'ISO/CEI 10026-3:1995 uniquement lors de l'établissement de l'association. Une fonctionnalité optionnelle permet de le spécifier sur la base d'un dialogue.
- h) **Diagnostics d'achèvement:** les services liés à l'engagement et au retour-arrière prennent maintenant en charge optionnellement les diagnostics.
- i) **Endiguement de comptes rendus heuristiques:** l'émission de comptes rendus heuristiques peut maintenant être supprimée au niveau du dialogue.

L'Amendement 1 a introduit certaines contraintes de service globales, c'est-à-dire des contraintes au niveau d'un nœud qui dépendent d'un autre nœud. Il est prévu que l'application assure le respect de ces contraintes. Ces dernières sont nécessaires pour éviter des collisions entre des données d'application et un signal "prêt" qui pourraient se produire lors de l'utilisation de la préparation implicite avec la commande partagée. L'application doit utiliser uniquement la commande polarisée avec préparation implicite si elle ne peut pas ou ne veut pas imposer le respect de ces contraintes.





## SÉRIES DES RECOMMANDATIONS UIT-T

|                |   |
|----------------|---|
| Série A        | Organisation du travail de l'UIT-T  |
| Série B        | Moyens d'expression: définitions, symboles, classification  |
| Série C        | Statistiques générales des télécommunications   |
| Série D        | Principes généraux de tarification  |
| Série E        | Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains  |
| Série F        | Services de télécommunication non téléphoniques   |
| Série G        | Systèmes et supports de transmission, systèmes et réseaux numériques  |
| Série H        | Systèmes audiovisuels et multimédias  |
| Série I        | Réseau numérique à intégration de services  |
| Série J        | Transmission des signaux radiophoniques, télévisuels et autres signaux multimédias  |
| Série K        | Protection contre les perturbations   |
| Série L        | Construction, installation et protection des câbles et autres éléments des installations extérieures  |
| Série M        | RGT et maintenance des réseaux: systèmes de transmission, de télégraphie, de télécopie, circuits téléphoniques et circuits loués internationaux |
| Série N        | Maintenance: circuits internationaux de transmission radiophonique et télévisuelle  |
| Série O        | Spécifications des appareils de mesure  |
| Série P        | Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux   |
| Série Q        | Commutation et signalisation  |
| Série R        | Transmission télégraphique  |
| Série S        | Equipements terminaux de télégraphie  |
| Série T        | Terminaux des services télématiques   |
| Série U        | Commutation télégraphique   |
| Série V        | Communications de données sur le réseau téléphonique  |
| <b>Série X</b> | <b>Réseaux pour données et communication entre systèmes ouverts</b>   |
| Série Y        | Infrastructure mondiale de l'information  |
| Série Z        | Langages de programmation   |