



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

X.219

**INTERCONNEXION DES SYSTÈMES OUVERTS
DÉFINITIONS DES SERVICES**

**OPÉRATIONS DISTANTES: MODÈLE,
NOTATION ET DÉFINITION DU SERVICE**

Recommandation UIT-T X.219

(Extrait du *Livre Bleu*)

NOTES

1 La Recommandation X.219 de l'UIT-T a été publiée dans le fascicule VIII.4 du Livre Bleu. Ce fichier est un extrait du Livre Bleu. La présentation peut en être légèrement différente, mais le contenu est identique à celui du Livre Bleu et les conditions en matière de droits d'auteur restent inchangées (voir plus loin).

2 Dans la présente Recommandation, le terme «Administration» désigne indifféremment une administration de télécommunication ou une exploitation reconnue.

© UIT 1988, 1993

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.

Recommandation X.219

OPÉRATIONS DISTANTES: MODÈLE, NOTATION ET DÉFINITION DU SERVICE¹⁾

(Melbourne, 1988)

Le CCITT,

considérant

(a) que la Recommandation X.200 définit le Modèle de Référence de base pour l'interconnexion des systèmes ouverts (OSI) pour les applications du CCITT;

(b) que la Recommandation X.210 définit les conventions de service nécessaires à la description des services du Modèle de Référence OSI;

(c) que la Recommandation X.216 définit le service de couche de présentation;

(d) que la Recommandation X.217 définit le service de contrôle d'association;

(e) que la Recommandation X.218 définit le service de transfert fiable;

(f) que la Recommandation X.229 définit le protocole des opérations distantes;

(g) que des opérations distantes communes doivent prendre en charge plusieurs applications,

recommande à l'unanimité

la définition du service d'opérations distantes et la notation de l'interconnexion de systèmes ouverts pour les applications du CCITT comme spécifié par l'objet et le domaine d'application de la présente Recommandation.

SOMMAIRE

0	<i>Introduction</i>
1	<i>Objet et domaine d'application</i>
2	<i>Références</i>
3	<i>Définitions</i>
4	<i>Abréviations</i>
5	<i>Conventions</i>
6	<i>Modèle d'opérations distantes</i>
7	<i>Vue d'ensemble de la notation et du service</i>
8	<i>Relation avec d'autres ASE et services de couche inférieure</i>
9	<i>Notation des opérations distantes</i>
10	<i>Définition des services</i>
11	<i>Mise en correspondance de la notation et des services</i>
12	<i>Information sur l'enchaînement</i>

Annexe A – Notation pour la spécification des éléments du service d'application et des contextes d'application

Annexe B – Directives aux concepteurs de protocoles d'application sur l'emploi du ROSE

¹⁾ La Recommandation X.219 et la norme ISO 9072-1 [Systèmes de traitement de l'information – Communication de textes – Téléopérations – Partie I: Modèle, notation et définition du service] ont été élaborées en étroite collaboration et harmonisées du point de vue technique.

0 Introduction

La présente Recommandation définit une notation et les services fournis par un élément du service d'application – l'élément du service d'opérations distantes (ROSE) – pour servir de base à des applications interactives dans un environnement de systèmes ouverts répartis. La présente Recommandation fait partie d'une série qui définit les éléments de service d'application couramment utilisés par un certain nombre d'applications.

Les interactions entre entités d'une application répartie sont modélisées comme des opérations distantes et définies à l'aide d'une notation d'opérations distantes. Une opération distante est demandée par une entité; l'autre entité cherche à exécuter l'opération distante et signale le résultat de sa tentative. Les opérations distantes sont prises en charge par le ROSE.

La présente Recommandation est techniquement alignée sur la norme ISO 9072-1.

1 Objet et domaine d'application

La présente Recommandation définit une notation d'opération distante (RO) pour définir les services fournis à des applications interactives. Elle définit encore les services fournis par les services de l'élément de service d'opérations distantes (ROSE). Les services ROSE sont fournis par emploi du protocole ROSE (Recommandation X.229) en liaison avec les services de l'élément de service de contrôle d'association (ACSE) (Recommandation X.217) et avec le protocole ACSE (Recommandation X.227), optionnellement avec les services de l'élément de service de transfert fiable (RTSE) (Recommandation X.218) et avec le protocole RTSE (Recommandation X.228) et avec le service de présentation (Recommandation X.216).

Aucune condition n'est spécifiée en ce qui concerne la conformité à la présente Recommandation.

2 Références

- X.200 Modèle de Référence pour l'interconnexion de systèmes ouverts pour les applications du CCITT (voir également la norme ISO 7498)
- X.208 Spécification de la notation de syntaxe abstraite (voir également la norme ISO 8824)
- X.209 Spécification des règles de codage fondamentales pour la notation de syntaxe abstraite (voir également la norme ISO 8825)
- X.210 Conventions relatives à la définition du service de couche pour l'interconnexion de systèmes ouverts (voir également la norme ISO/TR 8509)
- X.216 Définition du service de présentation pour l'interconnexion de systèmes ouverts pour les applications du CCITT (voir également la norme ISO 8822)
- X.217 Définition du service de contrôle d'association pour les applications du CCITT (voir également la norme ISO 8649)
- X.218 Transfert fiable: Modèle et définition du service (voir également la norme ISO 9066-1)
- X.227 Spécification du protocole de contrôle d'association pour les applications du CCITT (voir également la norme ISO 8650)
- X.228 Spécification du protocole de transfert fiable (voir également la norme ISO 9066-2)
- X.229 Spécification du protocole de téléopération (voir également la norme ISO 9072-2)

3 Définitions

3.1 Définitions relatives au Modèle de Référence

La présente Recommandation est fondée sur les concepts définis dans la Recommandation X.200 et utilise les termes suivants qui y sont définis:

- a) Couche Application,
- b) processus d'application,

- c) entité d'application,
- d) élément du service d'application,
- e) unité de données du protocole d'application,
- f) information de commande du protocole d'application,
- g) Couche Présentation,
- h) service de présentation,
- i) connexion de présentation,
- j) service de session,
- k) connexion de session,
- l) syntaxe de transfert,
- m) élément utilisateur.

3.2 *Définitions relatives aux conventions du service*

La présente Recommandation utilise les termes suivants qui sont définis dans la Recommandation X.210:

- a) fournisseur de service,
- b) utilisateur de service,
- c) service confirmé,
- d) service non confirmé,
- e) service engendré par le fournisseur,
- f) primitive de service, primitive,
- g) demande (primitive),
- h) indication (primitive),
- i) réponse (primitive),
- j) confirmation (primitive).

3.3 *Définitions relatives au service de présentation*

La présente Recommandation utilise les termes suivants qui sont définis dans la Recommandation X.216:

- a) syntaxe abstraite,
- b) nom de syntaxe abstraite,
- c) nom de syntaxe de transfert,
- d) contexte de présentation.

3.4 *Définitions relatives au contrôle d'association*

La présente Recommandation utilise les termes suivants qui sont définis dans la Recommandation X.217:

- a) association d'application, association,
- b) contexte d'application,
- c) élément du service de contrôle d'association.

3.5 *Définitions relatives au transfert fiable*

La présente Recommandation utilise le terme suivant qui est défini dans la Recommandation X.218:

- a) élément du service de transfert fiable.

3.6 *Définitions relatives au ROSE*

Aux fins de la présente Recommandation, il convient d'appliquer les définitions suivantes:

3.6.1 **entité d'application engendrant l'application, demandeur d'association**

Entité d'application qui engendre l'association d'application.

3.6.2 **entité d'application répondant à la demande d'association, répondeur d'association**

Entité d'application qui répond au lancement d'une association d'application par une autre AE.

3.6.3 **entité d'application appelante, demandeur**

Entité d'application qui appelle l'opération distante.

3.6.4 **entité exécutrice de l'application, exécutant**

Entité d'application qui exécute une opération distante appelée par l'autre entité d'application.

3.6.5 **demandeur**

Partie d'une entité d'application qui émet une primitive de demande pour un service ROSE particulier.

3.6.6 **accepteur**

Partie d'une entité d'application qui reçoit la primitive d'indication pour un service ROSE particulier.

3.6.7 **opérations liées**

Ensemble d'opérations formé par une opération mère et une ou plusieurs opérations filles.

3.6.8 **opération mère**

Opération pendant l'exécution de laquelle l'exécutant peut appeler des opérations filles liées devant être exécutées par le demandeur de l'opération mère.

3.6.9 **opération fille**

Opération qui pourrait être appelée par l'exécutant de l'opération mère liée pendant l'exécution de l'opération mère et qui est exécutée par le demandeur de l'opération mère.

3.6.10 **opérations distantes**

- 1) Concept et notation qui servent de base à la spécification d'une communication interactive entre entités d'application. Englobe l'élément du service d'opérations distantes et la mise en correspondance de la notation avec les primitives de service des éléments du service d'application utilisés.
- 2) Ensemble d'opérations liantes, d'opérations déliantes et d'opérations.

3.6.11 **notation RO**

Notation utilisée pour la spécification des opérations distantes, définie dans la présente Recommandation.

3.6.12 **utilisateur d'ASE**

Fonction spécifique de l'application qui met en correspondance l'opération liante et l'opération déliante de la notation RO avec l'ACSE.

3.6.13 **élément du service d'opérations distantes**

Élément du service d'application défini dans la présente Recommandation.

3.6.14 **fournisseur de ROSE**

Fournisseur des services de l'élément de service d'opérations distantes.

3.6.15 **utilisateur de ROSE**

Fonction spécifique de l'application qui met en correspondance les opérations et erreurs de la notation RO avec le ROSE.

3.6.16 utilisateur du RTSE

Fonction spécifique de l'utilisateur qui met en correspondance l'opération liante et l'opération déliante de la notation RO avec le RTSE.

3.6.17 interface d'opération

Dans une entité d'application, interface entre l'élément utilisateur et les éléments du service d'application, définie comme un ensemble de services d'éléments du service d'application (opérations distantes) offerts à l'élément utilisateur en notation RO.

4 Abréviations

AE	entité d'application
ACSE	élément du service de contrôle d'application
ASE	élément du service d'application
APDU	unité de données du protocole d'application
OSI	interconnexion de systèmes ouverts
RO (ou ROS)	opérations distantes
ROSE	élément du service d'opérations distantes
RT (ou RTS)	transfert fiable
RTSE	élément du service de transfert fiable

5 Conventions

La présente Recommandation définit les services correspondant au ROSE conformément aux conventions descriptives définies dans la Recommandation X.210. Le § 10 comporte, dans la définition de chaque service ROSE, un tableau qui énumère les paramètres de ses primitives. Pour une primitive donnée, la présence de chaque paramètre est décrite par l'une des valeurs suivantes:

néant inapplicable

M	obligatoire
U	option pour l'utilisateur
C	conditionnel
O	la présence est une option pour le fournisseur du service ROSE

De plus, la notation (=) indique qu'une valeur de paramètre est sémantiquement égale à la valeur qui figure sur sa gauche dans le tableau.

6 Modèle d'opérations distantes

Dans l'environnement OSI, la communication entre processus d'application est représentée sous la forme d'une communication entre deux entités d'application (AE) utilisant le service de présentation. La communication entre certaines entités d'application est intrinsèquement interactive. Typiquement, une entité demande l'exécution d'une opération particulière; l'autre entité cherche à exécuter l'opération et signale le résultat de sa tentative. Le présent paragraphe présente le concept d'opérations distantes comme un véhicule pour la prise en charge d'applications interactives.

La structure générique d'une opération est une interaction demande/réponse élémentaire. Les opérations sont exécutées dans le contexte d'une association d'application.

La figure 1/X.219 représente un modèle de cette conception.

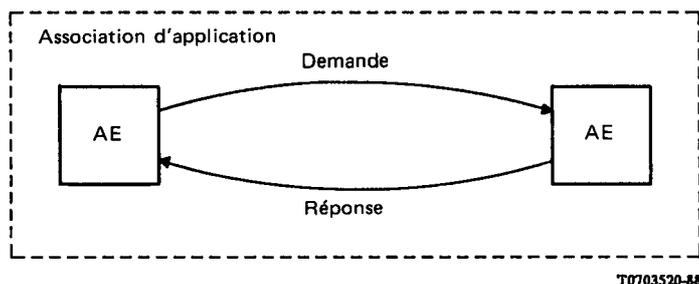


FIGURE 1/X.219

Modèle d'opération distante

Les opérations appelées par une AE (appelante) sont exécutées par l'autre AE (l'exécutant). Ces opérations peuvent être classées selon que l'on attend que leur exécutant en signale le résultat:

- en cas de succès ou d'échec (un résultat positif est retourné en réponse si l'opération a réussi, un résultat négatif si l'opération échoue);
- en cas d'échec seulement (aucune réponse n'est donnée si l'opération réussit, un résultat négatif est donné si l'opération échoue);
- en cas de réussite seulement (un résultat positif est donné si l'opération réussit. Aucune réponse ne l'est si l'opération échoue);
- pas de réponse (il n'est retourné ni résultat positif ni résultat négatif, que l'opération réussisse ou échoue).

Les opérations peuvent aussi être classées selon les deux modes de fonctionnement possibles: mode synchrone, dans lequel l'appelant doit obtenir une réponse de l'exécutant avant d'appeler une autre opération; et en mode asynchrone dans lequel le demandeur peut continuer à appeler de nouvelles opérations sans attendre une réponse.

Les classes d'opérations suivantes sont définies:

- Classe d'opérations 1: synchrone, signale la réussite ou l'échec (résultat positif ou négatif)
- Classe d'opérations 2: asynchrone, signale la réussite ou l'échec (résultat positif ou négatif)
- Classe d'opérations 3: asynchrone, signale seulement un échec (résultat négatif) éventuel
- Classe d'opérations 4: asynchrone, signale seulement une réussite (résultat positif)
- Classe d'opérations 5: asynchrone, résultat non signalé.

La classe d'opérations de chaque opération doit être convenue entre les entités d'application (par exemple, dans une Recommandation relative au protocole d'application).

Dans certains cas, il est utile de grouper des opérations en une série d'opérations liées, série qui est constituée d'une opération mère et d'une ou de plusieurs opérations filles. L'exécutant de l'opération mère peut appeler zéro, une ou plusieurs opérations filles pendant l'exécution de l'opération mère. Celui qui a appelé l'opération mère est l'exécutant des opérations filles. Une opération fille peut être l'opération mère d'une autre série d'opérations liées de manière récursive. La figure 2/X.219 représente un modèle de ce concept.

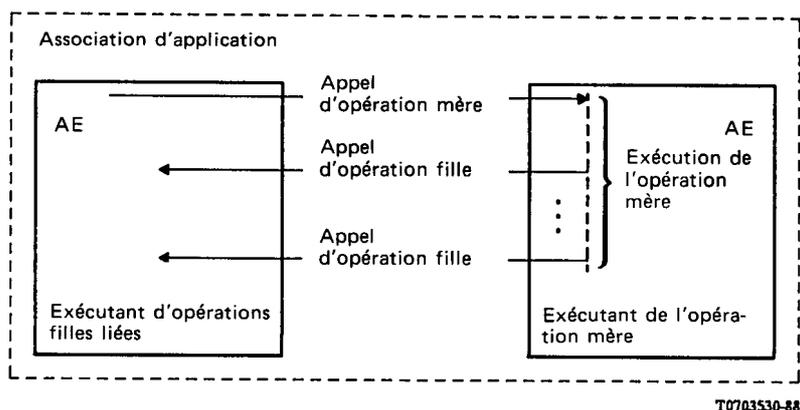


FIGURE 2/X.219
Opérations liées

Une association d'application définit la relation entre deux AE; elle est formée par l'échange d'informations de commande de protocole d'application, par utilisation de services de présentation. L'AE qui engendre une association d'application est appelée AE initiatrice d'association ou demandeur d'association, tandis que l'AE qui répond à la demande d'initiation d'une association d'application par une autre AE est dite AE qui répond à la demande d'association ou répondeur d'association. Seule l'AE initiatrice d'association peut procéder à la terminaison d'une association d'application établie.

Les associations d'application sont classées en fonction de l'entité d'application qui est autorisée à appeler des opérations:

- Classe d'associations 1: seule l'entité d'application qui engendre l'association peut appeler des opérations.
- Classe d'associations 2: seule l'entité d'application qui répond à la demande d'association peut appeler des opérations.
- Classe d'associations 3: l'entité d'application qui engendre l'association comme celle qui y répond peut appeler des opérations.

Les opérations liées impliquent la classe d'association 3.

La classe d'association doit être convenue entre les entités d'application (par exemple, dans une Recommandation relative au protocole d'application).

La fonctionnalité d'une AE est divisée en un élément utilisateur et un ensemble d'éléments du service d'application (ASE). Chaque ASE peut lui-même être divisé en un ensemble d'ASE (plus primitives). L'interaction entre AE est décrite par la forme selon laquelle elles utilisent les ASE.

La combinaison spécifique d'un élément utilisateur et de l'ensemble d'ASE qui forment une AE définit le contexte d'application.

La figure 3/X.219 illustre un exemple d'un contexte d'application qui fait intervenir l'élément de service d'opérations distantes (ROSE). A noter que cette figure n'entend pas montrer que l'application est symétrique. Les applications interactives sont souvent intrinsèquement asymétriques, c'est-à-dire qu'une seule ou les deux AE peuvent être habilitées à appeler des opérations et que les opérations que ces AE peuvent appeler peuvent être différentes. Les règles servant à spécifier l'AE qui peut appeler des opérations et le type d'opérations que peut appeler une AE sont définis au moyen de la notation RO dans une Recommandation relative au protocole d'application, et déterminent le contexte d'application.

L'ensemble d'ASE disponible pour l'élément utilisateur de l'AE à l'interface d'opération est défini au moyen de la notation d'opérations distantes (RO). Cette notation se fonde sur le concept de macro défini dans la Recommandation X.208. La complexité d'un ensemble particulier d'ASE dépend des besoins de l'application et n'est pas limitée par le concept d'opérations distantes.

Une importante caractéristique des opérations distantes est qu'elles fournissent des applications indépendantes des services de communication OSI. Etant donné que la notation est fondée sur des principes de programmation établis axés sur l'objet, il est possible de concevoir des outils automatiques pour lier les opérations distantes dans l'environnement d'exécution des applications.

Les ASE disponibles pour l'élément utilisateur nécessitent des communications sur une association d'application. Le contrôle de cette association d'application (établissement, terminaison, coupure) est exécuté soit par l'élément de service d'association (ACSE) défini dans la Recommandation X.217, soit par l'élément de service de transfert fiable défini dans la Recommandation X.218 et par l'élément du service de contrôle d'association (ACSE). La communication sur l'association d'application est exécutée par l'élément de service d'opérations distantes (ROSE) défini dans la présente Recommandation.

Une fonction spécifique de l'application met en correspondance les opérations disponibles pour l'élément utilisateur soit avec les services ACSE, soit avec les services RTSE et les services ROSE. La correspondance est définie dans la présente Recommandation. La fonction qui met en correspondance les opérations avec les services ACSE, ou les services RTSE, et les services ROSE, est dite utilisateur de l'ACSE, du RTSE et du ROSE ou utilisateur de l'ACSE, utilisateur du RTSE et utilisateur du ROSE.

Si le RTSE est inclus dans le contexte d'application, la fonction de mise en correspondance est un utilisateur du RTSE et un utilisateur du ROSE, le ROSE est utilisateur du RTSE, le RTSE est un utilisateur de l'ACSE et un utilisateur du service de présentation, et l'ACSE est un utilisateur du service de présentation.

Si le RTSE est exclu du contexte d'application, la fonction de mise en correspondance est un utilisateur de l'ACSE et un utilisateur du ROSE, le ROSE est un utilisateur du service de présentation et l'ACSE est un utilisateur du service de présentation.

7 Vue d'ensemble de la notation et du service

7.1 Vue d'ensemble de la notation

La présente Recommandation définit la notation RO pour la spécification d'un contexte d'application et de la composante de syntaxe abstraite correspondante du contexte de présentation.

La fonctionnalité d'un contexte d'application est fournie à l'élément utilisateur au moyen d'opérations distantes et d'erreurs qui constituent l'interface d'opération.

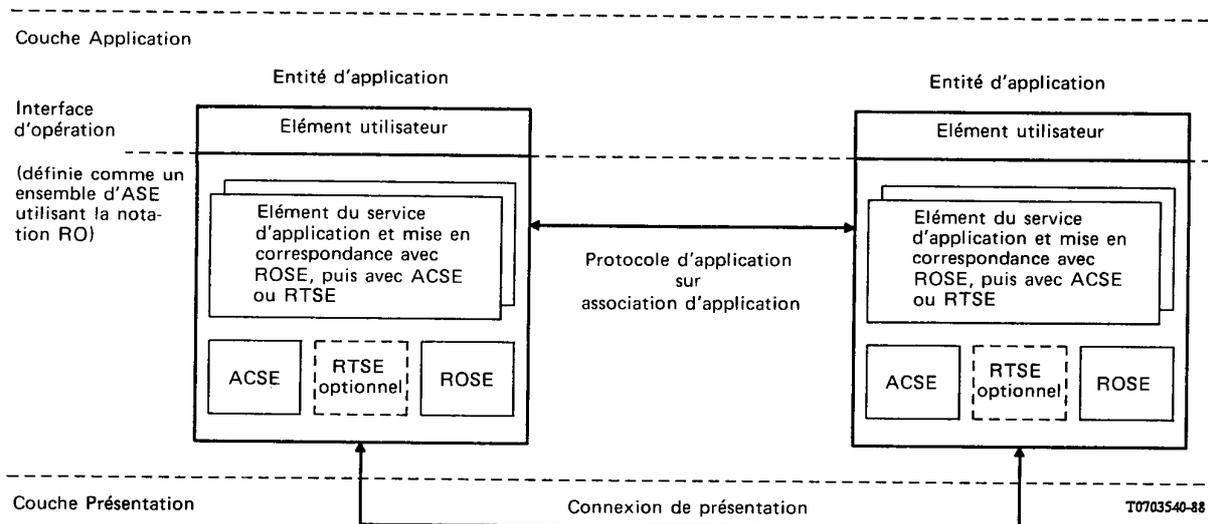


FIGURE 3/X.219

Modèle d'un contexte d'application avec opérations distantes

Les types suivants d'opérations distantes constituent une interface d'opération:

- une opération liante pour établir une association d'application,
- un ensemble d'opérations et, pour chaque opération, une liste de situations d'erreur (réponse négative),
- une opération déliante pour terminer une association d'application.

La notation de syntaxe abstraite de la Recommandation X.208 est utilisée pour la définition des macros suivantes:

- a) BIND (liante);
- b) UNBIND (déliante);
- c) OPERATION (opération); et
- d) ERROR (erreur).

Ces macros fournissent à la fois une notation de type et une notation de valeur pour opérations distantes et erreurs.

La notation de type de la macro BIND permet d'échanger la spécification d'un type d'opération liante et les types d'éventuelles valeurs de données d'utilisateur au cours de la phase d'établissement d'une association d'application. La notation de valeur de la macro BIND permet d'échanger la spécification de valeurs de données d'utilisateur éventuelles au cours de la phase d'établissement d'une association d'application.

La notation de type de la macro UNBIND permet d'échanger la spécification d'un type d'opération déliante et de types de valeurs éventuelles de données d'utilisateur au cours de la phase de terminaison d'une application d'association. La notation de valeur de la macro UNBIND permet d'échanger la spécification de valeurs éventuelles de données d'utilisateur au cours de la phase de terminaison d'association d'application.

La notation de type de la macro OPERATION permet d'échanger la spécification d'une opération et des types de données d'utilisateur pour une demande et une réponse positive. De plus, la notation de type permet de spécifier une liste de situations de réponses négatives valides. S'il s'agit d'une opération mère, la notation de type permet de spécifier une liste d'opérations filles liées. La notation de valeur de la macro OPERATION permet la spécification de l'identificateur d'une opération.

La notation de type de la macro ERROR permet d'échanger la spécification des types de données d'utilisateur dans une situation de réponse négative. La notation de valeur de la macro ERROR permet de spécifier l'identificateur d'une erreur.

Des macros additionnelles servant de base à la notation pour la spécification d'éléments de service d'application et de contextes d'application sont définies dans l'annexe A.

7.2 *Vue d'ensemble du service*

La présente Recommandation définit les services ROSE suivants:

- a) APPEL DE RO
- b) RÉSULTAT DE RO
- c) ERREUR DE RO
- d) REFUS DE RO PAR L'UTILISATEUR
- e) REFUS DE RO PAR LE FOURNISSEUR

Le service APPEL DE RO permet à une AE demandeur de demander l'exécution d'une opération par l'AE exécutant.

Le service RÉSULTAT DE RO permet à l'AE exécutant de renvoyer la réponse positive d'une opération réussie à l'AE demandeur.

Le service ERREUR DE RO permet à l'AE exécutant de renvoyer la réponse négative d'une opération ayant échoué à l'AE demandeur.

Le service REFUS DE RO PAR L'UTILISATEUR permet à une AE de rejeter la demande ou la réponse de l'autre AE si l'utilisateur du ROSE a détecté une difficulté.

Le service REFUS DE RO PAR LE FOURNISSEUR permet à l'utilisateur du ROSE d'être informé de la détection d'une difficulté par le fournisseur du ROSE.

7.3 *Mise en correspondance de la notation et des services*

A noter que la fonction qui met en correspondance les macros OPERATION et les macros ERROR de la notation RO avec les services ROSE est appelée l'utilisateur du ROSE, alors que la fonction qui met en correspondance les macros BIND et UNBIND de la notation RO avec les services ACSE ou RTSE respectivement est appelée respectivement l'utilisateur de l'ACSE ou l'utilisateur du RTSE.

La spécification de la mise en correspondance de la notation RO avec les services utilisés d'ACSE, RTSE et ROSE est précisée au § 11. En conséquence, les Recommandations qui utilisent la notation RO pour la spécification du protocole ne doivent pas nécessairement spécifier la mise en correspondance avec ces services utilisés.

8 Relation avec d'autres ASE et services de couche inférieure

8.1 *Autres éléments du service d'application*

Le ROSE est conçu pour être utilisé avec d'autres ASE afin de prendre en charge des tâches interactives spécifiques de traitement de l'information. En conséquence, il est prévu que le ROSE soit inclus dans un grand nombre de spécifications de contextes d'application.

L'ensemble constitué par le ROSE et les autres ASE inclus dans un contexte d'application doit utiliser les facilités du service de présentation de manière coordonnée.

Le ROSE nécessite une association d'application existante contrôlée par l'ACSE.

Un élément du service de transfert fiable (RTSE) est inclus pour certaines spécifications de contextes d'application.

Une spécification de protocole d'utilisateur de ROSE utilise la notation RO. Elle définit une ou plusieurs syntaxes abstraites et fournit des noms de syntaxes abstraites uniques d'identificateur d'objet type pour chaque syntaxe abstraite.

Si une syntaxe abstraite nommée spécifie des opérations et des erreurs, les APDU de ROSE définies dans la Recommandation X.229 sont incluses dans cette syntaxe abstraite nommée. Si des syntaxes abstraites nommées multiples sont définies pour des opérations et des erreurs, les APDU de ROSE sont incluses dans chaque syntaxe abstraite nommée.

Si une syntaxe abstraite nommée spécifie une opération liante, les APDU spécifiées par la notation de valeur de la macro BIND sont incluses dans cette syntaxe abstraite nommée. Si le RTSE est inclus dans le contexte d'application, les APDU pour l'opération liante partagent une unique syntaxe abstraite nommée avec les APDU de RTSE définies dans la Recommandation X.228.

Si une syntaxe abstraite nommée spécifie une opération déliante, les APDU spécifiées par la notation de valeur de la macro UNBIND sont incluses dans cette syntaxe abstraite nommée.

Les APDU résultant de la spécification d'une opération liante, d'une opération déliante, d'opérations et d'erreurs, et les APDU de RTSE peuvent partager une unique syntaxe abstraite nommée.

8.2 *Service de présentation*

Si un contexte d'application incluant le RTSE et le ROSE est défini, les services ROSE n'utilisent pas le service de présentation.

Si un contexte d'application incluant le ROSE mais excluant le RTSE est défini, les services ROSE doivent avoir accès au service de TRANSFERT DE DONNÉES DE PRÉSENTATION et doivent pouvoir utiliser l'unité fonctionnelle duplex du service de présentation. Les services ROSE n'utilisent ni n'imposent l'utilisation d'aucun autre service de présentation.

Une syntaxe abstraite nommée associée à une syntaxe de transfert compatible (négociée par la Couche Présentation) constitue un contexte de présentation.

Une valeur d'identificateur d'objet {joint-iso-ccitt asn1(1) codage-de-base (1)} spécifiée dans la Recommandation X.209 peut être utilisée comme nom de syntaxe de transfert. Dans ce cas, la spécification du protocole utilisateur de ROSE ne doit pas nécessairement nommer ni spécifier une syntaxe de transfert.

9 Notation des opérations distantes

9.1 Considérations générales

La notation utilisée dans la présente Recommandation est définie comme suit:

- la notation de syntaxe de données et la notation macro sont définies dans la Recommandation X.208,
- les macros d'opération distante sont définies au § 9.2 de la présente Recommandation.

Une opération liante définit le point où commence un objet liant (établissement d'une association d'application). Si une telle liaison est établie, des opérations peuvent être appelées. Une opération déliante définit le point où se termine une liaison d'objet.

Un protocole interactif est spécifié au moyen des types de données Opération Distante et Erreur. La présente section définit ces types. Elle explique également la définition notationnelle d'une opération distante particulière et les erreurs particulières qu'elle peut signaler. La notation est définie au moyen de la facilité macro définie dans la Recommandation X.208. Cette définition de macro permet une spécification généralisée de la mise en correspondance avec divers environnements d'exécution.

Les macros qui permettent la spécification des opérations liantes, des opérations déliantes, des opérations et des erreurs sont indiquées dans la figure 4/X.219.

9.2 Spécification d'opérations liantes

Une valeur de données unique, l'argument de l'opération liante, peut accompagner la demande d'établissement de l'association d'application. Certaines opérations liantes signalent leur résultat, qu'il s'agisse d'une réussite (c'est-à-dire le résultat normal) ou d'un échec (c'est-à-dire le résultat exceptionnel). D'autres opérations liantes ne signalent leur résultat qu'en cas d'échec, d'autres encore ne le signalent jamais. Une valeur de données unique, le résultat de l'opération liante, peut accompagner la réponse positive. Une valeur de données unique, l'erreur liante de l'opération liante, peut accompagner la réponse négative.

La notation pour un type d'opération liante est le mot clé BIND, facultativement suivi du mot clé ARGUMENT et du type d'argument de l'opération liante, du nom de référence qui lui est facultativement attribué et de la nature de la signalisation du résultat de l'opération (éventuel). Si le résultat signalé de l'opération liante est la réussite, le mot clé RESULT et le type de ce résultat ainsi que le nom de référence qui lui est facultativement assigné sont spécifiés. Si le résultat signalé de l'opération est l'échec, le mot clé BIND-ERROR et le type de l'information d'erreur qu'il signale ainsi que le nom de référence qui lui est facultativement assigné sont spécifiés.

La notation de valeur pour une opération liante est soit une valeur d'argument, soit une valeur de résultat, soit une valeur d'erreur. La notation de valeur pour une valeur (éventuelle) d'argument est le mot clé ARGUMENT suivi par une valeur du type d'argument. La notation de valeur pour une valeur (éventuelle) de résultat est le mot clé RESULT suivi d'une valeur du type de résultat. La notation de valeur pour une valeur (éventuelle) d'erreur est le mot clé ERROR suivi d'une valeur du type d'erreur.

```

Notation-d'Opération-Distante {joint-iso-ccitt opérations-distantes(4) notation(0)}
DEFINITIONS ::=
BEGIN

EXPORTS BIND, UNBIND, OPERATION, ERROR;

-- définition de macro pour opérations liantes

BIND MACRO ::=
BEGIN

TYPE NOTATION ::= Argument Résultat Erreur
VALUE NOTATION ::= valeur-d'Argument | valeur-de-Résultat | valeur-d'Erreur

Argument ::= vide | "ARGUMENT" Type de nom (type-d'Argument)
-- Attend un type quelconque d'ASN.1 et l'attribue au type d'Argument
variable

Résultat ::= vide | "RESULT" Type de nom (type-de-Résultat)
-- Attend un type quelconque d'ASN.1 et l'attribue au type de Résultat variable

Erreur ::= vide | "BIND-ERROR" Type de nom (type-d'Erreur)
-- Attend un type quelconque d'ASN.1 et l'attribue au type d'Erreur variable

Nom ::= vide | identificateur

Valeur-d'argument ::= vide | valeur "ARGUMENT" (valeur-Arg type-d'Argument)
-- Attend une valeur pour le type dans le type-d'Argument et l'attribue à la
-- valeur-Arg variable
<VALUE [16] EXPLICIT type-d'Argument ::= valeur-Arg>
-- Renvoie la valeur finale comme type explicitement marqué

Valeur-de-résultat ::= vide | valeur "RESULT" (valeur-Res type-de-Résultat)
-- Attend une valeur pour le type dans le type-de-Résultat et l'attribue à la
-- valeur-Res variable
<VALUE [17] EXPLICIT type-de-Résultat ::= valeur-Res>
-- Renvoie la valeur finale comme type explicitement marqué

Valeur-d'erreur ::= vide | valeur "ERROR" (valeur-Err type-d'Erreur)
-- Attend une valeur pour le type dans le type-d'Erreur et l'attribue à la
-- valeur-Err variable
<VALUE [18] EXPLICIT type-d'Erreur ::= valeur-Err>
-- Renvoie la valeur finale comme type explicitement marqué

END
-- Notation d'Opérations Distantes, à suivre

```

FIGURE 4/X.219 (feuillet 1 sur 3)

Définition formelle des types de données d'Opérations Distantes

- Notation d'Opérations Distantes, suite
- Définition de macro pour opérations-déliantes

UNBIND MACRO ::=

BEGIN

TYPE NOTATION ::= Argument Résultat Erreurs

VALUE NOTATION ::= valeur-d'Argument | valeur-de-Résultat | valeur-d'Erreur

Argument ::= vide | "ARGUMENT" Type de nom (type-d'Argument)
 -- Attend un type quelconque d'ASN.1 et l'attribue au type d'Argument variable

Résultat ::= vide | "RESULT" Type de nom (type-de-Résultat)
 -- Attend un type quelconque d'ASN.1 et l'attribue au type de Résultat variable

Erreur ::= vide | "UNBIND-ERROR" Type de nom (type-d'Erreur)
 -- Attend un type quelconque d'ASN.1 et l'attribue au type d'Erreur variable

Nom ::= vide | identificateur

Valeur-d'argument ::= vide | valeur "ARGUMENT" (valeur-Arg type-d'Argument)
 -- Attend une valeur pour le type dans le type-d'Argument et l'attribue à la
 -- valeur-Arg variable
<VALUE [19] EXPLICIT type-d'Argument ::= valeur-Arg >
 -- Renvoie la valeur finale comme type explicitement marqué

Valeur-de-résultat ::= vide | valeur "RESULT" (valeur-Res type-de-Résultat)
 -- Attend une valeur pour le type dans le type-de-Résultat et l'attribue à la
 -- valeur-Res variable
<VALUE [20] EXPLICIT type-de-Résultat ::= valeur-Res >
 -- Renvoie la valeur finale comme type explicitement marqué

Valeur-d'erreur ::= vide | valeur "ERROR" (valeur-Err type-d'Erreur)
 -- Attend une valeur pour le type dans le type-d'Erreur et l'attribue à la
 -- valeur-Err variable
<VALUE [21] EXPLICIT type-d'Erreur ::= valeur-Err >
 -- Renvoie la valeur finale comme type explicitement marqué

END

- Notation d'Opérations Distantes, à suivre

FIGURE 4/X.219 (feuillet 2 sur 3)

Définition formelle des types de données d'Opérations Distantes

```

-- Notation d'Opérations Distantes, suite
-- Définition de macro pour opérations

OPERATION MACRO ::=

BEGIN

TYPE NOTATION ::= ArgumentRésultatErreursOpérationsLiées
VALUE NOTATION ::= valeur (VALUE CHOICE {
                        Valeurlocale INTEGER,
                        Valeurglobale OBJECT IDENTIFIER})

Argument ::= "ARGUMENT" TypeNommé | vide
Résultat ::= "RESULT" TypeRésultat | vide
TypeRésultat ::= TypeNommé | vide
Erreurs ::= "ERRORS" "{NomsErreur}" | vide
OpérationsLiées ::= "LINKED" "{NomsOpérationsLiées}" | vide
NomsErreur ::= ListeErreurs | vide
ListeErreurs ::= Erreur | ListeErreurs "," Erreur
Erreur ::= valeur (ERROR) -- dénote une valeur d'erreur
                | type -- dénote un type d'erreur si aucune valeur d'erreur n'est spécifiée
NomsOpérationsLiées ::= ListeOpérations | vide
ListeOpérations ::= Opération | ListeOpérations "," Opération
Opération ::= valeur (OPERATION) -- dénote une valeur d'opération
                | type -- dénote un type d'opération si aucune valeur d'opération n'est spécifiée
TypeNommé ::= type d'identificateur | type

END
-- définition de macro pour erreurs d'opérations

ERROR MACRO ::=

BEGIN

TYPE NOTATION ::= Paramètre
VALUE NOTATION ::= valeur (VALUE CHOICE {
                        Valeurlocale INTEGER,
                        Valeurglobale OBJECT IDENTIFIER})

Paramètre ::= "PARAMETER" TypeNommé | vide
TypeNommé ::= type d'identificateur | type

END

END -- Fin de Notation d'Opérations Distantes

```

FIGURE 4/X.219 (feuillet 3 sur 3)
Définition formelle des types de données d'Opérations Distantes

9.3 Spécification d'opérations déliantes

Une valeur de données unique, l'argument de l'opération déliante, peut accompagner la demande de terminaison de l'association d'application. Certaines opérations déliantes signalent leur résultat, qu'il s'agisse de réussite (résultat normal) ou d'échec (résultat exceptionnel). D'autres opérations déliantes ne signalent leur résultat qu'en cas d'échec et d'autres ne le signalent jamais. Une valeur de données unique, le résultat de l'opération déliante, ou une valeur de données unique, l'erreur déliante de l'opération déliante, peuvent accompagner la réponse.

La notation pour un type d'opération déliante est le mot clé UNBIND, facultativement suivi du mot clé ARGUMENT et du type d'argument de l'opération déliante, du nom de référence qui lui est facultativement attribué et de la nature de la signalisation éventuelle du résultat de l'opération déliante. Si l'opération déliante signale la réussite, le mot clé RESULT et le type de son résultat ainsi que le nom de référence qui lui est facultativement assigné sont spécifiés. Si l'opération déliante signale un échec, le mot clé UNBIND-ERROR et le type de l'information d'erreur qu'elle signale ainsi que le nom de référence qui lui est facultativement assigné sont spécifiés.

La notation de valeur pour une opération déliante est soit une valeur d'argument, soit une valeur de résultat, soit une valeur d'erreur. La notation de valeur pour une valeur (éventuelle) d'argument est le mot clé ARGUMENT suivi d'une valeur du type d'argument. La notation de valeur pour une valeur (éventuelle) de résultat est le mot clé RESULT suivi d'une valeur du type de résultat. La notation de valeur pour une valeur (éventuelle) d'erreur est le mot clé ERROR suivi d'une valeur du type d'erreur.

9.4 Spécification des opérations

Une valeur de données d'opération type représente l'identificateur pour une opération dont un utilisateur de ROSE dans un système ouvert peut demander l'exécution par un utilisateur homologue de ROSE dans un autre système ouvert. Une valeur de données unique, l'argument de l'opération, peut accompagner la demande. Certaines opérations signalent leur résultat, qu'il s'agisse de réussite (résultat normal) ou d'échec (résultat exceptionnel). D'autres opérations ne signalent leur résultat qu'en cas d'échec, certaines ne signalant aucun résultat. Une valeur de données unique, le résultat de l'opération, accompagne un rapport de réussite: un rapport d'échec identifie la condition exceptionnelle qui a été rencontrée.

La notation correspondant à un type d'opération est le mot clé OPERATION, facultativement suivi du mot clé ARGUMENT et du type de l'argument de l'opération, du nom de référence qui lui est facultativement assigné et de la nature de la signalisation (éventuelle) du résultat de l'opération. Si l'opération signale une réussite, le mot clé RESULT et, facultativement, le type de son résultat et le nom de référence qui lui est facultativement assigné sont spécifiés. Si l'opération signale un échec, le mot clé ERROR et les noms de référence des valeurs d'erreur ou des types d'erreur signalés sont spécifiés. Si l'opération est l'opération mère d'un ensemble d'opérations liées, le mot clé LINKED-OPERATIONS et les noms de référence des valeurs d'opérations filles liées ou les types d'opérations filles sont spécifiés. La référence aux valeurs d'erreur ou aux valeurs d'opérations filles est préférée, mais les références à des types seront utilisées et ces valeurs sont définies autre part (voir le § 9.6).

La notation correspondant à une valeur d'opération est l'identificateur de l'opération. Si un identificateur localement unique (valeur locale) suffit, l'identificateur est du type INTEGER (entier). Si un identificateur globalement unique (valeur globale) est nécessaire pour permettre une identification univoque des opérations utilisées dans plusieurs syntaxes abstraites, l'identificateur est du type OBJECT IDENTIFIER (identificateur d'objet).

Les opérations filles et erreurs auxquelles se réfère une opération spécifique doivent partager une syntaxe abstraite à appellation unique (voir le § 8.1) avec cette opération, si les opérations filles ou les erreurs sont identifiées par des valeurs locales. L'emploi de valeurs globales n'est pas sujet à limitation.

9.5 Spécification d'erreurs

Une valeur de données d'erreur type représente l'identificateur pour une condition d'anomalie pouvant être signalée par un utilisateur de ROSE dans un système ouvert à un autre utilisateur de ROSE dans un autre système ouvert, lorsque la condition d'anomalie signale un résultat exceptionnel d'une opération demandée préalablement. Une valeur de données unique, le paramètre de l'erreur, peut accompagner le rapport de signalisation.

La notation correspondant à un type d'erreur est le mot clé ERROR, facultativement suivi du mot clé PARAMETER, du type du paramètre d'erreur et du nom de référence qui lui est facultativement assigné.

La notation correspondant à une valeur d'erreur est l'identificateur de l'erreur. Si un identificateur localement unique (valeur locale) suffit, l'identificateur est du type INTEGER (entier). Si un identificateur globalement unique (valeur globale) est nécessaire pour permettre une identification univoque des erreurs utilisées dans plusieurs syntaxes abstraites, l'identificateur est du type OBJECT-IDENTIFIER.

9.6 Exportation et importation d'opérations et d'erreurs

Les valeurs d'opération et les valeurs d'erreur doivent être uniques à l'intérieur d'une syntaxe abstraite désignée. Si des opérations et des erreurs sont spécifiées dans plusieurs modules ASN.1 et sont importées dans un module spécifiant une syntaxe abstraite nommée spécifique, l'une des règles suivantes devra être appliquée:

- 1) Si des valeurs locales sont utilisées et exportées, il appartient au concepteur du module importateur d'assurer l'unicité.
- 2) Un module peut spécifier et exporter des types d'opération et des types d'erreur. Les valeurs d'opération et les valeurs d'erreur sont attribuées dans le module qui importe les types. Une valeur unique doit être attribuée à chaque type d'opération ou type d'erreur.
- 3) Si des valeurs globales sont attribuées et exportées, leur unicité est assurée.

Cependant, des syntaxes abstraites différentes pourraient être utilisées pour des valeurs locales contradictoires.

10 Définition des services

Les services ROSE sont indiqués dans le tableau 1/X.219.

TABLEAU 1/X.219

Services ROSE

Service	Type
APPEL-RO	Non confirmé
RÉSULTAT-RO	Non confirmé
ERREUR-RO	Non confirmé
REFUS-RO PAR L'UTILISATEUR	Non confirmé
REFUS-RO PAR LE FOURNISSEUR	Engendré par le fournisseur

L'identification de la syntaxe abstraite nommée utilisée est admise par hypothèse pour tous les services ROSE. Il s'agit cependant d'une question d'initiative locale qui ne relève pas de la présente Recommandation.

10.1 Service APPEL-RO

Ce service est utilisé par un utilisateur de ROSE (l'appelant) pour provoquer l'appel d'une OPERATION devant être exécutée par un autre utilisateur de ROSE (l'exécutant). Il s'agit d'un service non confirmé.

La structure de service correspondante se compose de deux primitives de service, comme l'indique la figure 5/X.219.

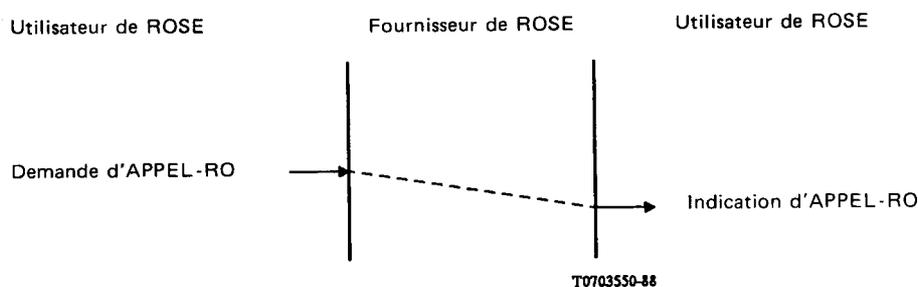


FIGURE 5/X.219

Primitives du service APPEL-RO

10.1.1 Paramètres du service APPEL-RO

Le tableau 2/X.219 énumère les paramètres du service APPEL-RO.

TABLEAU 2/X.219

Paramètres APPEL-RO

Nom de paramètre	Demande	Indication
Valeur d'opération	M	M (=)
Classe d'opération	U	
Argument	U	C (=)
ID d'appel	M	M (=)
ID-lié	U	C (=)
Priorité	U	

10.1.1.1 Valeur d'opération

Ce paramètre est l'identificateur de l'opération à appeler. Sa valeur doit être convenue entre les utilisateurs de ROSE. Ce paramètre doit être fourni par le demandeur du service.

10.1.1.2 Classe d'opération

Ce paramètre précise si l'on attend une réponse synchrone ou asynchrone et la nature de la réponse attendue, c'est-à-dire le résultat et/ou l'erreur – ou aucune réponse (voir § 6). Devant être fourni par le demandeur du service, il n'est utilisé que pour optimiser la gestion de tour (voir § 8.1.1, Recommandation X.229).

10.1.1.3 Argument

Ce paramètre est l'argument de l'opération appelée. Son type doit être convenue entre les utilisateurs de ROSE. Il doit être fourni par le demandeur du service.

10.1.1.4 ID-d'appel

Ce paramètre identifie la demande d'un service APPEL-RO et il sert à mettre en corrélation cette demande avec les réponses correspondantes (services RÉSULTAT-RO, ERREUR-RO, REFUS-RO PAR L'UTILISATEUR et REFUS-RO PAR LE FOURNISSEUR) ou avec l'appel d'opérations filles liées (APPEL-RO). Il doit être fourni par le demandeur du service.

Ce paramètre distingue plusieurs demandes du service que le demandeur peut avoir en cours (opérations asynchrones). Le demandeur peut commencer à réutiliser des valeurs d'ID-d'appel à tel instant qu'il choisit, sous réserve qu'il ne peut réutiliser une valeur d'ID-d'appel qui aurait été préalablement assignée à une demande du service pour lequel il attend – sans l'avoir encore reçu(e) – une réponse ou l'appel d'une opération fille liée.

L'utilisateur de ROSE auquel une indication d'APPEL-RO est donnée, admet qu'une valeur d'ID-d'appel violant la règle ci-dessus est une répétition; de ce fait, il n'exécute pas l'opération appelée et rejette l'appel répété.

Si les classes d'opération 3, 4 ou 5 sont utilisées, le demandeur du service peut réutiliser une valeur d'ID-d'appel après un délai raisonnable, ou si la réponse est acheminée par d'autres moyens (par exemple, le résultat d'une opération avez-vous terminé?).

Dans certains contextes d'application, des utilisateurs homologues de ROSE peuvent communiquer des valeurs ID-d'appel. Pour permettre cette situation, le type du paramètre ID-d'appel est exporté par le module qui définit la syntaxe abstraite des opérations distantes (§ 9 de la Recommandation X.229).

10.1.1.5 ID-lié

Si ce paramètre est présent, l'opération appelée est une opération fille et le paramètre identifie l'appel de l'opération mère liée. Il doit être fourni par le demandeur du service et sa valeur est celle du paramètre ID-d'appel de la primitive d'indication d'APPEL-RO de l'opération mère.

10.1.1.6 Priorité

Ce paramètre définit la priorité assignée au transfert de l'APDU correspondante par rapport aux autres APDU qui doivent être échangées entre AE. Plus faible est sa valeur, plus forte est la priorité. Si plusieurs APDU ayant la même priorité attendent leur transfert, elles sont transférées selon leur ordre d'arrivée.

Remarque 1 – Le paramètre de priorité produit un effet dans le cas d'une association bidirectionnelle à l'alternat, car il donne la priorité à l'émission d'APDU et il peut être utilisé pour déterminer l'instant auquel il faut demander le Tour pour émettre les APDU. Le paramètre de priorité peut aussi avoir un effet local dans le cas d'une association bidirectionnelle simultanée.

Remarque 2 – La priorité d'une réponse (RÉSULTAT-RO, ERREUR-RO et REFUS-RO PAR L'UTILISATEUR) devrait normalement être plus forte (valeur plus faible) que la priorité de l'appel qui lui correspond.

10.2 Service RÉSULTAT-RO

Le service RÉSULTAT-RO sert à un utilisateur de ROSE pour répondre à une indication préalable d'APPEL-RO lorsqu'une opération a été exécutée avec succès. Il s'agit d'un service non confirmé.

La structure de service correspondante se compose de deux primitives de service, comme le montre la figure 6/X.219.

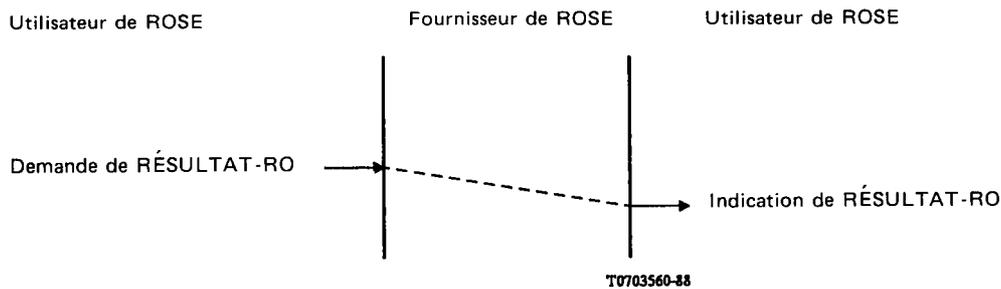


FIGURE 6/X.219

Primitives du service RÉSULTAT-RO

10.2.1 Paramètres RÉSULTAT-RO

Le tableau 3/X.219 indique les paramètres du service RÉSULTAT-RO

TABLEAU 3/X.219

Paramètres du service RÉSULTAT-RO

Nom de paramètre	Demande	Indication
Valeur d'opération	U	C (=)
Résultat	U	C (=)
ID d'appel	M	M (=)
Priorité	U	

10.2.1.1 Valeur d'opération

Ce paramètre est l'identificateur d'une opération appelée et exécutée avec succès. Il doit être fourni par le demandeur du service. Sa valeur est celle de la primitive d'indication d'APPEL-RO correspondante. Il ne doit être présent que si le paramètre Résultat est présent.

10.2.1.2 Résultat

Ce paramètre est le résultat d'une opération appelée et exécutée avec succès. Son type doit être convenu entre les utilisateurs de ROSE; il doit être fourni par le demandeur du service.

10.2.1.3 ID-d'appel

Ce paramètre identifie l'appel correspondant (voir le § 10.1.1.4). Il doit être fourni par le demandeur du service. Sa valeur est celle de la primitive d'indication d'APPEL-RO correspondante.

10.2.1.4 Priorité

Ce paramètre définit la priorité attribuée au transfert de l'APDU correspondante (voir le § 10.1.1.6).

10.3 Service ERREUR-RO

Le service ERREUR-RO sert à un utilisateur de ROSE pour répondre à une indication préalable d'APPEL-RO si l'exécution d'une opération a échoué. Il s'agit d'un service non confirmé.

La structure de service correspondante se compose de deux primitives de service, comme le montre la figure 7/X.219.

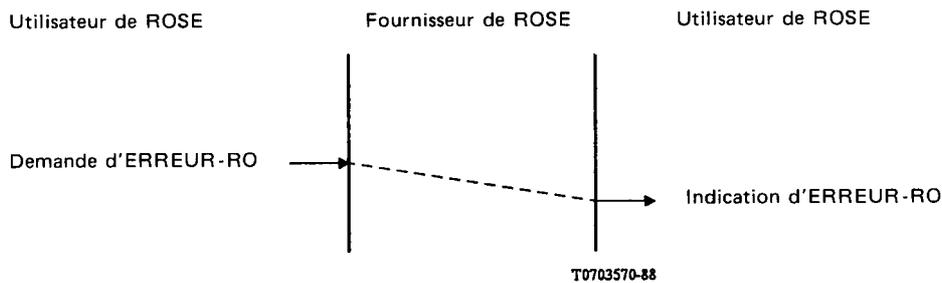


FIGURE 7/X.219

Primitives du service ERREUR-RO

10.3.1 Paramètres du service ERREUR-RO

Le tableau 4/X.219 indique les paramètres du service ERREUR-RO.

TABLEAU 4/X.219

Paramètres du service ERREUR-RO

Nom de paramètre	Demande	Indication
Valeur d'erreur	M	M (=)
Paramètre d'erreur	U	C (=)
ID d'appel	M	M (=)
Priorité	U	

10.3.1.1 *Valeur d'erreur*

Ce paramètre identifie l'erreur qui s'est produite pendant l'exécution de l'opération. Sa valeur doit être convenue entre les utilisateurs de ROSE. Il doit être fourni par le demandeur du service.

10.3.1.2 *Paramètre d'erreur*

Ce paramètre fournit une information additionnelle au sujet de l'erreur. Son type (éventuel) doit être convenu entre les utilisateurs de ROSE. Il doit être fourni par le demandeur du service.

10.3.1.3 *ID-d'appel*

Ce paramètre identifie l'appel correspondant (voir le § 10.1.1.4). Il doit être fourni par le demandeur du service. Sa valeur est celle de la primitive d'indication d'APPEL-RO correspondante.

10.3.1.4 *Priorité*

Ce paramètre définit la priorité attribuée au transfert de l'APDU correspondante (voir le § 10.1.1.6).

10.4 *Service REFUS-RO PAR L'UTILISATEUR*

Le service REFUS-RO PAR L'UTILISATEUR sert à un utilisateur de ROSE pour refuser une demande (indication d'APPEL-RO) de l'autre utilisateur de ROSE s'il a décelé une difficulté. Ce service peut aussi être utilisé par un utilisateur de ROSE pour refuser une réponse (indication de RÉSULTAT-RO, indication d'ERREUR-RO) en provenance de l'autre utilisateur de ROSE. Mais, pour éviter le viol des règles de mise en séquence d'autres ASE dans certains contextes d'application, un utilisateur de ROSE peut choisir de ne pas utiliser le service REFUS-RO PAR L'UTILISATEUR pour refuser des réponses. Il s'agit d'un service non confirmé.

La structure de service correspondante se compose de deux primitives de service, comme le montre la figure 8/X.219.

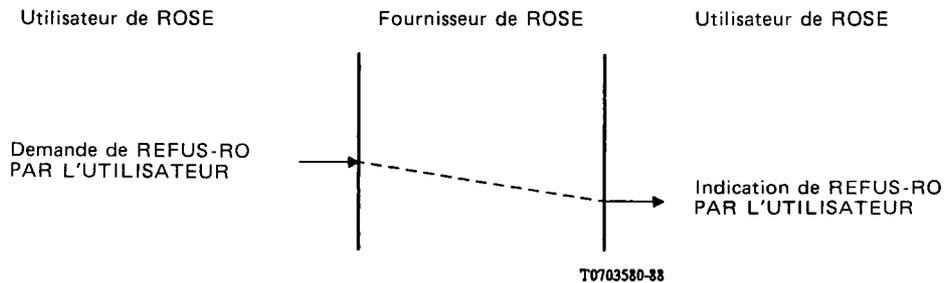


FIGURE 8/X.219

Primitives du service REFUS-RO PAR L'UTILISATEUR

10.4.1 *Paramètres du service REFUS-RO PAR L'UTILISATEUR*

Le tableau 5/X.219 indique les paramètres du service REFUS-RO PAR L'UTILISATEUR.

TABLEAU 5/X.219

Paramètres du service REFUS-RO PAR L'UTILISATEUR

Nom de paramètre	Demande	Indication
Raison de refus	M	M (=)
ID d'appel	M	M (=)
Priorité	U	

10.4.1.1 *Raison de refus*

Ce paramètre spécifie la raison du refus de la façon suivante:

- a) *Problème d'appel*: refus par l'utilisateur d'une primitive d'indication d'APPEL-RO ayant les valeurs:
 - appel-doublé:
signifie que le paramètre ID-d'appel viole les règles d'attribution du § 10.1.1.4
 - opération-non-reconnue:
signifie que l'opération n'est pas l'une de celles convenues entre utilisateurs de ROSE
 - argument-mal-typé:
signifie que le type de l'argument d'opération fourni n'est pas celui qui a été convenu entre utilisateurs de ROSE
 - limitation-de-ressources:
l'utilisateur de ROSE exécutant n'est pas en mesure d'exécuter l'opération appelée en raison de limitation de ressources
 - libération-par-le-demandeur:
le demandeur de l'association ne désire pas exécuter l'opération appelée car il est sur le point de procéder à la terminaison de l'association d'application
 - ID-lié-non-reconnu:
signifie qu'aucune opération en cours n'a un ID-d'appel égal à l'ID-lié spécifié
 - réponse-liée-non-attendue:
signifie que l'opération appelée à laquelle se réfère l'ID-lié n'est pas une opération mère
 - opération-fille-non-attendue:
signifie que l'opération fille appelée n'est pas l'une de celles que permet l'opération mère à laquelle se réfère l'ID-lié.
- b) *Problème de résultat positif*: refus par l'utilisateur d'une primitive d'indication de RÉSULTAT-RO ayant les valeurs:
 - appel-non-reconnu:
signifie qu'aucune opération ayant l'ID-d'appel spécifié n'est en cours
 - résultat-positif-non-attendu:
signifie que l'opération appelée ne signale pas de résultat
 - résultat-mal-typé:
signifie que le type du paramètre Résultat fourni n'est pas celui qui a été convenu entre utilisateurs de ROSE.
- c) *Problème de résultat négatif*: refus par l'utilisateur d'une primitive d'indication d'ERREUR-RO ayant les valeurs:
 - appel-non-reconnu:
signifie qu'aucune opération ayant l'ID-d'appel spécifié n'est en cours
 - réponse-d'erreur-non-attendue:
signifie que l'opération appelée ne signale pas d'échec
 - erreur-non-reconnue:
signifie que l'erreur signalée n'est pas l'une de celles convenues entre utilisateurs de ROSE
 - erreur-non-attendue:
signifie que l'erreur signalée n'est pas l'une de celles que l'opération appelée peut signaler
 - paramètre-mal-typé:
signifie que le type de paramètre d'erreur fourni n'est pas celui convenu entre utilisateurs de ROSE.

Ce paramètre doit être fourni par le demandeur du service.

10.4.1.2 ID-d'appel

Ce paramètre identifie l'appel correspondant (voir le § 10.1.1.4). Il doit être fourni par le demandeur du service. Sa valeur est celle de l'indication d'APPEL-RO refusée, de l'indication de RÉSULTAT-RO refusée ou de la primitive d'indication d'ERREUR-RO refusée.

10.4.1.3 Priorité

Ce paramètre définit la priorité attribuée au transfert de l'APDU correspondante (voir le § 10.1.1.6).

10.5 Service REFUS DE RO PAR LE FOURNISSEUR

Le service REFUS DE RO PAR LE FOURNISSEUR sert à annoncer à un utilisateur de ROSE qu'un problème a été détecté par le fournisseur du ROSE. Ce service est un service engendré par le fournisseur.

La structure de service correspondante se compose d'une seule primitive de service, comme le montre la figure 9/X.219.

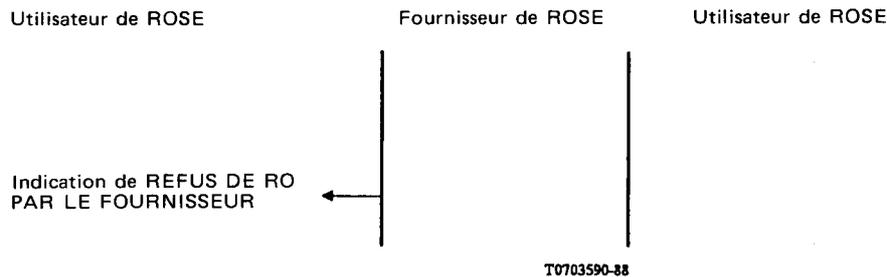


FIGURE 9/X.219

Primitive du service REFUS DE RO PAR LE FOURNISSEUR

10.5.1 Paramètres du service REFUS DE RO PAR LE FOURNISSEUR

Le tableau 6/X.219 énumère les paramètres du service REFUS DE RO PAR LE FOURNISSEUR.

TABLEAU 6/X.219

Paramètres du service REFUS-RO PAR LE FOURNISSEUR

Nom de paramètre	Indication
ID d'appel	O
Paramètres retournés	O
Raison de refus	O

10.5.1.1 ID-d'appel

Ce paramètre identifie l'appel correspondant (voir le § 10.1.1.4). Il est fourni par le fournisseur de ROSE. Sa valeur est celle de la demande d'APPEL-RO refusée, de la demande d'ERREUR-RO refusée ou des primitives de demande de REFUS DE RO PAR L'UTILISATEUR refusées. Il peut être omis si un ID-d'appel n'est pas disponible.

10.5.1.2 Paramètres retournés

Ce paramètre contient les paramètres de la demande d'APPEL-RO, de la demande de RÉSULTAT-RO, de la demande d'ERREUR-RO ou des primitives de demande de REFUS DE RO PAR L'UTILISATEUR si l'APDU correspondante n'a pu être transférée par le fournisseur de ROSE. Ce paramètre et le paramètre Raison de refus s'excluent mutuellement.

10.5.1.3 *Raison de refus*

Ce paramètre spécifie la raison du refus de la façon suivante:

- d) *Problème général*: refus par le fournisseur d'une APDU ayant les valeurs:
 - APDU-non-reconnue:
signifie que le type d'APDU, tel qu'il ressort de son identificateur de type, n'est pas l'un des quatre définis par la Recommandation X.229
 - APDU-mal-typée:
signifie que la structure de l'APDU n'est pas conforme à la Recommandation X.229
 - APDU-mal-structurée:
signifie que la structure de l'APDU n'est pas conforme à la notation et au codage normalisés définis dans les Recommandations X.208 et X.209.

Ce paramètre est fourni par le fournisseur de ROSE. Ce paramètre et les paramètres retournés s'excluent mutuellement.

11 **Mise en correspondance de la notation et des services**

11.1 *Contexte d'application et opérations*

La présente section décrit de quelle manière un contexte d'application est spécifié au moyen de la notation fournie par les macros définies au § 9.

Une telle spécification de contexte d'application se compose:

- a) d'une opération liante spécifiée à l'aide de la macro BIND, et
- b) d'une opération déliante spécifiée à l'aide de la macro UNBIND, et
- c) d'un ensemble d'opérations spécifié à l'aide de la macro OPERATION, et
- d) d'un ensemble d'erreurs lié aux opérations et spécifié au moyen de la macro ERROR.

Les opérations distantes (opération liante, opération déliante et opérations) peuvent être appelées par l'élément utilisateur.

Un élément utilisateur engendrant une association établit une association d'application en appelant une opération liante. Si l'association d'application est établie, l'élément utilisateur peut appeler des opérations. Lorsque l'élément utilisateur qui a engendré l'association veut terminer l'association d'application, il appelle une opération déliante.

11.2 *Mise en correspondance d'opérations avec les services ACSE, les services RTSE et les services ROSE*

L'opération liante et l'opération déliante sont mises en correspondance avec les services ACSE ou les services RTSE.

Les opérations sont mises en correspondance avec les services ROSE.

11.2.1 *Mise en correspondance avec les services ACSE*

L'opération liante est mise en correspondance avec le service d'ÉTABLISSEMENT D'ASSOCIATION, l'opération déliante avec le service de TERMINAISON D'ASSOCIATION.

11.2.1.1 *Mise en correspondance d'une opération liante*

Une opération liante est mise en correspondance avec le service d'ÉTABLISSEMENT D'ASSOCIATION.

11.2.1.1.1 *Appel d'une opération liante*

L'appel d'une opération liante est mis en correspondance avec les primitives de service de demande d'ÉTABLISSEMENT D'ASSOCIATION et d'indication d'ÉTABLISSEMENT D'ASSOCIATION.

La valeur d'argument de l'opération liante est mise en correspondance avec le paramètre Information d'utilisateur des primitives de service.

11.2.1.1.2 *Réponse d'une opération liante*

La réponse d'une opération liante est mise en correspondance avec les primitives de service de réponse à une demande d'ÉTABLISSEMENT D'ASSOCIATION et de confirmation d'ÉTABLISSEMENT D'ASSOCIATION.

Si l'opération liante a été exécutée avec succès, le paramètre Résultat des primitives de service est "accepté" et la valeur de résultat de l'opération liante est mise en correspondance avec le paramètre Information d'utilisateur des primitives de service.

Si l'opération liante n'a pas été exécutée avec succès, le paramètre Résultat est "refusé (définitivement)" et la valeur d'erreur de l'opération liante est mise en correspondance avec le paramètre Information d'utilisateur des primitives de service.

11.2.1.2 *Mise en correspondance d'une opération déliante*

Une opération déliante est mise en correspondance avec le service de TERMINAISON D'ASSOCIATION.

11.2.1.2.1 *Appel d'une opération déliante*

L'appel d'une opération déliante est mis en correspondance avec les primitives de service de demande de TERMINAISON D'ASSOCIATION et d'indication de TERMINAISON D'ASSOCIATION.

La valeur d'argument de l'opération déliante est mise en correspondance avec le paramètre Information d'utilisateur des primitives de service. La valeur du paramètre Raison des primitives de service est "normal".

11.2.1.2.2 *Réponse d'une opération déliante*

La réponse d'une opération déliante est mise en correspondance avec les primitives de service de réponse à une demande de TERMINAISON D'ASSOCIATION et de confirmation de TERMINAISON D'ASSOCIATION.

Si l'opération déliante a été exécutée avec succès, la valeur du paramètre Raison des primitives de service est "normal", la valeur du résultat de l'opération déliante est mise en correspondance avec le paramètre Information d'utilisateur des primitives de service, et le paramètre Résultat des primitives de service est "affirmatif".

Si l'opération déliante n'a pas été exécutée avec succès, la valeur du paramètre Raison des primitives de service est "non fini", la valeur d'erreur de l'opération déliante est mise en correspondance avec le paramètre Information d'utilisateur des primitives de service, et le paramètre Résultat des primitives de service est "affirmatif".

11.2.2 *Mise en correspondance avec les services RTSE*

L'opération liante est mise en correspondance avec le service d'ÉTABLISSEMENT D'ASSOCIATION et l'opération déliante avec le service de FERMETURE D'ASSOCIATION.

11.2.2.1 *Application d'une opération liante*

Une opération liante est mise en correspondance avec le service d'ÉTABLISSEMENT D'ASSOCIATION.

11.2.2.1.1 *Appel d'une opération liante*

L'appel d'une opération liante est mis en correspondance avec les primitives de service de demande d'ÉTABLISSEMENT D'ASSOCIATION et d'indication d'ÉTABLISSEMENT D'ASSOCIATION.

La valeur d'argument de l'opération liante est mise en correspondance avec le paramètre Données d'utilisateur des primitives de service. La valeur du paramètre du mode Dialogue est "bidirectionnel à l'alternat".

11.2.2.1.2 *Réponse d'une opération liante*

La réponse d'une opération liante est mise en correspondance avec les primitives de service de réponse à une demande d'ÉTABLISSEMENT D'ASSOCIATION et de confirmation d'ÉTABLISSEMENT D'ASSOCIATION.

Si l'opération liante a été exécutée avec succès, le paramètre Résultat des primitives de service est "accepté" et la valeur du résultat de l'opération liante est mise en correspondance avec le paramètre Données d'utilisateur des primitives de service.

Si l'opération liante n'a pas été exécutée avec succès, la valeur du paramètre Résultat des primitives de service est "refusé (définitivement)" et la valeur d'erreur de l'opération liante est mise en correspondance avec le paramètre Données d'utilisateur des primitives de service.

11.2.2.2 *Mise en correspondance d'une opération déliante*

Une opération déliante est mise en correspondance avec le service de FERMETURE D'ASSOCIATION.

11.2.2.2.1 *Appel d'une opération déliante*

L'appel d'une opération déliante est mis en correspondance avec les primitives de service de demande de FERMETURE D'ASSOCIATION et d'indication de FERMETURE D'ASSOCIATION.

La valeur d'argument de l'opération déliante est mise en correspondance avec le paramètre Données d'utilisateur des primitives de service. La valeur du paramètre Raison des primitives de service est "normal".

11.2.2.2.2 *Réponse d'une opération déliante*

La réponse d'une opération déliante est mise en correspondance avec les primitives de service de réponse à une demande de FERMETURE D'ASSOCIATION et de confirmation de FERMETURE D'ASSOCIATION.

Si l'opération déliante a été exécutée avec succès, la valeur du paramètre Raison des primitives de service est "normal" et la valeur du résultat de l'opération déliante est mise en correspondance avec le paramètre Données d'utilisateur des primitives de service.

Si l'opération déliante n'a pas été exécutée avec succès, la valeur du paramètre Raison des primitives de service est "non fini" et la valeur d'erreur de l'opération déliante est mise en correspondance avec le paramètre Données d'utilisateur des primitives de service.

11.2.3 *Mise en correspondance avec les services ROSE*

Une opération est mise en correspondance avec les services ROSE.

11.2.3.1 *Appel d'une opération*

L'appel d'une opération est mis en correspondance avec le service APPEL-RO.

La valeur attribuée à l'opération est mise en correspondance avec le paramètre Valeur d'opération de ce service. La valeur du Type-nommé dans la clause ARGUMENT de la macro OPERATION est mise en correspondance avec le paramètre Argument de ce service.

11.2.3.2 *Réponse d'une opération*

Si une opération a été exécutée avec succès, la réponse est mise en correspondance avec le service RÉSULTAT-RO.

La valeur du Type-nommé dans la clause RESULT de la macro OPERATION est mise en correspondance avec le paramètre Résultat de ce service.

Si une opération n'a pas été exécutée avec succès, la réponse est mise en correspondance avec le service ERREUR-RO.

Dans ce cas, l'une des erreurs figurant dans la liste d'identificateurs de noms d'erreurs dans la clause ERROR de la macro OPERATION peut être appliquée. La valeur attribuée à l'erreur appliquée est mise en correspondance avec le paramètre Erreur de ce service. La valeur du Type-nommé dans la clause PARAMETER de la macro ERROR de l'erreur appliquée est mise en correspondance avec le paramètre Paramètre-d'Erreur de ce service.

12 Information sur l'enchaînement

Ce paragraphe définit l'interaction entre opérations distantes et l'interaction entre les services ACSE et les services ROSE.

12.1 *Information sur l'enchaînement pour les opérations distantes*

12.1.1 *Opération liante*

12.1.1.1 *Restrictions d'utilisation*

Une opération liante n'est pas utilisée sur une association d'application établie. Une opération liante exécutée avec succès établit une association d'application.

12.1.1.2 *Opération distante susceptible d'être interrompue*

L'opération liante n'interrompt aucune opération distante.

12.1.1.3 *Opérations distantes susceptibles de l'interrompre*

Aucune opération distante ne peut interrompre cette opération.

12.1.1.4 *Collisions*

Il y a collision d'opérations liantes lorsque les éléments utilisateurs des deux AE appellent simultanément et mutuellement une opération liante. Dans ce cas, deux associations d'application indépendantes sont établies.

12.1.2 *Opération déliante*

12.1.2.1 *Restrictions d'utilisation*

Une opération déliante n'est utilisée que dans le cas d'une association d'application établie. Elle sert uniquement à l'élément utilisateur qui a appelé l'opération liante, si aucune réponse en provenance d'opérations des classes d'opérations 1 ou 2 n'est attendue.

L'association d'application cesse d'être établie, que l'opération déliante ait abouti ou non.

12.1.2.2 *Opérations distantes susceptibles d'être interrompues*

Une opération déliante n'interrompt pas des opérations distantes dans le cas d'opérations de la classe d'association 1 et des classes d'opérations 1 ou 2.

Dans tous les autres cas, une opération déliante peut interrompre des opérations. Toutefois, si les classes d'opérations 3, 4 ou 5 et/ou les classes d'association 2 ou 3 sont utilisées dans un contexte spécifique d'application, on admet que l'interruption est acceptable ou que le contexte d'application fournit des opérations permettant d'éviter l'interruption.

12.1.2.3 *Opérations distantes susceptibles de l'interrompre*

Aucune opération distante n'interrompt l'opération déliante.

12.1.2.4 *Collisions*

Du fait que seul l'initiateur d'une association peut procéder à la terminaison de l'association d'application, il n'y a pas de collision.

12.1.3 *Opérations*

12.1.3.1 *Restrictions d'utilisation*

Les opérations sont utilisées uniquement sur une association d'application établie.

12.1.3.2 *Opérations distantes susceptibles d'être interrompues*

Les opérations n'interrompent aucune opération distante.

12.1.3.3 *Opérations distantes susceptibles de les interrompre*

Des opérations peuvent être interrompues par n'importe quelle opération déliante (voir § 12.1.2.2).

12.1.3.4 *Collisions*

Il n'y a pas de collisions d'opérations.

12.1.4 *Autres informations sur l'enchaînement*

Les services susceptibles d'interrompre des opérations ne sont pas visibles à l'interface d'opération. Cependant, des opérations peuvent être interrompues par certains services (voir le § 12.2).

Les opérations liantes et les opérations déliantes sont interrompues par les services de COUPURE D'ASSOCIATION, de COUPURE D'ASSOCIATION PAR LE FOURNISSEUR, de COUPURE D'ASSOCIATION PAR L'UTILISATEUR (RT) et de COUPURE D'ASSOCIATION PAR LE FOURNISSEUR (RT).

Les opérations sont interrompues par les services de COUPURE D'ASSOCIATION, de COUPURE D'ASSOCIATION PAR LE FOURNISSEUR, de COUPURE D'ASSOCIATION PAR L'UTILISATEUR (RT), de COUPURE D'ASSOCIATION PAR LE FOURNISSEUR (RT), de REFUS-RO PAR L'UTILISATEUR et de REFUS-RO PAR LE FOURNISSEUR.

De plus, une opération peut être interrompue par le service de TERMINAISON D'ASSOCIATION. Cependant, cela ne concerne que l'interruption d'une opération par une opération déliante. Les opérations interrompues ne sont pas prises en considération dans les services interrompus du § 12.2.

Du fait que toutes les opérations distantes sont mises en correspondance avec des services et que les opérations distantes interruptrices (opérations déliantes) sont représentées par des services, aucune opération distante interruptrice n'est prise en considération dans les dispositions relatives aux services interrupteurs du § 12.2.

12.2 *Informations sur l'enchaînement intéressant les services*

12.2.1 *Services ACSE*

Dans le cas des services ACSE, les informations sur l'enchaînement sont données dans la Recommandation X.217. Des renseignements complémentaires sont indiqués ci-dessous.

12.2.1.1 *Services ROSE susceptibles d'être interrompus*

Outre les services interrompus définis dans la Recommandation X.217, tous les services ROSE, exception faite du service REFUS-RO PAR LE FOURNISSEUR, sont interrompus par les services de COUPURE D'ASSOCIATION et de COUPURE D'ASSOCIATION PAR LE FOURNISSEUR et peuvent être interrompus par le service de TERMINAISON D'ASSOCIATION (voir le § 12.2.3.6).

12.2.1.2 *Services ROSE susceptibles de les interrompre*

Il n'y a pas de services ROSE interrupteurs.

12.2.2 *Services RTSE*

Les informations sur l'enchaînement pour les services RTSE sont données dans la Recommandation X.218. Des renseignements complémentaires sont indiqués ci-dessous.

12.2.2.1 *Services ROSE susceptibles d'être interrompus*

Outre les services interrompus définis dans la Recommandation X.218, tous les services ROSE, exception faite du service REFUS-RO PAR LE FOURNISSEUR, sont interrompus par les services de COUPURE D'ASSOCIATION PAR L'UTILISATEUR (RT) et de COUPURE D'ASSOCIATION PAR LE FOURNISSEUR (RT) ainsi que par le service de confirmation de TRANSFERT FIABLE négative.

12.2.2.2 *Services ROSE susceptibles de les interrompre*

Il n'y a pas de services ROSE interrupteurs.

12.2.3 *Services ROSE*

On trouvera ci-dessous la description de l'interaction entre les services ROSE. Les interactions avec les services ACSE sont décrites au § 12.2.1 et celles avec les services RTSE au § 12.2.2.

12.2.3.1 *Service APPEL-RO*

12.2.3.1.1 *Type de service*

Le service APPEL-RO est un service non confirmé.

12.2.3.1.2 *Restrictions d'utilisation*

Le service APPEL-RO n'est utilisé que sur une association d'application établie.

12.2.3.1.3 *Services susceptibles d'être interrompus*

Le service APPEL-RO n'interrompt aucun service.

12.2.3.1.4 *Services susceptibles de l'interrompre*

Le service APPEL-RO est interrompu par le service REFUS-RO.

12.2.3.1.5 *Collision*

Il ne peut y avoir de collision avec le service APPEL-RO.

12.2.3.2 *RÉSULTAT-RO*

12.2.3.2.1 *Type de service*

Service non confirmé.

12.2.3.2.2 *Restrictions d'utilisation*

Le service RÉSULTAT-RO n'est utilisé que sur une association d'application établie et en réponse à un service APPEL-RO.

12.2.3.2.3 *Services susceptibles d'être interrompus*

Le service RÉSULTAT-RO n'interrompt aucun service.

12.2.3.2.4 *Service susceptibles de l'interrompre*

Le service RÉSULTAT-RO est interrompu par le service REFUS-RO PAR LE FOURNISSEUR.

12.2.3.2.5 *Collisions*

Il n'y a pas de collision avec le service RÉSULTAT-RO.

12.2.3.3 *Service ERREUR-RO*

12.2.3.3.1 *Type de service*

Service non confirmé.

12.2.3.3.2 *Restriction d'utilisation*

Le service ERREUR-RO n'est utilisé que sur une association d'application établie et en réponse à un service APPEL-RO.

12.2.3.3.3 *Services susceptibles d'être interrompus*

Le service ERREUR-RO n'interrompt aucun service.

12.2.3.3.4 *Services susceptibles de l'interrompre*

Le service ERREUR-RO est interrompu par le service REFUS-RO PAR LE FOURNISSEUR.

12.2.3.3.5 *Collisions*

Il n'y a pas de collision avec le service ERREUR-RO.

12.2.3.4 *Service REFUS-RO PAR L'UTILISATEUR*

12.2.3.4.1 *Type de service*

Service non confirmé.

12.2.3.4.2 *Restriction d'utilisation*

Le service REFUS-RO PAR L'UTILISATEUR n'est utilisé que sur une association d'application établie et en réponse aux services APPEL-RO, RÉSULTAT-RO et ERREUR-RO.

12.2.3.4.3 *Services susceptibles d'être interrompus*

Le service REFUS-RO PAR L'UTILISATEUR n'interrompt aucun service.

12.2.3.4.4 *Services susceptibles de l'interrompre*

Le service REFUS-RO PAR L'UTILISATEUR est interrompu par le service REFUS-RO PAR LE FOURNISSEUR.

12.2.3.4.5 *Collisions*

Il n'y a pas de collision avec le service REFUS-RO PAR L'UTILISATEUR.

12.2.3.5 *Service REFUS-RO PAR LE FOURNISSEUR*

12.2.3.5.1 *Type de service*

Service engendré par le fournisseur.

12.2.3.5.2 *Restriction d'utilisation*

Sans objet.

12.2.3.5.3 *Services susceptibles d'être interrompus*

Le service REFUS-RO PAR LE FOURNISSEUR interrompt tous les autres services ROSE.

12.2.3.5.4 *Services susceptibles de l'interrompre*

Le service REFUS-RO PAR LE FOURNISSEUR n'est interrompu par aucun autre service.

12.2.3.5.5 *Collision*

Si le service ACSE ou un service RTSE cause une coupure ou la terminaison d'une association d'application, il incombe à l'initiative locale de faire connaître à l'utilisateur du service les services REFUS-RO PAR LE FOURNISSEUR en instance pour les paramètres renvoyés.

12.2.3.6 *Informations additionnelles sur l'enchaînement*

Les restrictions d'utilisation relatives aux opérations déliantes (voir le § 12.1.2.1) et la mise en correspondance d'opérations déliantes à la suite du service de TERMINAISON D'ASSOCIATION évitent l'interruption des services ROSE si la classe d'association 1 et les classes d'opération 1 et 2 sont seules utilisées.

En cas d'utilisation des classes d'association 2 ou 3 ou des classes d'opération 3, 4 ou 5, le service de TERMINAISON D'ASSOCIATION peut interrompre les services ROSE. Dans ces cas, il appartient au concepteur du contexte d'application soit d'accepter cette interruption, soit de fournir les moyens (par exemple, des opérations) permettant de préparer la terminaison d'une association d'application.

ANNEXE A

(à la Recommandation X.219)

Notation pour la spécification des éléments du service d'application et des contextes d'application

La présente annexe fait partie intégrante de la présente Recommandation.

Cette annexe fournit une notation pour la spécification des éléments du service d'application et les contextes d'application qui sont spécifiés au moyen de la notation-RO. La notation-RO peut être utilisée pour spécifier l'opération liante et l'opération déliante d'un contexte d'application. De plus, la notation-RO peut servir à spécifier les types d'opération et les types d'erreur de divers éléments du service d'application (ASE de l'utilisateur de ROSE). Si la notation-RO est combinée avec d'autres notations, il devrait être possible d'utiliser d'autres outils de spécification définis ailleurs.

La présente annexe définit deux macros pour la spécification des éléments du service d'application et les contextes d'application. La définition formelle de ces macros est indiquée à la figure A-1/X.219.

A.1 *Éléments du service d'application (ASE)*

La notation permet l'identification univoque d'un ASE.

Si un ASE est un utilisateur de ROSE, la notation permet aussi la spécification des caractéristiques de l'ASE. L'interface d'opération et le protocole d'un ASE utilisateur de ROSE sont spécifiés par un ensemble de types d'opération et un ensemble de types d'erreur.

Le protocole spécifié pour un ASE donné peut-être intrinsèquement:

- a) symétrique, ou
- b) asymétrique.

Si le protocole est symétrique, les deux utilisateurs d'ASE peuvent appeler la même ensemble de types d'opération.

Si le protocole est asymétrique, un utilisateur d'ASE (appelé le fournisseur dans le contexte de cette annexe) fournit une fonctionnalité de traitement de l'information qui est utilisée par l'utilisateur d'ASE homologue (appelé le consommateur dans le contexte de la présente annexe). Dans ce cas, un type spécifique d'opération peut être:

- 1) appelé par le consommateur,
- 2) appelé par le fournisseur

de la fonctionnalité de traitement de l'information.

Remarque – Une attribution particulière des termes “fournisseur” et “consommateur” est souvent intuitive. On pourrait naturellement envisager un système de fichier qui serait, par exemple, appelé fournisseur, son utilisateur étant appelé consommateur. A strictement parler cependant, l'attribution des deux termes est arbitraire.

Si un ASE utilise le concept d'opérations liées, un type d'opération spécifique peut être appelé comme opération fille. Celui qui appelle l'opération fille est l'exécutant de l'opération mère liée. Les types d'opération exclusivement appelés comme opérations filles ne doivent pas figurer dans la spécification de l'ASE, ils le sont dans la spécification du type de leurs opérations mères.

Les types d'erreur que peuvent signaler les opérations ne figurent pas dans la spécification de l'ASE, ils sont énumérés dans la spécification de type des opérations.

L'ensemble des opérations restantes (opérations qui ne sont pas exclusivement appelées comme opérations filles) qui est autorisé à appeler ces opérations peut s'exprimer par une notation formelle spécifiant l'ASE utilisateur de ROSE.

A.1.1 *Spécification d'un élément du service d'application*

Un ASE peut être spécifié par une notation formelle basée sur la macro **APPLICATION SERVICE-ELEMENT** (voir la figure A-1/X.219).

```

extension-Notation-Opérations-Distantes {joint-iso-ccitt opérations-distantes(4) extension-notation(2)}

DEFINITIONS ::=

BEGIN

EXPORTS APPLICATION-SERVICE-ELEMENT, APPLICATION-CONTEXT, aCSE;

IMPORTS OPERATION, BIND, UNBIND FROM Notation-Opération-Distante
                                         {joint-iso-ccitt opérations-distantes(4) notation(0)};

-- définition de macro pour les ASE

APPLICATION-SERVICE-ELEMENT-MACRO ::=

BEGIN

TYPE NOTATION ::= AseSymétrique | ConsommateurAppelle FournisseurAppelle | vide

VALUE NOTATION ::= valeur (VALUE OBJECT IDENTIFIER)

AseSymétrique ::= "OPERATIONS" "{"ListeOpérations"}

ConsommateurAppelle ::= "CONSUMER INVOKES" "{"ListeOpérations"} | vide

FournisseurAppelle ::= "SUPPLIER INVOKES" "{"ListeOpérations"} | vide

ListeOpérations ::= Opération | ListeOpérations "," Opération

Opération ::= valeur (OPERATION)

END

aCSE APPLICATION-SERVICE-ELEMENT ::= {joint-iso-ccitt opérations-distantes(4) aseID-ACSE(4)}

-- extension de Notation d'Opérations Distantes, à suivre

```

FIGURE A-1/X.219 (feuillet 1 sur 2)

Définition formelle de types d'ASE et de données de contexte d'application

La notation de type de la macro **APPLICATION-SERVICE-ELEMENT** permet de spécifier un ASE. La notation de valeur du macro **APPLICATION-SERVICE-ELEMENT** permet de spécifier un identificateur univoque pour l'ASE.

La notation correspondant à un type d'ASE est le mot clé **APPLICATION-SERVICICE-ELEMENT**, facultativement suivi de la spécification d'opérations.

Si le protocole spécifié pour l'ASE est symétrique, le mot clé **OPERATIONS** et les noms de référence des opérations sont spécifiés.

Si le protocole spécifié pour l'ASE est asymétrique, les mots clés **CONSUMER INVOKES** est les noms de référence des opérations que le consommateur peut appeler, et/ou les mots clés **SUPPLIER INVOKES** et les noms de référence des opérations que le fournisseur peut appeler sont spécifiés.

A.2 Contextes d'application

Dans le cadre de la présente annexe, un contexte d'application identifie de façon explicite:

- a) une opération liante,
- b) une opération déliante,
- c) un ensemble d'éléments du service d'application

et nécessite une ou plusieurs syntaxes abstraites identifiées.

L'ensemble d'éléments du service d'application contient:

- 1) des ASE n'utilisant pas la notation-RO, à savoir des ACSE, facultativement des RTSE et, facultativement, d'autres ASE;
- 2) facultativement des ASE de ROSE et d'utilisateur de ROSE.

Si le contexte d'application contient des ASE d'utilisateur de ROSE avec un protocole asymétrique, la notation permet de spécifier si celui qui engendre l'association ou celui qui y répond est le consommateur d'un tel ASE.

L'identification de l'opération liante, de l'opération déliante, des éléments du service d'application et des syntaxes abstraites peut être exprimée par une notation formelle spécifiant le contexte d'application.

A.2.1 Spécification d'un contexte d'application

Un contexte d'application peut être spécifié par une notation basée sur la macro **APPLICATION-CONTEXT** (voir figure A-1/X.219).

La notation de type de la macro **APPLICATION-CONTEXT** permet de spécifier le contexte d'application. La notation de la valeur de la macro **APPLICATION-CONTEXT** permet la spécification d'un identificateur univoque pour le contexte d'application.

La notation correspondant à un type de contexte d'application est le mot clé **APPLICATION-CONTEXT** suivi des mots clés **APPLICATION SERVICE ELEMENTS** et des noms de référence des ASE qui n'utilisent pas la notation RO, suivis du mot clé **BIND** et du nom de référence du type d'opération liante, suivis du mot clé **UNBIND** et du nom de référence du type d'opération déliante, facultativement suivis de la spécification des ASE qui utilisent les opérations, suivie des mots clés **ABSTRACT SYNTAXES** et des noms de référence des syntaxes abstraites.

Si le contexte d'application contient des ASE utilisateurs de ROSE, on utilise les mots clés **REMOTE OPERATIONS** et le nom de référence du ROSE, suivis de la spécification des ASE à protocole symétrique et/ou de la spécification d'ASE à protocole asymétrique. Les ASE à protocole symétrique sont spécifiés par les mots clés **OPERATIONS OF** et les noms de référence de ces ASE. Les ASE à protocole asymétrique sont spécifiés par les mots clés **INITIATOR CONSUMER OF** suivis des noms de référence des ASE dont le consommateur est l'initiateur de l'association, et/ou des mots clés **RESPONDER CONSUMER OF** suivis des noms de référence des ASE dont le consommateur est celui qui répond à la demande d'association.

A.2.2 Mise en correspondance de la notation et du service

L'identificateur de contexte d'application et la liste de noms de syntaxes abstraites spécifiés au moyen de la macro **APPLICATION-CONTEXT** sont mis en correspondance soit avec les services d'ÉTABLISSEMENT D'ASSOCIATION (RT) de RTSE, si RTSE est inclus dans le contexte d'application, soit, dans les autres cas, avec les services d'ÉTABLISSEMENT D'ASSOCIATION de l'ACSE.

La valeur du contexte d'application est mise en correspondance avec le paramètre Nom du contexte d'application des services d'ÉTABLISSEMENT D'ASSOCIATION (RT) ou d'ÉTABLISSEMENT D'ASSOCIATION.

Les noms de syntaxes abstraites sont mis en correspondance avec le paramètre Liste de définitions de contexte de présentation et avec la Liste de résultats des définitions du contexte de présentation des services d'ÉTABLISSEMENT D'ASSOCIATION (RT) ou d'ÉTABLISSEMENT D'ASSOCIATION.

--	extension de Notation d'Opérations Distantes, suite
--	définition de macro pour contextes d'application
APPLICATION-CONTEXT MACRO ::=	
BEGIN	
TYPE NOTATION	::= élémentsNonRO élémentsRO Liants SyntaxesAbstraites
VALUE NOTATION	::= valeur (VALUE OBJECT IDENTIFIER)
élémentsNonRO	::= "APPLICATION SERVICE ELEMENTS" "{"ListeASE"}"
Liants	::= type "BIND" -- dénote une opération type liant type "UNBIND" -- dénote une opération type déliant
élémentsRO	::= "REMOTE OPERATIONS" "{"IDase"}" -- identifiant ROSE AseSymétriques AseAsymétriques vide
AseSymétriques	::= "OPERATIONS OF" "{"ListeASE"}" vide
AseAsymétriques	::= InitiateurConsommateurDe RépondeurConsommateurDe
InitiateurConsommateurDe	::= "INITIATOR CONSUMER OF" "{"ListeAse"}" vide
RépondeurConsommateurDe	::= "RESPONDER CONSUMER OF" "{"ListeAse"}" vide
SyntaxesAbstraites	::= "ABSTRACT SYNTAXES" "{"ListeDeSyntaxesAbstraites"}"
ListeAse	::= IDase ListeAse "," IDase
IDase	::= valeur (APPLICATION-SERVICE-ELEMENT)
ListeDeSyntaxesAbstraites	::= SyntaxeAbstraite ListeDeSyntaxesAbstraites "," SyntaxeAbstraite
SyntaxeAbstraite	::= valeur (OBJECT IDENTIFIER) -- syntaxe abstraite identifiante
END	
END	-- fin de l'extension de la Notation d'Opérations Distantes

FIGURE A-1/X.219 (feuillet 2 sur 2)

Définition formelle de types d'ASE et de données de contexte d'application

ANNEXE B

(à la Recommandation X.219)

Directives aux concepteurs de protocoles d'application sur l'emploi du ROSE

La présente annexe ne fait pas partie de la Recommandation X.219.

Elle propose des exemples et des directives à l'intention des concepteurs de protocoles d'application sur l'emploi du ROSE.

B.1 Exemples d'opérations et d'erreurs

Ce paragraphe contient des exemples relatifs à la définition des opérations et erreurs.

B.1.1 Classes d'opérations

On trouvera ci-dessous des exemples relatifs à la définition des opérations des classes d'opérations 1 à 5.

L'opération **Exemplé'opération12** des classes d'opérations 1 ou 2 (voir l'exemple ci-dessous) signale le succès (résultat du type **Typed'Argument12**) ou l'échec (erreurs **Exemplé'erreur1** ou **Exemplé'erreur2**). L'argument de l'opération **Exemplé'opération12** est du type **Typed'Argument12**. La valeur de l'opération **Exemplé'opération12** est **1**.

Exemple d'opération12	OPERATION
ARGUMENT	Typed'Argument12
RESULT	TypedeRésultat12
ERROR	{Exemplé'erreur1, Exemplé'erreur2}
::= 1	

L'opération **Exemplé'opération3** de la classe d'opérations 3 (voir l'exemple ci-dessous) signale uniquement l'échec (erreur **Exemplé'erreur1**) éventuel. L'argument de l'opération **Exemplé'opération2** est du type **Typed'Argument3**. La valeur de l'opération **Exemplé'opération3** est **2**.

Exemplé'opération3	OPERATION
ARGUMENT	Typed'Argument3
ERROR	{Exemplé'erreur}
::= 2	

L'opération **Exemplé'opération4** de la classe d'opérations 4 (voir l'exemple ci-dessous) signale le succès (résultat du type **TypedeRésultat4**) exclusivement. L'argument de l'opération **Exemplé'opération4** est du type **Typed'Argument4**. La valeur de l'opération **Exemplé'opération4** est **3**.

Exemplé'opération4	OPERATION
ARGUMENT	Typed'Argument4
RESULT	TypedeRésultat4
::= 3	

Les opérations **Exemplé'opération51** et **Exemplé'opération52** de la classe d'opérations 5 (voir l'exemple ci-dessous) ne signalent aucun résultat. L'argument de l'opération **Exemplé'opération51** est du type **Typed'Argument4**; l'opération **Exemplé'opération52** n'a pas d'argument. La valeur de l'opération **Exemplé'opération51** est **4**, celle de l'opération **Exemplé'opération52** est **5**.

Exemplé'opération51	OPERATION
ARGUMENT	Typed'argument4
::= 4	
Exemplé'opération52	OPERATION
::= 5	

B.1.2 Opérations liées

L'exemple ci-dessous montre la définition d'un ensemble d'opérations liées consistant en une opération mère **op-mère12** et en opérations filles **Exemplé'opération51** et **Exemplé'opération52**.

op-mère12	OPERATION
ARGUMENT	Typed'Argument12
RESULT	TypedeRésultat12
ERROR	{Exemplé'erreur1, Exemplé'erreur2}
LINKED	{Exemplé'opération51, Exemplé'opération52}
::= 6	

B.1.3 Erreurs

Les erreurs (voir **Exemplé'erreur1** et **Exemplé'erreur2** ci-dessous) signalent des échecs. Le paramètre de l'erreur **Exemplé'erreur1** est du type **TypedeParamètre1**, tandis que l'erreur **Exemplé'erreur2** n'a pas de paramètre. La valeur de l'erreur **Exemplé'erreur1** est **1**, celle de l'erreur **Exemplé'erreur2** est **2**.

Exemple'd'erreur1	ERROR PARAMETER ::= 1	TypedParamètre1
Exemple'd'erreur2	ERROR ::= 2	

B.2 Exemples d'opérations liantes et d'opérations déliantes

On trouvera ci-dessous des exemples concernant la définition des opérations liantes et des opérations déliantes.

B.2.1 Opérations liantes

Les opérations liantes servent à établir une association d'application.

La demande de l'opération liante **ExempleLiant1**, qui vise à établir une association d'application, est accompagnée de l'argument du type **Typed'ArgumentLiant1**. La réponse positive à l'établissement de l'association d'application est accompagnée du résultat du type **TypedeRésultatLiant1**. La réponse négative à l'établissement de l'association d'application s'accompagne de l'erreur-liante du type **Typed'ErreurLiant1**.

ExempleLiant1 ::=	BIND	
	ARGUMENT	Typed'ArgumentLiant1
	RESULT	TypedRésultatLiant1
	BIND-ERROR	Typed'ErreurLiant1

La demande d'opération-liante **ExempleLiant2**, qui vise à établir une association d'application, est accompagnée de l'argument du type **Typed'ArgumentLiant1**. La réponse positive à l'établissement de l'association d'application n'est pas accompagnée de données d'utilisateur. La réponse négative à l'établissement de l'association d'application s'accompagne de l'erreur-liante du type **Typed'ErreurLiant1**.

ExempleLiant2 ::=	BIND	
	ARGUMENT	Typed'ArgumentLiant1
	BIND-ERROR	Typed'ErreurLiant1

A noter que l'argument, le résultat et l'erreur-liante d'une opération liante sont facultatifs. Ni la demande de l'opération liante **ExempleLiant3** visant à établir une association d'application, ni la réponse à l'établissement de l'association d'application ne sont accompagnées de données d'utilisateur.

ExempleLiant3 ::=	BIND	
--------------------------	-------------	--

B.2.2 Opérations déliantes

Les opérations déliantes servent à terminer une association d'application.

La demande de l'opération déliante **ExempleDéliant1**, qui vise à terminer une association d'application, est accompagnée par l'argument du type **Typed'ArgumentDéliant1**. La réponse à la terminaison de l'association d'application s'accompagne soit du résultat du type **TypedeRésultatDéliant1**, soit de l'erreur déliante du type **Typed'ErreurDéliant1**.

ExempleDéliant1 ::=	UNBIND	
	ARGUMENT	Typed'ArgumentDéliant1
	RESULT	TypedeRésultatDéliant1
	UNBIND-ERROR	Typed'ErreurDéliant1

La demande de l'opération déliante **ExempleDéliant2**, qui vise à terminer une association d'application, est accompagnée par l'argument du type **Typed'ArgumentDéliant1**. La réponse à la terminaison de l'association d'application s'accompagne facultativement du type d'erreur déliante **Typed'ErreurDéliant1**.

ExempleDéliant2 ::=	UNBIND	
	ARGUMENT	Typed'ArgumentDéliant1
	UNBIND-ERROR	Typed'ErreurDéliant1

A noter le caractère facultatif de l'argument, du résultat et de l'erreur-déliante de l'opération déliante. Ni la demande de l'opération liante **ExempleDéliant3** visant à terminer une association d'application, ni la réponse à la terminaison de l'association d'application ne s'accompagnent de données d'utilisateur.

ExempleDéliant3 ::=	UNBIND	
----------------------------	---------------	--

B.3 Exportation et importation d'opérations et d'erreurs

On trouvera ci-dessous des exemples de la manière d'exporter et d'importer des opérations et des erreurs.

L'exemple ci-dessous montre comment exporter des opérations et des erreurs. L'**opération10** et l'**erreur10** ont des valeurs locales. Le **Typed'OpérationA** et le **Typed'ErreurA** sont des types, et des valeurs spécifiques doivent être assignées dans les modules importateurs. L'**opération11** et l'**erreur11** ont des valeurs globalement uniques.

```
ModuleExportateur {identificateurd'objet1} DEFINITION ::=
BEGIN

    EXPORTS          opération10, Typed'OpérationA, opération11, erreur10,
                    Typed'ErreurA, erreur11;
    IMPORTS          OPERATION, ERROR, BIND, UNBIND
    FROM Notation-d'Opération-Distante
                   {joint-iso-ccitt opérations-distantes(4) notation(0)};

    opération10     OPERATION
                   ARGUMENT          Typed'Argument10
                   RESULT             TypedeRésultat10
                   ERROR              {erreur10}
                   ::= 10

    Typed'OpérationA ::=
    OPERATION
    ARGUMENT          Typed'ArgumentA
    RESULT            TypedeRésultatA

    opération11     OPERATION
                   ARGUMENT          Typed'Argument11
                   RESULT            TypedeRésultat11
                   ERROR              {erreur11}
                   ::= {identificateurd'objet2composante11}

    erreur10        ERROR
                   PARAMETER         TypedeParamètre10
                   ::= 10

    Typed'ErreurA   ::=
    ERROR
    PARAMETER        TypedeParamètreA

    erreur11        ERROR
                   PARAMETER         TypedeParamètre11
                   ::= {identificateurd'objet2composante12}

END
```

L'exemple suivant montre comment importer des opérations et des erreurs. L'**opération10** et l'**erreur10** ont des valeurs locales définies dans le module exportateur. Le concepteur du module importateur garantit l'unicité de ces valeurs dans la syntaxe abstraite. L'opération **opération13** est du type **Typed'OpérationA** et la valeur **13** lui est assignée. L'erreur **erreur13** est du type **Typed'ErreurA** et la valeur **13** lui est assignée. L'**opération11** et l'**erreur11** ont des valeurs globalement uniques définies dans le module exportant.

```
ModuleImportateur {identificateurd'objet3} DEFINITIONS ::=
BEGIN

    IMPORTS          opération10, Typed'OpérationA, opération11, erreur10,
                    Typed'ErreurA, erreur11
    FROM ModuleExportateur {identificateurd'objet1};

    opération13     Typed'OpérationA   := 13
    erreur13        Typed'ErreurA      := 13

END
```

B.4 Définition des éléments du service d'application

On trouvera ci-dessous des exemples de la manière dont il convient de définir les éléments du service d'application. Ces exemples se rapportent aux opérations et erreurs définies dans les exemples du § B.1.

L'élément du service d'application **élément1** englobe les opérations **Exemplé'opération12** et **Exemplé'opération3** et les erreurs **Exemplé'erreur1** et **Exemplé'erreur2**. Noter que les erreurs sont indirectement incluses par la définition des opérations. L'élément du service d'application **élément1** est symétrique, les deux éléments utilisateurs pouvant appeler les opérations **Exemplé'opération12** et **Exemple d'opération3**.

```
élément1      APPLICATION-SERVICE-ELEMENT
              OPERATIONS{Exemplé'opération12, Exemple d'opération3}
              ::= {identificateur'objetDel'Elément1}
```

L'élément du service d'application **élément2** englobe les opérations **Exemplé'opération3** et **Exemplé'opération4** et l'erreur **Exemplé'erreur1**. L'élément du service d'application **élément2** est asymétrique, c'est-à-dire qu'un seul élément utilisateur (celui qui joue le rôle de consommateur) peut appeler les opérations **Exemplé'opération3** et **Exemplé'opération4**.

```
élément2      APPLICATION-SERVICE-ELEMENT
              CONSUMER INVOKES {Exemplé'opération3, Exemplé'opération4}
              ::= {identificateur'objetDel'Elément2}
```

L'élément du service d'application **élément3** englobe les opérations **Exemplé'opération12** et **Exemplé'opération51** et les erreurs **Exemplé'erreur1** et **Exemplé'erreur2**; il est asymétrique, en ce sens qu'un seul élément utilisateur (celui qui joue le rôle de fournisseur) peut appeler les opérations **Exemplé'opération12** et **Exemplé'opération51**.

```
élément3      APPLICATION-SERVICE-ELEMENT
              SUPPLIER INVOKES {Exemplé'opération12, Exemplé'opération51}
              ::= {identificateur'objetDel'Elément3}
```

L'élément du service d'application **élément4** englobe les opérations **op-mère12**, **Exemplé'opération51** et **Exemplé'opération52** et les erreurs **Exemplé'erreur1** et **Exemplé'erreur2**. A noter que les opérations filles **Exemplé'opération51** et **Exemplé'opération52** sont incluses indirectement par la définition de l'opération mère **op-mère12**. L'élément du service d'application **élément4** est asymétrique, en ce sens qu'un seul élément utilisateur (celui qui joue le rôle de consommateur) peut appeler l'opération **op-mère12** et que seul l'autre élément utilisateur (qui joue le rôle de fournisseur) peut appeler les opérations filles **Exemplé'opération51** et **Exemplé'opération52** pendant l'exécution de l'opération mère **op-mère12**.

```
élément4      APPLICATION-SERVICE-ELEMENT
              CONSUMER INVOKES {op-mère12}
              ::= {identificateur'objetDel'Elément4}
```

L'élément du service d'application **élément5** englobe les mêmes opérations et erreurs que l'élément du service d'application **élément4**. Leur seule différence est que l'élément utilisateur qui tient le rôle de fournisseur peut appeler l'opération **Exemplé'opération52** soit comme opération fille, soit en dehors d'un ensemble d'opérations liées (comme opération autre qu'une opération fille).

```
élément5      APPLICATION-SERVICE-ELEMENT
              CONSUMER INVOKES {op-mère12}
              SUPPLIER INVOKES {Exemplé'opération52}
              ::= {identificateur'objetDel'Elément5}
```

B.5 Définition des contextes d'application

On trouvera ci-dessous des exemples de la manière dont il convient de définir les contextes d'application de diverses classes d'association. Ces exemples se rapportent aux éléments du service d'application définis dans les exemples du § B.4 et aux opérations liantes et opérations déliantes définies dans les exemples du § B.2.

Le contexte d'application **contexte1** englobe les éléments du service d'application ACSE, RTSE, ROSE et **élément2**, l'opération liante **ExempleLiant1** et l'opération déliante **ExempleDéliant3**. L'initiateur de l'association peut appeler les opérations **Exemplé'opération3** et **Exemplé'opération4**. Celui qui répond à l'association n'est pas habilité à appeler des opérations (classe d'association 1).

contexte1 **APPLICATION-CONTEXT**
APPLICATION SERVICE ELEMENTS {aCSE, rTSE}
BIND **ExempleLiant1**
UNBIND **ExempleDéliant3**
REMOTE OPERATIONS {rOSE}
INITIATOR CONSUMER OF {élément2}
ABSTRACT SYNTAXES {{joint-iso-ccitt contrôle-d'association(2)
Syntaxeabstraite(1)
 apdus(0) version1(1)},
 identificateurd'objetDeSyntaxeAbstraite1}

::= {identificateurd'objetDuContexte1}

Le contexte d'application **contexte2** englobe les éléments du service d'application ACSE, ROSE, **élément2** et **élément3**, l'opération liante **ExempleLiant1** et l'opération déliante **ExempleDéliant3**. Celui qui répond à l'association peut appeler les opérations **Exemplé'opération3**, **Exemplé'opération4**, **Exemplé'opération12** et **Exemplé'opération51**. L'initiateur de l'association n'est habilité à appeler aucune opération (classe d'association 2).

contexte2 **APPLICATION-CONTEXT**
APPLICATION SERVICE ELEMENTS {aCSE}
BIND **ExempleLiant1**
UNBIND **ExempleDéliant3**
REMOTE OPERATIONS {rOSE}
INITIATOR CONSUMER OF {élément3}
RESPONDER CONSUMER OF {élément2}
ABSTRACT SYNTAXES {{ joint-iso-ccitt contrôle-d'association(2)
Syntaxeabstraite(1)
 apdus(0) version1(1)}
 identificateurd'objetDeSyntaxeAbstraite2}

::= {identificateurd'objetDuContexte2}

Le contexte d'application **contexte3** englobe les éléments du service d'application ACSE, RTSE, ROSE et **élément2**, l'opération liante **ExempleLiant1** et l'opération déliante **ExempleDéliant3**. Celui qui répond à l'association peut appeler les opérations **Exemplé'opération3** et **Exemplé'opération4**. L'initiateur de l'association n'est pas habilité à appeler des opérations (classe d'association 2).

contexte3 **APPLICATION-CONTEXT**
APPLICATION SERVICE ELEMENTS {aCSE, RTSE}
BIND **ExempleLiant1**
UNBIND **ExempleDéliant3**
REMOTE OPERATIONS {rOSE}
RESPONDER CONSUMER OF {élément2}
ABSTRACT SYNTAXES {**identificateurd'objetDeSyntaxeAbstraite3**}

::= {identificateurd'objetDuContexte3}

Le contexte d'application **contexte4** englobe les éléments du service d'application ACSE, ROSE et **élément2**, l'opération liante **ExempleLiant3** et l'opération déliante **ExempleDéliant3**. Le contexte d'application **contexte3** est symétrique en ce sens que l'initiateur comme le répondeur de l'association peuvent appeler les opérations **Exemplé'opération12** et **Exemplé'opération3** (classe d'association 3).

contexte4 **APPLICATION-CONTEXT**
APPLICATION SERVICE ELEMENTS {aCSE}
BIND **ExempleLiant3**
UNBIND **ExempleDéliant3**
REMOTE OPERATIONS {rOSE}
OPERATIONS OF {élément1}
ABSTRACT SYNTAXES {**identificateurd'objetDeSyntaxeAbstraite4**}

::= {identificateurd'objetDuContexte4}

Les deux derniers exemples se fondent sur l'hypothèse d'une seule syntaxe abstraite nommée.

B.6 Terminaison régulière des associations d'application

B.6.1 Introduction

La présente Recommandation définit cinq classes d'opération fondées sur le résultat qu'elles signalent et trois classes d'association fondées sur le critère suivant: les opérations peuvent être appelées par l'initiateur de l'association, par le répondeur à la demande d'association, ou par les deux. Le présent paragraphe définit les règles qui permettent d'assurer une terminaison régulière des associations d'application et des classes d'opération appelées sur ces associations.

B.6.2 Objectif

Les règles contenues dans le présent paragraphe visent à réaliser l'un des deux objectifs suivants, selon la situation:

- a) *objectif une fois exactement*: théoriquement, une entité d'application devrait pouvoir compter sur le fait que l'appel d'une opération provoque exactement une fois l'exécution de cette opération, c'est-à-dire qu'elle n'est ni exécutée plusieurs fois ni pas du tout exécutée;
- b) *objectif une fois au maximum*: dans certaines circonstances, il n'est pas possible d'atteindre l'objectif "une fois exactement". Un objectif inférieur, mais néanmoins utile, est que l'appel d'une opération en provoque l'exécution une fois au maximum, c'est-à-dire qu'elle n'est peut-être pas toujours exécutée, mais qu'elle n'est jamais exécutée deux fois.

B.6.3 Définitions des règles

Les **règles générales** suivantes s'appliquent dans toutes les circonstances:

- G.1 L'exécutant signale le résultat ou l'erreur de chaque opération confirmée sur la même association d'application à l'aide de laquelle l'opération a été appelée.
- G.2 L'initiateur ne doit pas terminer l'association d'application tant que toutes les opérations appelées n'ont pas été confirmées.

Les **règles spécifiques** suivantes s'appliquent dans certaines circonstances:

- S.1 Chaque fois qu'il utilise le service APPEL-RO, l'appelant doit fournir un ID-d'appel différent (à moins qu'il ne répète une tentative d'appel), même dans une séquence d'associations d'application. L'exécutant est ainsi en mesure de réaliser l'objectif "une fois au maximum" en supprimant les répétitions.
- S.2 Si l'exécutant rencontre un ID-d'appel répété dans le service APPEL-RO, il doit utiliser le service REFUS-RO PAR L'UTILISATEUR avec appel répété comme raison de refus. Cela aide à réaliser l'objectif une fois exactement.
- S.3 L'initiateur de l'association doit rejeter tout appel qu'il n'a pas exécuté avant de terminer l'association d'application.
- S.4 L'initiateur de l'association doit répondre à tout appel qu'il a exécuté avant de terminer l'association d'application.

B.6.4 Applications des règles

Les règles générales sont applicables en toutes circonstances. Les règles spécifiques régissent comme suit les associations d'application de classes d'association particulières et les opérations (appelées à l'aide de telles associations) de classes d'opération particulières:

- a) *Associations d'application de la classe d'association 1*: pour les opérations des classes d'opération 1 et 2, aucune règle spécifique ne s'applique. Pour les opérations des classes d'opération 3,4 et 5, les règles spécifiques S.1 et S.2 s'appliquent.
- b) *Associations d'application des classes d'association 2 et 3* : pour les classes d'opération 1 et 2, les règles spécifiques S.3 et S.4 s'appliquent. (Tout appel émis par le répondeur à la demande d'association, après que son initiateur a émis une terminaison, est perdu. Les refus de réponses peuvent également être perdus). Pour les opérations des classes d'opération 3,4 et 5, les règles spécifiques S.1 et S.2, S.3 et S.4 s'appliquent.

Dans le cas des protocoles qui ne contiennent que des opérations appartenant aux classes d'opération 1 ou 2, la seule contrainte imposée aux valeurs des ID-d'appel fournis par l'appelant au service APPEL-RO est que ces valeurs doivent être distinctes pendant toute la durée de l'association d'application.

Les entités d'application rendent univoques les ID-d'appel destinés à un appelant et transmis sur des associations d'application consécutives, en échangeant les adresses de présentation au moment de l'établissement de l'association d'application et, pour chaque adresse de présentation, en transformant les ID-d'appel en nombres entiers qui augmentent de façon monotone pendant un laps de temps suffisamment long.

Pour garantir qu'une opération des classes d'opération 3, 4 ou 5 a été exécutée exactement une fois, une entité d'application doit expliciter la raison de Refus d'appel répété en appelant l'opération à deux reprises (ou davantage) avec le même ID- d'appel. Autrement, l'objectif "une fois au maximum" est le seul à pouvoir être garanti, ce qui laisse à penser que, en moyenne, l'appelant ne se soucie pas de l'exécution d'une opération non confirmée.

B.6.5 *Observations*

Chaque opération de classes d'opération 1 ou 2 est exécutée une fois exactement.

L'utilité des opérations non confirmées ou des opérations confirmées sous conditions peut dépendre de l'application particulière. Les concepteurs de protocole devraient être avisés de ne définir que des opérations des classes d'opération 1 ou 2, à moins qu'il n'existe des conditions très particulières.