

---

---

**Technologies de l'information —  
Traitement réparti ouvert — Modèle de  
référence: Sémantique architecturale**

*Information technology — Open Distributed Processing — Reference  
Model: Architectural semantics*

**iTeh STANDARD PREVIEW**  
**(standards.iteh.ai)**

[ISO/IEC 10746-4:1998](https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aaf6-fbc389642f1a/iso-iec-10746-4-1998)

[https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aaf6-  
fbc389642f1a/iso-iec-10746-4-1998](https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aaf6-fbc389642f1a/iso-iec-10746-4-1998)

## Sommaire

	<i>Page</i>
1	Domaine d'application..... 1
2	Références normatives..... 2
3	Définitions ..... 3
3.1	Termes définis dans l'ISO/CEI 8807 ..... 3
3.2	Termes définis dans la Recommandation UIT-T Z.100 ..... 3
3.3	Termes définis dans "The Z Base Standard" ..... 3
3.4	Termes définis dans l'ISO/CEI 9074 ..... 3
4	Interprétation des concepts de modélisation..... 3
4.1	Sémantique architecturale en LOTOS..... 3
4.2	Sémantique architecturale en ACT ONE..... 10
4.3	Sémantique architecturale en SDL-92..... 17
4.4	Sémantique architecturale en Z ..... 23
4.5	Sémantique architecturale en ESTELLE..... 29

## iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 10746-4:1998](https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aa6f-fbc389642f1a/iso-iec-10746-4-1998)

<https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aa6f-fbc389642f1a/iso-iec-10746-4-1998>

© ISO/CEI 1998

Droits de reproduction réservés. Sauf prescription différente, 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'ISO à l'adresse ci-après ou du comité membre de l'ISO dans le pays du demandeur.

ISO copyright office  
Case postale 56 • CH-1211 Geneva 20  
Tel. + 41 22 749 01 11  
Fax. + 41 22 734 10 79  
E-mail [copyright@iso.ch](mailto:copyright@iso.ch)  
Web [www.iso.ch](http://www.iso.ch)

Version française parue en 2000

Imprimé en Suisse

## Avant-propos

L'ISO (Organisation internationale de normalisation) et la CEI (Commission électrotechnique internationale) forment le système spécialisé de la normalisation mondiale. Les organismes nationaux membres de l'ISO ou de la CEI participent au développement de Normes internationales par l'intermédiaire des comités techniques créés par l'organisation concernée afin de s'occuper des domaines particuliers de l'activité technique. Les comités techniques de l'ISO et de la CEI collaborent dans des domaines d'intérêt commun. D'autres organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'ISO et la CEI participent également aux travaux.

Dans le domaine des technologies de l'information, l'ISO et la CEI ont créé un comité technique mixte, l'ISO/CEI JTC 1. Les projets de Normes internationales adoptés par le comité technique mixte sont soumis aux organismes nationaux pour vote. Leur publication comme Normes internationales requiert l'approbation de 75 % au moins des organismes nationaux votants.

La Norme internationale ISO/CEI 10746-4 a été élaborée par le comité technique mixte ISO/CEI JTC 1, *Technologies de l'information*, sous-comité SC 33, *Services d'applications distribuées*, en collaboration avec l'UIT-T. Le texte identique est publié en tant que Recommandation UIT-T X.904.

L'ISO/CEI 10746 comprend les parties suivantes, présentées sous le titre général *Technologies de l'information — Traitement réparti ouvert — Modèle de référence*:

- *Partie 1: Aperçu général*
- *Partie 2: Fondements*
- *Partie 3: Architecture*
- *Partie 4: Sémantique architecturale*

ITIH STANDARD PREVIEW

(standards.iteh.ai)

[ISO/IEC 10746-4:1998](#)

<https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aa6f-fbc389642f1a/iso-iec-10746-4-1998>

## Introduction

La présente Recommandation | Norme internationale fait partie intégrante du modèle de référence du traitement réparti ouvert (ODP, *open distributed processing*). Elle contient une formalisation des concepts de modélisation ODP définis dans les articles 8 et 9 de la Rec. UIT-T X.902 | ISO/CEI 10746-2. La formalisation est obtenue par l'interprétation de chaque concept en fonction des constructions des différentes techniques de description formelle normalisées.

La présente Recommandation | Norme internationale est accompagnée d'un amendement et d'un rapport technique. L'amendement s'intéresse à la formalisation du langage du point de vue informatique contenu dans la Rec. UIT-T X.903 | ISO/CEI 10746-3. Le rapport technique associé présente des exemples sur la manière possible d'appliquer des formalismes du modèle de référence ODP au développement de spécifications.

## iTeh STANDARD PREVIEW (standards.iteh.ai)

[ISO/IEC 10746-4:1998](https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aaf6-fbc389642f1a/iso-iec-10746-4-1998)

<https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aaf6-fbc389642f1a/iso-iec-10746-4-1998>

## NORME INTERNATIONALE

## RECOMMANDATION UIT-T

## TECHNOLOGIES DE L'INFORMATION – TRAITEMENT RÉPARTI OUVERT – MODÈLE DE RÉFÉRENCE: SÉMANTIQUE ARCHITECTURALE

### 1 Domaine d'application

La croissance rapide des applications réparties a fait naître le besoin d'un cadre pour coordonner la normalisation du traitement réparti ouvert (ODP, *open distributed processing*). Le modèle de référence du traitement réparti ouvert fournit ce cadre. Il établit une architecture qui permet d'intégrer la répartition, l'interfonctionnement, l'interopérabilité et la portabilité.

Le modèle de référence de base du traitement réparti ouvert (RM-ODP, *reference model of open distributed processing*) (voir les Rec. UIT-T X.901 à X.904 | ISO/CEI 10746) repose sur des concepts précis issus des développements récents dans le domaine du traitement réparti et s'appuie, dans la mesure du possible, sur l'utilisation des techniques de description formelle pour la spécification de l'architecture.

Le modèle RM-ODP se compose:

- de la Rec. UIT-T X.901 | ISO/CEI 10746-1: **Vue d'ensemble**: elle contient un aperçu général du modèle RM-ODP, en précise les motivations, le champ d'application et la justification, et propose une explication des concepts clés, ainsi qu'une présentation de l'architecture des systèmes ODP. Ce texte n'est pas normatif;
- de la Rec. UIT-T X.902 | ISO/CEI 10746-2: **Fondements**: elle contient la définition des concepts ainsi que le cadre analytique et la notation à utiliser pour la description normalisée de systèmes de traitement réparti (arbitraires). Elle s'en tient à un niveau de détail suffisant pour étayer la Rec. UIT-T X.903 | ISO/CEI 10746-3 et pour établir les prescriptions applicables à de nouvelles techniques de spécification. Ce texte est normatif;
- de la Rec. UIT-T X.903 | ISO/CEI 10746-3: **Architecture**: elle contient la spécification des caractéristiques requises pour qu'un système de traitement réparti puisse être qualifié d'ouvert. Il s'agit des contraintes que les normes ODP doivent respecter. Ce texte, qui utilise les techniques descriptives de la Rec. UIT-T X.902 | ISO/CEI 10746-2. Ce texte est normatif;
- de la Rec. UIT-T X.904 | ISO/CEI 10746-4: **Sémantique architecturale**: elle contient une formalisation des concepts de modélisation ODP définis dans les articles 8 et 9 de la Rec. UIT-T X.902 | ISO/CEI 10746-2 et une formalisation des langages de point de vue définis dans la Rec. UIT-T X.903 | ISO/CEI 10746-3. La formalisation est obtenue par l'interprétation de chaque concept en fonction des constructions des différentes techniques de description formelle normalisées. Ce texte est normatif.

La présente Recommandation | Norme internationale a pour objet de fournir une sémantique architecturale pour les systèmes ODP, ce qui se traduit par une interprétation des concepts de modélisation de base et de spécification définis dans la Rec. UIT-T X.902 | ISO/CEI 10746-2 et des langages de point de vue définis dans la Rec. UIT-T X.903 | ISO/CEI 10746-3; elle utilise les diverses caractéristiques de différents langages de spécification formelle. Une sémantique architecturale est élaborée pour quatre différents langages de spécification formelle: LOTOS, ESTELLE, SDL et Z, ce qui conduit à une formalisation de l'architecture des systèmes ODP. Un processus d'élaboration itérative et de retour a permis d'améliorer la cohérence des Rec. UIT-T X.902 | ISO/CEI 10746-2 et UIT-T X.903 | ISO/CEI 10746-3.

La mise au point d'une sémantique architecturale présente les avantages supplémentaires suivants:

- favoriser l'élaboration harmonieuse et uniforme des descriptions formelles de systèmes ODP;
- permettre une comparaison uniforme et cohérente des descriptions formelles de la même norme dans différents langages de spécification formelle.

## ISO/CEI 10746-4 : 1998 (F)

La présente Recommandation | Norme internationale contient une interprétation, non pas de tous les concepts définis dans la Rec. UIT-T X.902 | ISO/CEI 10746-2, mais uniquement des concepts les plus fondamentaux. Pour les concepts architecturaux de niveau supérieur, une sémantique est fournie indirectement par la définition de ces concepts en fonction des concepts ODP fondamentaux.

Des exemples d'utilisation de certains des langages de spécification formelle dont il est question dans la présente Spécification sont donnés dans TR 10167 (Principes directeurs pour l'application d'ESTELLE, LOTOS et SDL).

Dans les articles suivants, la numérotation des paragraphes relatifs aux concepts est conforme à la numérotation utilisée dans la Rec. UIT-T X.902 | ISO/CEI 10746-2.

La présente Recommandation | Norme internationale, qui contient la définition d'une sémantique architecturale pour les systèmes ODP, permet:

- de fournir une formalisation des concepts de modélisation ODP;
- de favoriser une élaboration harmonieuse et uniforme des descriptions formelles des normes applicables aux systèmes répartis;
- de relier les concepts de modélisation ODP et les modèles sémantiques des langages de spécification LOTOS, SDL, ESTELLE et Z;
- de constituer une base pour une comparaison uniforme et cohérente des descriptions formelles de la même norme dans les langages de spécification utilisés pour élaborer une sémantique architecturale.

Ce texte est normatif.

## 2 Références normatives

Les Recommandations et les Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation | Norme internationale. Au moment de la publication, les éditions indiquées étaient en vigueur. Toutes Recommandations et Normes sont sujettes à révision et les parties prenantes aux accords fondés sur la présente Recommandation | Norme internationale sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes Internationales indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes Internationales en vigueur. Le Bureau de la normalisation des télécommunications de l'UIT tient à jour une liste des Recommandations de l'UIT-T en vigueur.

- ISO/CEI 8807:1989, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – LOTOS – Technique de description formelle basée sur l'organisation temporelle de comportement observationnel*.
- Recommandation UIT-T Z.100 (1993), *Langage de description et de spécification du CCITT*.
- ISO/CEI TR 10167:1991, *Technologies de l'information – Interconnexion de systèmes ouverts – Principes directeurs pour l'application d'Estelle, LOTOS et SDL*.
- ISO/CEI 13568<sup>1)</sup>, *Information technology – Programming Languages their Environments and System Software Interfaces, Z Specification language*.
- *The Z Notation, A Reference Manual*, J.M. Spivey, *International Series in Computer Science, Second Edition*, Prentice-Hall International, 1992.
- ISO/CEI 9074:1997, *Technologies de l'information – Interconnexion de systèmes ouverts (OSI) – Estelle: Technique de description formelle basée sur un modèle de transition d'état étendu*.

1) Actuellement à l'état de projet.

### 3 Définitions

#### 3.1 Termes définis dans l'ISO/CEI 8807

La présente Recommandation | Norme internationale utilise les termes ci-après dont les équivalents anglais sont définis dans l'ISO/CEI 8807:

*activation, actualisation de paramètres, choix, composition parallèle, conformité, définition de processus, définition de type, définition de type paramétré, dénotation d'action, désactivation, enrichissement, entrelacement, équation, événement, événement interne observable, expression de comportement, extension, garde, instanciation, liste de paramètres formels, liste de paramètres de valeurs, liste de portes formelles, prédicat de sélection, occultation de porte, opération, porte, réduction, sorte, synchronisation.*

#### 3.2 Termes définis dans la Recommandation UIT-T Z.100

La présente Recommandation | Norme internationale utilise les termes ci-après définis dans la Rec. UIT-T Z.100:

*actif, action statement, arrêt, call, canal, clause "atleast", condition de validation, content parameter, créer, entrée, état suivant, export, finalized, import, initialisation, maintenant ("now"), procédure, procédure distante, procédure exportée, porte, "provided", réinitialisation, retard, retour, route de signaux (acheminement de signal), signal, signal continu, sortie, tâche, temporisateur, temps, transition, type de bloc, type de processus, type de service, type de système, type redéfini, type virtuel, variable exportée, variable importée, variable révélée, variable visualisée, visualisation.*

#### 3.3 Termes définis dans "The Z Base Standard"

La présente Recommandation | Norme internationale utilise les termes ci-après dont les équivalents anglais sont définis dans "The Z base standard":

*affinement de données, affinement d'opération, calcul de schéma, composition de schéma, conjonction, description axiomatique, invariant, postcondition, précondition, remplacement, schéma (opération, état, encadrement).*

<https://standards.iteh.ai/catalog/standards/sist/f4db7ed8-4e6f-45b6-aa6f-b389642f1a/iso-iec-10746-4-1998>

#### 3.4 Termes définis dans l'ISO/CEI 9074

La présente Recommandation | Norme internationale utilise les termes ci-après dont les équivalents anglais sont définis dans l'ISO/CEI 9074:

*activité, activité-système, bloc de transition, canal, clause DELAY, clause de transition, clause FROM, clause PROVIDED, clause TO, clause WHEN, connexion, déconnexion, définition de canal, définition de corps de module, définition d'en-tête de module, détachement, état de contrôle, fonction, initialisation, instance de module, instance mère, instanciation, instruction d'affectation, interaction, libération, point d'interaction, point d'interaction externe, procédure, procédure primitive, processus-système, rattachement, rôle, sortie, transition, variable exportée.*

### 4 Interprétation des concepts de modélisation

#### 4.1 Sémantique architecturale en LOTOS

Le LOTOS est un langage de spécification formelle (FSL, *formal specification language*) normalisé (ISO/CEI 8807). Un didacticiel est donné dans la norme.

Le présent paragraphe contient une explication de la manière dont les concepts de modélisation de base peuvent être exprimés en LOTOS (voir ISO/CEI 8807). Il convient de signaler qu'il existe, en LOTOS, deux principales façons de modéliser les concepts définis dans la Rec. UIT-T X.902 | ISO/CEI 10746-2. Elles sont basées sur les parties algèbre de processus et typage de données ACT ONE du langage. Comme la formalisation ACT ONE des concepts s'applique aussi au langage SDL-92, elle est donnée dans un paragraphe indépendant. Voir 4.2.

Afin d'éviter toute confusion entre les terminologies ODP et LOTOS, les termes propres à la terminologie LOTOS sont en *italique* dans les paragraphes qui suivent.

#### 4.1.1 Concepts de modélisation de base

##### 4.1.1.1 Objet

Un objet est une *instanciation* de *définition de processus* LOTOS qui peut être identifiée de manière univoque.

##### 4.1.1.2 Environnement (d'un objet)

L'environnement d'un objet est la partie d'un modèle qui ne fait pas partie de l'objet. En LOTOS, l'environnement d'un objet dans une spécification à un certain instant est donné par l'environnement de la spécification et par les autres *expressions de comportement* qui sont composées avec cet objet dans la spécification concernée à l'instant en question.

NOTE – L'environnement d'une spécification est vide si la spécification n'est pas paramétrée.

##### 4.1.1.3 Action

En LOTOS, les actions sont modélisées sous forme d'*événements internes* ou d'*événements observables*. En LOTOS, tous les événements sont atomiques. Une action interne peut être donnée explicitement par le symbole d'*événement interne*, *i*, ou par l'occurrence d'un événement dont la *porte* associée est *cachée* du côté environnement.

En LOTOS, une interaction est représentée par une *synchronisation* entre deux *expressions de comportement* ou plus associées à des objets en un point d'interaction commun (*porte*). Les interactions peuvent être du type:

- *synchronisation* pure en une *porte* commune, sans offre: il ne se produit aucun passage de valeurs entre les objets;
- ! et ! pour une *synchronisation* pure: aucune valeur n'est échangée entre les objets;
- ! et ? pour un passage de valeur sous réserve que l'événement ? contienne l'événement !; autre façon de voir: l'événement ! choisit une valeur parmi un ensemble de valeurs associées à l'événement;
- ? et ? pour un établissement de valeur: ici, une valeur est déterminée d'un commun accord à partir de l'intersection des ensembles de valeurs. Si cette intersection est l'ensemble vide, il ne se produit aucune *synchronisation* et donc aucune interaction.

Si une granularité non atomique des actions est exigée, on peut avoir recours à l'affinement d'événement, ce qui permet de modéliser les actions non instantanées et les actions se chevauchant. Il convient de noter que l'affinement d'événement est une opération complexe, en particulier lorsqu'il s'agit de maintenir la compatibilité de comportement.

En LOTOS, il n'existe aucune construction qui permette d'exprimer les relations de cause à effet, même si une représentation non formelle est parfois possible.

##### 4.1.1.4 Interface

Une interface est une abstraction du comportement d'un objet, constituée par un sous-ensemble des actions observables de cet objet. En LOTOS, toutes les actions observables d'un objet exigent des *portes* qui leur permettent de se synchroniser avec l'environnement, on obtient donc généralement un sous-ensemble d'actions observables par une subdivision de l'ensemble des *portes* données dans la *définition de processus* associée à l'objet. Pour obtenir une interface, il est possible de *cache* les *portes* non requises par l'interface considérée. Autre solution: il est possible d'effectuer une *synchronisation* uniquement sur un sous-ensemble des *portes* associées à un objet. Dans ce cas, les actions se produisant aux *portes* de la *définition de processus* qui n'appartiennent pas à l'ensemble sur lequel a porté la synchronisation, peuvent être considérées comme des actions internes à l'objet dans la mesure où l'environnement sur lequel sont synchronisées les portes constituant l'interface est concerné.

Il convient de noter que dans le cadre de cette définition, les interfaces d'un objet doivent utiliser des noms de porte différents, car il est impossible de faire la distinction entre des interfaces utilisant la même porte.

##### 4.1.1.5 Activité

Une activité est un graphe d'actions acyclique orienté à racine unique, chaque noeud du graphe représentant un état du système et chaque arc une action. Pour qu'une action puisse se produire, elle doit satisfaire les préconditions de l'état du système.



#### 4.1.1.6 Comportement (d'un objet)

Le comportement d'un objet est défini par l'*expression de comportement* LOTOS associée à la *définition de processus* qui constitue le squelette d'objet. Une *expression de comportement* peut être composée d'une séquence d'offres d'événements visibles de l'extérieur et d'*événements internes*. Le comportement effectif d'un objet, tel qu'il pourrait être enregistré dans une trace, dépend de l'*expression de comportement* associée à l'objet et du mode de configuration avec l'environnement. Le comportement effectif que l'objet présente dépend de l'*expression de comportement* de l'objet et du mode de synchronisation avec l'environnement. Un objet peut présenter un comportement non déterministe.

#### 4.1.1.7 Etat (d'un objet)

C'est la condition d'un objet déterminant l'ensemble de toutes les séquences d'actions auxquelles l'objet peut participer. Cette condition est fonction de l'*expression de comportement* définie dans le squelette d'objet à partir duquel l'objet a été créé et éventuellement des relations courantes entre d'éventuelles variables locales existantes.

#### 4.1.1.8 Communication

Une communication est une transmission d'information (par passage de valeur) entre deux objets en interaction ou plus. Il est impossible d'écrire directement des relations de cause à effet. Il convient aussi de signaler que la *synchronisation* peut être interprétée comme une communication.

#### 4.1.1.9 Position dans l'espace

En LOTOS, l'abstraction correspondante est très éloignée de la notion de position dans l'espace. On peut uniquement faire correspondre l'espace à la structure du modèle de spécification. Pour les interactions en LOTOS, la position d'un événement – la position structurelle par rapport au modèle de spécification – est donnée par une *porte*. En LOTOS, l'abstraction correspondant à la notion de position dans l'espace à laquelle un *événement interne* peut se produire est très éloignée de cette notion. Cette abstraction est obtenue implicitement au moyen de la construction LOTOS *hide ... in* (cacher ... dans) qui permet une utilisation interne des *portes* dans un processus invisible pour l'environnement du processus, ou explicitement au moyen du symbole d'*événement interne*.

En LOTOS, il est possible qu'une même position dans l'espace soit employée pour plusieurs points d'interaction, car différentes *dénotations d'action* peuvent être associées à une même porte.

La position d'un objet est donnée par la réunion des positions des *portes* associées à cet objet, c'est-à-dire la réunion de toutes les positions des actions auxquelles l'objet participe.

#### 4.1.1.10 Position dans le temps

En LOTOS, l'abstraction correspondante est très éloignée du concept de temps, seul l'ordre temporel est pris en considération, il n'existe donc pas de *position absolue dans le temps*. Toutefois, un positionnement dans le temps serait possible, si on utilisait une forme étendue de LOTOS dans laquelle des aspects temporels seraient incorporés.

#### 4.1.1.11 Point d'interaction

Un point d'interaction est une *porte* à laquelle est associée un ensemble éventuellement vide de valeurs.

NOTE – Dans une spécification, des changements de position peuvent se traduire par des modifications des valeurs associées.

### 4.1.2 Concepts de spécification

#### 4.1.2.1 Composition

- **d'objets:** un objet composite est un objet décrit au moyen d'un ou de plusieurs opérateurs de combinaison LOTOS, qui comprennent:
  - l'*opérateur d'entrelacement* ( $\parallel$ );
  - les *opérateurs de composition parallèle* ( $\parallel$  et  $\parallel$ [liste-de-portes]);
  - l'*opérateur d'activation* ( $\gg$ );
  - l'*opérateur de désactivation* ( $\gg$ );
  - l'*opérateur de choix* ( $\square$ ).
- **de comportements:** il s'agit de la composition des *expressions de comportement* associées aux objets composants qui entrent dans la création d'un objet composite par composition. Les opérateurs servant à la composition de comportements sont les mêmes que ceux qui servent à la composition d'objets.

#### 4.1.2.2 Objet composite

Un objet composite est un objet décrit au moyen d'un ou de plusieurs des *opérateurs* LOTOS d'*entrelacement*, de *composition parallèle*, de *désactivation*, d'*activation* et de *choix*.

#### 4.1.2.3 Décomposition

- **d'objets:** il s'agit de l'expression d'un objet donné sous forme d'*objet composite*. Toutefois, il peut exister plusieurs façons de décomposer un objet composite.
- **de comportements:** il s'agit de l'expression d'un comportement donné sous forme de comportement composite. Il peut exister plusieurs façons de décomposer un comportement composite.

NOTE – On pourrait aussi considérer que la notion de décomposition de comportements est fournie de manière inhérente par les *opérations* et les équations ACT ONE associées à une *sorte*. En effet, ces *opérations* et *équations* fournissent toutes les combinaisons possibles de comportement. Par conséquent, une composition séquentielle pourrait par exemple être générée par des *opérations* appliquées séquentiellement. Chaque *opération* de la séquence doit satisfaire les équations nécessaires pour pouvoir se produire. Mais le fait de parler de composition de comportements est discutable, étant donné que les *opérations* et les équations existent déjà et définissent tous les comportements possibles.

#### 4.1.2.4 Compatibilité de comportements

En LOTOS, des théories spécifiques ont été établies pour vérifier la compatibilité de comportement. Il n'existe pas d'éléments syntaxiques propres au LOTOS pour construire et garantir une compatibilité de comportement de manière générale. Toutefois, la norme LOTOS définit la notion de *conformité* qui constitue une base pour la prise en considération de la compatibilité de comportement.

Pour pouvoir déterminer si les comportements de deux objets sont compatibles, il faut introduire la notion de *conformité*. La *conformité* a pour objet l'évaluation de la fonctionnalité d'une réalisation par rapport à sa spécification; on peut considérer ici que le terme réalisation désigne une description moins abstraite d'une spécification.

Si **P** et **Q** sont deux processus LOTOS, l'instruction "**Q** est conforme à **P**" (s'écrivant **Q conf P**) signifie que **Q** est une réalisation valable de **P**. Autrement dit, si **P** peut exécuter une certaine trace  $\sigma$  et ensuite se comporter comme un certain processus **P'** et si **Q** peut aussi exécuter la trace  $\sigma$  et ensuite se comporter comme **Q'**, alors les conditions suivantes sur **P'** et **Q'** doivent être satisfaites: chaque fois que **Q'** peut refuser d'exécuter tous les événements d'un ensemble donné A d'actions observables, alors **P'** doit aussi être capable de refuser d'exécuter tous les événements de A.

Par conséquent, **Q conf P** si et seulement si, placé dans un environnement dont les traces sont limitées à celles de **P**, **Q** ne peut pas parvenir à un blocage lorsque **P** ne peut pas parvenir à un blocage. Autre définition possible: **Q** présente les blocages de **P** dans un environnement dont les traces sont limitées à celles de **P**.

Un objet peut être rendu compatible sur le plan du comportement avec un second objet après une certaine modification de son comportement, qui peut consister en une *extension* (ajouter un comportement supplémentaire) ou en une *réduction* (restreindre le comportement). Ce processus de modification d'un objet est appelé affinement (voir 4.1.2.5).

#### 4.1.2.5 Affinement

L'affinement est le processus permettant de modifier un objet, par l'extension ou la réduction de son comportement ou encore par une combinaison des deux, de sorte qu'il soit conforme à un autre objet. Soient **P** et **Q** deux processus LOTOS, une *extension* de **P** par **Q** (s'écrivant **Q extends P**) signifie que **Q** a autant ou plus de traces que **P**, mais que dans un environnement dont les traces sont limitées à celles de **P**, **Q** présente les mêmes blocages. Une *réduction* de **P** par **Q** (s'écrivant **Q reduces P**) signifie que **Q** a autant ou moins de traces que **P**, mais que dans un environnement dont les traces sont limitées à celles de **Q**, **P** présente les mêmes blocages.

#### 4.1.2.6 Trace

Une trace du comportement d'un objet depuis son état instancié initial jusqu'à un certain autre état est un enregistrement de la séquence finie d'interactions (*événements observables*) entre l'objet et son environnement.

#### 4.1.2.7 Type d'un <X>

Les types qui peuvent être consignés explicitement en LOTOS pour des objets et des interfaces sont des types de squelette. En LOTOS, il n'existe aucune construction explicite qui permette de modéliser les types d'action. Une spécification LOTOS est constituée par une *expression de comportement* elle-même composée de *dénotations d'actions* (squelettes d'action). Soit ces squelettes d'action se produisent dans le cadre du comportement du système, auquel cas leur occurrence peut abusivement être considérée comme l'instanciation du squelette d'action, soit ils ne se produisent pas, auquel cas le squelette d'action reste non instancié. Les squelettes d'action peuvent être donnés par le symbole d'événement interne, *i*, ou par des offres d'événement en des *portes* qui peuvent ou non comporter une séquence finie de déclarations de valeurs ou de variables.

Le LOTOS ne permet pas de caractériser les actions directement; toutefois, une forme limitée de caractérisation des actions est intégrée dans l'élément *synchronisation* du LOTOS. En effet, on pourrait considérer que les *dénotations d'actions* (squelettes d'action) synchronisées doivent satisfaire le même type d'action pour que l'action puisse se produire. Toutefois, le LOTOS ne classe pas les éléments de caractérisation de ces *dénotations d'action* arbitraires et il est donc impossible d'attribuer un type formel à une action donnée. Un rôle de cause à effet pourrait être attribué de manière non formelle aux offres d'événement intervenant dans une interaction, mais ce n'est généralement pas le cas. Voir 4.1.1.8.

Le symbole d'événement interne peut servir à représenter un type d'action, la caractéristique commune de cet ensemble d'actions étant que ces actions n'ont pas de caractéristique.

Il convient de noter que si on affirme qu'en LOTOS, le seul prédicat possible pour les objets (et les interfaces) est qu'ils satisfassent à leur type de squelette, les concepts de type et de type de squelette donnés dans la Rec. UIT-T X.902 | ISO/CEI 10746-2 en sont réduits à la même technique de modélisation en LOTOS. Il n'existe donc pas de distinction en LOTOS entre un type dans son sens de caractérisation général et un type de squelette dans son sens plus restrictif d'instanciation de squelette.

#### 4.1.2.8 Classe d'un <X> iTeh STANDARD PREVIEW

La notion de classe dépend du prédicat de type de caractérisation auquel les membres de la classe satisfont. Les objets, les interfaces et les actions peuvent satisfaire à de nombreux prédicats de type de caractérisation arbitraires. Un type pouvant être consigné est un type de squelette. Lorsque c'est le cas, la classe d'objets, d'interfaces ou d'actions associée à ce type est la classe de squelette.

NOTE – Il convient de noter que, si on affirme qu'en LOTOS, on ne peut classer les objets, interfaces et actions que par rapport à leur type de squelette, les concepts de classe et de classe de squelette donnés dans la Rec. UIT-T X.902 | ISO/CEI 10746-2 en sont réduits à la même technique de modélisation en LOTOS. Il n'existe donc pas de distinction en LOTOS entre une classe dans son sens de classification général et une classe de squelette dans son sens plus restrictif d'ensemble d'instances d'un type de squelette donné.

#### 4.1.2.9 Sous-type/super-type

Etant donné qu'en LOTOS, les types qui peuvent être consignés pour les objets, les interfaces et dans une moindre mesure les actions sont des types de squelette, une relation de sous-typage en LOTOS est une relation qui peut exister entre types de squelette. Toutefois, il n'existe pas en LOTOS d'élément direct permettant de consigner directement des relations de sous-typage. Si le sous-typage est exigé, on peut alors avoir recours à une *extension* pour donner une relation de sous-typage sur la base d'une capacité de substitution; toutefois, ce n'est pas un élément explicitement prévu en LOTOS.

#### 4.1.2.10 Sous-classe/superclasse

Etant donné qu'en LOTOS, les types qui peuvent être consignés pour les objets, les interfaces et dans une moindre mesure les actions sont des types de squelette, une relation de sous-classe existe entre deux classes lorsqu'une relation de sous-typage existe entre leurs types de squelette correspondants.

#### 4.1.2.11 Squelette de <X>

- **Squelette d'objet:** c'est une *définition de processus* avec un certain moyen permettant de l'identifier de manière univoque une fois instanciée. Si aucune liste de paramètres de valeurs n'est donnée, l'identification d'objet sera alors impossible si plusieurs objets sont instanciés à partir du squelette d'objet.

Concernant la combinaison de squelettes d'objet en LOTOS, il n'existe pas d'opérateur de combinaison sauf pour une forme limitée de domaine d'application utilisant le terme LOTOS *where*.

- **Squelette d'interface:** il s'agit de tout comportement obtenu à partir d'une *définition de processus* en ne considérant que les interactions au niveau d'un sous-ensemble des *portes* associées à la *définition de processus*. On obtient ce sous-ensemble de *portes* en *cachant* les *portes* qui ne sont pas requises pour les interactions considérées.

Concernant la combinaison de squelettes d'interface en LOTOS, il n'existe pas d'opérateur de combinaison sauf pour une forme limitée de domaine d'application utilisant le terme LOTOS *where*.

- **Squelette d'action:** c'est une *dénotation d'action*, qui peut être un symbole d'*événement interne*, un identificateur de porte ou un identificateur de porte suivi par une séquence finie de déclarations de valeurs ou de variables.

NOTE – La définition de *dénotation d'action* donnée ici est imaginée car le LOTOS ne prend pas vraiment en charge le concept de squelette d'action. En LOTOS, les comportements possibles sont spécifiés par la combinaison de *dénotations d'action* sous une certaine forme. En LOTOS, tout ce qu'on puisse faire, c'est relier un squelette à une *dénotation d'action*. Toutefois, d'après le texte de la Rec. UIT-T X.902 | ISO/CEI 10746-2, un squelette d'action doit regrouper les caractéristiques des actions. Ceci n'est pas vérifié en LOTOS car les offres d'événement (*dénotations d'action*) existent de façon isolée et il est impossible de les rassembler et de leur appliquer un squelette qui les caractérise.

La composition de squelettes d'action peut abusivement être comparée à une *synchronisation* avec passage de valeur ou génération de valeur. Dans ce cas, deux squelettes d'action (ou plus) déterminent un squelette d'action commun pour que la *synchronisation* puisse se produire, c'est-à-dire un squelette d'action avec les caractéristiques communes à tous les squelettes d'action intervenant dans la *synchronisation* (composition).

#### 4.1.2.12 Signature d'interface

La signature d'une interface, définie comme étant un ensemble de squelettes d'action associés aux interactions de l'interface, est représentée en LOTOS par un ensemble de *dénotations d'action*. Les membres de cet ensemble sont les *dénotations d'action* qui nécessitent une *synchronisation* avec l'environnement pour pouvoir se produire.

#### 4.1.2.13 Instanciation (d'un squelette de $\langle X \rangle$ )

- **d'un squelette d'objet:** c'est le résultat d'un processus qui utilise un squelette d'objet pour créer un nouvel objet dans son état initial. Ce processus met en jeu l'*actualisation* de la *liste de portes formelles* et des *paramètres formels* d'une *définition de processus* par un réétiquetage au cas par cas à partir d'une liste de portes spécifiées et d'une liste de paramètres effectifs. Les éléments de l'objet créé sont fonction du squelette d'objet et des éventuels paramètres utilisés pour l'instancier.
- **d'un squelette d'interface:** c'est le résultat d'un processus permettant de créer une interface à partir d'un squelette d'interface. L'interface créée peut ensuite être utilisée par l'objet auquel elle est associée en vue d'une interaction avec l'environnement. Les éléments de l'interface créée sont fonction du squelette d'interface et des éventuels paramètres utilisés pour l'instancier.
- **d'un squelette d'action:** il s'agit de l'occurrence d'une action en LOTOS, pouvant impliquer la réécriture d'expressions ACT ONE.

#### 4.1.2.14 Rôle

Un rôle est un nom associé à une *définition de processus* dans le squelette d'un objet composite (c'est-à-dire une composition LOTOS d'*expressions de comportement*). En tant que tels, les rôles ne peuvent être utilisés comme des paramètres. Toutefois, il est possible d'assigner des valeurs de données à chacun des rôles d'une composition afin de les distinguer ou d'y accéder de manière spécifique.

#### 4.1.2.15 Création (d'un $\langle X \rangle$ )

- **d'un objet:** c'est l'instanciation d'un squelette d'objet, entrant dans le cadre du comportement d'un objet existant.
- **d'une interface:** les objets et les interfaces étant modélisés de la même manière en LOTOS (via les *définitions de processus*), la création d'objet correspond à la création d'interface. La définition de la création d'interface est donc donnée par la définition de la création d'objet.

#### 4.1.2.16 Introduction (d'un objet)

Il s'agit de l'*instanciation* du comportement associé à une spécification LOTOS.