



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

T.435

(08/95)

TERMINAUX POUR LES SERVICES TÉLÉMATIQUES

**TRANSFERT ET MANIPULATION
DE DOCUMENTS – SERVICES ET
PROTOCOLES – DÉFINITION DU
SERVICE ABSTRAIT ET PROCÉDURES
POUR LA MANIPULATION DE DOCUMENT
AVEC CONFIRMATION**

Recommandation UIT-T T.435

(Antérieurement «Recommandation du CCITT»)

AVANT-PROPOS

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

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

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution n° 1 de la CMNT (Helsinki, 1^{er}-12 mars 1993).

La Recommandation UIT-T T.435, que l'on doit à la Commission d'études 8 (1993-1996) de l'UIT-T, a été approuvée le 11 août 1995 selon la procédure définie dans la Résolution n° 1 de la CMNT.

NOTE

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

© UIT 1996

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

TABLE DES MATIÈRES

		<i>Page</i>
1	Champ d'application.....	1
2	Références	2
3	Définitions	3
4	Abréviations	3
5	Conventions.....	4
6	Modèles abstraits.....	4
6.1	Environnement DTAM-DM.....	4
6.1.1	Client DTAM-DM.....	5
6.1.2	Serveur DTAM-DM.....	5
6.1.3	Accès DTAM-DM.....	5
6.1.4	Modèle d'information DTAM-DM.....	6
6.1.4.1	Identification des objets manipulables	6
6.1.4.2	Identification de documents complets	6
6.1.4.3	Identification de fragments documentaires.....	6
6.1.4.4	Identification des informations sur le contenu du document.....	7
6.2	Environnement combiné DTAM-DM/DTAM-TK	7
6.2.1	Client/serveur DTAM-DM/DTAM-TK	7
6.2.2	Accès dans l'environnement DTAM-DM/DTAM-TK	7
6.2.2.1	Accès DTAM-DM-SYM.....	7
6.2.2.2	Accès DTAM-TK.....	8
6.2.3	Utilisation d'un jeton d'application.....	8
6.3	Environnement combiné DTAM-DM/DFR.....	8
6.3.1	Le client DTAM-DM/DFR.....	10
6.3.2	Serveur DTAM-DM/DFR	10
6.3.3	Accès dans l'environnement DTAM-DM/DFR.....	10
7	Vue d'ensemble des services abstraits	10
7.1	Vue d'ensemble des services abstraits DTAM-DM	10
7.1.1	Service abstrait DM-DOCUMENT-OPEN (ouverture de document DM)	11
7.1.2	Service abstrait DM-DOCUMENT-SAVE (sauvegarde de document DM).....	11
7.1.3	Service abstrait DM-DOCUMENT-DISCARD (abandon de document DM)	11
7.1.4	Service abstrait DM-DOCUMENT-CLOSE (fermeture de document DM)	11
7.1.5	Service abstrait DM-DOCUMENT-LIST (listage de documents DM).....	11
7.1.6	Service abstrait DM-GET (lecture DM).....	11
7.1.7	Service abstrait DM-SEARCH (recherche DM)	11
7.1.8	Service abstrait DM-CREATE (création DM)	11
7.1.9	Service abstrait DM-DELETE (suppression DM).....	11
7.1.10	Service abstrait DM-MODIFY (modification DM)	11
7.1.11	Service abstrait DM-COPY (copie DM)	11
7.1.12	Service abstrait DM-MOVE (déplacement DM).....	12
7.1.13	Service abstrait DM-REPLACE (remplacement DM)	12
7.1.14	Service abstrait DM-RESERVE (réservation DM)	12
7.1.15	Service abstrait DM-UNRESERVE (déréservation DM)	12
7.1.16	Service abstrait DM-POINT (pointage DM)	12
7.1.17	Service abstrait DM-MACRO-CALL (appel de macro commande DM)	12
7.1.18	Service abstrait DM-GROUP-BEGIN (début de groupe DM).....	12
7.1.19	Service abstrait DM-GROUP-END (fin de groupe DM)	12

7.2	Vue d'ensemble des services abstraits d'échange de jeton DTAM-TK	12
7.2.1	Service abstrait TK-TOKEN-GIVE (cession de jeton)	12
7.2.2	Service abstrait TK-TOKEN-PLEASE (demande de jeton TK)	12
8	Etablissement d'une association d'application	13
8.1	Paramètres de rattachement et de détachement abstraits	13
8.1.1	Paramètres de rattachement abstrait	13
8.1.1.1	Paramètres d'argument de rattachement	13
8.1.1.2	Paramètres du résultat de rattachement	16
8.1.1.3	Paramètres d'erreur de rattachement.....	16
8.1.2	Paramètres ABSTRACT-UNBIND du détachement abstrait.....	17
8.2	Utilisation du service A-ABORT (Annulation d'association)	17
9	Définition des services abstraits	17
9.1	Types de données importés	17
9.2	Définition des types de données communs aux services abstraits	18
9.2.1	DocumentId – Identificateur de document	18
9.2.2	ManipulationObject – Objet manipulable	18
9.2.3	ObjectValue – Valeur d'objet	18
9.2.4	Oda-Expression – Expression ODA	18
9.2.5	ObjectPosition – Position d'objet	19
9.2.6	ObjectContent – Contenu d'objet	19
9.2.7	AssignedId – Identificateur affecté	19
9.2.8	NonPermanentIdentifier – Identificateur non permanent	19
9.3	Définition des services abstraits DTAM-DM	19
9.3.1	Service abstrait DM-DOCUMENT-OPEN (ouverture de document).....	20
9.3.1.1	Arguments	20
9.3.1.2	Résultat.....	20
9.3.1.3	Erreurs	20
9.3.2	Service abstrait DM-DOCUMENT-SAVE (sauvegarde).....	20
9.3.2.1	Arguments	21
9.3.2.2	Résultat.....	21
9.3.2.3	Erreurs	21
9.3.3	Service abstrait DM-DOCUMENT-DISCARD (abandon)	21
9.3.3.1	Arguments	21
9.3.3.2	Résultat.....	21
9.3.3.3	Erreurs	22
9.3.4	Service abstrait DM-DOCUMENT-CLOSE (fermeture).....	22
9.3.4.1	Arguments	22
9.3.4.2	Résultat.....	22
9.3.4.3	Erreurs	22
9.3.5	Service abstrait DM-DOCUMENT-LIST (listage)	22
9.3.5.1	Arguments	23
9.3.5.2	Résultat.....	23
9.3.5.3	Erreurs	23
9.3.6	Service abstrait DM-GET (lecture)	23
9.3.6.1	Arguments	24
9.3.6.2	Résultat.....	24
9.3.6.3	Erreurs	24
9.3.7	Service abstrait DM-SEARCH (recherche).....	24
9.3.7.1	Arguments	24
9.3.7.2	Résultat.....	24
9.3.7.3	Erreurs	25
9.3.8	Service abstrait DM-CREATE (création).....	25
9.3.8.1	Arguments	25
9.3.8.2	Résultat.....	25
9.3.8.3	Erreurs	25

	<i>Page</i>	
9.3.9	Service abstrait DM-DELETE (suppression)	26
	9.3.9.1 Arguments	26
	9.3.9.2 Résultat.....	26
	9.3.9.3 Erreurs	26
9.3.10	Service abstrait DM-MODIFY (modification).....	26
	9.3.10.1 Arguments	27
	9.3.10.2 Résultat.....	27
	9.3.10.3 Erreurs	27
9.3.11	Service abstrait DM-COPY (recopie)	27
	9.3.11.1 Arguments	28
	9.3.11.2 Résultat.....	28
	9.3.11.3 Erreurs	28
9.3.12	Service abstrait DM-MOVE (déplacement).....	28
	9.3.12.1 Arguments	29
	9.3.12.2 Résultat.....	29
	9.3.12.3 Erreurs	29
9.3.13	Service abstrait DM-REPLACE (remplacement).....	29
	9.3.13.1 Arguments	30
	9.3.13.2 Résultat.....	30
	9.3.13.3 Erreurs	30
9.3.14	Service abstrait DM-RESERVE (réservation)	30
	9.3.14.1 Arguments	31
	9.3.14.2 Résultat.....	31
	9.3.14.3 Erreurs	31
9.3.15	Service abstrait DM-UNRESERVE (déréservation).....	31
	9.3.15.1 Arguments	32
	9.3.15.2 Résultat.....	32
	9.3.15.3 Erreurs	32
9.3.16	Service abstrait DM-POINT (pointage)	32
	9.3.16.1 Arguments	32
	9.3.16.2 Résultat.....	32
	9.3.16.3 Erreurs	32
9.3.17	Service abstrait DM-MACRO-CALL (appel de macrocommande).....	32
	9.3.17.1 Arguments	33
	9.3.17.2 Résultat.....	33
	9.3.17.3 Erreurs	33
9.3.18	Service abstrait DM-GROUP-BEGIN (début de groupe)	33
	9.3.18.1 Arguments	33
	9.3.18.2 Résultat.....	33
	9.3.18.3 Erreurs	33
9.3.19	Service abstrait DM-GROUP-END (fin de groupe)	33
	9.3.19.1 Arguments	34
	9.3.19.2 Résultat.....	34
	9.3.19.3 Erreurs	34
9.4	Définitions des erreurs abstraites du service DTAM-DM.....	34
	9.4.1 Erreur d'accès DM-access-error	34
	9.4.2 Erreur d'attribut DM-attribute-error	35
	9.4.3 Erreur de mise à jour DM-update-error.....	35
	9.4.4 Erreur de document DM-document-error.....	36
	9.4.5 Erreur de fragment DM-fragment-error	36
	9.4.6 Erreur de liste DM-list-error.....	37
	9.4.7 Erreur de réservation DM-reservation-error.....	37
	9.4.8 Erreur de service DM-service-error	37
	9.4.9 Priorité des erreurs	38
9.5	Définition des services abstraits DTAM-TK	38
	9.5.1 Service abstrait TK-TOKEN-GIVE (cession de jeton).....	38
	9.5.1.1 Arguments	38
	9.5.1.2 Résultat.....	38
	9.5.1.3 Erreurs	38

	<i>Page</i>
9.5.2 Service abstrait TK-TOKEN-PLEASE (demande de jeton)	38
9.5.2.1 Arguments	39
9.5.2.2 Résultat.....	39
9.5.2.3 Erreurs	39
9.6 Définitions des erreurs abstraites du service DTAM-TK.....	39
9.6.1 L'erreur TK-service-error	39
Annexe A – Affectation formelle des identificateurs d'objet.....	40
Annexe B – Définition formelle des services abstraits DTAM-DM et DTAM-TK	41
Appendice I – Procédures de base pour l'utilisation des services abstraits DTAM-DM	54
I.1 Etablissement d'une association DTAM	54
I.2 Ouverture de document.....	54
I.3 Manipulation de document.....	54
I.4 Sauvegarde de document.....	54
I.5 Abandon de document	54
I.6 Fermeture de document.....	54
I.7 Terminaison d'association DTAM	54
Appendice II – Procédures de base pour l'utilisation combinée des services abstraits DTAM-DM et DFR	56
II.1 Etablissement d'association DTAM-DM/DFR.....	56
II.2 Archivage et consultation de documents (utilisation des services abstraits DFR)	56
II.3 Manipulation de document (utilisation des services abstraits DTAM-DM)	56
II.4 Terminaison de l'association DTAM-DM/DFR.....	56

RÉSUMÉ

La présente Recommandation définit les services abstraits d'accès à distance aux documents et de présentation, d'édition et de transfert de documents. Elle contient deux nouveaux éléments de service d'application pour la manipulation interactive de documents avec confirmation. Ces éléments sont (par conception) compatibles avec une utilisation intégrée dans des applications définies conformément au modèle d'applications bureautiques décentralisées (DOAM) (*distributed office application model*). Ces deux éléments de service d'application sont le DTAM-TM et le DTAM-TK.

L'élément de service DTAM-DM permet d'effectuer tout type de manipulation à distance de documents complets ou fragmentaires. Il est conçu pour être associé à l'interface abstraite ODA pour la manipulation de documents (Recommandation T.413). Si nécessaire, l'élément DTAM-DM permet d'utiliser pleinement les fonctions d'archivage et de consultation de documents définies dans ISO/CEI 10166 (DFR) dans le cadre de l'association en mode DTAM. L'élément de service DTAM-TK fournit une fonction de jeton d'application pour permettre à deux utilisateurs ou plus de communiquer en conférence.

**TRANSFERT ET MANIPULATION DE DOCUMENTS –
SERVICES ET PROTOCOLES – DÉFINITION DU SERVICE ABSTRAIT
ET PROCÉDURES POUR LA MANIPULATION
DE DOCUMENT AVEC CONFIRMATION**

(Genève, 1995)

1 Champ d'application

L'UIT-T (anciennement CCITT) a spécifié le modèle de référence à sept couches pour l'interconnexion des systèmes ouverts (OSI), Recommandation X.200. L'objectif global de ce modèle OSI est de définir des normes permettant aux systèmes de communiquer entre eux.

La couche application de l'OSI comporte les éléments de service d'application (ASE) (*application service elements*) qui sont destinés à fournir aux applications des services spécifiques.

Un de ces éléments de service d'application est le transfert et la manipulation de documents (DTAM). L'élément DTAM est conçu pour le traitement des documents, c'est-à-dire le transfert de documents ainsi que l'accès et la manipulation de fragments de documents entre entités d'application dans le cadre de systèmes télématiques ouverts. L'élément DTAM est défini dans les Recommandations de la série T.430. La Recommandation T.431 introduit et décrit les principes généraux de l'élément DTAM. Les Recommandations T.432 et T.433 définissent l'élément de service d'application pour le transfert brut de documents et pour la manipulation de document sans confirmation.

La présente Recommandation, conjointement avec les spécifications de protocole contenues dans la Recommandation T.436, définit deux nouveaux éléments de service d'application pour la manipulation de document interactive avec confirmation. Ces éléments sont alignés (et prévus) pour une utilisation intégrée dans des applications définies conformément au modèle d'applications bureautiques réparties (DOAM) (*distributed office application model*), ISO/CEI 10031.

Ces deux éléments de service d'application, définis dans la présente Recommandation, sont les suivants:

- 1) DTAM-DM – Élément de service d'application DTAM de manipulation de document (DM) (*document manipulation*) améliorée; et
- 2) DTAM-TK – Élément de service d'application DTAM d'échange de jetons (*token*) d'application.

Les services abstraits de l'élément DTAM-DM offrent des fonctions permettant d'effectuer des opérations distantes sur tout type de documents ou de fragments de document. Si les documents considérés sont des informations structurées conformément à l'architecture de document ouverte (ODA) (*open document architecture*), les fragments de document représentent des constituants ODA, Recommandations des séries T.410 et T.420. Ses autres types de documents peuvent comporter d'autres moyens d'identification des fragments de document. Les documents et fragments de document sont considérés comme des objets manipulables dans le cadre de la présente Recommandation.

L'élément DTAM-DM peut se combiner à l'élément de service d'archivage et de consultation de documents (DFR), ISO/CEI 10166, en ce qui concerne les applications d'accès et de manipulation des documents. Les deux éléments DFR et DTAM-DM sont des applications qui permettent d'effectuer des opérations distantes sur tous types de documents. Alors que l'élément DTAM-DM manipule des fragments de documents, l'élément DFR porte sur l'archivage et la consultation de documents entiers. Lorsqu'il est nécessaire de manipuler à la fois des documents entiers et fragmentaires conservés en mémoire documentaire, on combinera l'utilisation des services abstraits DTAM-DM et DFR.

L'élément de service DTAM-TK assure le traitement d'un jeton d'application, qui permet de structurer le dialogue lorsque deux utilisateurs ou plus participent à des applications de type conférence. L'élément DTAM-TK est prévu pour être utilisé avec d'autres éléments de service d'application, par exemple avec l'élément DTAM-DM.

Les services abstraits DTAM-DM et DTAM-TK sont fournis conjointement avec l'élément de service de contrôle d'association (ACSE) (*association control service element*), Recommandation X.217, l'élément de service d'opérations distantes (ROSE) (*remote operation service element*) Recommandation X.219, et le service de couche présentation, Recommandation X.216.

La présente Recommandation ne spécifie pas de mises en œuvre ou de produits particuliers. Elle ne décrit pas non plus la mise en œuvre d'entités ou d'interfaces à l'intérieur d'un système télématique.

2 Références

Les Recommandations et autres références suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables de la présente Recommandation. Au moment de la publication, les éditions indiquées étaient en vigueur. Toute Recommandation ou autre référence est sujette à révision; tous les utilisateurs de la présente Recommandation sont donc invités à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et autres références indiquées ci-après. Une liste des Recommandations UIT-T en vigueur est publiée régulièrement.

- Recommandation UIT-T T.411 (1993) | ISO/CEI 8613-1:1994, *Technologie de l'information – Architecture ouverte des documents (ODA) et format d'échange – Introduction et principes généraux.*
- Recommandation UIT-T T.412 (1993) | ISO/CEI 8613-2:1995, *Technologie de l'information – Architecture ouverte de document et format d'échange – Structure des documents.*
- Recommandation UIT-T T.413 (1994) | ISO/CEI 8613-3 (1994), *Technologie de l'information – Architecture ouverte des documents (ODA) et format d'échange – Interface abstraite pour la manipulation de documents à architecture ODA.*
- Recommandation UIT-T T.414 (1993) | ISO/CEI 8613-4:1994, *Technologie de l'information – Architecture de document ouverte et format de transfert – Profil de document.*
- Recommandation UIT-T T.415 (1993) | ISO/CEI 8613-5:1994, *Technologie de l'information – Architecture de document ouverte et format de transfert – Format ouvert d'échange de documents.*
- Recommandation UIT-T T.416 (1993) | ISO/CEI 8613-6:1994, *Technologies de l'information – Architecture de document ouverte (ODA) et format de transfert – Architectures de contenu de type caractères.*
- Recommandation UIT-T T.417 (1993) | ISO/CEI 8613-7:1994, *Technologie de l'information – Architecture de document ouverte et format de transfert – Architecture de contenu graphique en points.*
- Recommandation UIT-T T.418 (1993) | ISO/CEI 8613-8:1994, *Technologie de l'information – Architecture de document ouverte et format de transfert – Architecture de contenu graphique géométrique.*
- Recommandation UIT-T T.422 (1995) | ISO/CEI 8613-12 (1994), *Technologie de l'information – Architecture de document ouverte et format de transfert – Identification des fragments de documents.*
- Recommandation T.431 du CCITT (1992), *Transfert et manipulation de documents – Services et protocoles – Introduction et principes généraux.*
- Recommandation T.432 du CCITT (1992), *Transfert et manipulation de documents – Services et protocoles – Définition du service.*
- Recommandation T.433 du CCITT (1992), *Transfert et manipulation de documents – Services et protocoles – Spécification de protocole.*
- Recommandation T.434 du CCITT (1992), *Format de transfert de fichiers binaires pour les services télématiques.*
- Recommandation UIT-T T.436 (1995), *Transfert et manipulation de documents – Services et protocoles – Spécification de protocoles de manipulation confirmée de documents.*
- Recommandation X.200 du CCITT (1988), *Technologie de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: le modèle de référence de base.*
- Recommandation X.208 du CCITT (1988), *Spécification de la syntaxe abstraite numéro un (ASN.1).*
- Recommandation X.209 du CCITT (1988), *Spécification des règles de codage de base pour la notation de syntaxe abstraite numéro un (ASN.1).*
- Recommandation X.215 du CCITT (1988), *Technologie de l'information – Interconnexion des systèmes ouverts – Définition du service de session.*
- Recommandation X.216 du CCITT (1988), *Technologie de l'information – Interconnexion des systèmes ouverts – Définition du service de présentation.*
- Recommandation X.217 du CCITT (1992), *Technologie de l'information – Interconnexion des systèmes ouverts – Définition de service applicable à l'élément de service de contrôle d'association.*
- Recommandation X.219 du CCITT (1988), *Opérations distantes: modèle, notation et définition du service.*

- ISO/CEI 10021-3: 1990, *Technologies de l'information – Communication de texte – Systèmes d'échange de texte en mode message – Partie 3: Conventions relatives à la définition de service abstrait.*
- ISO/CEI 10031-1: 1991, *Technologies de l'information – Bureautique – Modèle d'application pour bureau distribué – Partie 1: Modèle général.*
- ISO/CEI 10166-1: 1991, *Technologies de l'information – Bureautique – Classement et récupération de documents – Partie 1: Procédures et définition du service abstrait.*
- ISO/CEI 10166-2: 1991, *Technologies de l'information – Bureautique – Classement et récupération de documents – Partie 2: Spécification du protocole.*

3 Définitions

Pour les besoins de la présente Recommandation les définitions de la Recommandation T.431 sont utilisées.

De plus, la présente Recommandation définit les termes suivants:

- 3.1 client DTAM-DM:** Objet de l'environnement DTAM-DM qui consomme les services abstraits DTAM-DM fournis par un serveur DTAM-DM.
- 3.2 serveur DTAM-DM:** Objet de l'environnement DTAM-DM qui fournit les services abstraits DTAM-DM.
- 3.3 accès DTAM-DM:** Connexion entre client DTAM-DM et serveur DTAM-DM qui fournit les services abstraits, c'est-à-dire le moyen de communiquer dans l'environnement DTAM-DM.
- 3.4 protocole DTAM-DM:** Ensemble de règles régissant la réalisation des services abstraits de l'accès DTAM-DM. Cet ensemble de règles est normalement décrit dans une spécification de protocole distincte.
- 3.5 environnement DTAM-DM:** Ensemble d'objets communicants raccordés à un accès DTAM-DM. La procédure de communication suit les règles déterminées par le protocole DTAM-DM.
- 3.6 client/serveur DTAM-TK:** Objet qui exécute les services abstraits DTAM-TK.
- 3.7 client/serveur DTAM-DM/DTAM-TK:** Objet de l'environnement DTAM-DM/DTAM-TK qui exécute les services abstraits DTAM-DM et DTAM-TK.
- 3.8 accès DTAM-TK:** Connexion entre entités d'application qui fournit les services abstraits DTAM-TK.
- 3.9 protocole DTAM-TK:** Ensemble de règles régissant la réalisation des services abstraits d'accès à l'élément DTAM-TK. Cet ensemble de règles est normalement décrit dans une spécification de protocole distincte.
- 3.10 environnement DTAM-DM/DTAM-TK:** Application de communication qui permet l'utilisation combinée des services abstraits DTAM-DM et DTAM-TK.
- 3.11 client DTAM-DM/DFR:** Utilisateur des services abstraits fournis par un serveur DTAM-DM/DFR.
- 3.12 serveur DTAM-DM/DFR:** Partie d'un environnement DTAM-DM/DFR qui permet l'utilisation combinée des services abstraits DTAM-DM et DFR.
- 3.13 environnement DTAM-DM/DFR:** Application de communication qui permet l'utilisation combinée des services abstraits DTAM-DM et DFR.

4 Abréviations

Pour les besoins de la présente Recommandation, les abréviations définies dans la Recommandation T.431 sont utilisées.

La présente Recommandation utilise en outre les abréviations suivantes:

DM	Manipulation de document (<i>document manipulation</i>)
TK	Transaction de jeton (<i>token exchange</i>)
DFR	Archivage et consultation de document (<i>document filing and retrieval</i>)
MO	Objet manipulable (<i>manipulation object</i>)
UPI	Identificateur permanent unique (<i>unique-permanent-identifier</i>)

5 Conventions

La présente Recommandation spécifie les services abstraits DTAM-DM et DTAM-TK au moyen de la notation de syntaxe abstraite numéro un, Recommandations X.208 et X.209 et des conventions ISO/CEI 10021 relatives à la définition d'un service abstrait.

L'applicabilité d'un service ou d'un paramètre de service sera exprimée par l'une des valeurs suivantes:

- | | |
|--------|---|
| <vide> | Non applicable. |
| M | Obligatoire. L'item sera présent dans tous les cas. |
| U | Option d'utilisateur. La présence de l'item dépend des capacités dont dispose l'utilisateur. L'item peut être absent. |
| C | Présence conditionnelle. La présence de l'item dépend de conditions indiquées dans la présente Recommandation. |
| FS | Pour complément d'étude. L'applicabilité n'est pas encore clairement fixée. |

NOTE – Les spécifications ASN.1 de la présente Recommandation sont écrites avec la convention d'étiquetage implicite «IMPLICIT TAGS», c'est-à-dire avec l'omission systématique lors du codage de toutes les étiquettes imbriquées inutiles, en particulier celles qui se déduisent par des étiquettes propres au contexte.

6 Modèles abstraits

Un environnement de client-serveur est une application de communication asymétrique qui met en jeu deux objets: le client et le serveur. Ces deux objets sont des entités d'application raccordées par un accès. Le serveur attend d'être consulté par le client pour lui fournir un service.

L'accès fournit les services abstraits, c'est-à-dire les moyens de communiquer dans cet environnement. L'ensemble des règles qui régissent la réalisation des services abstraits de l'accès est normalement décrit dans une spécification de protocole distincte.

La présente Recommandation décrit les environnements abstraits de communication suivants, au moyen de la notation du modèle client-serveur:

- Environnement DTAM-DM;
- Environnement combiné DTAM-DM/DTAM-TK;
- Environnement combiné DTAM-DM/DFR.

L'environnement combiné DTAM-DM offre des services abstraits permettant d'effectuer des manipulations sur des morceaux internes au document, appelés fragments de document.

L'environnement combiné DTAM-DM/DTAM-TK offre tous les services abstraits de l'environnement DTAM-DM plus la possibilité pour le client et le serveur d'échanger un jeton. Cet environnement combiné peut servir à des applications de communication en mode conférence.

L'environnement combiné DTAM-DM/DFR offre tous les services abstraits de l'environnement DTAM-DM plus la possibilité de gestion d'une base documentaire. Les services abstraits fournis par l'élément DFR permettent de traiter les documents pris dans leur entier et non pas des fragments de document.

Ces environnements de communication abstraits seront décrits dans les paragraphes suivants.

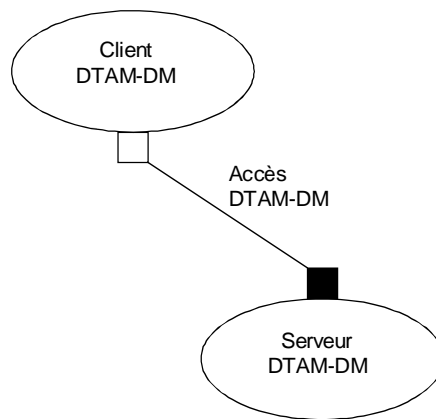
Les spécifications de protocole correspondantes figurent dans la Recommandation T.436.

6.1 Environnement DTAM-DM

Ce paragraphe présente le modèle fonctionnel abstrait de l'environnement DTAM-DM. Celui-ci permet de manipuler des fragments de document.

L'environnement DTAM-DM comporte deux objets: le client DTAM-DM et le serveur DTAM-DM. Ces deux objets sont reliés par l'accès DTAM-DM.

La Figure 1 représente l'environnement DTAM-DM.



T0817970-94/d01

FIGURE 1/T.435
Environnement DTAM-DM

6.1.1 Client DTAM-DM

Le client DTAM-DM est modélisé sous la forme d'un objet atomique. Le client DTAM-DM consomme les services abstraits de l'accès DTAM-DM fournis par le serveur DTAM-DM.

La définition formelle de l'objet client DTAM-DM est la suivante:

```

dtam-dm-client    OBJECT
                    PORTS { dtam-dm-port [ C ] }
                    ::= id-ot-dtam-dm-client

```

6.1.2 Serveur DTAM-DM

Le serveur DTAM-DM est modélisé sous la forme d'un objet distinct. Il fournit les services abstraits de l'accès DTAM-DM au client DTAM-DM.

La définition formelle de l'objet serveur DTAM-DM est la suivante:

```

dtam-dm-server  OBJECT
                    PORTS { dtam-dm-port [ S ] }
                    ::= id-ot-dtam-dm-server

```

6.1.3 Accès DTAM-DM

L'accès DTAM-DM permet aux objets de communiquer dans l'environnement DTAM-DM.

L'accès DTAM-DM est défini comme suit:

```

dtam-dm-port    PORT
                    CONSUMER INVOKES
                    {
                        DM-DOCUMENT-OPEN,
                        DM-DOCUMENT-SAVE,
                        DM-DOCUMENT-DISCARD,
                        DM-DOCUMENT-CLOSE,
                        DM-DOCUMENT-LIST,
                        DM-GET,
                        DM-SEARCH,
                        DM-CREATE,
                        DM-DELETE,
                        DM-MODIFY,
                        DM-COPY,
                        DM-MOVE,

```

```

DM-REPLACE,
DM-RESERVE,
DM-UNRESERVE,
DM-POINT,
DM-MACRO-CALL,
DM-GROUP-BEGIN,
DM-GROUP-END
}
SUPPLIER INVOKES
{
}
::= id-pt-dtam-dm-port

```

6.1.4 Modèle d'information DTAM-DM

Un serveur DTAM-DM offre des opérations de client DTAM-DM applicables aux documents ainsi qu'à l'environnement dont dépendent les capacités de manipulation de document. Les documents, qui sont des objets sous-structurés de type manipulable, prendront par exemple la forme suivante:

- documents à architecture ODA, sous-structurés en plusieurs types d'objets, de classes, de styles, de contenus, etc.;
- contraintes propres à des documents ODA, définies dans des profils d'application de document (DAP) (*document application profiles*). Ces contraintes permettent de traiter différentes caractéristiques sémantiques, telles que des segments, des alinéas, des notes de bas de page, des sommaires, etc.;
- documents SGML, sous-structurés en régions délimitées par des étiquettes;
- tous autres types de données de forme structurée (y compris les formats non ouverts et les données multimédias).

L'environnement de manipulation de document consiste à déterminer les capacités de manipulation d'un document (par exemple un profil d'opérations) et les capacités nécessaires pour traiter les documents (par exemple des profils particuliers, des jeux de caractères donnés ou des caractéristiques secondaires).

6.1.4.1 Identification des objets manipulables

Le traitement de documents par élément DTAM comporte les opérations nécessaires pour traiter un document préalablement identifié à différentes fins, comme le manipuler ou le lire partiellement. Ces opérations s'inscrivent dans l'un des trois groupes suivants:

- traitement de document complet;
- traitement de fragments documentaires structurés;
- traitement d'informations relatives au contenu d'un document structuré.

6.1.4.2 Identification de documents complets

L'identification du document a lieu en début et en fin de manipulation. Pour identifier un document qui fera l'objet d'une manipulation ultérieure ou pour sauvegarder, abandonner ou clore une manipulation, on dispose des services abstraits d'ouverture de document, de sauvegarde de document, d'abandon de document et de fermeture de document. Un service abstrait de listage simple est prévu pour les applications qui ne disposent pas des capacités d'archivage et consultation documentaires.

Plusieurs documents peuvent être ouverts en même temps. Dans ce cas, chaque opération de manipulation identifiera le document auquel elle s'applique.

Les documents qui peuvent être identifiés par les services abstraits d'ouverture, de sauvegarde, d'abandon, de listage ou de fermeture correspondent aux différents types de documents normalisés ou de données identifiables.

Le mécanisme d'identification peut s'appuyer, par exemple, sur le concept d'identificateur permanent unique (UPI) utilisé en archivage DFR, ISO/CEI 10166.

6.1.4.3 Identification de fragments documentaires

L'identification des fragments de document a lieu au cours des opérations de manipulation de document. Les opérations qui font intervenir cette identification seront donc celles de lecture (*get*), de recherche (*search*), de réservation (*reserve*), de déréservation (*unreserve*), de création (*create*), de suppression (*delete*), de modification (*modify*), de remplacement (*replace*), de déplacement (*move*), d'appel de macrocommande (*macro-call*) et de pointage (*point*).

On peut identifier les fragments d'un document structuré au moyen du mécanisme d'identification propre au type d'information structurée correspondant, par exemple le modèle de localisation des documents ODA , Recommandation T.413.

Une deuxième méthode d'identification, faisant intervenir un paramètre distinct, consiste à repérer un fragment de document au moyen d'une expression permettant de cerner le ou les fragments à manipuler.

6.1.4.4 Identification des informations sur le contenu du document

Le contenu des documents peut être identifié par l'opération de recherche. La manipulation subséquente de parties non structurées d'un document appelle un complément d'étude.

6.2 Environnement combiné DTAM-DM/DTAM-TK

En ce qui concerne la manipulation des documents répartis, il a été établi qu'un système OSI aura à jouer aussi bien le rôle du client manipulant les documents mémorisés sur d'autres serveurs, que celui du serveur pour les documents stockés dans son propre système OSI. C'est pourquoi les services abstraits DTAM-DM/DTAM-TK ont été définis de manière symétrique, permettant ainsi à chaque entité d'être tant client que serveur. Il existe toutefois des dépendances qui déterminent le rôle (client ou serveur) d'une entité à un moment donné.

L'environnement combiné DTAM-DM/DTAM-TK comporte un type d'objet: le client/serveur DTAM-DM/DTAM-TK. Les objets de ce type sont reliés entre eux par deux accès: l'accès DTAM-DM-SYM et l'accès DTAM-TK. Voir la Figure 2.

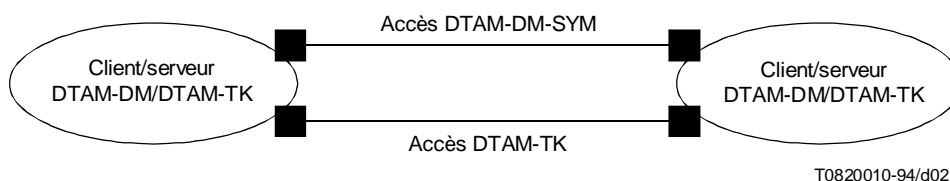


FIGURE 2/T.435

Environnement combiné DTAM-DM/DTAM-TK

6.2.1 Client/serveur DTAM-DM/DTAM-TK

Le client/serveur DTAM-DM/DTAM-TK est modélisé sous la forme d'un objet atomique. Il prend en charge les opérations abstraites sur les accès DTAM-DM-SYM et DTAM-TK .

La définition formelle de l'objet client/serveur DTAM-DM/DTAM-TK est la suivante:

```

dtam-dm-tk-client-server    OBJECT
                            PORTS { dtam-dm-sym-port, dtam-tk-port }
                            ::= id-ot-dtam-dm-tk-client-server

```

6.2.2 Accès dans l'environnement DTAM-DM/DTAM-TK

Les accès dans l'environnement DTAM-DM/DTAM-TK sont les accès DTAM-DM-SYM et DTAM-TK. L'accès DTAM-DM-SYM prend en charge les mêmes opérations abstraites que l'accès DTAM-DM, mais d'une manière symétrique. Les accès DTAM-DM-SYM et DTAM-TK seront définis dans le paragraphe suivant.

6.2.2.1 Accès DTAM-DM-SYM

L'accès DTAM-DM-SYM autorise un client/serveur à manipuler des fragments de documents stockés dans d'autres bases documentaires et à effectuer des manipulations dans sa propre base.

L'accès DTAM-DM-SYM est défini de la manière suivante:

```
dtam-dm-sym-port      PORT  
  ABSTRACT OPERATIONS {  
    DM-DOCUMENT-OPEN,  
    DM-DOCUMENT-SAVE,  
    DM-DOCUMENT-DISCARD,  
    DM-DOCUMENT-CLOSE,  
    DM-DOCUMENT-LIST,  
    DM-GET,  
    DM-SEARCH,  
    DM-CREATE,  
    DM-DELETE,  
    DM-MODIFY,  
    DM-COPY,  
    DM-MOVE,  
    DM-REPLACE,  
    DM-RESERVE,  
    DM-UNRESERVE,  
    DM-POINT,  
    DM-MACRO-CALL,  
    DM-GROUP-BEGIN,  
    DM-GROUP-END }  
  ::= id-pt-dtam-dm-sym-port
```

6.2.2.2 Accès DTAM-TK

L'accès DTAM-TK autorise le client/serveur DTAM-DM/DTAM-TK à échanger des jetons d'application qui déterminent à tout moment son rôle client ou serveur. Le paragraphe 6.2.3 spécifie l'utilisation des jetons d'application.

L'accès DTAM-TK est défini comme suit:

```
dtam-tk-port        PORT  
  ABSTRACT OPERATIONS {  
    TK-TOKEN-GIVE,  
    TK-TOKEN-PLEASE }  
  ::= id-pt-dtam-tk-port
```

6.2.3 Utilisation d'un jeton d'application

L'échange d'un jeton d'application entre client et serveur entraîne l'échange des rôles tenus par les entités d'application dans cet environnement. La Figure 3 décrit cet échange.

L'utilisateur 1 et l'utilisateur 2 sont deux entités d'application qui sont en mesure de jouer alternativement les rôles de client et de serveur.

Dans la configuration de départ, on suppose que l'utilisateur 1 possède le jeton d'application. Il joue le rôle de client, tandis que l'utilisateur 2 joue le rôle de serveur. Lorsque l'utilisateur 1 exécute le service abstrait TK-TOKEN-GIVE du protocole DTAM-TK, le jeton d'application est remis à l'utilisateur 2. L'utilisateur 1 passe alors à l'état «serveur», tandis que l'utilisateur 2 passe en mode de fonctionnement «client» .

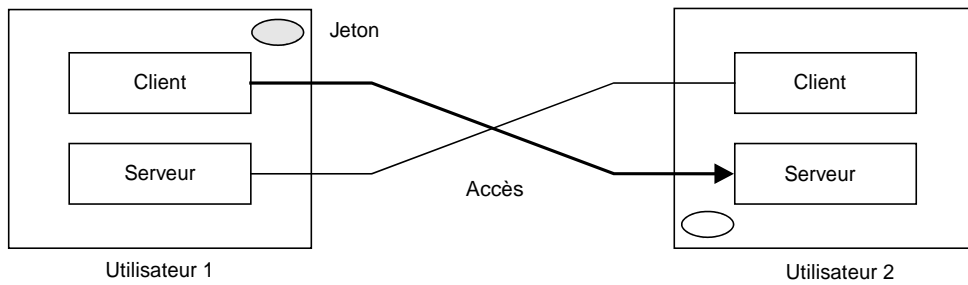
Le serveur en titre peut demander au client de lui passer le jeton d'application au moyen du service abstrait TK-TOKEN-PLEASE.

6.3 Environnement combiné DTAM-DM/DFR

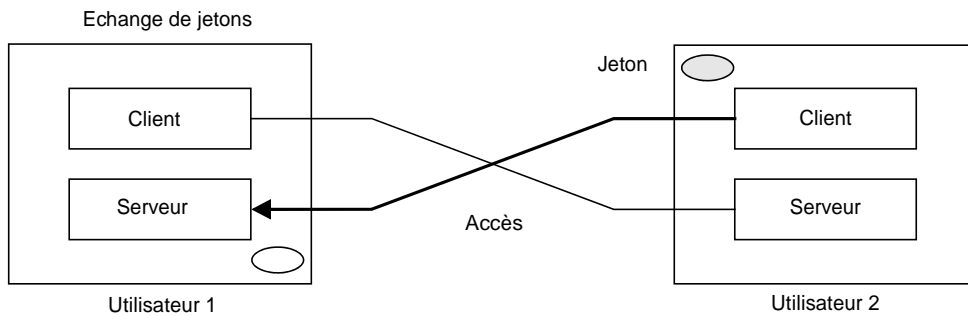
Le présent paragraphe décrit un modèle fonctionnel abstrait de l'environnement combiné DTAM-DM/DFR. Cet environnement fournit tous les services abstraits de l'environnement DTAM-DM, plus la possibilité de gérer une base documentaire.

L'environnement DTAM-DM/DFR comporte deux objets, le client DTAM-DM/DFR et le serveur DTAM-DM/DFR. Ces deux objets sont reliés par deux accès: l'accès DTAM-DM et l'accès DFR.

La Figure 4 décrit l'environnement combiné DTAM-DM/DFR.



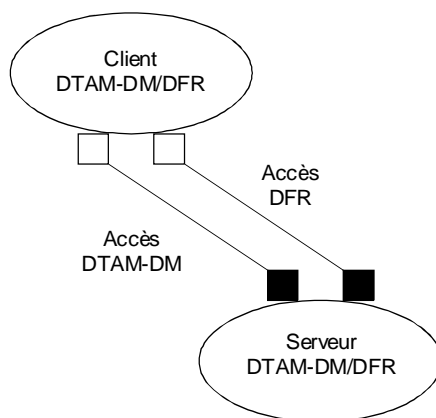
a) Configuration de départ



b) Configuration finale

T0820020-94/d03

FIGURE 3/T.435
Utilisation d'un jeton d'application



T0817980-94/d04

FIGURE 4/T.435
Environnement combiné DTAM-DM/DFR

6.3.1 Le client DTAM-DM/DFR

Le client DTAM-DM/DFR est modélisé sous la forme d'un objet atomique. Il consomme les services abstraits de l'accès DTAM-DM et de l'accès DFR fournis par le serveur DTAM-DM/DFR.

La définition formelle de l'objet client DTAM-DM/DFR est la suivante:

```
dtam-dm-dfr-client OBJECT
PORTS { dtam-dm-port [ C ], dfr-port [ C ] }
::= id-ot-dtam-dm-dfr-client
```

6.3.2 Serveur DTAM-DM/DFR

Le serveur DTAM-DM/DFR est modélisé sous la forme d'un objet distinct. Il fournit les services abstraits de l'accès DTAM-DM et de l'accès DFR au client DTAM-DM/DFR.

La définition formelle de l'objet serveur DTAM-DM/DFR est la suivante:

```
dtam-dm-dfr-server OBJECT
PORTS { dtam-dm-port [ S ], dfr-port [ S ] }
::= id-ot-dtam-dm-dfr-server
```

6.3.3 Accès dans l'environnement DTAM-DM/DFR

Les accès dans l'environnement DTAM-DM/DFR sont l'accès DTAM-DM et l'accès DFR. L'accès DTAM-DM a déjà été défini ci-dessus. L'accès DFR est défini dans ISO/CEI 10166 et importé par la présente Recommandation.

NOTE – L'utilisation du service abstrait DM-Document-LIST n'est pas autorisée dans l'environnement DTAM-DM/DFR. On utilisera à leur place les services abstraits DFR correspondants.

7 Vue d'ensemble des services abstraits

Cet article présente les services abstraits DTAM-DM et DTAM-TK. La définition formelle de ces services abstraits est contenue dans l'Annexe B «Définition formelle des services abstraits».

7.1 Vue d'ensemble des services abstraits DTAM-DM

La présente Recommandation définit les services abstraits DTAM-DM suivants:

- DM-DOCUMENT-OPEN;
- DM-DOCUMENT-SAVE;
- DM-DOCUMENT-DISCARD;
- DM-DOCUMENT-CLOSE;
- DM-DOCUMENT-LIST;
- DM-GET;
- DM-SEARCH;
- DM-CREATE;
- DM-DELETE;
- DM-MODIFY;
- DM-COPY;
- DM-MOVE;
- DM-REPLACE;
- DM-RESERVE;
- DM-UNRESERVE;
- DM-POINT;
- DM-MACRO-CALL;
- DM-GROUP-BEGIN;
- DM-GROUP-END.

Ces services sont décrits dans les paragraphes suivants.

7.1.1 Service abstrait DM-DOCUMENT-OPEN (ouverture de document DM)

Le service abstrait d'ouverture de document est utilisé par le client DTAM-DM pour sélectionner et ouvrir des documents avant de les manipuler. Aucune manipulation interne ne peut être effectuée sur des documents qui n'ont pas été préalablement ouverts. Plusieurs documents peuvent être ouverts en même temps.

7.1.2 Service abstrait DM-DOCUMENT-SAVE (sauvegarde de document DM)

Le service abstrait de sauvegarde de document est utilisé par le client DTAM-DM pour sauvegarder des documents à la suite de manipulations documentaires, effectuées depuis la dernière utilisation des services abstraits DM-DOCUMENT-OPEN, DM-DOCUMENT-DISCARD, ou DM-DOCUMENT-SAVE.

7.1.3 Service abstrait DM-DOCUMENT-DISCARD (abandon de document DM)

Le service abstrait d'abandon de document est utilisé par le client DTAM-DM pour abandonner les manipulations documentaires effectuées depuis la dernière utilisation du service abstrait DM-DOCUMENT-SAVE, DM-DOCUMENT-OPEN ou DM-DOCUMENT-DISCARD.

7.1.4 Service abstrait DM-DOCUMENT-CLOSE (fermeture de document DM)

Le service abstrait de fermeture de document est utilisé par le client DTAM-DM pour refermer un document manipulé et s'en dessaisir. Toutes les manipulations de document effectuées depuis la dernière utilisation d'un des services abstraits DM-DOCUMENT-OPEN, DM-DOCUMENT-SAVE, DM-DOCUMENT-DISCARD sont abandonnées. Pour sauvegarder le contenu d'un document avant sa fermeture, on utilisera le service abstrait DM-DOCUMENT-SAVE.

7.1.5 Service abstrait DM-DOCUMENT-LIST (listage de documents DM)

Le service abstrait de listage de documents est utilisé par le client DTAM-DM pour demander au serveur DTAM-DM une liste d'identificateurs de document répondant aux critères indiqués. L'utilisation du service de listage de documents est limitée aux applications qui ne disposent que de capacités restreintes d'archivage et de consultation. Mais si une application dispose du service DFR (archivage et consultation) en plus du service DTAM-DM, il est recommandé qu'elle utilise ces services de manière combinée pour de telles opérations.

7.1.6 Service abstrait DM-GET (lecture DM)

Le service abstrait de lecture est utilisé par le client DTAM-DM pour lire des fragments de document.

7.1.7 Service abstrait DM-SEARCH (recherche DM)

Le service abstrait de recherche est utilisé par le client DTAM-DM pour rechercher des fragments spécifiés d'un document. Le client DTAM-DM spécifie des domaines de recherche (par exemple ne chercher que dans le premier chapitre) et des critères de recherche. Le service abstrait de recherche peut servir à identifier des fragments de document, des identificateurs étant ensuite utilisés par d'autres opérations abstraites du type modification, création, etc.

7.1.8 Service abstrait DM-CREATE (création DM)

Le service abstrait de création est utilisé par le client DTAM-DM pour ajouter des fragments de documentaires à un document existant. Ce service peut véhiculer le fragment documentaire, ainsi que les valeurs qui lui sont applicables. Le client DTAM-DM doit toujours spécifier l'emplacement d'insertion du fragment dans le document.

7.1.9 Service abstrait DM-DELETE (suppression DM)

Le service abstrait de suppression est utilisé par le client DTAM-DM pour supprimer des fragments de document.

7.1.10 Service abstrait DM-MODIFY (modification DM)

Le service abstrait de modification est utilisé par le client DTAM-DM pour modifier des fragments de document déjà existants. Ce service affecte de nouvelles valeurs aux attributs du fragment de document identifié.

7.1.11 Service abstrait DM-COPY (copie DM)

Le service abstrait de copie est utilisé par le client DTAM-DM pour recopier des fragments de document dans un autre emplacement du même document ou d'un autre document.

7.1.12 Service abstrait DM-MOVE (déplacement DM)

Le service abstrait de déplacement est utilisé par le client DTAM-DM pour modifier la position de fragments de document à l'intérieur d'un même document ou pour les transférer d'un document à un autre.

7.1.13 Service abstrait DM-REPLACE (remplacement DM)

Le service abstrait de remplacement est utilisé par le client DTAM-DM pour remplacer des fragments de document. Ce service supprime un fragment identifié et le remplace par un nouveau fragment.

7.1.14 Service abstrait DM-RESERVE (réservation DM)

Le service abstrait de réservation est utilisé par le client DTAM-DM pour réserver un fragment de document donné. L'opération de réservation n'altère pas la propriété ou les droits d'accès aux fragments de documents (les droits d'accès dans le profil de document d'un document ODA par exemple).

7.1.15 Service abstrait DM-UNRESERVE (déréservation DM)

Le service abstrait de déréservation est utilisé par le client DTAM-DM pour libérer un fragment documentaire précédemment réservé par le service abstrait de réservation.

7.1.16 Service abstrait DM-POINT (pointage DM)

Le service abstrait de pointage est utilisé par les clients DTAM-DM dans des applications multiutilisateur comme la multiédition ou le multiaffichage.

7.1.17 Service abstrait DM-MACRO-CALL (appel de macro commande DM)

Ce service fera l'objet d'un complément d'étude.

7.1.18 Service abstrait DM-GROUP-BEGIN (début de groupe DM)

Le service abstrait de début de groupe est utilisé par le client DTAM-DM pour indiquer le début d'un ensemble de services abstraits DTAM-DM que le serveur doit exécuter et auxquels il doit répondre en groupe.

Ce service fera l'objet d'un complément d'étude.

7.1.19 Service abstrait DM-GROUP-END (fin de groupe DM)

Le service abstrait de fin de groupe est utilisé par le client DTAM-DM pour indiquer la fin d'un ensemble de services abstraits DTAM-DM que le serveur doit exécuter et auxquels il doit répondre en groupe.

Ce service fera l'objet d'un complément d'étude.

7.2 Vue d'ensemble des services abstraits d'échange de jeton DTAM-TK

La présente Recommandation définit les services abstraits DTAM-TK suivants:

- TK-TOKEN-GIVE;
- TK-TOKEN-PLEASE.

Ces services sont décrits dans les paragraphes suivants.

7.2.1 Service abstrait TK-TOKEN-GIVE (cession de jeton)

Le service abstrait de cession de jeton fait passer le jeton d'application du client au serveur DTAM-TK et intervertit les rôles client et serveur des entités d'application engagées dans le processus de communication. Le jeton d'application fait appel aux services d'opérations distantes de l'élément ROSE, Recommandation X.219, et n'a donc pas de rapport avec les jetons de communication utilisés dans les autres couches OSI, par exemple les jetons de la couche session.

7.2.2 Service abstrait TK-TOKEN-PLEASE (demande de jeton TK)

Le service abstrait de demande de jeton est utilisé par le serveur DTAM-TK pour réclamer à un client DTAM-TK le jeton d'application, c'est-à-dire pour demander l'interversion des rôles client et serveur des entités d'application engagées dans le processus de communication.

8 Etablissement d'une association d'application

Les éléments de service d'application DTAM-DM et DTAM-TK sont utilisés conjointement avec les services ACSE, ainsi qu'avec la notation formelle des opérations abstraites de rattachement et de détachement spécifiées dans la Recommandation X.407. Le mappage des opérations de rattachement et de détachement sur les services ACSE est décrit dans la Recommandation X.219.

Dans les développements qui suivent, les termes «client DTAM-DM» et «serveur DTAM-DM» ne prennent leur sens que dans les environnements DTAM-DM et DTAM-DM/DFR. Dans l'environnement DTAM-DM/DTAM-TK, ces termes seront respectivement interprétés comme «client/serveur demandeur» et «client/serveur répondeur». Dans l'environnement DTAM-DM/DFR ces termes seront respectivement interprétés comme «client DTAM-DM/DFR» et «serveur DTAM-DM/DFR».

8.1 Paramètres de rattachement et de détachement abstraits

8.1.1 Paramètres de rattachement abstrait

L'opération abstraite DTAM-DMBind rattache l'accès DTAM-DM du client DTAM-DM au serveur DTAM-DM. Dans l'environnement DTAM-DM/DTAM-TK, elle rattache les accès DTAM-DM-SYM et DTAM-TK des clients/serveurs DTAM-DM/DTAM-TK. Dans l'environnement DTAM-DM/DFR, elle rattache les accès DTAM-DM et DFR du client DTAM-DM/DFR au serveur DTAM-DM/DFR.

DTAM-DMBind ::= ABSTRACT-BIND

TO {dtam-dm-port[S], dfr-port[S], dtam-dm-sym-port, dtam-tk-port}

BIND

ARGUMENT DtamDmBindArgument

RESULT DtamDmBindResult

BIND-ERROR DtamDmBindError

8.1.1.1 Paramètres d'argument de rattachement

Le paramètre *DtamDmBindArgument* permet de convenir des capacités utilisées durant l'association.

DtamDmBindArgument ::= [0] SEQUENCE {

dtamProtocolVersion	[0] BIT STRING { version-1 (0) } DEFAULT {version-1},
dtamManipulationCapabilities	[1] ManipulationCapabilities,
odaApplicationCapabilities	[2] SEQUENCE OF OdaApplicationCapabilities OPTIONAL,
dfrCapabilities	[3] DfrCapabilities OPTIONAL,
bilateralInformation	[4] SEQUENCE OF BilateralInformation OPTIONAL,
applicationRequirements	[5] ApplicationRequirements OPTIONAL }

- a) **dtamProtocolVersion**: ce paramètre indique les versions du protocole DTAM-DM demandées par le client DTAM-DM. C'est une chaîne binaire de longueur variable où chaque bit de valeur 1 indique que la version correspondante du protocole DTAM-DM est demandée. Plusieurs bits peuvent être mis à 1 pour indiquer la prise en charge de plusieurs versions. La chaîne binaire ne comportera aucun bit surnuméraire au-delà de la version la plus récente de la présente Recommandation prise en charge par le client. En d'autres termes, le dernier bit de la chaîne est égal à 1. En l'absence de ce paramètre, on adoptera la valeur par défaut «version-1».

Si le serveur DTAM-DM prend en charge une version du protocole demandé, l'opération abstraite de rattachement DTAM-DM se poursuit normalement. Sinon le serveur DTAM-DM rejettera l'association avec un message d'erreur adéquat (voir 8.1.1.3)

La version-1 est définie dans la Recommandation T.436.

- b) **dtamManipulationCapabilities**: ce paramètre indique les capacités DTAM-DM que le client DTAM-DM souhaite utiliser durant l'association. Les capacités sont négociées avec le serveur DTAM-DM. S'il peut les prendre en charge, l'opération de rattachement continue normalement. Sinon il rejettera l'association avec un message d'erreur adéquat (voir 8.1.1.3)

Il est possible de sélectionner un profil DTAM-DM normalisé ou de convenir de conventions bilatérales connues. Une autre possibilité est d'indiquer le niveau de manipulation.

Le Tableau 1 présente une vue d'ensemble du rapport existant entre les services abstraits DTAM-DM et les valeurs possibles du paramètre *manipulationLevelSelection* prises dans le paramètre *ManipulationCapabilities*.

ManipulationCapabilities ::= CHOICE {
 profileSelection [0] OBJECT IDENTIFIER,
 manipulationLevelSelection [1] ENUMERATED {
 basicReadOnlyLevel (0),
 basicManipulationLevel (1),
 extendedLevel (2) },
 bilateralAgreement [2] IMPLICIT OCTET STRING }

- c) **odaApplicationCapabilities**: dans l'opération de rattachement, ce paramètre est facultatif et ne sera utilisé que dans les applications de type conférence, où chaque partenaire dispose de capacités symétriques (comportant un client et un serveur). Ce paramètre spécifie le profil d'application du document et, optionnellement, deux autres paramètres décrits ci-après comme étant les capacités en réception du client/serveur DTAM-DM/DTAM-TK. Le déroulement de ce service n'est garanti que si l'interlocuteur agit dans le cadre des capacités en réception du client/serveur DTAM-DM/DTAM-TK.

Les caractéristiques *NonBasicOdaDocumentCharacteristics* spécifient les caractéristiques secondaires de documents caractéristiques accessibles au serveur en tant que capacités en réception. Les attributs et les valeurs qui peuvent être spécifiés par ce paramètre sont limités à ceux qu'autorise le profil d'application de document correspondant. Le format de ce paramètre est décrit dans la Recommandation T.414.

Les caractéristiques *NonBasicOdaStructuralCharacteristics* spécifient les caractéristiques structurelles secondaires, accessibles au serveur en tant que capacités en réception. Les attributs et les valeurs qui peuvent être spécifiés par ce paramètre sont limités à ceux qu'autorise le profil d'application de document correspondant. Le format de ce paramètre est décrit dans la Recommandation T.414.

OdaApplicationCapabilities ::= SEQUENCE {
 odaDocumentApplicationProfile [0] OBJECT IDENTIFIER,
 nonBasicOdaDocumentCharacteristics [1] Non-Basic-Doc-Characteristics
 OPTIONAL,
 nonBasicOdaStructuralCharacteristics [2] Non-Basic-Struc-Characteristics
 OPTIONAL }

- d) **dfrCapabilities**: dans l'opération de rattachement, ce paramètre est facultatif et ne sera utilisé que dans l'environnement DTAM-DM/DFR. Il indique les capacités DFR que demande le client DTAM-DM/DFR dans l'environnement DTAM-DM/DFR. Il représente une version du protocole DFR et, optionnellement, un profil DFR.

La version du protocole DFR est représentée par une chaîne binaire de longueur variable où chaque bit de valeur 1 correspond à une version demandée du protocole DFR. Plusieurs bits peuvent être mis à 1 pour indiquer la prise en charge de plusieurs versions. La chaîne binaire ne comportera aucun bit surnuméraire au-delà de la version la plus récente de la présente Recommandation prise en charge par le client. En d'autres termes, le dernier bit de la chaîne est égal à 1.

Si le serveur DTAM-DM/DFR prend en charge les capacités DFR demandées, le service de rattachement abstrait DTAM-DM se poursuit normalement. Sinon le serveur DTAM-DM/DFR rejettera l'association avec un message d'erreur adéquat (voir 8.1.1.3).

DfrCapabilities ::= SEQUENCE {
 dfrProtocolVersion [0] DfrProtocolVersion DEFAULT {version-1},
 dfrProfile [1] OBJECT IDENTIFIER OPTIONAL }

DfrProtocolVersion ::= BIT STRING {
 version-1 (1),
 version-2 (2) }

- La version 1 est définie par ISO/CEI 10166-1:1991
- La version 2 est définie par ISO/CEI 10166-1:1991 et de l'Amendement 1 de celle-ci.

- e) **bilateralInformation**: ce paramètre véhicule une information bilatérale entre le client et le serveur DTAM-DM. Le format de cette information est le suivant:

BilateralInformation ::= SEQUENCE {
 serverName [0] ServerName,
 information [1] EXTERNAL }

ServerName ::= DistinguishedName

- f) **applicationRequirements**: ce paramètre indique les prescriptions supplémentaires d'application négociées entre le serveur et le client DTAM. Sa signification dépend de l'application. Si le serveur DTAM peut prendre en charge ces prescriptions, l'opération de rattachement continue normalement. Sinon, le serveur DTAM rejettera l'association avec un message d'erreur adéquat (voir 8.1.1.3).

ApplicationRequirement ::= CHOICE {
[0] OCTET STRING,
[1] OBJECT IDENTIFIER }

TABLEAU 1/T.435

Relation entre les services abstraits et les valeurs de manipulationLevelSelection

Services abstraits DTAM-DM	manipulationLevelSelection		
	basicReadOnlyLevel (lecture seulement)	basicManipulationLevel (manipulation de base)	extendedLevel (niveau étendu)
DM-DOCUMENT-OPEN (Note 1)	M	M	M
DM-DOCUMENT-SAVE (Note 1)		M	M
DM-DOCUMENT-DISCARD (Note 1)		M	M
DM-DOCUMENT-CLOSE (Note 1)	M	M	M
DM-POINT (Note 1)	U	U	U
DM-DOCUMENT-LIST (Note 2)	C	C	C
DM-GET (Note 3)	M	M	M
DM-SEARCH (Note 3)	M	M	M
DM-CREATE (Note 4)		M	M
DM-DELETE (Note 4)		M	M
DM-MODIFY (Note 4)		M	M
DM-COPY (Note 4)		M	M
DM-MOVE (Note 5)			M
DM-REPLACE (Note 5)			M
DM-RESERVE (Note 5)			M
DM-UNRESERVE (Note 5)			M
DM-MACRO-CALL (Note 6)	FS	FS	FS
DM-GROUP-BEGIN (Note 6)	FS	FS	FS
DM-GROUP-END (Note 6)	FS	FS	FS

NOTES

- 1 Les services abstraits DM-DOCUMENT-OPEN et DM-DOCUMENT-CLOSE doivent être pris en charge par toutes les valeurs de manipulationLevelSelection. L'utilisation du service DM-POINT est facultative.
- 2 L'utilisation du service abstrait DM-DOCUMENT-LIST n'est pas autorisé dans les applications combinées DTAM-DM/DFR. On utilisera dans ce cas le service abstrait DFR de listage (C).
- 3 En plus des services abstraits énumérés dans la Note 1, les services abstraits DM-GET et DM-SEARCH seront pris en charge si manipulationLevelSelection prend la valeur basicReadOnlyLevel (M); les autres services ne s'appliquent pas (<blanc>).
- 4 En plus des services abstraits énumérés dans la Note 3, les services abstraits DM-DOCUMENT-SAVE, DM-DOCUMENT-DISCARD, DM-CREATE, DM-DELETE, DM-MODIFY et DM-COPY seront pris en charge si manipulationLevelSelection prend la valeur basicManipulationLevel (M); les autres services abstraits ne s'appliquent pas (<blanc>).
- 5 En plus des services abstraits énumérés dans la Note 4, les services abstraits DM-MOVE, DM-REPLACE, DM-RESERVE, DM-UNRESERVE et DM-POINT seront pris en charge si manipulationLevelSelection prend la valeur extendedLevel (M).
- 6 La prise en charge des services abstraits DM-MACRO-CALL, DM-GROUP-BEGIN et DM-GROUP-END est à l'étude (FS) pour les différents niveaux manipulationLevelSelection.

8.1.1.2 Paramètres du résultat de rattachement

Le paramètre *DtamDmBindResult* indique les capacités prises en charges par le serveur DTAM-DM.

```
DtamDmBindResult ::= [1] SEQUENCE {
    serverSelectedDtamProtocolVersion  [0] BIT STRING
                                         { version-1 (0) } DEFAULT {version-1},
    odaApplicationCapabilities          [1] SEQUENCE OF OdaApplicationCapabilities,
    serverSelectedDfrVersion           [2] BIT STRING
                                         { version-1 (1),
                                           version-2 (2) } OPTIONAL,
    bilateralInformation                [3] SEQUENCE OF
                                         BilateralInformation OPTIONAL }
```

- a) **serverSelectedDtamProtocolVersion**: ce paramètre indique la version du protocole DTAM-DM choisie par le serveur DTAM-DM parmi celles que le client DTAM-DM a proposées. C'est une chaîne binaire de longueur variable où seule la version choisie est mise à 1. La valeur de ce paramètre définit la version du protocole à utiliser pour cette association.
- b) **odaApplicationCapabilities**: ce paramètre spécifie le profil d'application du document et optionnellement, deux autres paramètres comme étant les capacités en réception du serveur DTAM-DM. Le déroulement du service n'est garanti que si le client DTAM-DM agit dans le cadre des capacités en réception du serveur DTAM-DM.
- c) **serverSelectedDfrProtocolVersion**: ce paramètre est facultatif et ne sera utilisé que dans l'environnement DTAM-DM/DFR. Il indique la version du protocole DFR choisie par le serveur DTAM-DM/DFR parmi celles que le client DTAM-DM/DFR a proposées. C'est une chaîne binaire de longueur variable où seule la version choisie est mise à 1. La valeur de ce paramètre définit la version du protocole à utiliser pour cette association.
- d) **bilateralInformation**: ce paramètre véhicule une information bilatérale entre le serveur et le client DTAM-DM.

8.1.1.3 Paramètres d'erreur de rattachement

Le paramètre *DtamDMBindError* signale les erreurs susceptibles de survenir lors de l'établissement d'une association.

Le paramètre *DmServiceProblem* signale que le serveur DTAM-DM ne peut pas établir l'association à cause de certaines erreurs relatives à l'exploitation.

Le paramètre *ProtocolProblem* indique que le serveur DTAM-DM ne prend pas en charge les versions de protocole demandées. Dans ce cas, le serveur peut optionnellement indiquer les versions du protocole DFR ou DTAM-DM qu'il pourrait prendre en charge.

Le paramètre *ProfileProblem* indique que le serveur ne prend pas en charge les profils de protocole demandés. Dans ce cas, le serveur peut optionnellement indiquer le profil DTAM-DM ou DFR qu'il pourrait prendre en charge.

Le paramètre *DmServiceProblem*, qui est aussi utilisé dans les opérations DTAM-DM, est défini au 9.4.8.

Le paramètre *ApplicationProblem* indique que le serveur DTAM-DM ne prend pas en charge les prescriptions d'application demandées. Dans ce cas, le serveur peut optionnellement indiquer les prescriptions d'application qu'il pourrait prendre en charge.

```
DtamDmBindError ::= [2] CHOICE {
    serviceError          [0] DmServiceProblem,
    protocolError         [1] ProtocolProblem,
    profileError          [2] ProfileProblem,
    applicationError      [3] ApplicationProblem }

ProtocolProblem ::= SEQUENCE {
    protocol-version-not-supported  [0] SEQUENCE OF ENUMERATED {
        dtam-version-not-supported (0),
        dfr-version-not-supported (1) },
    supportedDtamProtocolVersions  [1] BIT STRING
        { version-1 (0) } OPTIONAL,
    supportedDfrProtocolVersions   [2] BIT STRING
        { version-1 (0),
          version-2 (1) } OPTIONAL }
```



```

ProfileProblem ::= SEQUENCE {
    profile-not-supported [0] SEQUENCE OF ENUMERATED {
        dtam-manipulation-level-not-supported (0),
        dfr-profile-not-supported (1) },
    supportedDtamManipulationLevels [1] SEQUENCE OF ManipulationCapabilities
        OPTIONAL,
    supportedDfrProfiles [2] SEQUENCE OF OBJECT IDENTIFIER
        OPTIONAL }

```

ApplicationProblem ::= SEQUENCE OF ApplicationRequirements OPTIONAL

8.1.2 Paramètres ABSTRACT-UNBIND du détachement abstrait

Ce paragraphe définit et décrit les paramètres des opérations de détachement concernant les accès DTAM-DM, DTAM-DM-SYM et DTAM-TK selon l'environnement. Il n'existe ni argument ni erreur associés à l'opération de détachement.

```

DTAM-DMUnBind ::= ABSTRACT-UNBIND
    FROM {dtam-dm-port[S], dfr-port[S], dtam-dm-sym-port, dtam-tk-port}

```

8.2 Utilisation du service A-ABORT (Annulation d'association)

Le service A-ABORT est lancé par le client ou par le serveur afin de dissoudre sans condition une association. Le codage du paramètre *DTAM-DMAbortInfo*, qui est transcrit dans le paramètre «information d'utilisateur» de A-ABORT, est défini par le type de données ASN.1 externe (EXTERNAL).

Le paramètre *DtamDmAbortSource* indique la source d'annulation de l'association si l'association a été annulée à l'initiative du fournisseur ou de l'utilisateur du service DTAM-DM. Le paramètre *DtamDmAbortReason* indique le motif de l'annulation. La valeur "temporary-problem" indique qu'aucune tentative de rétablissement de l'association ne devra être effectuée pendant un certain temps déterminé par une règle locale.

```

DTAM-DMAbortInfo ::= [3] SEQUENCE {
    abortSource [0] DtamDmAbortSource,
    abortReason [1] DtamDmAbortReason OPTIONAL,
    userInformation [2] EXTERNAL OPTIONAL }

```

```

DtamDmAbortSource ::= ENUMERATED {
    dtamDmServiceUser (0),
    dtamDmServiceProvider (1) }

```

```

DtamDmAbortReason ::= ENUMERATED {
    local-system-problem (0),
    temporary-problem (1),
    protocol-error (2) }

```

9 Définition des services abstraits

Cet article définit les services abstraits DTAM-DM et DTAM-TK. Leur définition formelle complète figure dans l'Annexe B.

Certains types de données, utilisés dans ces définitions, sont définis dans d'autres Recommandations et normes internationales; ils sont importés par la présente Recommandation.

L'argument de tout service abstrait DTAM-DM ou DTAM-TK se présente sous la forme d'une séquence de paramètres, certains étant obligatoires et d'autres facultatifs pour le service en cause. Certains de ces paramètres peuvent être communs à tous les services ou à la plupart d'entre eux: ces paramètres sont alors définis comme étant des types de données communs.

Les définitions des services abstraits DTAM-DM et DTAM-TK sont données dans d'autres paragraphes.

9.1 Types de données importés

A partir de l'architecture ODA (Recommandation T.413):

```

    ConstituentType,
    ConstituentIdentifiant,
    AttributeValuesType,
    QueryType

```

A partir de l'architecture ODA (Recommandation T.415)

**Non-Basic-Doc-Characteristics,
Non-Basic-Struc-Characteristics,
Document-Reference**

A partir de l'architecture ODA (Recommandation T.422):

**Location-expression,
Basic-location-expression,
Constituent-Locator**

A partir de l'élément DFR (ISO/CEI 10166):

DfrEntryName,

A partir du système MHS (Recommandation X.407)

**ABSTRACT-BIND,
ABSTRACT-UNBIND,
ABSTRACT-ERROR,
ABSTRACT-OPERATION,
OBJECT,
PORT**

A partir du système DS (Recommandation X.500)

DistinguishedName

9.2 Définition des types de données communs aux services abstraits

La présente Recommandation définit les types de données suivants comme des types de données communs:

**DocumentId
ManipulationObject
ObjectValue
ObjectPosition
ObjectContent
AssignedId
NonPermanentIdentifier**

9.2.1 DocumentId – Identificateur de document

**DocumentId ::= CHOICE {
 dfrName [0] DfrEntryName,
 documentReference [1] Document-Reference,
 documentName [2] OCTET STRING,
 nonPermanentIdentifier [3] NonPermanentIdentifier }**

9.2.2 ManipulationObject – Objet manipulable

**ManipulationObject ::= SEQUENCE {
 documentIdentifier [0] DocumentId OPTIONAL,
 objectValue [1] ObjectValue OPTIONAL }**

9.2.3 ObjectValue – Valeur d'objet

**ObjectValue ::= CHOICE {
 odaExpression [0] Oda-expression,
 otherExpression [1] OCTET STRING,
 anyExpression [2] ANY }**

9.2.4 Oda-Expression – Expression ODA

**Oda-Expression ::= CHOICE {
 locationExpression [0] Location-Expression,
 basicLocationExpression [1] basic-Location-Expression,
 constituentLocator [2] Constituent-Locator,
 ConstituentIdentifier [3] Constituent-Identifier }**

9.2.5 ObjectPosition – Position d'objet

```
ObjectPosition ::= ENUMERATED {  
    before          (0),  
    afterLastChild (1) }
```

9.2.6 ObjectContent – Contenu d'objet

```
ObjectContent ::= CHOICE {  
    odaContent          [0] ConstituentType,  
    octetAligned        [1] OCTET STRING,  
    anyContent          [2] ANY }
```

9.2.7 AssignedId – Identificateur affecté

```
AssignedId ::= CHOICE {  
    odaIdentifier       [0] ConstituentIdentifier,  
    nonOdaIdentifier    [1] INTEGER,  
    any                 [2] ANY }
```

9.2.8 NonPermanentIdentifier – Identificateur non permanent

```
NonPermanentIdentifier ::= CHOICE {  
    [0] INTEGER,  
    [1] OCTET STRING }
```

9.3 Définition des services abstraits DTAM-DM

La présente Recommandation définit les services abstraits DTAM-DM suivants:

- DM-DOCUMENT-OPEN;
- DM-DOCUMENT-SAVE;
- DM-DOCUMENT-DISCARD;
- DM-DOCUMENT-CLOSE;
- DM-DOCUMENT-LIST;
- DM-GET;
- DM-SEARCH;
- DM-CREATE;
- DM-DELETE;
- DM-MODIFY;
- DM-COPY;
- DM-MOVE;
- DM-REPLACE;
- DM-RESERVE;
- DM-UNRESERVE;
- DM-POINT;
- DM-MACRO-CALL;
- DM-GROUP-BEGIN;
- DM-GROUP-END.

Les services abstraits DTAM-DM sont décrits selon la définition des macro instructions des opérations abstraites figurant dans la Recommandation X.407. Dans l'environnement DTAM-DM ou DTAM-DM/DFR, toutes les opérations abstraites sont invoquées par le consommateur (client DTAM-DM ou DTAM-DM/DFR). Dans l'environnement DTAM-DM/DTAM-TK, toutes les opérations sont invoquées par l'entité qui tient le rôle du client (le jeton d'application détermine le rôle client ou serveur tenu par l'entité client/serveur DTAM-DM/DTAM-TK à un moment donné). Chaque opération abstraite signale toujours soit un succès soit une erreur. Les opérations abstraites sont mappées sur les services ROSE définis dans la Recommandation X.219.

Les services abstraits DTAM-DM sont définis dans les paragraphes suivants.

9.3.1 Service abstrait DM-DOCUMENT-OPEN (ouverture de document)

Le service abstrait d'ouverture de document est utilisé par le client DTAM-DM pour sélectionner et ouvrir des documents avant de les manipuler. Aucune manipulation interne ne peut être effectuée sur des documents qui n'ont pas été préalablement ouverts. Plusieurs documents peuvent être ouverts en même temps.

DM-DOCUMENT-OPEN ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-DOCUMENT-OPEN-Arguments
RESULT            DM-DOCUMENT-OPEN-Result
ERRORS
{
    DM-DocumentError,
    DM-AccessError,
    DM-ServiceError
}
```

DM-DOCUMENT-OPEN-Arguments ::= SEQUENCE

```
{
    documentId      [0] DocumentId,
    mode            [1] ModeType OPTIONAL
}
```

ModeType ::= ENUMERATED{ read (0), modify (1), create (2) }

DM-DOCUMENT-OPEN-Result ::= CHOICE

```
{
    noValue          [0] NULL,
    nonPermanentIdentifier [1] NonPermanentIdentifier
}
```

9.3.1.1 Arguments

Le paramètre *document-Id* est obligatoire et identifie le document sélectionné pour manipulation ultérieure. Le paramètre *mode* est facultatif et peut prendre trois valeurs: "*read*" (lecture), "*modify*" (modification) et "*create*" (création). Si un document est ouvert en «mode lecture», il ne sera accessible qu'en lecture. S'il est ouvert en «mode modification», il sera possible de lui appliquer toutes les opérations de manipulation. Ce «mode création» signifie que le document est à créer; dans ce cas, l'identificateur permanent est l'identificateur à attribuer au document.

9.3.1.2 Résultat

Le résultat *DM-DOCUMENT-OPEN-Result* sera retourné en cas de succès. Il véhicule soit un identificateur non permanent *nonPermanentIdentifier*, qui servira jusqu'à l'opération de fermeture correspondante *DM-DOCUMENT-CLOSE* qui attribuera au document un nouvel identificateur permanent, ou alors il ne véhicule aucun paramètre, ce qui implique que l'identificateur permanent du document est utilisé durant toute la manipulation.

9.3.1.3 Erreurs

Les motifs d'erreur d'ouverture de document sont les suivants:

- **DM-DocumentError:** le paramètre *documentId* ne pointe pas sur un document existant.
- **DM-AccessError:** le document est déjà ouvert par l'utilisateur, est réservé ou correspond à des droits d'accès dont ne dispose pas l'utilisateur.
- **DM-ServiceError:** le document ne peut pas être ouvert pour des raisons locales.

9.3.2 Service abstrait DM-DOCUMENT-SAVE (sauvegarde)

Le service abstrait de sauvegarde de document est utilisé par le client DTAM-DM pour sauvegarder des documents après les avoir manipulés. Il est essentiel d'effectuer cette opération avant l'opération de fermeture *DM-DOCUMENT-CLOSE* si on veut conserver les modifications.

```

DM-DOCUMENT-SAVE ::= ABSTRACT-OPERATION
  ARGUMENT          DM-DOCUMENT-SAVE-Arguments
  RESULT            DM-DOCUMENT-SAVE-Result
  ERRORS
  {
    DM-AccessError,
    DM-DocumentError,
    DM-ServiceError
  }

```

```

DM-DOCUMENT-SAVE-Arguments ::= SEQUENCE
{
  documentId        [0] DocumentId
}

```

DM-DOCUMENT-SAVE-Result ::= NULL

9.3.2.1 Arguments

Le paramètre *documentId* est obligatoire et identifie le document sélectionné à sauvegarder.

9.3.2.2 Résultat

Le résultat DM-DOCUMENT-SAVE-Result sera retourné en cas de succès. Il ne comporte aucun paramètre.

Dans ce cas, toutes les manipulations qui ont été appelées depuis la dernière opération DM-DOCUMENT-OPEN, DM-DOCUMENT-SAVE ou DM-DOCUMENT-DISCARD sont enregistrées de manière définitive.

9.3.2.3 Erreurs

Les motifs d'erreur de sauvegarde sont les suivants:

- DM-AccessError: le document n'est pas ouvert, est réservé par un autre utilisateur ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- DM-DocumentError: le paramètre *document-ID* ne pointe pas vers un document existant.
- DM-ServiceError: le document ne peut pas être sauvegardé pour des raisons locales.

9.3.3 Service abstrait DM-DOCUMENT-DISCARD (abandon)

Le service abstrait d'abandon est utilisé par le client DTAM-DM pour abandonner les manipulations de document qui ont été effectuées depuis l'utilisation du dernier service abstrait de sauvegarde, d'ouverture ou d'abandon de document.

```

DM-DOCUMENT-DISCARD ::= ABSTRACT-OPERATION
  ARGUMENT          DM-DOCUMENT-DISCARD-Arguments
  RESULT            DM-DOCUMENT-DISCARD-Result
  ERRORS
  {
    DM-AccessError,
    DM-DocumentError,
    DM-ServiceError
  }

```

```

DM-DOCUMENT-DISCARD-Arguments ::= SEQUENCE
{
  documentId        [0] DocumentId
}

```

DM-DOCUMENT-DISCARD-Result ::= NULL

9.3.3.1 Arguments

Le paramètre *documentId* est obligatoire et identifie le document sélectionné pour le service d'abandon. Les modifications apportées au document depuis la dernière opération de sauvegarde, d'ouverture ou d'abandon sont abandonnées.

9.3.3.2 Résultat

Le résultat DM-DOCUMENT-DISCARD-Result sera retourné en cas de succès. Il n'y a aucun paramètre.

9.3.3.3 Erreurs

Les motifs d'erreur d'abandon sont les suivants:

- **DM-AccessError:** le document n'est pas ouvert, est réservé par un autre utilisateur ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- **DM-DocumentError:** le paramètre *document-ID* ne pointe pas vers un document existant.
- **DM-ServiceError:** le document ne peut pas être abandonné pour des raisons locales.

9.3.4 Service abstrait DM-DOCUMENT-CLOSE (fermeture)

Le service abstrait de fermeture de document est utilisé par le client DTAM-DM pour refermer et libérer des documents après les avoir manipulés. Les modifications effectuées depuis la dernière utilisation du service abstrait DM-DOCUMENT-OPEN, DM-DOCUMENT-SAVE ou DM-DOCUMENT-DISCARD sont abandonnées. Pour sauvegarder le contenu d'un document avant sa fermeture, il est nécessaire d'utiliser le service abstrait DM-DOCUMENT-SAVE.

DM-DOCUMENT-CLOSE ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-DOCUMENT-CLOSE-Arguments
RESULT            DM-DOCUMENT-CLOSE-Result
ERRORS
{
    DM-AccessError,
    DM-DocumentError,
    DM-ServiceError
}
```

DM-DOCUMENT-CLOSE-Arguments ::= SEQUENCE

```
{
    documentId      [0] DocumentId
}
```

DM-DOCUMENT-CLOSE-Result ::= CHOICE

```
{
    noValue          NULL,
    permanentId     [0] DocumentId
}
```

9.3.4.1 Arguments

Le paramètre *documentId* est obligatoire et identifie le document sélectionné à refermer.

9.3.4.2 Résultat

En cas de succès, le résultat DM-DOCUMENT-CLOSE-Result véhicule un identificateur permanent attribué par le serveur au document manipulé. Si l'identificateur utilisé pendant les différentes manipulations depuis l'ouverture du document était provisoire *nonPermanentIdentifier*. Si l'identificateur utilisé est l'identificateur permanent, aucune valeur n'est retournée.

9.3.4.3 Erreurs

Les motifs d'erreur de fermeture sont les suivants:

- **DM-AccessError:** le document n'est pas ouvert, est réservé, ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- **DM-DocumentError:** l'identificateur de document ne pointe pas vers un document existant.
- **DM-ServiceError:** le document ne peut pas être fermé pour des raisons locales.

9.3.5 Service abstrait DM-DOCUMENT-LIST (listage)

Le service abstrait de listage des documents est utilisé par le client DTAM-DM pour demander au serveur DTAM-DM une liste de documents répondant à des critères donnés. L'utilisation du service de listage de documents est réservé aux applications qui ne disposent que de capacités limitées d'archivage et de consultation. Lorsqu'une application prend en charge le service DFR (archivage/consultation) en plus du service de manipulation, il est recommandé d'utiliser ces deux services d'une manière combinée.

```

DM-DOCUMENT-LIST ::= ABSTRACT-OPERATION
  ARGUMENT          DM-DOCUMENT-LIST-Arguments
  RESULT            DM-DOCUMENT-LIST-Result
  ERRORS
  {
    DM-ListError,
    DM-ServiceError
  }

```

```

DM-DOCUMENT-LIST-Arguments ::= CHOICE
  {
    odaCriteria      [0] QueryType,
    otherCriteria    [1] OtherCriteria,
    anyCriteria      [2] ANY
  }

```

```

OtherCriteria ::= CHOICE {
  all              [0] BOOLEAN,
  prefix           [1] OCTET STRING,
  suffix           [2] OCTET STRING }

```

```

DM-DOCUMENT-LIST-Result ::= SEQUENCE OF DocumentId

```

9.3.5.1 Arguments

Le paramètre *odaCriteria* (critères ODA) est une requête établie sous la forme d'une expression logique comportant des attributs évalués relatifs à un profil de document ODA. Il est défini dans la Recommandation T.413.

Le paramètre *OtherCriteria* (autres critères) s'applique à tout type de document. Il peut être constitué de trois critères de sélection différents. Le paramètre *all* demande la liste de tous les documents. Le paramètre *prefix* demande la liste de tous les documents dont le nom commence par une chaîne de caractère identique à la valeur *prefix*. Le paramètre *suffix* demande l'énumération de tous les documents dont le nom se termine par une chaîne identique à la valeur *suffix*.

Le paramètre *anyCriteria* demande la liste de tous les documents dont les noms correspondent au critère donné.

9.3.5.2 Résultat

Le serveur envoie comme résultat, une liste d'identificateurs de document. Les noms de document correspondant à ces identificateurs répondent aux critères spécifiés.

9.3.5.3 Erreurs

Les motifs d'erreur dans le listage sont les suivants:

- DM-ListError: les critères de liste spécifiés ne sont pas acceptés.
- DM-ServiceError: le listage ne peut pas être effectué pour des raisons locales.

9.3.6 Service abstrait DM-GET (lecture)

Le service abstrait de lecture est utilisé par le client DTAM-DM pour lire des fragments de document.

```

DM-GET ::= ABSTRACT-OPERATION
  ARGUMENT          DM-GET-Arguments
  RESULT            DM-GET-Result
  ERRORS
  {
    DM-AccessError,
    DM-FragmentError,
    DM-DocumentError,
    DM-ServiceError
  }

```

```

DM-GET-Arguments ::= SEQUENCE
  {
    object          [0] ManipulationObject
  }

```

```

DM-GET-Result ::= SEQUENCE OF SEQUENCE OF ObjectContent

```

9.3.6.1 Arguments

Le paramètre *object* est obligatoire et comporte une expression de localisation (de type "Localisation-expression") permettant d'identifier le document et le fragment de document sélectionnés pour le service de lecture.

9.3.6.2 Résultat

En cas de succès de l'opération de lecture, le paramètre *result* retourne le contenu de l'objet.

9.3.6.3 Erreurs

Les motifs d'erreur de lecture sont les suivants:

- DM-AccessError: le document n'est pas ouvert ou est réservé par un autre utilisateur.
- DM-FragmentError: le paramètre *object* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.
- DM-ServiceError: l'opération de lecture ne peut pas être effectuée pour des raisons locales.

9.3.7 Service abstrait DM-SEARCH (recherche)

Le service abstrait de recherche est utilisé par le client DTAM-DM pour rechercher des fragments de document spécifiés. Le client DTAM-DM spécifie des domaines de recherche (par exemple ne chercher que dans le premier chapitre) et des critères de recherche. Le service abstrait de recherche peut servir à identifier des fragments de document pour les soumettre par la suite à des opérations abstraites, par exemple de modification ou de création.

DM-SEARCH ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-SEARCH-Arguments
RESULT            DM-SEARCH-Result
ERRORS
{
    DM-AccessError,
    DM-FragmentError,
    DM-DocumentError,
    DM-ServiceError
}
```

DM-SEARCH-Arguments ::= SEQUENCE

```
{
    object      [0] ManipulationObject,
    limit       [1] INTEGER OPTIONAL
}
```

DM-SEARCH-Result ::= SEQUENCE

```
{
    objectList  [0] SEQUENCE OF DocFragment-Id,
    number      [1] INTEGER
}
```

DocFragment-Id ::= ConstituentIdentifier

9.3.7.1 Arguments

Le paramètre *object* est obligatoire et comporte une expression de localisation (de type "Location-expression" permettant d'identifier le document et le fragment de document sélectionnés pour le service de recherche. Le paramètre *limit* est facultatif et indique le nombre maximal d'objets que le service de recherche pourra renvoyer.

9.3.7.2 Résultat

Le paramètre *objectList* contient les identificateurs de tous les objets trouvés par le service de recherche. Le paramètre *number* indique le nombre d'objets trouvés. Si une limite a été indiquée, ce nombre n'est pas supérieur à cette limite. Si le nombre d'objets répondant aux critères de recherche est supérieur à la limite indiquée, l'opération s'arrête.

9.3.7.3 Erreurs

Les motifs d'erreur de recherche sont les suivants:

- DM-AccessError: le document n'est pas ouvert ou est réservé par un autre utilisateur.
- DM-FragmentError: le paramètre *object* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.
- DM-ServiceError: l'opération de recherche ne peut pas être effectuée pour des raisons locales.

9.3.8 Service abstrait DM-CREATE (création)

Le service abstrait de création est utilisé par le client DTAM-DM pour ajouter des fragments de document à un document existant. Ce service permet de véhiculer le fragment de document, ainsi que les valeurs d'attribut qui s'y rapportent. Le client DTAM-DM doit toujours spécifier l'emplacement d'insertion du fragment dans le document de destination.

DM-CREATE ::= ABSTRACT-OPERATION

ARGUMENT	DM-CREATE-Arguments
RESULT	DM-CREATE-Result
ERRORS	
{	
DM-AccessError,	
DM-FragmentError,	
DM-DocumentError,	
DM-AttributeError,	
DM-UpdateError,	
DM-ServiceError	
}	

DM-CREATE-Arguments ::= SEQUENCE

{		
destination	[0] ManipulationObject,	
position	[1] ObjectPosition,	
content	[2] SEQUENCE OF ObjectContent	OPTIONAL
}		

DM-CREATE-Result ::= CHOICE

{		
noValue	[0] NULL,	
assignedID	[1] SEQUENCE OF AssignedId	
}		

9.3.8.1 Arguments

Le paramètre *destination* identifie le document et le fragment de document sélectionnés pour le service de création. Il comporte un localisateur de constituant (de type "Constituent-Locator"). Le paramètre *position* identifie la position du fragment de document dans le document choisi. Le paramètre *content* contient l'information du fragment de document à inclure dans le document choisi.

Tous les paramètres définis en argument sont normalement spécifiés. Toutefois, le paramètre "content" peut ne pas l'être si DM-CREATE est utilisé pour créer un nouveau paragraphe sans y insérer un fragment de document, c'est-à-dire en instanciant une structure subordonnée se référant à des structures génériques ou à des règles par défaut spécifiées dans le document ODA mémorisé sur le serveur.

9.3.8.2 Résultat

Le résultat DM-DOCUMENT-CREATE-Result peut ne véhiculer aucun paramètre ou la séquence des nouveaux identificateurs de fragment.

9.3.8.3 Erreurs

Les motifs d'erreur de création sont les suivants:

- DM-AccessError: l'identificateur de document ne pointe pas vers un document ouvert, ou l'utilisateur n'a pas les droits d'accès nécessaires ou le document est réservé par un autre utilisateur.
- DM-FragmentError: le paramètre *destination* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.

- DM-AttributeError: il est impossible de créer la valeur d'attribut.
- DM-UpdateError: il est impossible de créer un fragment de document.
- DM-ServiceError: le fragment de document ne peut pas être créé pour des raisons locales.

9.3.9 Service abstrait DM-DELETE (suppression)

Le service abstrait de suppression est utilisé par le client DTAM-DM pour supprimer des fragments de document¹⁾.

```
DM-DELETE ::= ABSTRACT-OPERATION
  ARGUMENT      DM-DELETE-Arguments
  RESULT        DM-DELETE-Result
  ERRORS
  {
    DM-AccessError,
    DM-FragmentError,
    DM-DocumentError,
    DM-AttributeError,
    DM-UpdateError,
    DM-ServiceError
  }
```

```
DM-DELETE-Arguments ::= SEQUENCE
{
  object      [0] ManipulationObject
}
```

```
DM-DELETE-Result ::= NULL
```

9.3.9.1 Arguments

Le paramètre *object* est obligatoire et comporte une expression de localisation (de type "Location-expression") permettant d'identifier le document et le fragment de document sélectionnés pour le service de suppression.

9.3.9.2 Résultat

Le résultat de *DM-DELETE-Result* est retourné en cas de succès. Il n'y a aucun paramètre.

9.3.9.3 Erreurs

Les motifs d'erreur de suppression sont les suivants:

- DM-AccessError: le document n'est pas ouvert, est réservé par un autre utilisateur, ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- DM-FragmentError: le paramètre *object* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.
- DM-AttributeError: il est impossible de détruire la valeur d'attribut.
- DM-UpdateError: il est impossible de détruire le fragment de document
- DM-ServiceError: le fragment de document ne peut pas être supprimé pour des raisons locales.

9.3.10 Service abstrait DM-MODIFY (modification)

Le service abstrait de modification est utilisé par le client DTAM-DM pour modifier des fragments de document déjà existants. Ce service affecte de nouvelles valeurs à des attributs du fragment de document identifié.

¹⁾ Dans certains cas, le service abstrait DM-DELETE peut nécessiter que le serveur mémorise les fragments de documents supprimés afin de les réutiliser ailleurs. Toutefois, la mémorisation ou non de ces fragments supprimés sort du cadre de la présente Recommandation.

DM-MODIFY ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-MODIFY-Arguments
RESULT            DM-MODIFY-Result
ERRORS
{
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-AttributeError,
    DM-UpdateError,
    DM-ServiceError
}
```

DM-MODIFY-Arguments ::= SEQUENCE

```
{
    object          [0] ManipulationObject,
    modifications   [1] ModificationsType
}
```

DM-MODIFY-Result ::= NULL

ModificationsType ::= CHOICE

```
{
    odaModifications   [0] OdaModifications,
    otherModifications [1] OCTET STRING,
    anyModifications   [2] ANY
}
```

OdaModifications ::= SEQUENCE

```
{
    attributeValue   [0] AttributeValuesType,
    deleting         [1] BOOLEAN DEFAULT FALSE
}
```

9.3.10.1 Arguments

Le paramètre *object* est obligatoire et comporte une expression de localisation (de type Location-expression) permettant d'identifier le document et le fragment de document sélectionnés pour le service de modification. Le paramètre *modifications* contient les nouvelles valeurs d'attribut du fragment de document identifié.

9.3.10.2 Résultat

En cas de succès, le résultat DM-MODIFY-Result sera retourné. Il n'y a aucun paramètre.

9.3.10.3 Erreurs

Les motifs d'erreur de modification sont les suivants:

- DM-AccessError: le document n'est pas ouvert, est réservé par un autre utilisateur, ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- DM-FragmentError: le paramètre *object* ne pointe pas sur un fragment de document existant.
- DM-AttributeError: il est impossible de modifier la valeur d'attribut.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.
- DM-UpdateError: il est impossible de modifier le fragment de document.
- DM-ServiceError: le fragment de document ne peut pas être supprimé pour des raisons locales.

9.3.11 Service abstrait DM-COPY (recopie)

Le service abstrait de recopie est utilisé par le client DTAM-DM pour recopier des fragments de document en un autre emplacement du même document ou d'un autre document.

DM-COPY ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-COPY-Arguments
RESULT            DM-COPY-Result
ERRORS
{
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-AttributeError,
    DM-UpdateError,
    DM-ServiceError
}
```

DM-COPY-Arguments ::= SEQUENCE

```
{
    source          [0] ManipulationObject OPTIONAL,
    destination     [1] ManipulationObject OPTIONAL,
    position        [2] ObjectPosition OPTIONAL
}
```

DM-COPY-Result ::= SEQUENCE { newConstituent [0] ManipulationObject }

9.3.11.1 Arguments

Le paramètre *source* comporte une expression de localisation de base (de type "Basic-location-expression") permettant d'identifier le document et le fragment de document sélectionnés à recopier. Le paramètre *destination* identifie le document et le fragment de document destinataires de la copie. Il comporte un localisateur de constituant (de type Constituent-Locator). Le paramètre *position* identifie la position du fragment de document dans le document destinataire, celui-ci pouvant être le document d'origine ou un document différent.

Les paramètres définis dans les arguments DM-COPY-arguments doivent normalement être tous utilisés. Exceptionnellement, les paramètres définis dans les arguments de DM-COPY-arguments peuvent être omis, dans les cas suivants.

- le paramètre «source» peut ne pas être utilisé, si les fragments à insérer sont à extraire de la mémoire tampon.
- les deux paramètres «destination» et «position» peuvent ne pas être utilisés, si le fragment de document à recopier est à stocker en mémoire tampon sans modification du document cible.

9.3.11.2 Résultat

En cas de succès, le résultat DM-COPY-Result sera retourné. Le paramètre "newConstituent" identifie le nouvel objet au moyen d'un identificateur de constituant (de type "constituent-identifiant").

9.3.11.3 Erreurs

Les motifs d'erreur de recopie sont les suivants:

- DM-AccessError: le document source ou de destination n'est pas ouvert, est réservé par un autre utilisateur, ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- DM-FragmentError: les paramètres *source* ou *destination* ne pointent pas sur des fragments de document existants.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document source ou destinataire existant.
- DM-UpdateError: il est impossible de copier un fragment de document.
- DM-AttributeError: il est impossible de copier les valeurs des attributs.
- DM-ServiceError: l'opération de copie ne peut pas être effectuée pour des raisons locales.

9.3.12 Service abstrait DM-MOVE (déplacement)

Le service abstrait de déplacement est utilisé par le client DTAM-DM pour modifier la position de fragments de document à l'intérieur d'un même document ou pour les transférer d'un document à un autre.

DM-MOVE ::= ABSTRACT-OPERATION

```
ARGUMENT      DM-MOVE-Arguments
RESULT        DM-MOVE-Result
ERRORS
{
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-AttributeError,
    DM-UpdateError,
    DM-ServiceError
}
```

DM-MOVE-Arguments ::= SEQUENCE

```
{
    source      [0] ManipulationObject,
    destination [1] ManipulationObject,
    position    [2] ObjectPosition
}
```

DM-MOVE-Result ::= SEQUENCE

```
{
    newConstituent [0] ManipulationObject
}
```

9.3.12.1 Arguments

Le paramètre *source* comporte une expression de localisation de base (de type "Basic-location-expression") permettant d'identifier le document sélectionné et le fragment de document à déplacer. Le paramètre *destination* identifie le document et le fragment de document destinataires du déplacement. Il comporte un localisateur de constituant (de type "Constituent-Locator"). Le paramètre *position* identifie la position du fragment de document dans le document destinataire, qui peut être le document d'origine ou un document différent.

9.3.12.2 Résultat

En cas de succès, le résultat DM-MOVE-Result sera retourné. Le paramètre "newConstituent" identifie le nouvel objet au moyen d'un identificateur de constituant (de type "Constituent-identifiant").

9.3.12.3 Erreurs

Les motifs d'erreur de déplacement sont les suivants:

- DM-AccessError: l'identification de document source ou destinataire n'est pas ouvert, est réservé par un autre utilisateur, ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- DM-FragmentError: le paramètre *source* ou *destination* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document source ou destinataire existant.
- DM-AttributeError: il est impossible de déplacer les valeurs d'attributs.
- DM-UpdateError: il est impossible de déplacer un fragment de document.
- DM-ServiceError: l'opération de déplacement ne peut pas être effectuée pour des raisons locales.

9.3.13 Service abstrait DM-REPLACE (remplacement)

Le service abstrait de remplacement est utilisé par le client DTAM-DM pour remplacer des fragments de document. Ce service supprime le fragment identifié et le remplace par le nouveau fragment de document.

DM-REPLACE ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-REPLACE-Arguments
RESULT            DM-REPLACE-Result
ERRORS
{
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-AttributeError,
    DM-UpdateError,
    DM-ServiceError
}
```

DM-REPLACE-Arguments ::= SEQUENCE

```
{
    object          [0] ManipulationObject,
    content         [1] ObjectContent
}
```

DM-REPLACE-Result ::= NULL

9.3.13.1 Arguments

Le paramètre *object* est obligatoire et comporte un localisateur de constituant (de type "Constituent-Locator") permettant d'identifier le document et le fragment de document sélectionnés pour le remplacement. Le paramètre *content* est obligatoire et contient le nouveau fragment de document à insérer.

9.3.13.2 Résultat

En cas de succès, le résultat *DM-REPLACE-Result* est retourné; il ne comporte aucun paramètre.

9.3.13.3 Erreurs

Les motifs d'erreur de remplacement sont les suivants:

- DM-AccessError: le document n'est pas ouvert, est réservé par un autre utilisateur, ou nécessite des droits d'accès dont ne dispose pas l'utilisateur.
- DM-FragmentError: le paramètre *object* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.
- DM-UpdateError: il est impossible de remplacer un fragment de document.
- DM-AttributeError: il est impossible de remplacer des valeurs des attributs.
- DM-ServiceError: l'opération de remplacement ne peut pas être effectuée pour des raisons locales.

9.3.14 Service abstrait DM-RESERVE (réservation)

Le service abstrait de réservation est utilisé par le client DTAM-DM pour réserver un fragment de document spécifié. Ce service place le fragment de document en accès exclusif, ce qui signifie que les autres utilisateurs ne pourront plus appliquer à ce fragment les services abstraits DM-GET, DM-DELETE, DM-MODIFY, DM-COPY, DM-MOVE, DM-REPLACE, DM-RESERVE, DM-UNRESERVE, DM-SEARCH et DM-POINT.

Le fragment de document est libéré implicitement par l'opération DM-CLOSE (fermeture) et explicitement par l'opération DM-UNRESERVE (déréservation).

Le processus de réservation ne s'applique pas à la propriété ou aux droits d'accès des fragments de document qui peuvent déjà avoir été définis. Pour les documents ODA, ces aspects sont pris en charge par le profil ou le sous-profil du document ODA (voir Recommandation T.414).

```

DM-RESERVE ::= ABSTRACT-OPERATION
  ARGUMENT      DM-RESERVE-Arguments
  RESULT        DM-RESERVE-Result
  ERRORS
  {
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-ReservationError,
    DM-ServiceError
  }

```

```

DM-RESERVE-Arguments ::= SEQUENCE
{
  object      [0] ManipulationObject
}

```

```

DM-RESERVE-Result ::= NULL

```

9.3.14.1 Arguments

Le paramètre *object* est obligatoire et comporte une expression de localisation (de type "Location-expression") permettant d'identifier le document et le fragment de document sélectionnés pour le service de réservation.

9.3.14.2 Résultat

En cas de succès, le résultat *DM-RESERVE-Result* est retourné; il ne comporte aucun paramètre.

9.3.14.3 Erreurs

Les motifs d'erreur de réservation sont les suivants:

- DM-AccessError: l'utilisateur n'a pas ouvert le document ou ne dispose pas des droits d'accès nécessaires.
- DM-FragmentError: le paramètre *object* ne pointe pas sur un fragment de document existant.
- DM-DocumentError: l'identificateur de document ne pointe pas vers un document existant.
- DM-ReservationError: le fragment de document est réservé.
- DM-ServiceError: l'opération de mise en réserve ne peut pas être effectuée pour des raisons locales.

9.3.15 Service abstrait DM-UNRESERVE (déréservation)

Le service abstrait de déréservation est utilisé par le client DTAM-DM pour annuler la réservation effectuée antérieurement par le service abstrait de réservation sur un fragment de document.

```

DM-UNRESERVE ::= ABSTRACT-OPERATION
  ARGUMENT      DM-UNRESERVE-Arguments
  RESULT        DM-UNRESERVE-Result
  ERRORS
  {
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-ReservationError,
    DM-ServiceError
  }

```

```

DM-UNRESERVE-Arguments ::= SEQUENCE
{
  object      [0] ManipulationObject
}

```

```

DM-UNRESERVE-Result ::= NULL

```

9.3.15.1 Arguments

Le paramètre *object* est obligatoire et comporte une expression de localisation (de type "Location-expression") permettant d'identifier le document et le fragment de document sélectionnés pour le service de déréserveation.

9.3.15.2 Résultat

En cas de succès, le résultat *DM-UNRESERVE-Result* est retourné; il ne comporte aucun paramètre.

9.3.15.3 Erreurs

Les motifs d'erreur de déréserveation sont les suivants:

- **DM-AccessError:** l'utilisateur n'a pas ouvert le document ou ne dispose pas des droits d'accès nécessaires.
- **DM-FragmentError:** le paramètre *object* ne pointe pas sur un fragment de document existant.
- **DM-DocumentError:** l'identificateur de document ne pointe pas vers un document existant.
- **DM-ReservationError:** le fragment de document n'a pas été réservé.
- **DM-ServiceError:** l'opération de déréserveation ne peut pas être effectuée pour des raisons locales.

9.3.16 Service abstrait DM-POINT (pointage)

Le service abstrait de pointage est utilisé par des clients DTAM-DM dans des applications multiutilisateurs comme la multi-édition ou le multi-affichage.

DM-POINT ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-POINT-Arguments
RESULT            DM-POINT-Result
ERRORS
{
    DM-AccessError,
    DM-DocumentError,
    DM-FragmentError,
    DM-ServiceError
}
```

DM-POINT-Arguments ::= SEQUENCE

```
{
    object          [0] ManipulationObject
}
```

DM-POINT-Result ::= NULL

9.3.16.1 Arguments

Le paramètre *object* est obligatoire et identifie le document et le fragment de document sélectionnés pour le service de pointage.

9.3.16.2 Résultat

En cas de succès, le résultat *DM-POINT-Result* est retourné; il ne comporte aucun paramètre.

9.3.16.3 Erreurs

Les motifs d'erreur de pointage sont les suivants:

- **DM-AccessError:** le document n'est pas ouvert ou est réservé par un autre utilisateur.
- **DM-FragmentError:** le paramètre *object* ne pointe pas sur un fragment de document existant.
- **DM-DocumentError:** l'identificateur de document ne pointe pas vers un document existant.
- **DM-ServiceError:** l'opération de pointage ne peut pas être effectuée pour des raisons locales.

9.3.17 Service abstrait DM-MACRO-CALL (appel de macrocommande)

Ce service fera l'objet d'un complément d'étude.

DM-MACRO-CALL ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-MACRO-CALL-Arguments
RESULT            DM-MACRO-CALL-Result
ERRORS
{
    DM-ServiceError
}
```

DM-MACRO-CALL-Arguments ::= NULL

DM-MACRO-CALL-Result ::= NULL

9.3.17.1 Arguments

Aucun.

9.3.17.2 Résultat

Aucun.

9.3.17.3 Erreurs

Les motifs d'erreur d'appel de macrocommande sont les suivants:

- **DM-ServiceError:** l'opération ne peut pas être effectuée pour des raisons locales.

9.3.18 Service abstrait DM-GROUP-BEGIN (début de groupe)

Le service abstrait de début de groupe est utilisé par le client DTAM-DM pour indiquer le début d'un ensemble de services abstraits DTAM-DM qui doivent être traités et exécutés en groupe.

Ce service fera l'objet d'un complément d'étude.

DM-GROUP-BEGIN ::= ABSTRACT-OPERATION

```
ARGUMENT          DM-GROUP-BEGIN-Arguments
RESULT            DM-GROUP-BEGIN-Result
ERRORS
{
    DM-ServiceError
}
```

DM-GROUP-BEGIN-Arguments ::=NULL

DM-GROUP-BEGIN-Result ::= NULL

9.3.18.1 Arguments

Néant.

9.3.18.2 Résultat

Néant.

9.3.18.3 Erreurs

Les motifs d'erreur de ce service abstrait sont les suivants:

- **DM-ServiceError:** l'opération de début de groupe ne peut pas être effectuée pour des raisons locales.

9.3.19 Service abstrait DM-GROUP-END (fin de groupe)

Le service abstrait de fin de groupe est utilisé par le client DTAM-DM pour indiquer la fin d'un ensemble de services abstraits DTAM-DM qui doivent être traités et exécutés en groupe.

Ce service fera l'objet d'un complément d'étude.

```

DM-GROUP-END ::= ABSTRACT-OPERATION
  ARGUMENT          DM-GROUP-END-Arguments
  RESULT            DM-GROUP-END-Result
  ERRORS
  {
    DM-ServiceError
  }

```

DM-GROUP-END-Arguments ::= NULL

DM-GROUP-END-Result ::= NULL

9.3.19.1 Arguments

Néant.

9.3.19.2 Résultat

Néant.

9.3.19.3 Erreurs

Les motifs d'erreur de ce service abstrait sont les suivants:

- **DM-ServiceError**: l'opération de fin de groupe ne peut pas être effectuée pour des raisons locales.

9.4 Définitions des erreurs abstraites du service DTAM-DM

Ce paragraphe définit et décrit les erreurs abstraites susceptibles d'être signalées en réponse à l'appel d'opérations abstraites à l'accès DTAM-DM.

Si une erreur quelconque survient durant l'appel d'une opération abstraite, le serveur DTAM-DM s'assurera qu'aucune modification n'a été apportée par l'opération abstraite.

Les erreurs spécifiées dans ce paragraphe sont les suivantes:

- **DM-AccessError** (erreur d'accès);
- **DM-AttributeError** (erreur d'attribut);
- **DM-UpdateError** (erreur de mise à jour);
- **DM-DocumentError** (erreur de document);
- **DM-FragmentError** (erreur de fragment);
- **DM-ListError** (erreur de liste);
- **DM-ReservationError** (erreur de réservation);
- **DM-ServiceError** (erreur de service).

Les erreurs non identifiées sont signalées par le paramètre approprié du service **DM-Service-Error**.

9.4.1 Erreur d'accès **DM-access-error**

Une erreur *DM-AccessError* signale un problème rencontré en tentant d'accéder à un document ou à un fragment de document spécifié dans l'argument de l'opération abstraite.

```

DM-AccessError ::= ABSTRACT-ERROR
  PARAMETER CHOICE {
    docAccessError    [0] DocAccessError,
    fragAccessError  [1] FragAccessError }

```

```

DocAccessError ::= SEQUENCE {
  documentId          [0] DocumentId,
  dmDocAccProblem     [1] DmDocAccProblem }

```

```

DmDocAccProblem ::= ENUMERATED {
  document-already-open (1),
  document-not-opened   (2),
  improper-access-rights (3) }

```

- a) **document-already-open**: le document à ouvrir a déjà été ouvert par le même utilisateur;
- b) **document-not-opened**: le document ou le fragment de document à modifier, fermer ou abandonner n'a pas encore été ouvert;
- c) **improper-access-rights**: une tentative d'accès au document a été effectuée par un client DTAM-DM ne disposant pas de droits accès suffisants pour cette entrée. Si un client DTAM-DM n'a pas de droit d'accès en lecture au document, une erreur *DM-DocumentError* sera signalée afin de cacher l'existence de ce document.

```
FragAccessError ::= SEQUENCE {
    entry                [0] ManipulationObject,
    dmFragAccProblem    [1] DmFragAccProblem }
```

```
DmFragAccProblem ::= ENUMERATED {
    reserved-by-a-user    (1),
    improper-access-rights (2) }
```

- d) **reserved-by-a-user**: le fragment de document à modifier est réservé par un autre utilisateur.
- e) **improper-access-rights**: une tentative d'accès au fragment de document a été effectuée par un client DTAM-DM ne disposant pas de droits d'accès suffisants pour cette entrée. Si un client DTAM-DM n'a pas de droit d'accès en lecture au fragment de document, une erreur *DM-DocumentError* sera signalée afin de cacher l'existence de ce document.

9.4.2 Erreur d'attribut DM-attribute-error

Une erreur *DM-AttributeError* signale un problème rencontré en tentant de modifier les attributs d'un fragment de document.

```
DM-AttributeError ::= ABSTRACT-ERROR
PARAMETER SEQUENCE {
    entry                [0] ManipulationObject,
    problems            [1] SEQUENCE OF SEQUENCE {
        problem          [0] DmAttributeProblem,
        type             [1] TypeOfAttributeValue OPTIONAL } }
```

```
DmAttributeProblem ::= ENUMERATED {
    invalid-syntax      (1),
    constraint-violation (2),
    illegal-violation   (3) }
```

```
TypeOfAttributeValue ::= CHOICE {
    odaAttributeValue   [0] AttributeValueType,
    anyAttributeValue  [1] ANY }
```

Le paramètre *problems* spécifie un ou plusieurs problèmes relatifs aux attributs. Chaque problème signalé est facultativement accompagné par la valeur de l'attribut. Les différents problèmes sont identifiés ci-dessous

- a) **invalid-syntax**: une valeur, spécifiée dans le paramètre *ModificationsType* de l'argument de l'opération abstraite, ne se conforme pas à la syntaxe abstraite du type de l'attribut;
- b) **constraint-violation**: une valeur d'attribut fournie dans l'argument de l'opération abstraite contrevient aux contraintes imposées par une norme ou par le profil d'application du document (la valeur dépasse la taille maximale autorisée par exemple);
- c) **illegal-violation**: une tentative a été effectuée pour modifier un attribut qui ne peut pas être modifié par une opération abstraite. Pour des documents ODA, les attributs "*identifier*" (identificateur), "*subordinates*" (subordonnées) et "*content portions*" (portions de contenu) d'un constituant ne sont pas modifiables (voir 7.3.3/T.413).

9.4.3 Erreur de mise à jour DM-update-error

Une erreur *DM-UpdateError* signale un problème rencontré en tentant de modifier un fragment de document. La suppression, la création, le déplacement, la copie et le remplacement de fragments de document sont aussi considérés ici comme des modifications.

```
DM-UpdateError ::= ABSTRACT-ERROR
PARAMETER SEQUENCE {
    entry                [0] ManipulationObject,
    problem             [1] DmUpdateProblem }
```

```
DmUpdateProblem ::= ENUMERATED { illegal-modification (1) }
```

Le paramètre *DmUpdateProblem* prendra l'une des valeurs suivantes:

- a) **illegal-modification**: une tentative a eu lieu pour modifier un fragment de document, qui aurait amené le document à un état incohérent. (Par exemple créer un objet logique dans la structure de mise en page d'un document ODA.)

9.4.4 Erreur de document DM-document-error

Une erreur *DM-DocumentError* signale un problème lié au nom du document spécifié dans l'argument de l'opération abstraite. Le document à l'origine du problème est signalé tel qu'il a été spécifié, avec indication du problème rencontré. Si un client DTAM-DM ne dispose pas de l'accès en lecture au document, une erreur *DM-DocumentError* sera signalée afin de cacher l'existence de ce document.

DM-DocumentError ::= ABSTRACT-ERROR

```
PARAMETER SEQUENCE {
  entry           [0] DocumentId,
  problem         [1] DmDocumentProblem }
```

DmDocumentProblem ::= ENUMERATED {

```
  invalid-upi           (1),
  invalid-path-name     (2),
  ambiguous-path-name   (3),
  invalid-document-reference (4),
  invalid-document-name (5),
  invalid-non-permanent-id (6) }
```

- a) **invalid-upi**: l'identificateur permanent unique (UPI) fourni en argument de l'opération abstraite ne désigne aucun document de la base documentaire.
- b) **invalid-path-name**: le nom *DfrPathName* (absolu ou relatif) fourni en argument de l'opération abstraite ne correspond à aucun document de la base documentaire.
- c) **ambiguous-path-name**: le nom *DfrEntryName* fourni en argument de l'opération abstraite est ambigu et correspond à plus d'un document.
- d) **invalid-document-reference**: la référence du document *Document-Reference* fournie en argument de l'opération abstraite ne correspond à aucun document existant dans la base documentaire.
- e) **invalid-document-name**: le nom *DocumentName* ne se rapporte à aucun document de la base documentaire.
- f) **invalid-non-permanent-id**: l'identificateur provisoire fourni en argument de l'opération abstraite ne se rapporte à aucun document de la base documentaire.

9.4.5 Erreur de fragment DM-fragment-error

L'erreur *DM-fragmentError* signale un problème lié au nom du fragment documentaire ou à la valeur de l'expression de la localisation spécifiée dans l'argument d'une opération abstraite. La valeur *objectValue* à l'origine du problème est signalée telle qu'elle était spécifiée, suivie d'une indication précisant le problème rencontré.

DM-FragmentError ::= ABSTRACT-ERROR

```
PARAMETER SEQUENCE {
  entry           [0] ObjectValue,
  problem         [1] DmFragmentProblem }
```

DmFragmentProblem ::= ENUMERATED {

```
  invalid-location-expression (1),
  invalid-expression          (2) }
```

- a) **invalid-location-expression**: le paramètre *Location-expression* fourni en argument de l'opération abstraite ne correspond à aucun fragment de document.
- b) **invalid-expression**: le paramètre *otherExpression* ou *anyexpression* fourni en argument de l'opération abstraite ne correspond à aucun fragment de document.

9.4.6 Erreur de liste DM-list-error

L'erreur DM-ListError signale un problème lié à la recherche de documents disponibles dans une base documentaire. Le type de recherche à l'origine du problème est signalé tel qu'il a été spécifié.

DM-ListError ::= ABSTRACT-ERROR
PARAMETER ListProblem

ListProblem ::= CHOICE {
 all [0] **BOOLEAN,**
 prefix [1] **OCTET STRING,**
 suffix [2] **OCTET STRING,**
 typeOfQuery [3] **QueryType,**
 anyCriteria [4] **ANY }**

9.4.7 Erreur de réservation DM-reservation-error

L'erreur *DM-ReservationError* signale un problème survenu lors d'une tentative de réservation ou de déréserveation d'un fragment de document.

DM-ReserveError ::= ABSTRACT-ERROR
PARAMETER SEQUENCE {
 entry [0] **ManipulationObject,**
 problem [1] **DmReservationProblem }**

DmReservationProblem ::= ENUMERATED {
 already-reserved (1),
 not-yet-reserved (2) }

Le paramètre *DmReservationProblem* prendra l'une des valeurs suivantes:

- a) **already-reserved**: une tentative de réservation ou de déréserveation a été effectuée sur un fragment de document déjà réservé par un autre utilisateur.
- b) **not-yet-reserved**: une tentative de réservation a été effectuée sur un document non réservé.

9.4.8 Erreur de service DM-service-error

L'erreur *DM-ServiceError* signale un problème lié à la fourniture du service et qui n'est dû ni à la demande incorrecte d'une opération abstraite, ni aux droits d'accès du demandeur.

DM-ServiceError ::= ABSTRACT-ERROR
PARAMETER DmServiceProblem

DmServiceProblem ::= ENUMERATED {
 server-busy (1),
 server-unavailable (2),
 resource-limit-exceeded (3),
 operation-too-complex (4),
 unclassified-server-error (5) }

Le paramètre *DmServiceProblem* prendra l'une des valeurs suivantes:

- a) **server-busy**: le serveur DTAM-DM est à cet instant trop occupé pour effectuer l'opération abstraite demandée, mais pourra l'effectuer un peu plus tard;
- b) **server-unavailable**: le serveur DTAM-DM n'est pas disponible à cet instant. L'opération proposée est de détacher le serveur DTAM-DM du client DTAM-DM;
- c) **resource-limit-exceeded**: les ressources du serveur DTAM-DM peuvent être dépassées par exemple lorsqu'un fragment de document très grand est créé ou recopié dans la base documentaire;
- d) **operation-too-complex**: l'opération abstraite demandée est syntaxiquement ou sémantiquement trop complexe (par exemple, le paramètre *location-expression* qui désigne un fragment de document ne peut pas être résolu).
- e) **unclassified-server-error**: erreur qui ne peut être classée ailleurs. Le motif de cette erreur est propre à l'application.

9.4.9 Priorité des erreurs

Si plusieurs erreurs se produisent simultanément pour une même opération abstraite DTAM-DM, une seule sera signalée au demandeur. L'ordre de priorité de ces erreurs dans l'ordre décroissant est le suivant:

- DM-ServiceError;
- DM-DocumentError;
- DM-FragmentError;
- DM-AccessError;
- DM-ReservationError;
- DM-AttributeError;
- DM-UpdateError;
- DM-ListError.

9.5 Définition des services abstraits DTAM-TK

La présente Recommandation définit les services abstraits DTAM-TK suivants.

- TK-TOKEN-GIVE (cession de jeton);
- TK-TOKEN-PLEASE (demande de jeton).

Ces services sont définis dans les paragraphes suivants.

9.5.1 Service abstrait TK-TOKEN-GIVE (cession de jeton)

Le service abstrait de cession de jeton fait passer le jeton d'application d'un client/serveur DTAM-TK qui tient temporairement le rôle de client, au client/serveur DTAM-TK qui tient temporairement le rôle de serveur. Cette opération abstraite intervertit les rôles client et serveur des entités d'application engagées dans le processus de communication.

TK-TOKEN-GIVE ::= ABSTRACT-OPERATION

ARGUMENT	TK-TOKEN-GIVE-Arguments
RESULT	TK-TOKEN-GIVE-Result
ERRORS	
{	
TK-ServiceError	
}	

TK-TOKEN-GIVE-Arguments ::= NULL

TK-TOKEN-GIVE-Result ::= NULL

9.5.1.1 Arguments

Néant.

9.5.1.2 Résultat

En cas de succès, le résultat TK-TOKEN-GIVE-Result est retourné; il ne comporte aucun paramètre.

9.5.1.3 Erreurs

Les motifs d'erreur de ce service abstrait sont les suivants:

- TK-ServiceError: l'opération de cession de jeton ne peut pas être effectuée pour des raisons locales.

9.5.2 Service abstrait TK-TOKEN-PLEASE (demande de jeton)

Le service abstrait de demande de jeton est utilisé par le client/serveur DTAM-TK, qui tient temporairement le rôle de serveur, pour demander le jeton d'application à un client/serveur DTAM-TK, qui tient temporairement le rôle client. Cette requête intervertit les rôles des entités d'application engagées dans le processus de communication.

S'il accepte l'échange, le client/serveur DTAM-TK répond par une opération de cession de jeton TK-TOKEN-GIVE. S'il refuse, il ne répond pas. Le client/serveur DTAM-TK peut répondre immédiatement après la réception de la demande de jeton ou appeler une ou plusieurs autres opérations DTAM-DM avant de décider de céder le jeton d'application.

TK-TOKEN-PLEASE ::= ABSTRACT-OPERATION

```
ARGUMENT      TK-TOKEN-PLEASE-Arguments
RESULT        TK-TOKEN-PLEASE-Result
ERRORS
{
    TK-ServiceError
}
```

TK-TOKEN-PLEASE-Arguments ::= NULL

TK-TOKEN-PLEASE-Result ::= NULL

9.5.2.1 Arguments

Néant.

9.5.2.2 Résultat

En cas de succès, le résultat TK-TOKEN-PLEASE-Result est retourné; il ne comporte aucun paramètre.

9.5.2.3 Erreurs

Les motifs d'erreur de ce service abstrait sont les suivants:

- **TK-ServiceError:** l'opération de demande de jeton ne peut pas être effectuée pour des raisons locales.

9.6 Définitions des erreurs abstraites du service DTAM-TK

Ce paragraphe définit et décrit les erreurs abstraites du service DTAM-TK qui peuvent être signalées en réponse à l'invocation d'opérations abstraites à l'accès DTAM-TK.

9.6.1 L'erreur TK-service-error

L'erreur TK-ServiceError signale un problème lié à la fourniture du service qui n'est dû ni à une demande d'opération abstraite incorrecte ni aux droits d'accès du demandeur.

TK-ServiceError ::= ABSTRACT-ERROR

PARAMETER TKServiceProblem

```
TKServiceProblem ::= ENUMERATED {
    server-busy          (1),
    server-unavailable   (2),
    unclassified-server-error (3) }
```

Le paramètre *TKServiceProblem* prendra l'une des valeurs suivantes:

- a) **server-busy:** le serveur DTAM-TK est à cet instant trop occupé pour effectuer l'opération abstraite demandée, mais pourra l'effectuer un peu plus tard.
- b) **server-unavailable:** le serveur DTAM-TK n'est pas disponible à cet instant.
- c) **unclassified-server-error:** erreur qui ne peut être classée dans les catégories précédentes. Le motif de cette erreur est propre à l'application.

Annexe A

Affectation formelle des identificateurs d'objet

(Cette annexe fait partie intégrante de la présente Recommandation)

Tous les identificateurs d'objet utilisés dans la présente Recommandation sont affectés formellement dans la présente annexe au moyen de la notation ASN.1.

```
DTAM-MAN-ObjectIdentifiers { ccitt recommendation t 435 arch(3) modules(0) object-identifiers(0) }
DEFINITIONS ::=
BEGIN
-- Prologue
-- Exporter tout
-- Ne rien emporter

ID ::= OBJECT IDENTIFIER
    id-dtam-dm          ID ::= { ccitt recommendation t 435 dm(1) } -- manipulation de document
    id-dtam-tk          ID ::= { ccitt recommendation t 435 tk(2) } -- manipulation de jeton
    id-arch             ID ::= { ccitt recommendation t 435 arch(3) } -- architecture générale

-- Catégories
    id-mod              ID ::= { id-arch 0 } -- modules

-- Modules
    id-object-identifiers ID ::= { id-mod 0 }

END -- fin du module DTAM-MAN-ObjectIdentifiers

DTAM-DM-ObjectIdentifiers { ccitt recommendation t 435 dm(1) modules(0) object-identifiers(0) }
DEFINITIONS ::=
BEGIN
-- Exporter tout

IMPORTS
    ID, id-dtam-dm
    FROM DTAM-MAN-ObjectIdentifiers { ccitt recommendation t 435 arch(3) modules(0) object-identifiers(0) };

-- Catégories
    id-mod              ID ::= { id-dtam-dm 0 } -- modules
    id-ot               ID ::= { id-dtam-dm 1 } -- objets
    id-pt               ID ::= { id-dtam-dm 2 } -- accès

-- Modules
    id-mod-object-identifiers ID ::= { id-mod 0 }
    id-mod-abstract-services ID ::= { id-mod 1 }

-- Objets
    id-ot-dtam-dm-client ID ::= { id-ot 0 }
    id-ot-dtam-dm-server ID ::= { id-ot 1 }
    id-ot-dtam-dm-dfr-client ID ::= { id-ot 2 }
    id-ot-dtam-dm-dfr-server ID ::= { id-ot 3 }

-- Accès
    id-pt-dtam-dm-port ID ::= { id-pt 0 }
    id-pt-dtam-dm-sym-port ID ::= { id-pt 1 } -- seulement pour l'environnement DTAM-DM/TK

END -- fin du module DTAM-DM-ObjectIdentifiers
```



```

DTAM-TK-ObjectIdentifiers { ccitt recommendation t 435 tk(2) modules(0) object-identifiers(0) }
DEFINITIONS ::=
BEGIN
-- Exporter tout
IMPORTS
    ID, id-dtam-tk
    FROM DTAM-MAN-ObjectIdentifiers { ccitt recommendation t 435 arch(3) modules(0) object-identifiers(0) };

-- Catégories
    id-mod          ID ::= { id-dtam-tk 0 } -- modules
    id-ot           ID ::= { id-dtam-tk 1 } -- objets
    id-pt           ID ::= { id-dtam-tk 2 } -- accès

-- Modules
    id-mod-object-identifiers ID ::= { id-mod 0 }
    id-mod-abstract-services ID ::= { id-mod 1 }

-- Objets
    id-ot-dtam-dm-tk-client-server ID ::= { id-ot 0 }

-- Accès
    id-pt-dtam-tk-port ID ::= { id-pt 0 }

END -- fin du module DTAM-TK-ObjectIdentifiers

```

Annexe B

Définition formelle des services abstraits DTAM-DM et DTAM-TK

(Cette annexe fait partie intégrante de la présente Recommandation)

Tous les services abstraits utilisés dans la présente Recommandation sont formellement définis dans cette annexe en notation ASN.1.

```

DTAM-DM-AbstractServices { ccitt recommendation t 435 dm(1) modules(0) abstract-services(1) }
DEFINITIONS IMPLICIT TAGS ::=
BEGIN
-- Exporter tout
IMPORTS
    ABSTRACT-BIND, ABSTRACT-UNBIND, ABSTRACT-ERROR, ABSTRACT-OPERATION, OBJECT, PORT
    FROM AbstractServiceNotation
    { joint-iso-ccitt mhs-motis(6) asdc(2) modules(0) notation(1) }
    -- voir Recommandation X.407

id-ot-dtam-dm-client, id-ot-dtam-dm-server,
id-ot-dtam-dm-dfr-client, id-ot-dtam-dm-dfr-server,
id-pt-dtam-dm-port

    FROM DTAM-DM-ObjectIdentifiers
    { ccitt recommendation t 435 dm(1) modules(0) object-identifiers(0) }

Non-Basic-Doc-Characteristics, Non-Basic-Struc-Characteristics
    FROM Document-Profile-Descriptor { 2 8 1 5 6 }

```

ConstituentType, AttributeValueType, QueryType, ConstituentIdentifier

```
FROM Abstract-Operations { 2 8 1 3 0 }  
-- voir Recommandation T.413
```

Location-expression, Basic-location-expression, Constituent-locator

```
FROM Location-Model { 2 8 1 12 0 }  
-- voir Recommandation T.422
```

DistinguishedName

```
FROM InformationFramework  
{ joint-iso-ccitt ds(5) modules(1) informationFramework(1) }
```

Document-Reference

```
FROM Document-Profile-Descriptor { 2 8 1 5 6 }  
-- voir Recommandation T.415
```

DfrEntryName

```
FROM DFRAbstractService { iso standard 10166 part-1(1) modules(0) abstract-service(1) };  
-- voir ISO/CEI 10166-1
```

-- Objets

```
dtam-dm-client OBJECT  
  PORTS { dtam-dm-port [ C ] }  
  ::= id-ot-dtam-dm-client  
  
dtam-dm-server OBJECT  
  PORTS { dtam-dm-port [ S ] }  
  ::= id-ot-dtam-dm-server  
  
dtam-dm-dfr-client OBJECT  
  PORTS { dtam-dm-port [ C ], dfr-port [ C ] }  
  ::= id-ot-dtam-dm-dfr-client  
  
dtam-dm-dfr-server OBJECT  
  PORTS { dtam-dm-port [ S ], dfr-port [ S ] }  
  ::= id-ot-dtam-dm-dfr-server
```

-- Accès

```
dtam-dm-port PORT  
  CONSUMER INVOKES  
  {  
    DM-DOCUMENT-OPEN,  
    DM-DOCUMENT-SAVE,  
    DM-DOCUMENT-DISCARD,  
    DM-DOCUMENT-CLOSE,  
    DM-DOCUMENT-LIST,  
    DM-GET,  
    DM-SEARCH,  
    DM-CREATE,  
    DM-DELETE,  
    DM-MODIFY,  
    DM-COPY,  
    DM-MOVE,  
    DM-REPLACE,  
    DM-RESERVE,  
    DM-UNRESERVE,  
    DM-POINT,  
    DM-MACRO-CALL,  
    DM-GROUP-BEGIN,  
    DM-GROUP-END  
  }  
  SUPPLIER INVOKES  
  {  
  }  
  ::= id-pt-dtam-dm-port
```

-- seulement dans l'environnement DTAM-DM/DTAM-TK

```
dtam-dm-sym-port  PORT
  ABSTRACT OPERATIONS {
    DM-DOCUMENT-OPEN,
    DM-DOCUMENT-SAVE,
    DM-DOCUMENT-DISCARD,
    DM-DOCUMENT-CLOSE,
    DM-DOCUMENT-LIST,
    DM-GET,
    DM-SEARCH,
    DM-CREATE,
    DM-DELETE,
    DM-MODIFY,
    DM-COPY,
    DM-MOVE,
    DM-REPLACE,
    DM-RESERVE,
    DM-UNRESERVE,
    DM-POINT,
    DM-MACRO-CALL,
    DM-GROUP-BEGIN,
    DM-GROUP-END }
  ::= id-pt-dtam-dm-sym-port
```

-- Types de données

```
DocumentId ::= CHOICE {
  dfrName           [0] DfrEntryName,
  documentReference [1] Document-Reference,
  documentName      [2] OCTET STRING,
  nonPermanentIdentifier [3] NonPermanentIdentifier
}
```

```
ManipulationObject ::= SEQUENCE {
  documentIdentifier [0] DocumentId OPTIONAL,
  objectValue       [1] ObjectValue OPTIONAL
}
```

```
ObjectValue ::= CHOICE
{
  odaExpression [0] OdaExpression,
  otherExpression [1] OCTET STRING,
  anyExpression [2] ANY
}
```

```
OdaExpression ::= CHOICE
{
  locationExpression [0] LocationExpression,
  basicLocationexpression [1] BasicLocationExpression,
  constituentLocator [2] ConstituentLocator,
  ConstituentIdentifier [3] ConstituentIdentifier
}
```

```
ObjectPosition ::= ENUMERATED {
  before (0),
  afterLastChild (1)
}
```

```
ObjectContent ::= CHOICE {
  odaContent [0] ConstituentType,
  octetAligned [1] OCTET STRING,
  anyContent [2] ANY
}
```

```
AssignedId ::= CHOICE {
  odaIdentifier [0] ConstituentIdentifier,
  nonOdaIdentifier [1] INTEGER,
  any [2] ANY
}
```

```

NonPermanentIdentifier ::= CHOICE
{
    [0] INTEGER,
    [1] OCTET STRING
}

-- Rattachement abstrait

DTAM-DMbind ::= ABSTRACT-BIND
TO {dtam-dm-port[S], dfr-port[S], dtam-dm-sym-port, dtam-tk-port} -- selon l'environnement
BIND
    ARGUMENT    DtamDmBindArgument
    RESULT      DtamDmBindResult
    BIND-ERROR  DtamDmBindError

DtamDmBindArgument ::= [0] SEQUENCE
{
    dtamProtocolVersion    [0] BIT STRING {
                            version-1 (0)} DEFAULT {version-1},
    dtamManipulationCapabilities [1] ManipulationCapabilities,
    odaApplicationCapabilities [2] SEQUENCE OF OdaApplicationCapabilities OPTIONAL,
    dfrCapabilities        [3] DfrCapabilities OPTIONAL,
    bilateralInformation    [4] SEQUENCE OF BilateralInformation OPTIONAL,
    applicationRequirements [5] ApplicationRequirements OPTIONAL
}

OdaApplicationCapabilities ::= SEQUENCE
{
    odaDocumentApplicationProfile    [0] OBJECT IDENTIFIER,
    nonBasicOdaDocumentCharacteristics [1] Non-Basic-Doc-Characteristics OPTIONAL,
    nonBasicOdaStructuralCharacteristics [2] Non-Basic-Struc-Characteristics OPTIONAL
}

ManipulationCapabilities ::= CHOICE
{
    profileSelection    [0] OBJECT IDENTIFIER,
    manipulationLevelSelection [1] ENUMERATED {
        basicReadOnlyLevel    (0),
        basicManipulationLevel (1),
        extendedLevel          (2) },
    bilateralAgreement [2] IMPLICIT OCTET STRING }

DfrCapabilities ::= SEQUENCE {
    dfrProtocolVersion [0] DfrProtocolVersion DEFAULT {version-1},
    dfrProfile          [1] OBJECT IDENTIFIER OPTIONAL }

DfrProtocolVersion ::= BIT STRING
{
    version-1 (1),
    version-2 (2)
}

BilateralInformation ::= SEQUENCE
{
    serverName    [0] ServerName,
    information   [1] EXTERNAL
}

ServerName ::= DistinguishedName

ApplicationRequirement ::= CHOICE
{
    [0] OCTET STRING,
    [1] OBJECT IDENTIFIER
}

```

```

DtamDmBindResult ::= [1] SEQUENCE
{
    serverSelectedDtamProtocolVersion    [0] BIT STRING
                                           { version-1 (0) } DEFAULT {version-1},
    odaApplicationCapabilities            [1] SEQUENCE OF OdaApplicationCapabilities,
    serverSelectedDfrVersion              [2] BIT STRING
                                           { version-1 (1),
                                           version-2 (2) } DEFAULT {version-1},
    bilateralInformation                  [3] SEQUENCE OF BilateralInformation OPTIONAL }

DtamDmBindError ::= [2] CHOICE
{
    serviceError          [0] DmServiceProblem,
    protocolError         [1] ProtocolProblem,
    profileError          [2] ProfileProblem,
    applicationError      [3] ApplicationProblem
}

ProtocolProblem ::= SEQUENCE
{
    protocol-version-not-supported    [0] SEQUENCE OF ENUMERATED {
        dtam-version-not-supported (0),
        dfr-version-not-supported (1) },
    supportedDtamProtocolVersions    [1] BIT STRING {
        version-1 (0) } OPTIONAL,
    supportedDfrProtocolVersions     [2] BIT STRING {
        version-1 (0),
        version-2 (1) } OPTIONAL
}

ProfileProblem ::= SEQUENCE
{
    profile-not-supported            [0] SEQUENCE OF ENUMERATED {
        dtam-manipulation-level-not-supported (0),
        dfr-profile-not-supported (1) },
    supportedDtamManipulationLevels [1] SEQUENCE OF ManipulationCapabilities OPTIONAL,
    supportedDfrProfiles            [2] SEQUENCE OF OBJECT IDENTIFIER OPTIONAL
}

ApplicationProblem ::= SEQUENCE OF ApplicationRequirements OPTIONAL

DTAM-DMUnBind ::= ABSTRACT-UNBIND
FROM {dtam-dm-port[S], dfr-port[S], dtam-dm-sym-port, dtam-tk-port}
-- dépendant de l'environnement

DTAM-DMAbortInfo ::= [3] SEQUENCE
{
    abortSource          [0] DtamDmAbortSource,
    abortReason          [1] DtamDmAbortReason OPTIONAL,
    userInformation      [2] EXTERNAL OPTIONAL }

DtamDmAbortSource ::= ENUMERATED
{
    dtamDmServiceUser    (0),
    dtamDmServiceProvider (1)
}

DtamDmAbortReason ::= ENUMERATED
{
    local-system-problem (0),
    temporary-problem (1),
    protocol-error (2)
}

```

-- *Opérations abstraites*

DM-DOCUMENT-OPEN ::= ABSTRACT-OPERATION

ARGUMENT **DM-DOCUMENT-OPEN-Arguments**
RESULT **DM-DOCUMENT-OPEN-Result**
ERRORS {
 DM-DocumentError,
 DM-AccessError,
 DM-ServiceError
}

DM-DOCUMENT-OPEN-Arguments ::= SEQUENCE

{
 documentId [0] **DocumentId**
 mode [1] **ModeType** **OPTIONAL**
}

ModeType ::= ENUMERATED

{
 read (0),
 modify (1),
 create (2)
}

DM-DOCUMENT-OPEN-Result ::= CHOICE {

noValue [0] **NULL**,
 nonPermanentIdentifier [1] **NonPermanentIdentifier**
}

DM-DOCUMENT-SAVE ::= ABSTRACT-OPERATION

ARGUMENT **DM-DOCUMENT-SAVE-Arguments**
RESULT **DM-DOCUMENT-SAVE-Result**
ERRORS {
 DM-AccessError,
 DM-DocumentError,
 DM-ServiceError
}

DM-DOCUMENT-SAVE-Arguments ::= SEQUENCE

{
 documentId [0] **DocumentId**
}

DM-DOCUMENT-SAVE-Result ::= NULL

DM-DOCUMENT-DISCARD ::= ABSTRACT-OPERATION

ARGUMENT **DM-DOCUMENT-DISCARD-Arguments**
RESULT **DM-DOCUMENT-DISCARD-Result**
ERRORS
{
 DM-AccessError,
 DM-DocumentError,
 DM-ServiceError
}

DM-DOCUMENT-DISCARD-Arguments ::= SEQUENCE

{
 documentId [0] **DocumentId**
}

DM-DOCUMENT-DISCARD-Result ::= NULL

DM-DOCUMENT-CLOSE ::= ABSTRACT-OPERATION

ARGUMENT **DM-DOCUMENT-CLOSE-Arguments**
RESULT **DM-DOCUMENT-CLOSE-Result**
ERRORS
{
 DM-AccessError,
 DM-DocumentError,
 DM-ServiceError
}

```

DM-DOCUMENT-CLOSE-Arguments ::= SEQUENCE
{
    documentId    [0] DocumentId
}

DM-DOCUMENT-CLOSE-Result ::= CHOICE
{
    noValue      NULL,
    permanentId [0] DocumentId
}

DM-DOCUMENT-LIST ::= ABSTRACT-OPERATION
    ARGUMENT      DM-DOCUMENT-LIST-Arguments
    RESULT        DM-DOCUMENT-LIST-Result
    ERRORS
    {
        DM-ListError,
        DM-ServiceError
    }

DM-DOCUMENT-LIST-Arguments ::= CHOICE
{
    odaCriteria    [0] QueryType,
    otherCriteria  [1] OtherCriteria,
    anyCriteria    [2] ANY
}

OtherCriteria ::= CHOICE {
    all            [0] BOOLEAN,
    prefix         [1] OCTET STRING,
    suffix         [2] OCTET STRING }

DM-DOCUMENT-LIST-Result ::= SEQUENCE OF DocumentId

DM-GET ::= ABSTRACT-OPERATION
    ARGUMENT      DM-GET-Arguments
    RESULT        DM-GET-Result
    ERRORS
    {
        DM-AccessError,
        DM-FragmentError,
        DM-DocumentError,
        DM-ServiceError
    }

DM-GET-Arguments ::= SEQUENCE
{
    object    [0] ManipulationObject
}

DM-GET-Result ::= SEQUENCE OF SEQUENCE OF ObjectContent

DM-SEARCH ::= ABSTRACT-OPERATION
    ARGUMENT      DM-SEARCH-Arguments
    RESULT        DM-SEARCH-Result
    ERRORS
    {
        DM-AccessError,
        DM-FragmentError,
        DM-DocumentError,
        DM-ServiceError
    }

DM-SEARCH-Arguments ::= SEQUENCE
{
    object    [0] ManipulationObject,
    limit     [1] INTEGER OPTIONAL
}

```

```

DM-SEARCH-Result ::=SEQUENCE
{
    objectList          [0] SEQUENCE OF DocFragment-Id,
    number              [1] INTEGER
}

DocFragment-Id ::= ObjectContent

DM-CREATE ::= ABSTRACT-OPERATION
    ARGUMENT          DM-CREATE-Arguments
    RESULT            DM-CREATE-Result
    ERRORS
    {
        DM-AccessError,
        DM-FragmentError,
        DM-DocumentError,
        DM-AttributeError,
        DM-UpdateError,
        DM-ServiceError
    }

DM-CREATE-Arguments ::= SEQUENCE
{
    destination        [0] ManipulationObject,
    position           [1] ObjectPosition,
    content             [2] SEQUENCE OF ObjectContent OPTIONAL
}

DM-CREATE-Result ::= CHOICE
{
    noValue            [0] NULL,
    assignedID         [1] SEQUENCE OF AssignedId
}

DM-DELETE ::= ABSTRACT-OPERATION
    ARGUMENT          DM-DELETE-Arguments
    RESULT            DM-DELETE-Result
    ERRORS
    {
        DM-AccessError,
        DM-FragmentError,
        DM-DocumentError,
        DM-AttributeError,
        DM-UpdateError,
        DM-ServiceError
    }

DM-DELETE-Arguments ::= SEQUENCE
{
    object             [0] ManipulationObject
}

DM-DELETE-Result ::= NULL

DM-MODIFY ::= ABSTRACT-OPERATION
    ARGUMENT          DM-MODIFY-Arguments
    RESULT            DM-MODIFY-Result
    ERRORS
    {
        DM-AccessError,
        DM-DocumentError,
        DM-FragmentError,
        DM-AttributeError,
        DM-UpdateError,
        DM-ServiceError
    }

```



```

DM-MODIFY-Arguments ::= SEQUENCE
{
    object                [0] ManipulationObject,
    modifications         [1] ModificationsType
}

DM-MODIFY-Result ::= NULL

ModificationsType ::= CHOICE
{
    odaModifications     [0] OdaModifications,
    otherModifications   [1] OCTET STRING,
    anyModifications     [2] ANY
}

OdaModifications ::= SEQUENCE
{
    attributeValue       [0] AttributeValuesType,
    deleting             [1] BOOLEAN DEFAULT FALSE
}

DM-COPY ::= ABSTRACT-OPERATION
    ARGUMENT          DM-COPY-Arguments
    RESULT            DM-COPY-Result
    ERRORS
    {
        DM-AccessError,
        DM-DocumentError,
        DM-FragmentError,
        DM-AttributeError,
        DM-UpdateError,
        DM-ServiceError
    }

DM-COPY-Arguments ::= SEQUENCE
{
    source                [0] ManipulationObject OPTIONAL,
    destination           [1] ManipulationObject OPTIONAL,
    position              [2] ObjectPosition OPTIONAL
}

DM-COPY-Result ::= SEQUENCE
{
    newConstituent       [0] ManipulationObject
}

DM-MOVE ::= ABSTRACT-OPERATION
    ARGUMENT          DM-MOVE-Arguments
    RESULT            DM-MOVE-Result
    ERRORS {
        DM-AccessError,
        DM-DocumentError,
        DM-FragmentError,
        DM-AttributeError,
        DM-UpdateError,
        DM-ServiceError
    }

DM-MOVE-Arguments ::= SEQUENCE
{
    source                [0] ManipulationObject,
    destination           [1] ManipulationObject,
    position              [2] ObjectPosition
}

```

DM-MOVE-Result ::= SEQUENCE

```
{  
    newConstituent      [0] ManipulationObject  
}
```

DM-REPLACE ::= ABSTRACT-OPERATION

```
ARGUMENT      DM-REPLACE-Arguments  
RESULT        DM-REPLACE-Result  
ERRORS  
{  
    DM-AccessError,  
    DM-DocumentError,  
    DM-FragmentError,  
    DM-AttributeError,  
    DM-UpdateError,  
    DM-ServiceError  
}
```

DM-REPLACE-Arguments ::= SEQUENCE

```
{  
    object          [0] ManipulationObject,  
    content         [1] ObjectContent  
}
```

DM-REPLACE-Result ::= NULL

DM-RESERVE ::= ABSTRACT-OPERATION

```
ARGUMENT      DM-RESERVE-Arguments  
RESULT        DM-RESERVE-Result  
ERRORS  
{  
    DM-AccessError,  
    DM-DocumentError,  
    DM-FragmentError,  
    DM-ReservationError,  
    DM-ServiceError  
}
```

DM-RESERVE-Arguments ::= SEQUENCE

```
{  
    object          [0] ManipulationObject  
}
```

DM-RESERVE-Result ::= NULL

DM-UNRESERVE ::= ABSTRACT-OPERATION

```
ARGUMENT      DM-UNRESERVE-Arguments  
RESULT        DM-UNRESERVE-Result  
ERRORS  
{  
    DM-AccessError,  
    DM-DocumentError,  
    DM-FragmentError,  
    DM-ReservationError,  
    DM-ServiceError  
}
```

DM-UNRESERVE-Arguments ::= SEQUENCE

```
{  
    object          [0] ManipulationObject  
}
```

DM-UNRESERVE-Result ::= NULL

DM-POINT ::= ABSTRACT-OPERATION

ARGUMENT **DM-POINT-Arguments**
RESULT **DM-POINT-Result**
ERRORS
{
 DM-AccessError,
 DM-DocumentError,
 DM-FragmentError,
 DM-ServiceError
}

DM-POINT-Arguments ::= SEQUENCE

{
 object **[0] ManipulationObject**
}

DM-POINT-Result ::= NULL

-- *Erreurs abstraites*

DM-AccessError ::= ABSTRACT-ERROR

PARAMETER CHOICE {
 docAccessError **[0] DocAccessError,**
 fragAccessError **[1] FragAccessError }**

DocAccessError ::= SEQUENCE {

documentId **[0] DocumentId,**
 dmDocAccProblem **[1] DmDocAccProblem }**

DmDocAccProblem ::= ENUMERATED {

document-already-open **(1),**
 document-not-opened **(2),**
 improper-access-rights **(3) }**

FragAccessError ::= SEQUENCE {

entry **[0] ManipulationObject,**
 dmFragAccProblem **[1] DmFragAccProblem }**

DmFragAccProblem ::= ENUMERATED {

reserved-by-a-user **(1),**
 improper-access-rights **(2) }**

DM-AttributeError ::= ABSTRACT-ERROR

PARAMETER SEQUENCE {
 entry **[0] ManipulationObject,**
 problems **[1] SEQUENCE OF SEQUENCE {**
 problem **[0] DmAttributeProblem,**
 type **[1] TypeOfAttributeValue OPTIONAL }**
}

DmAttributeProblem ::= ENUMERATED {

invalid-syntax **(1),**
 constraint-violation **(2),**
 illegal-violation **(3) }**

TypeOfAttributeValue ::= CHOICE {

odaAttributeValue **[0] AttributeValueType,**
 anyAttributeValue **[1] ANY }**

DM-UpdateError ::= ABSTRACT-ERROR

PARAMETER SEQUENCE {
 entry **[0] ManipulationObject,**
 problem **[1] DmUpdateProblem }**

DmUpdateProblem ::= ENUMERATED { **illegal-modification** **(1) }**

```

DM-DocumentError ::= ABSTRACT-ERROR
PARAMETER SEQUENCE {
    entry          [0] DocumentId,
    problem        [1] DmDocumentProblem }

DmDocumentProblem ::= ENUMERATED {
    invalid-upi          (1),
    invalid-path-name   (2),
    ambiguous-path-name (3),
    invalid-document-reference (4),
    invalid-document-name (5),
    invalid-non-permanent-id (6) }

DM-FragmentError ::= ABSTRACT-ERROR
PARAMETER SEQUENCE {
    entry          [0] ObjectValue,
    problem        [1] DmFragmentProblem }

DmFragmentProblem ::= ENUMERATED {
    invalid-location-expression (1),
    invalid-expression          (2) }

DM-ListError ::= ABSTRACT-ERROR
PARAMETER ListProblem

ListProblem ::= CHOICE {
    all          [0] BOOLEAN,
    prefix       [1] OCTET STRING,
    suffix       [2] OCTET STRING,
    typeOfQuery  [3] QueryType,
    anyCriteria  [4] ANY }

DM-ReserveError ::= ABSTRACT-ERROR
PARAMETER SEQUENCE {
    entry          [0] ManipulationObject,
    problem        [1] DmReservationProblem }

DmReservationProblem ::= ENUMERATED {
    already-reserved (1),
    not-yet-reserved (2) }

DM-ServiceError ::= ABSTRACT-ERROR
PARAMETER DmServiceProblem

DmServiceProblem ::= ENUMERATED
{
    server-busy          (1),
    server-unavailable   (2),
    resource-limit-exceeded (3),
    operation-too-complex (4),
    unclassified-server-error (5)
}

END -- fin du module DTAM-DM-AbstractServices

DTAM-TK-AbstractServices { ccitt recommendation t 435 tk(2) modules(0) abstract-services(1) }
DEFINITIONS IMPLICIT TAGS ::=
BEGIN
-- Exporter tout

IMPORTS

    ABSTRACT-ERROR, ABSTRACT-OPERATION, OBJECT, PORT

    FROM AbstractServiceNotation
    { joint-iso-ccitt mhs-motis(6) asdc(2) modules(0) notation(1) }

```

```

id-pt-dtam-dm-sym-port

    FROM DTAM-DM-AbstractServices
    { ccitt recommendation t 435 dm(1) modules(0) abstract-services(1) }

id-pt-dtam-tk-port,
id-ot-dtam-dm-tk-client-server

    FROM DTAM-TK-ObjectIdentifiers
    { ccitt recommendation t 435 tk(2) modules(0) object-identifiers(0) };

-- Objets

    dtam-dm-tk-client-server    OBJECT
                                PORTS { dtam-dm-sym-port, dtam-tk-port }
                                ::= id-ot-dtam-dm-tk-client-server

-- Accès

dtam-tk-port    PORT
ABSTRACT OPERATIONS {
    TK-TOKEN-GIVE,
    TK-TOKEN-PLEASE }
::= id-pt-dtam-tk-port

-- Opérations abstraites

TK-TOKEN-GIVE ::= ABSTRACT-OPERATION
    ARGUMENT      TK-TOKEN-GIVE-Arguments
    RESULT        TK-TOKEN-GIVE-Result
    ERRORS
    {
        TK-ServiceError
    }

TK-TOKEN-GIVE-Arguments ::= NULL
TK-TOKEN-GIVE-Result ::= NULL

TK-TOKEN-PLEASE ::= ABSTRACT-OPERATION
    ARGUMENT      TK-TOKEN-PLEASE-Arguments
    RESULT        TK-TOKEN-PLEASE-Result
    ERRORS
    {
        TK-ServiceError
    }

TK-TOKEN-PLEASE-Arguments ::= NULL
TK-TOKEN-PLEASE-Result ::= NULL

-- Erreurs abstraites

TK-ServiceError ::= ABSTRACT-ERROR
    PARAMETER TKServiceProblem

TKServiceProblem ::= ENUMERATED {
    server-busy          (1),
    server-unavailable   (2),
    unclassified-server-error (3) }

END -- fin du module DTAM-TK-AbstractServices

```

Appendice I

Procédures de base pour l'utilisation des services abstraits DTAM-DM

(Cet appendice ne fait pas partie intégrante de la présente Recommandation)

Les procédures de base suivantes pourraient être suivies pour utiliser les services abstraits DTAM-DM.

I.1 Etablissement d'une association DTAM

Le client DTAM-DM crée une association DTAM avec le serveur DTAM-DM au moyen du service ACSE A-ASSOCIATE, qui établit une liaison de communication. Au cours de cet établissement, le serveur DTAM communique ses capacités de couche application, y compris ses capacités de manipulation. Les capacités indiquées par le serveur sont interprétées comme étant les capacités en réception du serveur, et non comme des capacités à négocier.

I.2 Ouverture de document

Avant de manipuler les fragments d'un document, le client DTAM-DM doit d'abord ouvrir celui-ci au moyen du service abstrait d'ouverture de document. Le client DTAM-DM peut ouvrir plusieurs documents dans le cadre d'une même association DTAM.

I.3 Manipulation de document

Le client DTAM-DM peut, au moyen des services abstraits DM-* appropriés, manipuler les fragments d'un document préalablement ouvert. De plus, le client DTAM-DM peut effectuer des manipulations entre plusieurs documents, par exemple copier une partie d'un document dans un autre document.

I.4 Sauvegarde de document

Le client DTAM-DM peut enregistrer le résultat d'une séquence de manipulations effectuées sur les fragments d'un document en sauvegardant celui-ci.

I.5 Abandon de document

Le client DTAM-DM peut annuler une séquence de manipulations effectuées sur les fragments d'un document en abandonnant celui-ci. Toutes les manipulations effectuées depuis la précédente opération de sauvegarde sont alors annulées.

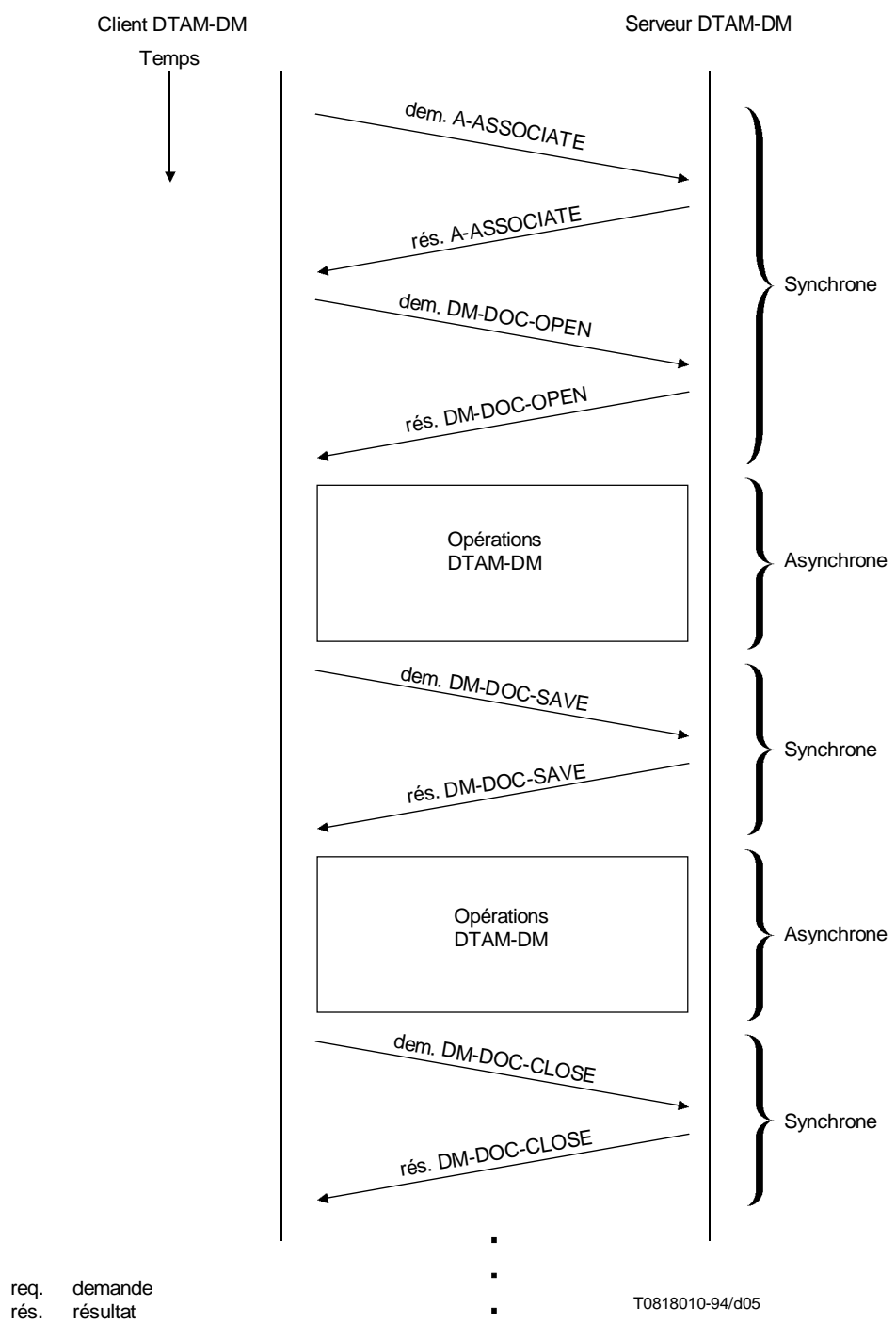
I.6 Fermeture de document

Une fois terminée la manipulation des fragments d'un document, le client DTAM-DM referme celui-ci.

I.7 Terminaison d'association DTAM

Une fois terminées toutes les activités de manipulation de documents, le client DTAM-DM met fin à l'association DTAM au moyen du service ACSE A-RELEASE qui libère la liaison de communication entre les utilisateurs DTAM.

La Figure I.1 montre un exemple d'utilisation de services abstraits DTAM-DM.



NOTE – Les seules opérations dont l’invocation et la transmission de résultat s’effectuent de façon asynchrone sont les opérations contenues dans l’unité fonctionnelle de manipulation de document DTAM-DM avec confirmation.

FIGURE I.1/T.435
Exemple d’utilisation de services abstraits DTAM-DM

Appendice II

Procédures de base pour l'utilisation combinée des services abstraits DTAM-DM et DFR

(Cet appendice ne fait pas partie intégrante de la présente Recommandation)

La configuration d'utilisation combinée des services DTAM-DM et DFR constitue un système client-serveur. Le client DTAM-DM/DFR fait appel aux services DTAM-DM et DFR tandis que le serveur DTAM-DM fournit les services DTAM-DM et DFR.

Les procédures suivantes s'appliquent à l'utilisation combinée des services abstraits DTAM-DM et DFR.

II.1 Etablissement d'association DTAM-DM/DFR

Le client DTAM-DM/DFR crée une association DTAM-DM/DFR avec le serveur DTAM-DM/DFR au moyen du service ACSE A-ASSOCIATE. Des sous-ensembles des services DTAM-DM et DFR peuvent être utilisés selon ce qui aura été convenu lors de l'établissement de l'association. Pendant l'établissement de l'association, le client et le serveur négocieront également leurs capacités d'application, y compris les capacités de manipulation des fragments de document.

II.2 Archivage et consultation de documents (utilisation des services abstraits DFR)

Après l'établissement de l'association DTAM-DM/DFR, le client DTAM-DM/DFR peut utiliser tous les services abstraits DFR, par exemple pour lire, rechercher, gérer ou supprimer des documents de base documentaire.

Le client DTAM-DM/DFR sélectionne un document au moyen d'un service DFR et en obtient l'identification. Avant de manipuler un document, le client DTAM-DM/DFR modifiera les droits d'accès des autres clients DTAM-DM/DFR à l'ensemble du document au moyen du service de réservation DFR, les empêchant ainsi d'y accéder.

II.3 Manipulation de document (utilisation des services abstraits DTAM-DM)

Le client DTAM-DM/DFR ouvrira un document avant de pouvoir en manipuler les fragments. Il peut manipuler les fragments de plusieurs documents.

Le client DTAM-DM/DFR utilisera les services abstraits DTAM-DM pour manipuler à distance les fragments de document. On trouvera de plus amples détails sur les services abstraits DTAM-DM dans l'Appendice I.

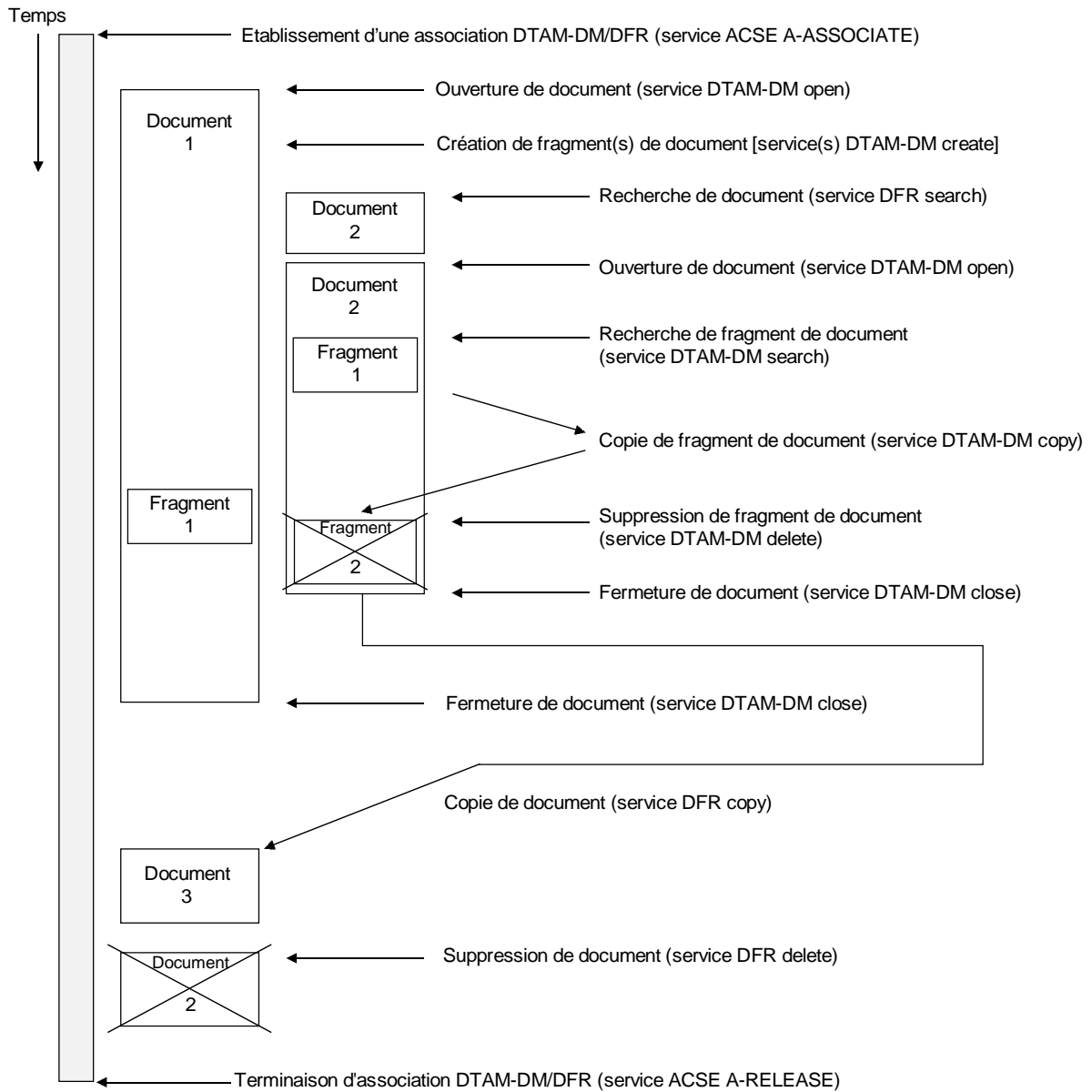
Au cours de la manipulation du document, le client DTAM-DM/DFR peut recourir à tous les services abstraits DFR qu'il jugera utiles.

Une fois terminées toutes les opérations de manipulation, le client DTAM-DM/DFR refermera le document et le déréservera au moyen du service DFR de déréserve, permettant ainsi à d'autres clients DTAM-DM/DFR d'y accéder.

II.4 Terminaison de l'association DTAM-DM/DFR

Une fois terminées toutes les activités d'archivage, de recherche et de manipulation, le client DTAM-DM/DFR met fin à l'association DTAM-DM/DFR au moyen du service ACSE A-RELEASE qui libère la liaison de communication DTAM entre client et serveur DTAM-DM/DFR.

La Figure II.1 montre un exemple d'utilisation combinée des services abstraits DTAM-DM et DFR.



T0820030-94/d06

FIGURE II.1/T.435

Exemple d'utilisation combinée des services abstraits DTAM-DM et DFR