

# Les Couches Hautes du Modèle OSI

## Notion de Services

## Sommaire

- I. Terminologie OSI**
- II. La couche Session
- III. La couche Présentation
- IV. La couche Application
- V. Les applications normalisées

## I. Terminologie OSI

### ➤ Concepts

- Sous-système (N)
  - ✓ L'élément d'un système ouvert qui correspond au niveau N et qui ne communique qu'avec les éléments de niveau N-1 et N+1
- Couche (N)
  - ✓ L'ensemble des sous-systèmes de niveau N
- Entité (N)
  - ✓ Un élément d'un sous-système (N) qui assure une partie des activités de communication du sous-système
- Entités paires
  - ✓ Entités de la même couche sur machines différentes
- La couche (N) est fournisseur de service
- La couche (N+1) est utilisatrice du service
  - ✓ Les entités de la couche (N) activent un service utilisé par la couche N+1
- Point d'accès à des services (N)
  - ✓ Un point de jonction entre une entité (N) et une entité (N+1), il est repéré par une adresse qui permet d'atteindre l'entité (N+1) à partir de la couche N
- Protocole (N)
  - ✓ L'ensemble des règles que doivent respecter les entités de la couche (N)

## I. Terminologie OSI

### ➤ Transfert des données, mode connecté/non connecté

- L'échange d'informations entre entités (N+1) paires utilise une association
  - ✓ Une connexion entre les entités (N+1)
  - ✓ Les points d'accès utilisés sont les terminaisons de cette connexion
- Le transfert de données sans connexion préalable est possible
  - ✓ Une entité (N+1) doit associer à chaque message les informations nécessaires à son acheminement

### ➤ Qualité de service

- Dans la plupart des couches, il est prévu des paramètres de qualité de service
- Une entité (N+1) peut demander à une entité (N) de garantir
  - ✓ Un débit
  - ✓ Un taux d'erreurs ...

# I. Terminologie OSI

## ➤ Les types d'échange de données

- Dans une architecture en couches, on distingue 2 types d'échange de données
  - ✓ Entre entités paires
  - ✓ Entre entités situées dans 2 couches adjacentes d'un même système
- Échange de PDU
  - ✓ Les échanges entre entités paires permettent le transfert des données d'un système vers un autre, ils utilisent les unités de données de protocole (PDU)
  - ✓ Chaque PDU comprend
    - les informations de contrôle de protocole (PCI) utilisées par les entités
    - les données de l'utilisateur (champ vide dans certaines PDU de service)

# I. Terminologie OSI

## ➤ Les types d'échange de données

- Échange de SDU
  - ✓ Les échanges entre entités de couches adjacentes permettent à une entité (N+1) d'utiliser les services offerts par une entité (N)
  - ✓ Une unité de données de service SDU est un ensemble de données transférées entre 2 entités (N+1) paires à travers une connexion dans la couche (N)
- Échange de IDU
  - ✓ L'unité d'information entre couches adjacentes est l'unité de données de l'interface, elle comprend
    - les informations de contrôle de l'interface (ICI) qui précise le service demandé
    - les données de l'interface qui sont constituées
      - soit par une SDU ou une partie de SDU fournie par l'entité (N+1) et qui sera transmise par l'entité (N) à l'une de ses paires
      - soit par une SDU ou une partie de SDU reçue par l'entité (N) et destinée à l'entité (N+1)

# I. Terminologie OSI

## ➤ Les services fournis par une couche

- Le service d'une couche est assuré par un fournisseur de service (logiciel spécifique)
- Le fournisseur de service met à la disposition de la couche immédiatement supérieure ou à l'utilisateur un ensemble de primitives (sous forme d'appel de fonctions)
- Ces services sont de deux types
  - ✓ Les services confirmés qui utilisent 4 primitives
    - demande (request)
      - demande transmise au fournisseur de service local
    - indication (indication)
      - indication d'événement reçue du fournisseur de service local
    - réponse (response)
      - acquittement au fournisseur de service local de l'indication reçue
    - confirmation (confirm)
      - indication par le fournisseur de service local de l'acquittement reçu par le fournisseur de service distant

# I. Terminologie OSI

## ➤ Les services fournis par une couche

- ✓ Les services non confirmés qui utilisent 2 primitives
  - demande (request)
    - demande transmise au fournisseur de service local
  - indication (indication)
    - indication d'événement reçue du fournisseur de service local
- Exemples
  - ✓ La connexion est toujours un service confirmé
    - (CONNECT.request, CONNECT.indication)
    - (CONNECT.response, CONNECT.confirm)
  - ✓ Le transfert de données n'est pas (toujours) confirmé
    - (DATA.request, DATA.indication)

# I. Terminologie OSI

## ➤ Les services fournis par une couche

- Ces services sont implantés sous forme d'automates, appelés machines de protocole, décrits par des tables
  - ✓ Table des états : elle contient la liste des différents états de l'automate
  - ✓ Table des événements entrants : il s'agit de primitives appelées par l'utilisateur, de primitives d'indication venant de la couche inférieure ou de l'arrivée de PDU
  - ✓ Table des événements sortants : il s'agit de primitives d'indication fournies à l'utilisateur, de primitives de demandes vers la couche inférieure ou de l'émission de PDU
  - ✓ Table des prédicats : conditions à vérifier lors de l'arrivée d'un événement dans un état donné de l'automate
  - ✓ Table des actions : elle décrit les différentes actions prévues
  - ✓ Table de fonctionnement : elle indique pour chaque état et chaque événement entrant, les prédicats à vérifier, les actions à entreprendre et le nouvel état

# Sommaire

- I. Terminologie OSI
- II. La couche Session**
- III. La couche Présentation
- IV. La couche Application
- V. Les applications normalisées

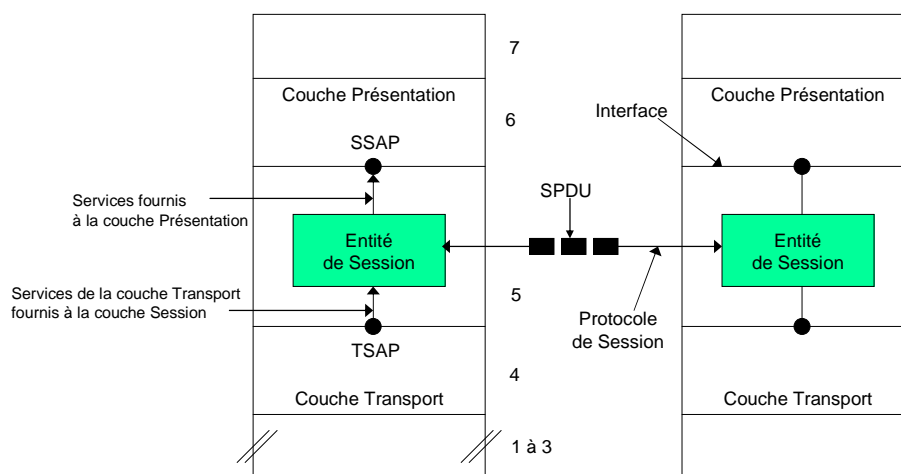
## II. La couche Session

### ➤ Rôle de la couche Session

- Fournir aux utilisateurs de session les moyens d'établir des connexions appelées *sessions* et d'y transférer des données en bon ordre
  - ✓ Exemples :
    - connexion à distance d'un terminal à une machine
    - transfert de fichiers
- Chargée des problèmes de synchronisation
  - ✓ Gérer le dialogue entre 2 utilisateurs
  - ✓ Permettre la reprise d'un traitement à la suite d'un incident survenu dans les couches basses (coupure de la connexion de Transport)
- Traitement découpé en activités et dialogues séparés par des points de synchronisation

## II. La couche Session

### ➤ Éléments de conception



## II. La couche Session

### ➤ Fonctionnalités de la couche Session

- Principales fonctionnalités :
  - ✓ Transfert de données
  - ✓ Gestion du dialogue
  - ✓ Synchronisation
  - ✓ Gestion des activités
  - ✓ Rapport d'anomalie
- La synchronisation entre les opérations de 2 utilisateurs du service de session est assuré par un système de **jetons**
  - ✓ Un jeton représente un **droit exclusif** acquis sur demande de l'une ou l'autre des entités correspondantes pour utiliser des services critiques
- quatre jetons (données, terminaison, synchronisation mineure, synchronisation majeure et activité) sont définis pour
  - ✓ Prévenir certaines collisions
  - ✓ Définir le maître temporaire d'un service
  - ✓ Associer certaines fonctions qui sont logiquement apparentées

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

- Un jeton est associé à un service, les services qui font intervenir les jetons sont tous optionnels
  - ✓ Un jeton est dit **disponible** lorsque le service correspondant a été choisi à l'établissement de la connexion
  - ✓ Dans ce cas, il peut être **attribué** à l'un des utilisateurs et celui qui le possède a seul le droit d'utiliser le service correspondant
  - ✓ Un utilisateur pour lequel le jeton disponible est **non attribué** doit s'interdire d'utiliser le service associé
  - ✓ Lorsqu'un service utilisant un jeton n'a pas été sélectionné à l'établissement de la connexion de session, le jeton associé est **indisponible**
    - les 2 utilisateurs ont les mêmes droits concernant le service
      - certains services sont à la disposition des 2 utilisateurs
      - d'autres ne sont utilisables par aucun des deux

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Conventions adoptées

- ✓ Les services de transferts de données normales et de terminaison peuvent être utilisés par les deux utilisateurs lorsque les jetons de données et de terminaison sont indisponibles
- ✓ L'indisponibilité des jetons de synchronisation et de gestion d'activité interdit l'utilisation des deux services aux utilisateurs
- ✓ Pour le service de transfert de données normales, l'indisponibilité du jeton de données signifie que l'interaction est bidirectionnelle simultanée
- ✓ Lorsque le jeton de données est disponible, l'échange se fait à l'alternat
  - l'utilisateur ne peut envoyer de données normales que si le jeton de données lui est attribué

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### A - Transfert de données

- ✓ On distingue trois phases :
  - établissement de la session
  - transfert de données
  - libération de la session
- ✓ Jeton de données
  - s'il est indisponible, la transmission s'effectue en full-duplex
  - s'il est disponible, seul l'utilisateur auquel il est attribué peut émettre des données (half-duplex)
- ✓ Jeton de terminaison
  - s'il est disponible, seul l'utilisateur qui le détient peut demander la terminaison normale de la connexion de session
  - s'il est indisponible, la terminaison normale peut être demandée par l'un quelconque des utilisateurs



## II. La couche Session

### ➤ Fonctionnalités de la couche Session

- La couche Session prévoit 4 types de données :
  - ✓ **Données normales** : leur transmission est soumise à la possession du jeton de données quand ce dernier est disponible
  - ✓ **Données exprès** : elles permettent d'échanger des messages urgents de taille limitée, indépendamment de l'état du jeton de données
  - ✓ **Données typées** : elles permettent de transférer des données sans limitation de taille, indépendamment de l'état du jeton de données (essentiellement utilisées pour le fonctionnement interne de la couche Présentation)
  - ✓ **Données de capacité** : elles permettent des échanges (en quantité limitée en v1, illimitée en v2) de données de contrôle avec confirmation avant une activité ou entre deux activités

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

- Au niveau de la couche Session, il n'y a pas de multiplexage
  - ✓ Il existe une seule connexion de Session à un instant donné sur une connexion de Transport
- La connexion de Session permet aux utilisateurs du service de négocier les unités fonctionnelles qui seront utilisées
- Les services fournis par la couche Session sont regroupés en 12 unités fonctionnelles
  - ✓ Ce sont des groupements logiques d'éléments de procédure pour permettre la négociation de leur utilisation au cours de l'établissement de la session
  - ✓ L'ensemble des unités fonctionnelles retenues déterminera les jetons disponibles et les jetons indisponibles

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les unités fonctionnelles

- ✓ Noyau :
  - cette unité fonctionnelle comprend les services indispensables au fonctionnement de la couche Session
- ✓ Terminaison négociée :
  - elle contrôle la demande de terminaison par le jeton de terminaison qui est alors disponible
- ✓ Transmission semi-duplex :
  - elle contrôle le transfert de données par le jeton de données; le transfert s'effectue dans ce cas à l'alternat et le jeton est disponible
- ✓ Transmission duplex :
  - elle autorise le transfert de données simultanément dans les 2 sens (le jeton de données est indisponible)

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les unités fonctionnelles

- ✓ Transfert de données exprès :
  - elle permet d'émettre des messages urgents de taille limitée (14 octets) non soumis au contrôle de flux des données normales ni au jeton de données
- ✓ Transfert de données typées :
  - elle fournit un flux de données indépendant des données normales et est utilisé par la couche Présentation
- ✓ Synchronisation mineure :
  - elle gère la pose de points de synchronisation mineure; le jeton de synchronisation mineure est alors disponible
- ✓ Synchronisation majeure :
  - elle gère la pose de points de synchronisation majeure ; le jeton de synchronisation majeure et d'activité est alors disponible

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les unités fonctionnelles

- ✓ Resynchronisation :
  - elle permet les reprises après un incident
- ✓ Gestion d'activités :
  - elle gère les opérations concernant les activités ainsi que la passation de contrôle; dans ce cas, le jeton de synchronisation majeure et d'activité est disponible
- ✓ Échange d'informations de capacité :
  - elle permet un échange d'information entre 2 activités; elle ne peut être utilisée que si l'unité fonctionnelle de gestion d'activités l'est aussi
- ✓ Signalisation d'anomalies :
  - elle permet à l'utilisateur et au fournisseur de signaler les problèmes rencontrés durant le traitement. Elle ne peut être utilisée que si l'unité fonctionnelle de transmission semi-duplex l'est aussi

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les profils

- ✓ Les unités fonctionnelles peuvent être regroupés en profils correspondant aux sous-ensembles d'unités fonctionnelles les plus utilisées. Trois profils sont possibles :
  - Profil de base qui contient
    - le noyau
    - la transmission semi-duplex ou la transmission duplex intégral
  - Profil avec synchronisation qui contient
    - le noyau
    - la transmission semi-duplex
    - les données typées
    - la synchronisation mineure
    - la synchronisation majeure
    - la resynchronisation
    - la terminaison négociée

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les profils

- Profil avec activités qui contient
  - Le noyau
  - La transmission semi-duplex
  - Les données avec confirmation
  - La synchronisation mineure
  - Les exceptions
  - La gestion des activités

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Établissement d'une connexion de session

- ✓ Lorsqu'un utilisateur émet la primitive S-CONNECT.request, le fournisseur de service de Session émet une primitive T-CONNECT.request pour établir la connexion de Transport
- ✓ L'ouverture d'une session nécessite la négociation de nombreux paramètres entre les utilisateurs des extrémités :
  - certains concernent la connexion de Transport et sont passés à la couche Transport sans aucune modification :
    - QoS (délai d'établissement d'une connexion, débit, temps de transit, taux d'erreurs résiduelles, ...)
    - existence ou non de données exprès, etc ...

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Établissement d'une connexion de session

➤ d'autres paramètres sont spécifiques à la couche Session :

- identificateur de la session (repérer une connexion sans ambiguïté, reprise en cas d'erreur)
- numéro de version du protocole de session
- SSAP de l'appelant et de l'appelé
- unités fonctionnelles souhaitées (autre que le noyau)
- attribution initiale des jetons
- valeur initiale du numéro de série des points de synchronisation
- données de l'utilisateur (taille max de TSUD)

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Établissement d'une connexion de session

- ✓ La primitive S-CONNECT.indication contient les mêmes paramètres que la requête, mais le fournisseur du service peut modifier les paramètres de qualité de service en fonction de ses possibilités
- ✓ La primitive S-CONNECT.response est utilisée pour l'acceptation ou le refus de la connexion et contient les paramètres suivants
  - identificateur de la session
  - numéro de version du protocole de session
  - SSAP de l'appelant et du répondeur
  - résultat de la connexion (acceptation, refus)
  - unités fonctionnelles à utiliser
  - attribution initiale des jetons (préciser les jetons disponibles)
  - valeur initiale du numéro de série des points de synchronisation
  - données de l'utilisateur (taille max de TSUD)

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Établissement d'une connexion de session

- ✓ La primitive S-CONNECT.confirm contient les mêmes paramètres que la réponse, en cas d'acceptation ou de refus
- ✓ En cas de refus par le fournisseur du service, le paramètre résultat est égal à "refus par le fournisseur du service" et peut être accompagné par l'une des réponses suivantes
  - raison non précisée
  - engorgement au niveau du fournisseur de service
  - adresse du SSAP appelé inconnue
  - utilisateur du service de session non connecté au SSAP

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Libération d'une connexion de session

- ✓ Elle peut être effectuée, en bon ordre, à l'initiative d'un quelconque utilisateur, par un service de terminaison normale
  - pas de perte de données
  - elle ne devient effective qu'après la bonne livraison de toutes les données en cours de transfert
  - un service plus évolué, de terminaison négociée, permet à l'utilisateur distant de refuser la libération de la connexion de Session
- ✓ Elle peut être effectuée de façon anormale par coupure (ou abandon) qui correspond à une libération rapide de la connexion de Session
  - qui se produit à l'initiative de l'un des utilisateurs
  - qui est déclenchée par le fournisseur du service de session à la suite d'un incident irrécupérable du réseau
- ✓ La coupure de la connexion de Session entraîne la perte des données qui n'ont pas été remises

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Libération d'une connexion de session

- ✓ La primitive S-RELEASE.request comporte un seul paramètre
  - les données de l'utilisateur
- ✓ La primitive S-RELEASE.indication ne peut contenir qu'un seul paramètre
  - les données de l'utilisateur
- ✓ La primitive S-RELEASE.response comporte 2 paramètres
  - résultat (acceptation ou refus)
  - les données de l'utilisateur
- ✓ La primitive S-RELEASE.confirm contient les mêmes paramètres que la réponse

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Libération anormale d'une connexion de session

- ✓ La terminaison anormale peut être demandée par l'utilisateur ou par le fournisseur de service en cas d'incident
  - elle peut entraîner une perte des données
- ✓ **Primitives de coupure par l'utilisateur**
  - la primitive S-U-ABORT.request comporte un seul paramètre
    - les données de l'utilisateur
  - la primitive S- U-ABORT.indication ne peut contenir que les données de l'utilisateur
- ✓ **Primitives de coupure par le fournisseur du service de session**
  - la primitive S- P-ABORT.indication comporte un paramètre qui est la raison de la coupure
    - coupure de la connexion de transport
    - erreur de protocole
    - cause indéterminée

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de transfert de données

- ✓ La couche Session peut gérer 4 types de données, seul l'échange d'informations de capacité est un service confirmé

#### ✓ Primitives de transfert de données normales

- la primitive S-DATA.request comporte un seul paramètre
  - les données de l'utilisateur
- la primitive S-DATA.indication ne peut contenir que les données reçues de l'utilisateur

#### ✓ Conditions

- en semi-duplex : jeton de données disponible et attribué au demandeur
- en duplex : aucune

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de transfert de données

#### ✓ Primitives de transfert de données exprès

- la primitive S-EXPEDITED-DATA.request comporte un seul paramètre
  - les données exprès limitées à 14 octets
- la primitive S-EXPEDITED-DATA.indication ne peut contenir que les données exprès reçues

#### ✓ Conditions

- unité fonctionnelle de transfert de données exprès

#### ✓ Primitives de transfert de données typées

- la primitive S-TYPED-DATA.request comporte un seul paramètre
  - les données typées dont la taille n'est pas limitée
- la primitive S-TYPED-DATA.indication ne peut contenir que les données typées reçues

#### ✓ Conditions

- unité fonctionnelle de transfert de données typées



## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de transfert de données

##### ✓ Primitives de transfert de données de capacité

- la primitive S-CAPABILITY-DATA.request avec comme seul paramètre les informations de capacité limitées à 512 octets en v1, illimitées en v2
- la primitive S-CAPABILITY-DATA.indication avec comme paramètre les données de capacité
- la primitive S-CAPABILITY-DATA.response avec éventuellement comme paramètre les données de capacité en réponse
- la primitive S-CAPABILITY-DATA.confirm contient le même paramètre que la réponse

##### ✓ Conditions

- unité fonctionnelle de transfert de données de capacité
- aucune activité en cours
- jeton de données indisponible ou attribué au demandeur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### B - Gestion du dialogue

- ✓ En principe, toutes les connexions ISO sont bidirectionnelles simultanées
  - des PDUs transférées dans les deux sens simultanément sur la même connexion
- ✓ Cependant, possibilité d'instaurer un tour de rôle
  - des communications bidirectionnelles à l'alternat
- ✓ Le contrôle de ce tour de rôle est appelé gestion du dialogue
  - c'est un service mis en œuvre grâce à l'utilisation du jeton de données

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Gestion des jetons

- ✓ Des primitives permettent l'échange des jetons entre les utilisateurs, il en existe 3 types
- ✓ **Primitives de cession de jetons** : elle permettent à un utilisateur de donner à son homologue un ou plusieurs jetons qu'il possède
  - la primitive S-TOKEN-GIVE.request comporte un seul paramètre indiquant le ou les jetons cédés
  - la primitive S-TOKEN-GIVE-DATA.indication avec comme paramètre les jetons cédés
- ✓ Conditions
  - les jetons cédés doivent être disponibles et attribués au demandeur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Gestion des jetons

- ✓ **Primitives de demande de jetons** : elles permettent à un utilisateur qui ne possède pas le ou les jetons requis pour effectuer une opération de les demander à son homologue
  - la primitive S-TOKEN-PLEASE.request comporte 2 paramètres
    - listes des jetons demandés : ils doivent être disponibles et attribués à l'autre demandeur
    - les données de l'utilisateur
  - la primitive S-TOKEN-PLEASE.indication comporte les mêmes paramètres que la requête
- ✓ Conditions
  - les jetons demandés doivent être disponibles et non attribués au demandeur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Gestion des jetons

- ✓ **Primitives de passation de contrôle** : elles permettent à un utilisateur de céder tous les jetons en sa possession à son homologue, ce service est lié aux services de gestion d'activités
  - la primitive S-CONTROL-GIVE.request comporte, à partir de la v2, un paramètre de données de l'utilisateur
  - la primitive S-CONTROL-GIVE.indication comporte les mêmes paramètres que la requête
- ✓ **Conditions**
  - unité fonctionnelle de gestion d'activités
  - aucune activité en cours
  - jeton de données indisponible ou attribué au demandeur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### C - Synchronisation

- ✓ Service utilisé pour mettre les entités dans un état connu des deux interlocuteurs en cas d'erreur ou de désaccord
- ✓ L'approche utilisée consiste à offrir aux utilisateurs la possibilité de repérer des instants significatifs de la communication par des **points de synchronisation** qui délimitent des **unités logiques** et permettent un pointage (checkpointing) portant sur des groupes d'unités logiques
  - chaque point de synchronisation est identifié par un numéro de série attribué par le fournisseur de service, et qui est le même aux deux extrémités.
- ✓ **Important**
  - c'est un service indispensable en cas d'erreur liées aux couches supérieures
  - nécessité d'insérer des points de synchronisation (majeure, mineure)

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Synchronisation

- ✓ On distingue deux types de points de synchronisation :
  - les **points de synchronisation majeure** : ils servent à découper les échanges de données en unités indépendantes appelées **les unités de dialogue**
  - les points de synchronisation majeure doivent être confirmés et séparent totalement les données de deux dialogues consécutifs :
    - après avoir émis une demande de pose de point de synchronisation majeure, une entité de session doit en attendre la confirmation sans émettre d'autres données
    - une entité qui reçoit une indication de pose de point de synchronisation majeure doit, après avoir mis en sécurité les données reçues au cours du dialogue, confirmer ce point de synchronisation

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Synchronisation

- ✓ Les **points de synchronisation mineure** : ils servent à poser des points de repère à l'intérieur d'un dialogue
  - leur confirmation est facultative et peut être demandé par l'utilisateur qui pose le point de synchronisation
  - à la différence des points de synchronisation majeure, l'entité de session peut continuer à envoyer des données après le point de synchronisation sans attendre de confirmation
    - ils ne garantissent pas une séparation nette entre les données échangées avant et les données échangées après
  - le point de synchronisation mineure est utilisé par certaines applications pour définir une fenêtre d'émission
    - une telle fenêtre est définie par le nombre maximal de points de synchronisation mineure qui peuvent être en attente de confirmation à un instant donné

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Synchronisation

- ✓ La synchronisation entre les opérations des deux utilisateurs du service de Session est assurée par un système de jetons :

- jeton de synchronisation majeur et d'activité

- pour qu'un utilisateur puisse poser un point de synchronisation majeur ou démarrer une activité, il est nécessaire que ce jeton soit disponible et en sa possession
- ce jeton est à la fois associé à la pose d'un point de synchronisation majeur et aux activités car la fin d'une activité implique la pose d'un point de synchronisation majeur

- jeton de synchronisation mineur

- pour qu'un utilisateur puisse poser un point de synchronisation mineur, il est nécessaire que ce jeton soit disponible et en sa possession

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Synchronisation

- ✓ Lorsqu'un échange est interrompu, il faut le reprendre dans un état cohérent
- ✓ Au cours des échanges, les utilisateurs peuvent définir des points de synchronisation qui serviront de repère lors d'une reprise après incident
  - c'est le rôle du service de resynchronisation qui permet une reprise de traitement en cours par retransmission des dernières données émises ou par abandon d'une partie de ce traitement selon l'option demandée
  - ceci implique une réallocation des jetons et la fixation d'un numéro de série pour les points de synchronisation futurs

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de synchronisation mineure

- ✓ La primitive S-SYNC-MINOR.request comporte les paramètres
  - type : précise si une confirmation est demandée ou non
  - premier numéro de série de point de synchronisation
  - données de l'utilisateur
- ✓ La primitive S-SYNC-MINOR.indication comporte les mêmes paramètres que la requête
- ✓ La primitive S-SYNC-MINOR.response comporte les paramètres
  - premier numéro de série de point de synchronisation
  - données de l'utilisateur
- ✓ La primitive S-SYNC-MINOR.confirm comporte les mêmes paramètres que la réponse
- ✓ Conditions
  - unité fonctionnelle de synchronisation mineure
  - activité en cours
  - jeton de données indisponible ou attribué au demandeur
  - jeton de synchronisation mineure disponible et attribué au demandeur

O. Baala - RE52

Couches Hautes - Notion de Services

43

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de synchronisation majeure

- ✓ La primitive S-SYNC-MAJOR.request comporte les paramètres
  - premier numéro de série de point de synchronisation
  - données de l'utilisateur
- ✓ La primitive S-SYNC-MAJOR.indication comporte les mêmes paramètres que la requête
- ✓ La primitive S-SYNC-MAJOR.response comporte les paramètres
  - données de l'utilisateur
- ✓ La primitive S-SYNC-MAJOR.confirm comporte les mêmes paramètres que la réponse
- ✓ Conditions
  - unité fonctionnelle de synchronisation majeure
  - activité en cours
  - jeton de données indisponible ou attribué au demandeur
  - jeton de synchronisation mineure indisponible ou attribué au demandeur
  - jeton de synchronisation majeure et d'activité disponible et attribué au demandeur

O. Baala - RE52

Couches Hautes - Notion de Services

44

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### D - Gestion des activités

- ✓ La communication sur une connexion de Session peut être organisée sous forme très structurée avec des **activités**
  - une activité représente une **unité logique de travail**, désigné par un identificateur défini par l'utilisateur
  - c'est un découpage logique du flot des messages en activités indépendantes
  - les activités sont délimitées par des points de synchronisation majeurs et peuvent regrouper **plusieurs unités de dialogue**
  - une activité peut être suspendue puis reprise ultérieurement à un point de synchronisation pour être poursuivie jusqu'à sa fin normale
    - utile pour les tâches plus urgentes
  - si une activité est interrompue avant sa fin normale, elle peut être reprise à partir du dernier point de synchronisation confirmé sans perte de données
    - cette opération s'apparente à une **resynchronisation avec redémarrage**

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de gestion d'activités

- ✓ Les primitives de gestion d'activités se répartissent en 5 groupes : démarrage, terminaison, interruption, reprise et abandon
  - les activités modifient la valeur du numéro de série du point de synchronisation
- ✓ **Primitives de démarrage d'activités**
  - La primitive S-ACTIVITY-START.request comporte les paramètres
    - identificateur de l'activité
    - données de l'utilisateur
  - La primitive S-ACTIVITY-START.indication comporte les mêmes paramètres que la requête
  - Conditions
    - unité fonctionnelle de gestion d'activités
    - aucune activité en cours
    - jeton de données indisponible ou attribué au demandeur
    - jeton de synchronisation mineure indisponible ou attribué au demandeur
    - jeton de synchronisation majeure et d'activité disponible et attribué au demandeur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de gestion d'activités

✓ **Primitives de terminaison d'activités** : la terminaison d'une activité correspond à la pose d'un point de MA

- La primitive S-ACTIVITY-END.request comporte les paramètres
  - premier numéro de série de point de synchronisation
  - données de l'utilisateur
- La primitive S-ACTIVITY-END.indication comporte les mêmes paramètres que la requête
- La primitive S-ACTIVITY-END.response comporte les paramètres
  - données de l'utilisateur
- La primitive S-ACTIVITY-END.confirm comporte les mêmes paramètres que la réponse
- Conditions
  - unité fonctionnelle de gestion d'activités
  - aucune activité en cours
  - jeton de données indisponible ou attribué au demandeur
  - jeton de synchronisation mineure indisponible ou attribué au demandeur
  - jeton de synchronisation majeure et d'activité disponible et attribué au demandeur

O. Baala - RE52

Couches Hautes - Notion de Services

47

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de gestion d'activités

✓ **Primitives d'interruption d'activités**

- La primitive S-ACTIVITY-INTERRUPT.request comporte les paramètres
  - raison (erreur locale, erreur de séquence, demande de jeton de données, capacité de réception de l'utilisateur compromise, ...)
  - données de l'utilisateur
- La primitive S-ACTIVITY-INTERRUPT.indication comporte les mêmes paramètres que la requête
- La primitive S-ACTIVITY-INTERRUPT.response comporte les paramètres
  - données de l'utilisateur
- La primitive S-ACTIVITY-INTERRUPT.confirm comporte les mêmes paramètres que la réponse
- Conditions
  - unité fonctionnelle de gestion d'activités
  - activité en cours
  - jeton de synchronisation majeure et d'activité disponible et attribué au demandeur

O. Baala - RE52

Couches Hautes - Notion de Services

48



## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de gestion d'activités

- ✓ **Primitives de reprise d'activités** : une activité peut être reprise après un incident ou une interruption volontaire durant la même connexion ou dans une autre connexion de session

- La primitive S-ACTIVITY-RESUME.request comporte les paramètres
  - nouvel identificateur de l'activité
  - ancien identificateur de l'activité
  - premier numéro de série de point de synchronisation
  - identificateur de l'ancienne connexion
  - données de l'utilisateur
- La primitive S-ACTIVITY-RESUME.indication comporte les mêmes paramètres que la requête
- Conditions
  - unité fonctionnelle de gestion d'activités
  - aucune activité en cours
  - jeton de données indisponible ou attribué au demandeur
  - jeton de synchronisation mineure indisponible ou attribué au demandeur
  - jeton de synchronisation majeure et d'activité disponible et attribué au demandeur

O. Baala - RE52

Couches Hautes - Notion de Services

49

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de gestion d'activités

- ✓ **Primitives d'abandon d'activités** : ce service est analogue à l'interruption d'activités mais indique qu'il n'y aura pas de reprise ultérieure

- La primitive S-ACTIVITY-DISCARD.request comporte les paramètres
  - raison
  - données de l'utilisateur
- La primitive S-ACTIVITY-DISCARD.indication comporte les mêmes paramètres que la requête
- La primitive S-ACTIVITY-DISCARD.response comporte les paramètres
  - données de l'utilisateur
- La primitive S-ACTIVITY-DISCARD.confirm comporte les mêmes paramètres que la réponse
- Conditions
  - unité fonctionnelle de gestion d'activités
  - activité en cours
  - jeton de synchronisation majeure et d'activité disponible et attribué au demandeur

O. Baala - RE52

Couches Hautes - Notion de Services

50

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Gestion des activités

- ✓ Les points de synchronisation peuvent servir de repère pour permettre la reprise (recovery) à la suite d'une erreur ou d'un incident
  - l'utilisateur doit posséder une copie des données transmises jusqu'à ce qu'il obtienne une confirmation de leur bonne réception (Ack du MA ou MI)
- ✓ Afin de faciliter la mise en œuvre de la reprise, la couche Session fournit un service de resynchronisation qui sert à mettre la connexion dans un état défini, il comprend
  - une fonction de réattribution des jetons
  - une fonction de sélection d'un nouveau numéro de série de point de synchronisation
- ✓ A la fin normale d'une resynchronisation
  - la connexion de Session est repositionnée sur un nouveau point de synchronisation
  - les données non remises sont purgées
  - les jetons sont réattribués

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Gestion des activités

- ✓ Le service de resynchronisation peut être offert avec 3 options
  - Redémarrage (Restart) : permet au utilisateurs de négocier le retour en un point de synchronisation antérieur convenu (ayant ou non fait l'objet d'une confirmation)
    - reprise à partir d'un point antérieur au dernier point MA confirmé
    - reprise correcte du traitement au niveau d'un point MA ou MI
    - la réémission des données suivant le point de synchronisation reste à la charge de l'utilisateur
  - Abandon (Abort) : sert à demander au fournisseur du service de Session de resynchroniser la connexion en un point dont le numéro de série est supérieur à la valeur courante
    - abandon du traitement en cours : certaines données sont irrémédiablement perdues
  - Choix ou Position (Set) : permet de repositionner la connexion de Session sur un point de synchronisation quelconque choisi par l'utilisateur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de resynchronisation

- ✓ La primitive S-RESYNCHRONIZE.request comporte les paramètres
  - premier type de resynchronisation (abandon, redémarrage, choix)
  - premier numéro de série de point de synchronisation
  - nouvel état des jetons
  - données de l'utilisateur
- ✓ La primitive S-RESYNCHRONIZE.indication comporte les mêmes paramètres que la requête
- ✓ La primitive S-RESYNCHRONIZE.response comporte les paramètres
  - premier type de resynchronisation (égal au type présent dans l'indication)
  - premier numéro de série de point de synchronisation
  - état des jetons
  - données de l'utilisateur
- ✓ La primitive S-RESYNCHRONIZE.confirm comporte les mêmes paramètres que la réponse
- ✓ Conditions
  - unité fonctionnelle de resynchronisation
  - activité en cours

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### E - Signalisation d'anomalies

- ✓ Bien que le service de Transport soit fiable, des incidents ou erreurs peuvent se manifester au niveau Session
  - déficience du fournisseur (équipement défectueux)
  - incapacité de l'utilisateur à traiter les données reçues
- ✓ Le protocole de Session doit donc comporter des mécanismes qui apportent une réponse appropriée aux différentes anomalies susceptibles de se produire
- ✓ Une stratégie simple consiste à
  - provoquer la coupure de la connexion de Session
  - et libérer la connexion de Transport lorsque se produit une erreur de protocole

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Signalisation d'anomalies

- ✓ La responsabilité d'une restauration ultérieure repose entièrement sur les utilisateurs
- ✓ La coupure de la connexion de Session et la libération de la connexion de Transport seront toujours utilisées en dernier ressort pour le traitement d'anomalies graves
  - moduler la réponse en fonction de la gravité de l'incident : tenter une resynchronisation en cas d'incident mineur
  - les utilisateurs reçoivent des informations détaillées sur la nature des anomalies ce qui leur permet d'élaborer les réponses appropriées

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Signalisation d'anomalies

- ✓ Pour un traitement efficace des anomalies, la couche Session fournit un service de signalisation d'anomalies (Exception Reporting)
  - service disponible sur la connexion de Session lorsque l'unité fonctionnelle correspondante a été sélectionnée au moment de l'établissement de la connexion
  - service lui-même subdivisé en
    - un service de signalisation d'anomalies par l'utilisateur
    - service de signalisation d'anomalie par le fournisseur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Signalisation d'anomalies

- ✓ Le service de signalisation d'anomalies par l'utilisateur permet à un des utilisateurs d'envoyer à l'utilisateur distant l'indication motivée d'une situation anormale
  - lorsque sa capacité à traiter correctement les données reçues est compromise
- ✓ Le traitement de l'anomalie peut être effectué par une des opérations suivantes, par ordre de gravité croissante :
  - cession du jeton de données
  - resynchronisation
  - interruption ou abandon d'activité
  - coupure par l'utilisateur

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Primitives de signalisation d'anomalies

##### ✓ Par l'utilisateur

- La primitive S-U-EXCEPTION-REPORT.request comporte les paramètres
  - raison
  - données de l'utilisateur
- La primitive S-U-EXCEPTION-REPORT.indication comporte les mêmes paramètres que la requête

##### ✓ Par le fournisseur du service de Session

- La primitive S-U-EXCEPTION-REPORT.indication comporte un seul paramètre
  - raison (erreur de protocole, erreur non spécifique)

##### ✓ Conditions

- unité fonctionnelle de signalisation d'anomalies
- unité fonctionnelle de transmission semi-duplex
- jeton de données disponible et non attribué au demandeur
- activité en cours

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les unités du protocole de Session

- ✓ Des SPDUs circulent entre entités paires sans que la couche de Session n'ait à se soucier des problèmes de séquençement, de détection d'erreurs et de régulation de flux
  - utilisation de SPDUs sans numéro de séquence
- ✓ Pas de multiplexage des connexions de Session sur les connexions de Transport
  - lorsqu'une SPDU arrive, l'entité de Session destinataire connaît sans ambiguïté la connexion de Session correspondante
- ✓ Le service de base de transfert de données peut mettre en œuvre les fonctions supplémentaires suivantes
  - segmentation
  - concaténation étendue
  - transfert de données exprès
- ✓ Le service de Transport peut imposer une limitation sur la TSDU
  - ceci amène à **segmenter** les SSDUs en plusieurs SPDUs
  - à l'arrivée, l'entité destinataire procède au **réassemblage**

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les unités du protocole de Session

- ✓ La **concaténation** consiste à rassembler plusieurs SPDUs en une seule TSDU de façon à accélérer les échanges
  - les SPDUs doivent être de types différents de sorte à pouvoir les séparer sans ambiguïté à la réception
- ✓ Du point de vue de la concaténation, les SPDUs sont divisées en trois catégories
  - **catégorie 0** comprend uniquement les SPDUs GT de remise de jeton et PT de demande de jeton
  - **catégorie 1** rassemble les SPDUs qui ne sont pas normalement suivies par un transfert de jetons (i.e. SPDUs de connexion, de données exprès et de terminaison)
  - **catégorie 2** regroupe les SPDUs qui peuvent être suivies par un transfert de jetons (i.e. SPDUs de données, d'accusé de réception de point de synchronisation majeur et de données de capacités)

## II. La couche Session

### ➤ Fonctionnalités de la couche Session

#### ▪ Les unités du protocole de Session

- ✓ Le protocole de Session doit mettre en œuvre la **concaténation de base** en respectant les règles suivantes
  - SPDUs de catégorie 1 doivent toujours être envoyés isolément
  - SPDUs de catégorie 2 doivent toujours être concaténés avec des SPDUs de catégorie 0
  - SPDUs de catégorie 0 sont soit envoyés isolément, soit concaténés avec des SPDUs de catégorie 2
- ✓ Le protocole de Session offre en option la possibilité de mettre en œuvre une concaténation étendue négociée au moment de l'établissement de la connexion sous forme de paramètre de QoS (transfert avec optimisation du dialogue)
  - la concaténation de base est étendue pour permettre de concaténer jusqu'à trois SPDUs différentes de catégorie 2 à une SPDU de catégorie 0

## Sommaire

- I. Terminologie OSI
- II. La couche Session
- III. La couche Présentation**
- IV. La couche Application
- V. Les applications normalisées

## III. La couche Présentation

### ➤ Rôle de la couche Présentation

- Elle est concernée par la représentation de l'information
  - ✓ Objectif : fournir un service assurant une représentation commune de l'information échangée entre systèmes hétérogènes
  - ✓ La représentation des données (chaînes de caractères, nombres entiers, nombres flottants) à l'intérieur d'un système réel dépend de son architecture (processeur)
    - systèmes informatiques hétérogènes codent les mêmes informations selon des règles différentes (codage des types, longueur, ordre des bits dans un octet, structures de données complexes, jeux d'instructions ...)
    - besoin de transférer de l'information **mutuellement intelligible**
- Elle doit assurer une **conversion** d'une **syntaxe réelle** vers une **syntaxe commune** et réciproquement

## III. La couche Présentation

### ➤ Rôle de la couche Présentation

- Il est nécessaire de définir une représentation **de référence**, une syntaxe commune et des règles de codage communes
  - ✓ souple : utilisable dans une large gamme d'applications
  - ✓ normalisée : même signification pour tout le monde
- La couche Présentation va s'appuyer sur la couche Session pour tous les problèmes concernant la transmission de données et la resynchronisation
- Elle fournit 2 services
  - ✓ la définition des types de données (syntaxes abstraites)
  - ✓ la définition des règles de codage



## III. La couche Présentation

### ➤ Le fonctionnement de la couche Présentation

#### ▪ Concepts utilisés par la couche Présentation

- ✓ La syntaxe concrète (locale) : syntaxe utilisée par une entité d'application dans un environnement matériel et logiciel particulier pour représenter l'information
- ✓ La syntaxe abstraite : syntaxe de haut niveau permettant la spécification d'information indépendamment de tout contexte de représentation
- ✓ ASN-1 : Notation de Syntaxe Abstraite
  - une notation pour écrire des syntaxes abstraites a été spécifiée et normalisée par l'ISO
  - ASN-1 est un méta-langage utilisé pour représenter les données et leurs structures

## III. La couche Présentation

### ➤ Le fonctionnement de la couche Présentation

#### ▪ Concepts utilisés par la couche Présentation

- ✓ Aperçu de ASN-1
  - notation basée sur la notion de **Type** d'information
  - définition de balises (tag) pour les types de bases et constructeurs de structures de données

INTEGER	entier
REAL	réel
BOOLEAN	logique
BIT STRING	suite de bits
OCTET STRING	suite d'octets
ANY	type quelconque
NULL	sans type
OBJECT IDENTIFIER	nom d'objet
SEQUENCE	suite ordonnée d'éléments de divers types
SEQUENCE OF	suite ordonnée d'éléments de même type
SET	ensemble d'éléments de divers types
SET OF	ensemble d'éléments de même type
CHOICE	sélection d'un type parmi une liste

## III. La couche Présentation

### ➤ Le fonctionnement de la couche Présentation

#### ▪ Concepts utilisés par la couche Présentation

- ✓ La syntaxe de transfert : définit la représentation concrète d'une syntaxe abstraite pour la communication entre entités distantes
  - pouvoir représenter l'information sous la forme d'une suite de bits d'une façon telle que la signification et la structure de l'information ne soient pas ambiguës (interprétation identique après le transfert)
  - BER (Basic Encoding Rules) : syntaxe de transfert normalisée
    - syntaxe de transfert a été spécifiée et normalisée par l'ISO pour être utilisée pour des des syntaxes abstraites décrites en ASN-1
- ✓ Les règles de codage : permettent de passer de la syntaxe concrète locale à la syntaxe de transfert et réciproquement

## III. La couche Présentation

### ➤ Le fonctionnement de la couche Présentation

#### ▪ Concepts utilisés par la couche Présentation

- ✓ L'association d'une syntaxe abstraite et d'une syntaxe de transfert constitue un contexte de présentation
  - il est négocié entre les 2 entités utilisateurs de Présentation lors de l'établissement de la connexion. Il est repéré durant toute la connexion par un identificateur entier
- ✓ Lors de l'établissement d'une connexion de Présentation, on peut négocier un ensemble de contextes de présentation qui constituent le DCS (**Defined Context Set**)
  - l'ensemble des contextes de présentation peut être modifié par ajout ou suppression pendant la connexion
  - il existe un contexte de présentation par défaut qui peut être choisi lors de l'établissement de la connexion
    - ce contexte par défaut permet le transfert de données même quand le DCS est vide

## III. La couche Présentation

### ➤ Le fonctionnement de la couche Présentation

- La couche Présentation comporte les mêmes unités fonctionnelles que la couche Session ainsi que 3 unités fonctionnelles spécifiques
  - ✓ Le noyau de présentation : permet d'utiliser le contexte par défaut ou les contextes négociés lors de l'ouverture de la connexion. Cette unité fonctionnelle est toujours utilisée
  - ✓ La gestion de contextes : permet d'ajouter ou de supprimer des contextes de présentation durant la connexion
  - ✓ La restauration de contextes : permet, lors d'une resynchronisation de présentation, de reprendre les contextes antérieurs
- ✓ L'unité fonctionnelle de gestion de contextes modifie le fonctionnement des activités
  - lors du démarrage, une copie du DCS courant lui est attachée
  - en cas de modification de contextes, seule cette copie sera modifiée
  - une terminaison, une interruption ou un abandon restaure le DCS à la valeur qu'il avait au moment du démarrage
  - une reprise d'activité restaure le DCS du point d'interruption

## III. La couche Présentation

### ➤ La gestion des contextes de Présentation

- Un utilisateur qui souhaite ouvrir une connexion de Présentation peut proposer une liste de contextes de présentation, défini chacun par
  - ✓ Un identificateur : nombre entier unique pour la connexion de Présentation
  - ✓ Une syntaxe abstraite
  - ✓ Un contexte par défaut défini par une syntaxe abstraite
- Le fournisseur du service de présentation appelant indique pour chaque syntaxe abstraite proposée la liste des syntaxes de transfert qu'il peut utiliser
  - ✓ Syntaxes de transfert représentées par des variables ASN-1 du type OBJECT IDENTIFIER
  - ✓ Si le contexte par défaut ne peut être utilisé, la demande est rejetée

## III. La couche Présentation

### ➤ La gestion des contextes de Présentation

- Le fournisseur du service de présentation appelé examine les contextes proposés et marque ceux qu'il peut utiliser
  - ✓ Si le contexte par défaut ne peut être utilisé, la demande est rejetée
- L'utilisateur qui accepte la connexion doit
  - ✓ Pour chaque contexte de présentation utilisable par le fournisseur du service de présentation, le rejeter ou l'accepter
    - le résultat de cette sélection sera retourné au demandeur sous la forme d'une liste de résultats de définitions de contextes de présentation
  - ✓ Pour le contexte par défaut, le rejeter ou l'accepter
    - si aucune syntaxe abstraite n'est acceptée par l'utilisateur distant (liste de contexte de présentation vide) et que le contexte par défaut est accepté, il sera le seul contexte de présentation disponible
    - si le contexte par défaut est rejeté par l'utilisateur, la connexion sera rejetée

## III. La couche Présentation

### ➤ Les primitives de la couche Présentation

- Les primitives de la couche Présentation sont calquées sur celles de la couche Session à l'exception des primitives de gestion de contextes
- La plupart des primitives du service de présentation permettent la transmission de données de l'utilisateur
  - ✓ Ces données sont codées conformément aux contextes de présentation disponibles en respectant les règles suivantes
    - les données exprès doivent toujours être représentées avec le contexte par défaut
    - si le DCS est vide et pas de confirmation de modification du DCS en attente, seul le contexte par défaut peut être utilisé
    - si le DCS n'est pas vide et pas de modification du DCS cours, les données peuvent être représentées par un contexte quelconque du DCS
    - si le DCS n'est pas vide et une confirmation de modification du DCS est en attente, seules les contextes dont la suppression n'a pas été demandée sont disponibles
    - s'il s'agit d'une primitive modifiant le DCS, les données doivent être représentées par un contexte du DCS résultant de la modification (par le contexte par défaut, si la modification laisse le DCS vide)

## III. La couche Présentation

### ➤ Les primitives de la couche Présentation

- La plupart des primitives du service de présentation "traversent" les entités de présentation vers les entités de session
  - ✓ Correspondance totale (T) ou directe (pas de P-PDU)
  - ✓ Correspondance partielle (P)
    - Les paramètres de la primitive de présentation
      - Paramètres de la primitive de session
      - P-PDU avec paramètres spécifiques de présentation

Établissement de connexion	P-CONNECT	P
Transfert de données (D, EX, TD, C)	P-DATA	P
Gestion des jetons	P-TOKEN	T
Pose de point MI	P-SYNC-MINOR	T
Pose de point MA	P-SYNC-MAJOR	T
Resynchronisation	P-SYNCHRONIZE	P
Gestion d'activités	P-ACTIVITY	T
Signalisation d'anomalies (U/P)	P-U/P-EXCEPTION-REPORT	T
Terminaison de connexion	P-RELEASE	T
Rupture (U/P)	P-U/P-ABORT	T

## III. La couche Présentation

### ➤ Les primitives de gestion de contextes

- Elle permettent la modification du DCS pendant la connexion
  - ✓ La primitive P-ALTER-CONTEXT.request comporte les paramètres
    - liste des contextes à ajouter : chaque contexte est défini par un identificateur et un nom de syntaxe abstraite
    - liste des contextes à supprimer : chaque contexte est défini par l'identificateur qui lui a été associé lors de la négociation
    - données de l'utilisateur
  - ➔ Le fournisseur du service du demandeur indique pour chaque syntaxe abstraite une liste de syntaxes de transfert utilisables
  - ✓ La primitive P-ALTER-CONTEXT.indication comporte les mêmes paramètres que la requête
    - le fournisseur peut y ajouter le paramètre liste de résultats d'addition de contextes de présentation pour indiquer les syntaxes abstraites qu'il ne peut utiliser

## III. La couche Présentation

### ➤ Les primitives de gestion de contextes

- Elle permettent la modification du DCS pendant la connexion
  - ✓ La primitive P-ALTER-CONTEXT.response comporte les paramètres
    - résultat de la négociation des contextes à ajouter : liste qui indique, pour chaque syntaxe abstraite proposée, si elle est acceptée ou rejetée. L'absence de ce paramètre signifie l'acceptation de tous les contextes proposés
    - résultat de la négociation des contextes à supprimer : liste qui indique, pour chaque contexte proposé, s'il peut être supprimé non. L'absence de ce paramètre signifie l'acceptation de toutes les suppressions
    - données de l'utilisateur
  - ➔ Le fournisseur du service de l'accepteur choisira pour chaque syntaxe abstraite acceptée une syntaxe de transfert parmi la liste proposée
  - ✓ La primitive P-ALTER-CONTEXT.confirm comporte les mêmes paramètres que la réponse

## III. La couche Présentation

### ➤ Le codage des PPDU

- Jusqu'à la couche session, les PDU de la couche n sont transmises comme données de la couche n-1
- Au niveau de la couche Présentation, la situation est différente
  - ✓ Il y a correspondance biunivoque entre connexion de Présentation et connexion de Session
  - ✓ La plupart des services de présentation correspondent à des services de session
- Trois cas sont à considérer
  - ✓ La primitive de présentation correspond exactement à une primitive de session
    - pas de PDU spécifique
    - seuls les données de l'utilisateur doivent être codées conformément aux contextes disponibles
    - ce cas concerne : la terminaison normale, le gestion des jetons, la pose de points de synchronisation, la signalisation d'anomalies et la gestion d'activités

## III. La couche Présentation

### ➤ Le codage des PPDU

- Trois cas sont à considérer

- ✓ La primitive de présentation comporte des paramètres spécifiques qui n'existent pas dans la primitive de session correspondante

- dans ce cas, les paramètres spécifiques sont insérés dans une PPDU transmise comme données de l'utilisateur de la primitive de session correspondante

- ce cas concerne :

- la demande de connexion (PPDU CP)
- l'acceptation (PPDU CPA) ou le refus de connexion (PPDU CPR)
- la coupure par l'utilisateur (PPDU ARU) ou le fournisseur (PPDU ARP)
- la resynchronisation (PPDU RS et PPDU de réponse RSA)
- le transfert de données (PPDU TD) (PPDU TE) (PPDU TTD) (PPDU TC et PPDU de réponse TCC)

## III. La couche Présentation

### ➤ Le codage des PPDU

- Trois cas sont à considérer

- ✓ La primitive de présentation n'a pas de correspondance au niveau de la couche session (primitives de gestion de contextes)

- dans ce cas : la PPDU est transmise en tant que **données typées** de session (fonctionnement de la couche Présentation indépendant du jeton de données)

- ce cas concerne :

- la gestion des contextes (PPDU AC et PPDU de réponse ACA)

- Ces règles accélèrent l'établissement ou la terminaison d'une connexion de présentation car l'établissement d'une connexion comporte les étapes suivantes

- ✓ Demande de connexion de Transport

- émission d'une TPDU CN

## III. La couche Présentation

### ➤ Le codage des PPDU

- ✓ Acceptation de la connexion de Transport
  - réception d'une TPDU CC
- ✓ Demande de connexion de Session et de Présentation
  - émission d'une TPDU DT contenant une SPDU CN transportant dans le champ *données de l'utilisateur* une PPDU CP de demande de connexion
- ✓ Acceptation des connexions de session et de présentation
  - réception d'une TPDU DT contenant une SPDU AC transportant dans le champ *données de l'utilisateur* une PPDU CPA d'acceptation de connexion