



UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

**UIT-T**

SECTEUR DE LA NORMALISATION  
DES TÉLÉCOMMUNICATIONS  
DE L'UIT

**T.171**

(10/96)

SÉRIE T: TERMINAUX DES SERVICES TÉLÉMATIQUES

---

**Protocoles pour les services audiovisuels  
interactifs: Représentation codée des objets  
multimédias et hypermédias**

Recommandation UIT-T T.171

(Antérieurement Recommandation du CCITT)

---

RECOMMANDATIONS UIT-T DE LA SÉRIE T  
**TERMINAUX DES SERVICES TÉLÉMATIQUES**

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

## AVANT-PROPOS

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

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

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

La Recommandation UIT-T T.171, que l'on doit à la Commission d'études 8 (1993-1996) de l'UIT-T, a été approuvée par la CMNT (Genève, 9-18 octobre 1996).

---

### NOTE

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

© UIT 1997

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

## TABLE DES MATIÈRES

	<i>Page</i>	
0	Introduction .....	1
0.1	Domaines d'application pour l'analyse des besoins .....	1
0.2	Besoins des applications multimédias et hypermédias .....	2
0.3	Justification de la normalisation des informations multimédias et hypermédias .....	4
0.4	Objectifs T.171 .....	5
0.5	Concepts T.171 .....	6
0.6	Interface d'application MHEG .....	10
0.7	Possibilités d'extension T.171 .....	10
1	Domaine d'application .....	10
1.1	Spécificité du domaine d'application.....	11
1.2	Questions sortant du cadre T.171.....	11
2	Conformité.....	11
2.1	Profils.....	11
2.2	Syntaxe.....	11
2.3	Sémantique.....	11
3	Références normatives .....	12
4	Définitions .....	12
5	Symboles et abréviations .....	19
	SECTION 1 – APERÇU GÉNÉRAL .....	21
6	Principales caractéristiques T.171 .....	21
6.1	Objets multimédias de transfert.....	21
6.2	Approche orientée objet.....	21
6.3	Caractéristiques techniques .....	23
7	Hypothèses relatives au moteur MHEG.....	27
7.1	Traitement et transfert des objets .....	27
7.2	Interface d'application MHEG .....	28
7.3	Traitement des exceptions.....	29
8	Méthodologie.....	29
8.1	Modularité.....	29
8.2	Méthodologie de représentation des objets MHEG .....	29
	SECTION 2 – MÉCANISMES DES FONCTIONNALITÉS GÉNÉRIQUES ET DES DÉFINITIONS UTILES	33
9	Mécanisme de présentation.....	33
9.1	Espace de présentation .....	34
9.2	Espace de présentation d'origine .....	37
9.3	Espace de présentation d'un canal .....	38
9.4	Espace de présentation relatif.....	39
9.5	Mappage CPS .....	39
10	Mécanisme d'identification générique .....	40
10.1	Identification externe .....	41
10.2	Identification interne .....	43
10.3	Identification symbolique.....	47
11	Mécanisme de référence générique.....	47
11.1	Référence générique au moyen des identifications génériques .....	47
11.2	Références prédéfinies .....	50
11.3	référence?.....	50

	<i>Page</i>
12	Valeur générique ..... 52
12.1	Booléen générique ..... 53
12.2	Numérique générique ..... 53
12.3	Entier générique ..... 53
12.4	Fraction générique ..... 53
12.5	Chaîne générique ..... 54
12.6	Référence générique ..... 54
12.7	Liste générique ..... 54
13	Mécanisme de macro ..... 54
14	Crochet ..... 56
15	Extensibilité de T.171 ..... 57
15.1	Catalogues ..... 57
15.2	Ajout de nouvelles classes d'objet MHEG ..... 58
15.3	Disposition d'extensibilité ..... 58
SECTION 3 – APERÇU GÉNÉRAL DES CLASSES MHEG ..... 58	
16	Aperçu général des classes d'objets MHEG ..... 58
17	Structure de la classe <i>MH-objet</i> ..... 59
17.1	identification de classe ..... 59
17.2	Identificateur MHEG ..... 59
17.3	Information générale sur les objets ..... 59
18	Structure de la classe <i>action</i> ..... 60
18.1	Actions élémentaires ..... 60
18.2	Objet <i>action de base</i> ..... 61
18.3	Objet <i>action enveloppante</i> ..... 61
18.4	Objet <i>macroaction</i> ..... 61
19	Structure de la classe <i>lien</i> ..... 61
19.1	Condition de lien ..... 62
19.2	Effet de lien ..... 71
19.3	Objet <i>lien de base</i> ..... 71
19.4	Objet <i>lien enveloppant</i> ..... 71
19.5	Objet <i>macrolien</i> ..... 72
20	Structure des classes <i>modèles</i> ..... 72
21	Structure de la classe <i>script</i> ..... 72
22	Structure de la classe <i>composant</i> ..... 72
23	Structure de la classe <i>contenu</i> ..... 73
24	Structure de la classe <i>contenu multiplexé</i> ..... 73
25	Structure de la classe <i>composite</i> ..... 74
25.1	Démarrage de disponibilité (availability start-up) ..... 75
25.2	Arrêt de disponibilité (Availability close-down) ..... 76
25.3	Démarrage de rt-disponibilité (rt-Availability Start-up) ..... 76
25.4	Arrêt de rt-disponibilité (rt-Availability Close-down) ..... 76
25.5	Élément de composition ..... 77
25.6	Exemple de composition ..... 77
26	Structure de la classe <i>conteneur</i> ..... 78
26.1	Démarrage de conteneur (container start-up) ..... 78
26.2	Arrêt du conteneur (container close-down) ..... 78
26.3	Élément de conteneur ..... 78

	<i>Page</i>
27	Structure de la classe <i>descripteur</i> ..... 79
27.1	Objet connexe ..... 79
27.2	Autre descripteur..... 81
27.3	«lisez-moi» ..... 81
27.4	Information lisibles par le système..... 81
27.5	Informations de canal..... 81
27.6	Informations sur les styles catalogués ..... 82
27.7	Informations sur les actions élémentaires étendues cataloguées ..... 82
27.8	Informations sur les attributs étendus catalogués ..... 82
SECTION 4 – COMPORTEMENT COMMUN AUX ENTITÉS MHEG..... 82	
28	Comportement de l'entité MHEG ..... 82
29	Définition des états des entités MHEG..... 82
29.1	Disponibilité d'objet MHEG ..... 83
29.2	Activation de liens ..... 84
29.3	Disponibilité de canal..... 85
29.4	Disponibilité de rt-objet ..... 86
29.5	Comportement de fonctionnement des rt-composants..... 88
29.6	Comportement de présentation du rt-composant..... 89
30	Cycle de vie des entités MHEG ..... 92
31	Mécanismes généraux des actions ..... 92
31.1	Application des actions ..... 92
31.2	Marche à suivre lors de l'application d'un effet de lien ..... 93
31.3	Traitement de base d'une action élémentaire..... 95
31.4	Résolution d'un ensemble de cibles..... 95
31.5	Précision arithmétique..... 95
32	Effets et traitements communs aux actions ..... 96
32.1	Actions élémentaires ..... 96
32.2	Actions Get ..... 96
32.3	Traitement recommandé pour les exceptions ..... 97
33	Comportement différé..... 97
33.1	Attributs et états du comportement ..... 97
33.2	Actions permettant de modifier le comportement ..... 97
34	Comportement de renvoi ..... 98
34.1	Attributs et états de comportement..... 98
34.2	Actions servant à modifier le comportement..... 98
35	Comportement de synonymie ..... 98
35.1	Attributs et états de comportement..... 98
35.2	Actions servant à modifier le comportement..... 98
36	Comportement d'extensibilité ..... 99
36.1	Attributs et états de comportement..... 99
36.2	Attribut catalogué..... 99
36.3	Actions servant à modifier le comportement..... 99
36.4	Actions servant à extraire le comportement ..... 100
SECTION 5 – COMPORTEMENT DES OBJETS MHEG ..... 101	
37	Comportement de disponibilité des objets MHEG ..... 101
37.1	Attributs et états du comportement ..... 101
37.2	Preparation Status (état de préparation) ..... 101
37.3	Actions de modification du comportement ..... 101
37.4	Actions de récupération du comportement..... 104

	<i>Page</i>	
38	Comportement d'activation des objets <i>liens</i> .....	104
	38.1 Attributs et états du comportement .....	104
	38.2 Activation Status (état d'activation) .....	105
	38.3 Actions de modification du comportement .....	105
	38.4 Actions de récupération du comportement.....	105
39	Comportement d'abandon des objets <i>liens</i> .....	106
	39.1 Attributs et états du comportement .....	106
	39.2 Actions de modification du comportement .....	106
40	Comportement de stockage de valeur générique de classe <i>contenu</i> .....	107
	40.1 Attributs et états du comportement .....	107
	40.2 Data (données) .....	107
	40.3 Actions de modification du comportement .....	107
	40.4 Actions de récupération du comportement.....	112
41	Comportement de copie de classe <i>contenu</i> .....	113
	41.1 Attributs et états du comportement .....	113
	41.2 Actions de modification du comportement .....	113
SECTION 6 – COMPORTEMENT DES RT-OBJETS .....		114
42	Comportement de disponibilité des rt-objets .....	114
	42.1 Attributs et états du comportement .....	114
	42.2 Rt-Availability Status (état de rt-disponibilité) .....	114
	42.3 Actions de modification du comportement .....	114
	42.4 Actions de récupération du comportement.....	116
43	Comportement d'exécution des rt-objets.....	117
	43.1 Attributs et états du comportement .....	117
	43.2 Running Status (état d'exécution).....	117
	43.3 Actions de modification du comportement .....	117
	43.4 Actions de récupération du comportement.....	119
44	Comportement de passage de paramètres aux rt-scripts .....	120
	44.1 Attributs et états du comportement .....	120
	44.2 Actions de modification du comportement .....	120
45	Comportement de terminaison des rt-scripts.....	121
	45.1 Attributs et états du comportement .....	121
	45.2 Termination Status (état de terminaison) .....	121
	45.3 Actions de récupération du comportement.....	121
46	Comportement de dynamisme de présentation et de structure des réceptacles .....	122
	46.1 Attributs et états du comportement .....	122
	46.2 Actions de modification du comportement .....	122
47	Comportement de navigation des rt-composites .....	124
	47.1 Attributs et états du comportement .....	124
	47.2 Rt-Composite Address (adresse de rt-composite) .....	124
	47.3 Navigation Command (commande de navigation) .....	124
	47.4 Child (enfant) .....	124
	47.5 EmptyChild (enfant vide).....	124
	47.6 Sibling (frère).....	124
	47.7 Ancestor (ancêtre).....	124
	47.8 Actions de récupération du comportement.....	125
48	Comportement d'affectation de rt-composants à un espace RPS .....	125
	48.1 Attributs et états du comportement .....	125
	48.2 RPS Assignment (affectation d'espace RPS).....	125
	48.3 Actions de modification du comportement .....	125
	48.4 Actions de récupération du comportement.....	126

	<i>Page</i>
49	Comportement de perceptibilité des rt-composants..... 127
49.1	Attributs et états du comportement ..... 127
49.2	Perceptability (perceptibilité)..... 127
49.3	Presentation Priority (priorité de présentation)..... 127
49.4	Actions de modification du comportement ..... 127
49.5	Actions de récupération du comportement..... 129
50	Comportement temporel des rt-composants..... 129
50.1	Attributs et états du comportement ..... 129
50.2	OD (durée d'origine) ..... 130
50.3	POD (durée d'origine projetée) ..... 130
50.4	OVD (durée de visualisation d'origine)..... 130
50.5	PVD (durée de visualisation projetée) ..... 130
50.6	Temporal Termination (terminaison temporelle) ..... 130
50.7	PVD Position (position de la durée PVD)..... 131
50.8	CTP (position temporelle courante) ..... 132
50.9	GTF (facteur temporel générique)..... 133
50.10	Timestone Status (état de marqueur temporel)..... 133
50.11	Timestone ID (identificateur de marqueur temporel)..... 133
50.12	Expected OVD Result (résultat de durée OVD attendue) ..... 133
50.13	Expected PVD Result (résultat de durée PVD attendue) ..... 133
50.14	Actions de modification du comportement ..... 133
50.15	Actions de récupération du comportement..... 136
51	Comportement spatial des rt-composants ..... 141
51.1	Attributs et états du comportement ..... 141
51.2	OS (taille d'origine)..... 141
51.3	POS (taille d'origine projetée)..... 141
51.4	Aspect Ratio (facteur de forme) ..... 141
51.5	Resizing Strategy (stratégie de redimensionnement)..... 142
51.6	OVS (taille de visualisation d'origine) ..... 143
51.7	OAP (point de rattachement de taille de visualisation d'origine) ..... 143
51.8	OVS Position (position de la taille OVS)..... 143
51.9	PVS (taille de visualisation projetée)..... 144
51.10	OVS Proj Strategy (stratégie de projection de taille OVS) ..... 144
51.11	PAP (point de rattachement de taille de visualisation projetée)..... 144
51.12	PVS Position (position de la taille PVS)..... 144
51.13	GSF (facteur spatial générique)..... 144
51.14	Spatial Control (commande spatiale) ..... 145
51.15	User Spatial Control (commande spatiale utilisateur) ..... 145
51.16	Expected Axis Result Param (paramètre de résultat d'axe attendu) ..... 145
51.17	Point Type Param (paramètre de type de point)..... 145
51.18	Actions de modification du comportement ..... 145
51.19	Actions de récupération du comportement..... 151
52	Comportement sonore des rt-composants ..... 157
52.1	Attributs et états du comportement ..... 157
52.2	OV (volume sonore d'origine)..... 157
52.3	CV (volume sonore courant)..... 157
52.4	PCV (volume sonore courant projeté)..... 157
52.5	GVF (facteur de volume sonore générique) ..... 158
52.6	Actions de modification du comportement ..... 158
52.7	Actions de récupération du comportement..... 159



	<i>Page</i>
53	Comportement de choix de flux d'un rt-mux ..... 160
	53.1 Attributs et états du comportement ..... 160
	53.2 Stream Choice (choix de flux) ..... 161
	53.3 Stream Chosen State (état de choix de flux) ..... 161
	53.4 Stream Identification (identification de flux) ..... 161
	53.5 Actions de modification du comportement ..... 161
	53.6 Actions de récupération du comportement ..... 161
54	Comportement d'interaction..... 162
	54.1 Attributs et états du comportement ..... 163
	54.2 Interaction Type (type d'interaction) ..... 163
	54.3 Interaction Status (état d'interaction)..... 163
	54.4 Selection Status (état de sélection)..... 163
	54.5 Modification Status (état de modification)..... 164
	54.6 Interaction Ability (capacité d'interaction)..... 164
	54.7 Selectability (sélectionnabilité)..... 164
	54.8 Modifiability (modifiabilité) ..... 164
	54.9 Min Interact Required (interaction minimale requise) ..... 164
	54.10 Max Interact Required (interaction maximale requise) ..... 165
	54.11 Number of Interacted Sockets (nombre de réceptacles en interaction) ..... 165
	54.12 Actions de modification du comportement ..... 166
	54.13 Actions de récupération du comportement..... 167
55	Comportement de style des rt-composants ..... 169
	55.1 Attributs et états du comportement ..... 170
	55.2 Style ..... 170
	55.3 Actions de modification du comportement ..... 170
	55.4 Actions de modification du comportement ..... 170
56	Comportement d'ancrage des rt-contenus ..... 171
	56.1 Attributs et états du comportement ..... 171
	56.2 Actions de modification du comportement ..... 171
	SECTION 7 – COMPORTEMENT DES CANAUX ..... 173
57	Comportement de disponibilité d'un canal..... 173
	57.1 Attributs et états du comportement ..... 173
	57.2 Channel Availability Status (état de disponibilité d'un canal)..... 173
	57.3 Actions de modification du comportement ..... 173
	57.4 Actions de récupération du comportement..... 174
58	Comportement de perceptibilité du canal ..... 175
	58.1 Attributs et états du comportement ..... 175
	58.2 Channel Perceptability (perceptibilité du canal) ..... 175
	58.3 Action de modification du comportement ..... 175
	58.4 Actions de récupération du comportement..... 176
59	Comportement de l'espace de présentation d'un canal ..... 176
	59.1 Attributs et états du comportement ..... 176
	59.2 Action de modification du comportement ..... 176
	SECTION 8 – COMPORTEMENT DES CANAUX ET RT-COMPOSANTS ..... 177
60	Comportement des événements des canaux et rt-composants..... 177
	60.1 Attributs et états du comportement ..... 177
	60.2 Event (événement) ..... 177
	60.3 Event Data (donnée d'événement)..... 177
	60.4 Action de modification du comportement ..... 177
	60.5 Action de récupération du comportement ..... 178

SECTION 9 – REPRÉSENTATION DÉTAILLÉE DES OBJETS MHEG .....	179
61 Attributs de représentation de la classe <i>objet MH</i> .....	179
61.1 MH-object Class (Classe <i>objet MH</i> ) .....	179
61.2 Class Identification (Identification de classe) .....	179
61.3 Class ID (Identificateur de classe) .....	180
61.4 Description (Description).....	180
61.5 Name (Nom) .....	180
61.6 Owner (Propriétaire) .....	180
61.7 Version (Version).....	180
61.8 Date (Date).....	180
61.9 Keywords (Mots clés) .....	180
61.10 Copyright (Droit d'auteur).....	180
61.11 Copyright ID (Identificateur de droit d'auteur) .....	180
61.12 Copyright Number (Numéro du droit d'auteur).....	180
61.13 Licence (Licence).....	180
61.14 Cache Priority (Priorité d'antémémoire).....	181
61.15 Comments (Observations).....	181
62 Attributs de représentation de la classe <i>action</i> .....	181
62.1 Action Class (Classe <i>action</i> ) .....	181
62.2 Synchro Indicator Param (Paramètre d'indicateur de synchronisation).....	181
62.3 Synchro Indicator Macro (Macro-paramètre d'indicateur de synchronisation) .....	181
62.4 Synchro Indicator (Indicateur de synchronisation) .....	182
62.5 Synchronised Action (Action synchronisée) .....	182
62.6 Action Object (Objet <i>action</i> ).....	182
63 Attributs de représentation de la classe <i>lien</i> .....	182
63.1 Link Class (Classe <i>lien</i> ) .....	182
63.2 Link Condition (Condition de lien) .....	183
63.3 Trigger Condition (Condition de déclenchement).....	183
63.4 Constraint Condition (Contrainte).....	183
63.5 Source Value (Valeur source) .....	183
63.6 Comparison Operation (Opération de comparaison).....	183
63.7 Comparison Value (Valeur de comparaison) .....	183
63.8 Previous Condition (Condition précédente).....	183
63.9 Current Condition (Condition courante) .....	183
63.10 Comparison Operator (Opérateur de comparaison) .....	183
63.11 Logical Combination (Combinaison logique) .....	184
63.12 Logical Operator (Opérateur logique).....	184
63.13 Condition (Condition) .....	184
63.14 Link Effect (Effet de lien) .....	184
63.15 Macro Parameter Resolution (Macro-résolution de paramètre) .....	184
63.16 Usage Value (Valeur d'utilisation) .....	184
64 Attributs de représentation de la classe <i>modèle</i> .....	184
64.1 Model Class (Classe <i>modèle</i> ).....	184
65 Attributs de représentation de la classe <i>script</i> .....	185
65.1 Script Class (Classe <i>script</i> ).....	185
65.2 Script Classification (Classification de scripts).....	185
65.3 Script Data (Données de script) .....	185
65.4 Script Inclusion (Inclusion de script) .....	185
65.5 Interchanged Script (Script transféré) .....	185
66 Attributs de représentation de la classe <i>composant</i> .....	185
66.1 Component Class (Classe <i>composant</i> ) .....	186

	<i>Page</i>
67	Attributs de représentation de la classe <i>contenu</i> ..... 186
67.1	Content Class (Classe <i>contenu</i> )..... 186
67.2	Content Data (Données de contenu)..... 186
67.3	Data Inclusion (Inclusion de données) ..... 186
68	Attributs de représentation de la classe <i>contenu multiplexé</i> ..... 186
68.1	Multiplexed Content Class (Classe <i>contenu multiplexé</i> )..... 187
68.2	Multiplexed Stream (Flux multiplexé) ..... 187
69	Attributs de représentation de la classe <i>composite</i> ..... 187
69.1	Composite Class (Classe <i>composite</i> )..... 187
69.2	Availability Start-up (Mise en disponibilité)..... 188
69.3	Availability Close-down (Retrait de disponibilité)..... 188
69.4	Rt-Availability Start-up (Mise en disponibilité d'exécution)..... 188
69.5	Rt-Availability Close-down (Retrait de disponibilité d'exécution)..... 188
69.6	Link Object (Objet <i>lien</i> )..... 188
69.7	Nb of Elements (Nombre d'éléments) ..... 188
69.8	Composition Element (Elément de composition)..... 188
69.9	Element Index (Indice d'élément)..... 189
69.10	Associated Model (Modèle associé) ..... 189
69.11	Label (Etiquette) ..... 189
70	Attributs de représentation de la classe <i>conteneur</i> ..... 189
70.1	Container Class (Classe <i>conteneur</i> ) ..... 189
70.2	Container Start-up (Mise en disponibilité du conteneur) ..... 189
70.3	Container Close-down (Retrait de disponibilité du conteneur) ..... 190
70.4	Container Element (Elément de conteneur)..... 190
71	Attributs de représentation de la classe <i>descripteur</i> ..... 190
71.1	Descriptor Class (Classe <i>descripteur</i> ) ..... 191
71.2	Related Object (Objet concerné)..... 191
71.3	Object Information (Information d'objet)..... 191
71.4	Object Size (Taille d'objet) ..... 191
71.5	Class Specific Information (Information spécifique de classe) ..... 191
71.6	Script Class Information (Information de classe <i>script</i> )..... 191
71.7	Content Class Information (Information de classe <i>contenu</i> ) ..... 192
71.8	Mux Content Class Info (Information de classe <i>contenu multiplexé</i> ) ..... 192
71.9	Number of Streams (Nombre de flux)..... 192
71.10	Stream Information (Information de flux)..... 192
71.11	Alternative Object (Objet de remplacement) ..... 192
71.12	Alternative Descriptor Object (Objet <i>descripteur</i> de remplacement)..... 192
71.13	Alternative Readme («Lisez-moi» de remplacement) ..... 192
71.14	Offset (Décalage) ..... 192
71.15	Other Descriptor (Autre descripteur) ..... 192
71.16	Readme («Lisez-moi»)..... 192
71.17	System Readable Material (Informations en langage machine) ..... 192
71.18	Channel Information (Information de canal)..... 193
71.19	X min (X min)..... 193
71.20	X max (X max) ..... 193
71.21	Y min (Y min)..... 193
71.22	Y max (Y max) ..... 193
71.23	Z min (Z min)..... 193
71.24	Z max (Z max) ..... 193
71.25	X Resolution (Résolution X)..... 193
71.26	Y Resolution (Résolution Y)..... 193

	<i>Page</i>
71.27 Z Resolution (Résolution Z).....	193
71.28 T Resolution (Résolution T) .....	193
71.29 F min (F min) .....	194
71.30 F max (F max).....	194
71.31 Audio Dynamic (Dynamique audio) .....	194
71.32 Channel Media Type (Type de support de canal) .....	194
71.33 Event Mapping (Mappage d'événement).....	194
71.34 Catalogued Style Information (Information de style catalogué).....	194
71.35 Cat Ext elementary action Info (Information d'action élémentaire étendue cataloguée) .....	194
71.36 Cat Ext Attribute Info (Information d'attribut étendu catalogué) .....	194
72 Comportement .....	194
72.1 Comportement de renvoi.....	194
72.2 Comportement de retour .....	195
72.3 Comportement des alias .....	196
72.4 Comportement d'extensibilité.....	196
72.5 Comportement de disponibilité des objets Mheg .....	197
72.6 Comportement d'activation des objets <i>lien</i> .....	197
72.7 Comportement d'abandon des objets <i>lien</i> .....	198
72.8 Comportement de stockage de la valeur générique de la classe <i>contenu</i> .....	198
72.9 Comportement de duplication de la classe <i>contenu</i> .....	199
72.10 Comportement de disponibilité des rt-objets.....	200
72.11 Comportement d'exécution des rt-objets .....	200
72.12 Comportement de passage de paramètre à un rt-script.....	201
72.13 Comportement de dynamisme de structure et de présentation des réceptacles .....	201
72.14 Comportement d'affectation d'espace RPS aux rt-composants.....	202
72.15 Comportement de perceptibilité des rt-composants .....	202
72.16 Comportement temporel des rt-composants .....	203
72.17 Comportement spatial des rt-composants.....	205
72.18 Comportement audible des rt-composants .....	207
72.19 Comportement de choix de flux rt-mux .....	208
72.20 Comportement d'interaction .....	209
72.21 Comportement de style des rt-composants.....	210
72.22 Comportement d'ancrage des rt-contenus.....	210
72.23 Comportement de disponibilité des canaux.....	211
72.24 Comportement de perceptibilité des canaux.....	211
72.25 Comportement d'espace de présentation des canaux.....	212
72.26 Comportement d'événement des canaux et des rt-composants .....	212
73 Actions élémentaires.....	213
73.1 Liste d'actions élémentaires.....	213
73.2 Entités MHEG, données, flux et macro-paramètres; définitions utiles concernant l'identification ...	214
73.3 Définitions utiles concernant les références .....	217
73.4 Définitions utiles concernant les cibles .....	222
73.5 Définitions utiles concernant les valeurs génériques.....	226
73.6 Définitions utiles concernant les valeurs évaluées .....	230
73.7 Crochets .....	238
73.8 Extensibilité .....	238
73.9 Définitions utiles concernant l'espace de présentation .....	242
73.10 Définitions utiles concernant les constantes .....	249
73.11 Constantes de valeurs de comparaison.....	256

	<i>Page</i>
Annexe A – Notations ASN.1 (Niveau C) Représentation codée (Niveau D) .....	257
Annexe B – Exemples de systèmes MHEG .....	291
B.1 Exemple de moteur MHEG.....	291
B.2 Exemples d'applications .....	295
Annexe C – Interfaçage avec les Normes et Recommandations relatives aux informations monomédias .....	302
C.1 Exemple d'objet <i>contenu</i> de type image fixe .....	302
C.2 Exemple d'objet <i>contenu</i> de type audio.....	303
Annexe D – Support hypertexte/hypermédia .....	303
D.1 Introduction.....	303
D.2 Mécanisme pour hypertexte et hypermédia.....	304
D.3 Exemple d'interactions entre un système MHEG et le logiciel de navigation du Web.....	305
Annexe E – Exemples de comportements spatiaux.....	306
E.1 Exemple 1 .....	306
E.2 Exemple 2 .....	311
Annexe F – Récapitulatif des identificateurs d'objets .....	313
Annexe G – Index .....	314

## **RÉSUMÉ**

La présente Recommandation fournit un codage pour les informations multimédias ou hypermédias utilisées et échangées par des applications dans un grand nombre de domaines.

# PROTOCOLES POUR LES SERVICES AUDIOVISUELS INTERACTIFS: REPRÉSENTATION CODÉE DES OBJETS MULTIMÉDIAS ET HYPERMÉDIAS

(Genève, 1996)

## 0 Introduction

La présente Recommandation définit un codage pour les informations multimédias et hypermédias utilisées et transférées par des applications appartenant à de nombreux domaines.

La présente Recommandation a été conçue sur la base des résultats d'une analyse des besoins des applications, menée afin d'identifier les aspects génériques de ces applications et de les intégrer dans un codage simple.

### 0.1 Domaines d'application pour l'analyse des besoins

Ces dernières années, on a assisté à une explosion des applications et des services multimédias dans de nombreux domaines. Le présent paragraphe illustre la très large variété d'applications audiovisuelles informatiques à partir desquelles s'est faite l'analyse des besoins.

- **Formation et enseignement (aussi appelé télé-enseignement):** l'extension du traitement des données et de la télématique aux informations audiovisuelles augmente l'intérêt d'un tel outil en enseignement en raison de l'amélioration de l'interface avec les étudiants. Un autre besoin qui s'est également dégagé est l'échange d'informations audiovisuelles entre outils et la réutilisation de ces informations par d'autres applications. Les informations audiovisuelles doivent être structurées de telle sorte qu'elles puissent être facilement mises à jour, modifiées et personnalisées.
- **Simulations et jeux:** l'utilisation de multiples supports sur systèmes informatiques permet d'en améliorer grandement le réalisme, par exemple dans les simulateurs de vol et les simulateurs maritimes; ces applications sont hautement interactives.
- **Ventes et publicité:** l'association d'informations audiovisuelles attrayantes mettant en valeur les produits, avec le processus de gestion et de commande des marchandises constitue une évolution commerciale naturelle.
- **Systèmes bureautiques, documentation technique:** la possibilité d'intégrer des dessins, des images, des informations audio et vidéo dans tous types de documents de bureau est un besoin ressenti depuis longtemps. De tels documents intégrés offrirait un support attrayant à la communication interne d'entreprise.
- **Culture:** le traitement informatisé des informations audiovisuelles permet de multiplier le public des théâtres, des musées et des événements culturels. Il rend également possibles de nouvelles formes d'art.
- **Publication électronique et livres électroniques:** de nombreux documents, actuellement publiés sur papier, pourraient être améliorés par l'intégration d'informations audiovisuelles et la présentation sous forme d'hypertextes sur un réseau de télécommunication (guides touristiques et pages jaunes par exemple); il pourrait s'agir d'une application mixte de ventes et publicité.
- **Information publique:** l'emploi d'interfaces audiovisuelles permet au public d'utiliser des systèmes de renseignement informatisés en temps réel, par exemple un kiosque consulté à distance par vidéotex ou localement sur disque optique.
- **Travail collectif informatisé sur documents multimédias:** il existe un besoin de constituer des équipes efficaces de personnes, géographiquement dispersées, mais qui souhaitent partager des informations textuelles, graphiques, iconographiques, audiographiques ou autres dans le cadre par exemple d'une télémaintenance, d'une publication conjointe ou d'une messagerie électronique multimédia.
- **Applications médicales:** il existe un besoin de rechercher des informations audiovisuelles dans les meilleurs délais, notamment des images médicales, des données médicales et des dossiers médicaux, de façon intégrée depuis un site distant. Il pourrait s'agir d'une application de travail conjoint multidisciplinaire faisant intervenir différents spécialistes.

---

<sup>1)</sup> Le présent texte est techniquement équivalent à l'ISO/IEC 13522-1.

- **Applications télévisuelles interactives:** l'apparition de systèmes de télévision numérique sur les réseaux de Terre, à satellites ou par câbles ainsi que l'existence de nouvelles techniques de modulation et de transmission sur le réseau téléphonique créent des conditions favorables pour une offre nouvelle de services multimédias interactifs basés sur la synergie des technologies télévisuelles et télématiques, permettant aux utilisateurs d'accéder interactivement aux documents vidéo.
- **Nouvelles classes d'applications:** la grande diffusion des moyens d'accès à distance aux systèmes de traitement de données audiovisuelles rend possible l'émergence de nouvelles classes d'applications: télévision interactive, actualités et archives personnalisées par exemple.

Les sous-paragraphes suivants examinent les besoins propres aux applications de communication multimédias et hypermédias appartenant à ces domaines et en déduisent une liste de besoins. Cet ensemble de besoins constitue le cadre général de la présente Recommandation.

## 0.2 Besoins des applications multimédias et hypermédias

Dans les divers domaines présentés ci-dessus, l'impact sur l'utilisateur des systèmes multimédias interactifs est plus important, les images et les informations vidéo et audio venant s'ajouter à l'information textuelle traditionnelle.

L'élaboration de ces systèmes repose sur la prise en charge des informations multimédias dans les domaines suivants:

- **Plates-formes:** les nouvelles normes de compression, comme les normes JPEG (ISO/CEI 10918) ou MPEG (ISO/CEI 11172 et ISO/CEI 13818), ont permis de développer des matériels spécialisés qui, dans le futur, seront proposés sous forme de ressource de système normalisée.
- **Mémorisation:** l'augmentation de la capacité des disques magnétiques et optiques et les techniques de compression permettent de stocker des quantités croissantes d'informations à large bande.
- **Communication:** la multiplication des supports de transmission numérique à large bande, comme le RNIS et les autres réseaux de diffusion et de télécommunication à large bande, permet de transférer convenablement de gros volumes de données.

Grâce à ces moyens, on prévoit que les applications multimédias seront conçues pour fonctionner sur des plates-formes hétérogènes et qu'elles seront interconnectées pour offrir des services multimédias.

Ces applications et services multimédias utiliseront de très nombreux objets multimédias structurés résidant dans des stations de travail, stockés sur un support de transfert numérique et extraits ou distribués à partir de sources distantes via un réseau. Ces données multimédias représenteront un investissement important, et il est essentiel que les informations soient utilisables dans un contexte d'évolution rapide des systèmes et des technologies. En particulier, il est important que les informations soient transférables entre les structures de données prises en charge par différentes applications.

Les besoins suivants ont été identifiés:

- interrogation de bases de données multimédias;
- fréquentes mises à jour des données multimédias;
- manipulation d'un ensemble d'éléments de données;
- création de documents multimédias sur une variété de stations de travail;
- composition de données multimédias dans le temps et dans l'espace;
- capacité à établir un lien entre une partie de document et une autre partie du même document ou d'un autre document dans un environnement ouvert d'hyperdocument;
- synchronisation;
- synchronisation élémentaire: synchronisation de deux objets par rapport au même instant de référence (mode parallèle) ou l'un par rapport à l'autre (mode séquentiel);
- synchronisation en chaîne: les objets d'un ensemble sont présentés l'un après l'autre à la chaîne;
- synchronisation cyclique: un ou plusieurs objets sont présentés de façon répétitive;
- synchronisation conditionnelle: la présentation d'un objet est liée à la satisfaction d'une condition;



- réutilisation de données multimédias par leur intégration dans différents documents;
- transfert de données multimédias entre systèmes hétérogènes;
- utilisateurs géographiquement dispersés, y compris les groupes fermés d'utilisateurs;
- interactivité en temps réel, y compris l'acquisition de données multimédias;
- télécommunication de données multimédias;
- diffusion de données multimédias;
- large éventail de volumes de données et de débits de transfert;
- contrôle d'accès, sécurité;
- tarification;
- droit d'auteur, octroi de licences;
- grande variété de documentations support;
- utilisation d'un large éventail de terminaux et de stations de travail, y compris de dispositifs à ressources minimales;
- ressources minimales: fonctionnalités que la syntaxe de transfert doit offrir pour qu'un ensemble d'objets donné, fonctionnant dans un environnement à ressources limitées, satisfasse aux spécifications fonctionnelles déterminées par le concepteur;
- facilité de spécification: pour spécifier les ressources recommandées minimales requises par l'application MHEG;
- facilité d'application: le processus de négociation doit pouvoir déterminer facilement les besoins minimaux d'une application;
- souplesse: le mécanisme doit rester fiable pour un large éventail de configurations de systèmes et de formats de supports;
- nature informative plutôt que restrictive: l'application utilisatrice doit déterminer en dernier lieu si un ensemble d'objets donné peut être interprété;
- extensibilité/ouverture: la représentation doit continuer à fonctionner correctement après l'ajout de nouveaux formats, de nouvelles classes ou de nouveaux paramètres dépendant de l'application;
- variabilité d'échelle: compromis entre différentes capacités de plate-forme d'application (amélioration de la résolution temporelle de présentation par exemple);
- composition: les ressources minimales prescrites pour une application MHEG donnée doivent pouvoir être combinées d'une manière cohérente et prévisible lorsque des applications sont combinées;
- progressivité de dégradation: la première préoccupation liée aux systèmes à ressources minimales est la façon dont la dégradation peut être traitée; une application doit être en mesure de définir sa politique concernant cette question, par exemple, par l'utilisation de mécanismes de variabilité d'échelle et de mécanismes connexes;
- transfert et présentation en temps réel;
- optimisation du placement des objets: les objets auxquels on accède simultanément de manière probable sont adjacents du point de vue du mécanisme d'accès;
- accès progressif aux objets: les images peuvent être récupérées et présentées avec une résolution croissante par les systèmes dont le temps de présentation est important; des objets à échelle réduite peuvent être représentés et récupérés par les systèmes dont les ressources sont insuffisantes pour une présentation complète;
- récupération partielle d'objet: les grands objets peuvent être récupérés en plusieurs portions, étant donné que la totalité du contenu ne sera pas présentée en même temps;
- séquençement des objets: l'ordre dans lequel on prévoit de présenter les objets doit être utilisé par le mécanisme d'accès lorsqu'un débit insuffisant conduirait à des durées inacceptables pour transférer l'objet complet;

- récupération séparée de la description d'objet et du contenu d'objet: on doit pouvoir récupérer la description d'objet sans nécessairement récupérer le contenu de sorte que le système puisse utiliser les informations relatives à un ensemble d'objets pour optimiser l'accès pour cet ensemble et que les ressources nécessaires pour l'accès puissent être préparées;
- index global des objets: une table comprenant tous les objets et leur position dans un ensemble d'objets doit être prévue afin de permettre une consultation rapide des objets;
- entrelacement d'objets: les objets à récupérer simultanément peuvent être entrelacés de sorte que les grands objets ne retardent pas les autres objets;
- besoins en ressources: les ressources nécessaires à la récupération et à la présentation par le système cible doivent pouvoir être déterminées par consultation plutôt que par dérivation;
- récupération groupée: les objets d'une collection peuvent être regroupés de sorte qu'ils puissent être récupérés en une seule fois plutôt qu'avec une série de demandes;
- capacité à faire référence à des objets monomédias particuliers de façon interne et externe;
- capacité à naviguer à la manière des hyperdocuments;
- capacité à prendre en charge des schémas transversaux d'hyper-liens arbitraires.

Ces besoins peuvent se résumer par un ensemble de besoins génériques:

- les concepteurs et les distributeurs d'application réclament la portabilité des applications dans un environnement multi-fournisseurs;
- les applications doivent traiter des informations multimédias dont la structure est telle qu'on puisse garantir une interactivité en temps réel (y compris l'acquisition de données multimédias) ainsi qu'un transfert en temps réel des données multimédias;
- les applications doivent composer et synchroniser les données multimédias dans l'espace et dans le temps;
- les applications doivent être en mesure de définir des liens entre les éléments de données multimédias;
- les applications doivent pouvoir réutiliser les données multimédias par intégration dans différents contextes;
- les applications doivent être capables de mettre à jour fréquemment les données multimédias et de manipuler un ensemble d'éléments de données;
- les applications doivent pouvoir être interprétées sur différents systèmes, depuis les systèmes à ressources minimales jusqu'aux systèmes à ressources non limitées;
- les applications doivent être transférées et présentées en temps réel.

Ces prescriptions ont débouché sur l'élaboration de la présente Recommandation relative aux informations multimédias et hypermédias.

### **0.3 Justification de la normalisation des informations multimédias et hypermédias**

L'élaboration d'une norme définissant les structures des informations multimédias et hypermédias se justifie comme suit.

- la normalisation au seul niveau des informations monomédias ne suffit pas à garantir la portabilité des applications. Les applications n'utilisent pas les données monomédias séparément, mais doivent définir un ensemble associé de paramètres nécessaires à la présentation. (Ces paramètres peuvent comprendre l'identificateur de l'algorithme de codage appliqué aux données, les paramètres de décodage correspondants et des attributs optionnels à utiliser pour la présentation.);
- la normalisation au seul niveau des informations monomédias ne suffit pas pour la conception et le transfert des informations multimédias et hypermédias. Les applications multimédias et hypermédias s'appuient sur des abstractions ou des structures de données qui prévoient par exemple des liens de synchronisation et des liens logiques entre données monomédias;
- la conception et la gestion des applications multimédias et hypermédias dans des environnements répartis seront facilitées si les détails internes de la présentation des informations sont masqués à l'application par l'utilisation d'abstractions appropriées. L'application ne doit avoir à traiter que des fonctions comme la gestion de la répartition des informations, la programmation de la présentation, la gestion du dialogue avec les utilisateurs et d'autres activités de haut niveau.

La présente Recommandation vise à définir des structures d'informations multimédias et hypermédias génériques qui répondent à ces considérations et qui en outre soient adaptées:

- à la présentation en temps réel à l'aide des fonctionnalités de présentation et de synchronisation multimédias disponibles sur la plate-forme d'application;
- au transfert en temps réel à l'aide des fonctionnalités de communication disponibles sur la plate-forme d'application;
- à la représentation sous forme finale dans laquelle les informations sont représentées et codées pour une présentation directe sans traitement supplémentaire de la structure.

## 0.4 Objectifs T.171

### 0.4.1 Transfert

La présente Recommandation vise à définir des fonctionnalités de transfert pour divers types de supports. Les données d'un support donné peuvent être codées conformément aux autres normes internationales (JPEG pour les images fixes, MPEG pour les informations vidéo par exemple), ou être encapsulées à l'aide de techniques de codage privées et hors normes. Les unités de transfert définies par la présente Recommandation sont traitées par le moteur MHEG sous le contrôle d'une application utilisatrice.

Pour prendre en charge le transfert multimédia – par opposition à de multiples transferts monomédias – la présente Recommandation prévoit des structures permettant de composer différents types de supports en une seule unité de transfert.

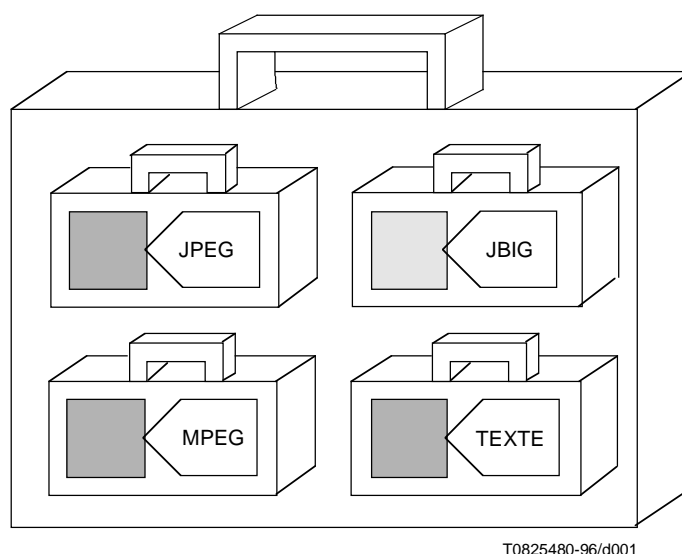
### 0.4.2 Présentation

La présente Recommandation vise à prendre en charge la présentation sous forme finale de multiples types de données. Elle prévoit des fonctionnalités permettant d'identifier la technique de codage afin de pouvoir utiliser les ressources de présentation appropriées sur une plate-forme particulière.

Pour prendre en charge la présentation multimédia – par opposition à de multiples présentations monomédias – la présente Recommandation prévoit des structures permettant de combiner différents types de support dans une présentation. Cette combinaison prend la forme de séquençement temporel, de positionnement spatial et d'interaction logique entre les supports.

Par ailleurs, la présente Recommandation prend en charge la modification et l'interaction avec les données monomédias avec leurs attributs de présentation associés (taille, position ou volume par exemple).

La Figure 0.1 montre l'utilisation de la présente Recommandation comme unité de transfert et comme identificateur d'informations multimédias codées selon différentes normes internationales.



**Figure 0.1/T.171 – Utilisation de la Recommandation T.171 pour le transfert et l'identification des informations multimédias**

### 0.4.3 Ressources minimales

La présente Recommandation vise à prendre en charge la spécification des ressources minimales nécessaires à la présentation des données codées, et prévoit des fonctionnalités pour le transfert des informations fournies par la source d'informations relatives à ces ressources.

### 0.4.4 Temps réel

La présente Recommandation vise à faciliter le transfert en temps réel d'informations multimédias et prévoit des fonctionnalités pour aider un système utilisateur à réaliser un flux d'informations approprié.

## 0.5 Concepts T.171

Le présent sous-paragraphe est une introduction générale aux concepts de la présente Recommandation et présente les concepts suivants:

- **classes MHEG, objets MHEG:** la présente Recommandation est une norme qui définit des classes d'objets. A partir de ces classes, des objets MHEG peuvent être instanciés par le concepteur d'objet (un logiciel par exemple) et transférés entre applications utilisatrices. Un nombre quelconque d'objets MHEG peuvent être instanciés à partir d'une classe MHEG donnée;
- **objets d'exécution (rt-objets):** la classe modèle MHEG est une classe abstraite MHEG, dont dérivent les classes MHEG suivantes: classes *script*, *contenu*, *contenu multiplexé* et *composite*. A partir de ces classes modèles, des objets modèles peuvent être instanciés et transférés par le concepteur d'objet. Pour pouvoir réutiliser les données transférées dans des objets modèles dans différents contextes, le concepteur d'objet peut créer à partir d'un objet modèle donné des objets d'exécution, aussi appelés rt-objets: les mêmes données d'un objet *contenu* peuvent par exemple être présentées deux fois au même moment avec des tailles différentes grâce à deux objets rt-contenus, mais l'objet *contenu* ne sera transféré qu'une seule fois. Un nombre quelconque de rt-objets peuvent être créés par le concepteur d'objet à partir d'un objet modèle donné;
- **canaux:** la présente Recommandation définit des espaces logiques dans lesquels les rt-objets sont présentés et perçus par l'utilisateur.

### 0.5.1 Orientation objet

La présente Recommandation définit une classification des structures correspondant aux unités d'informations multimédias et hypermédias à transférer. Cette classification repose sur une analyse du comportement commun et des propriétés communes des informations multimédias et hypermédias et prend la forme d'une approche orientée objet de la normalisation. Cette approche se traduit par des structures d'informations autonomes et réutilisables qui sont génériques pour les applications multimédias et hypermédias.

Dans le cadre de la présente Recommandation, le terme «classe» désigne une structure d'informations multimédias et hypermédias transférables à partir de laquelle des objets MHEG peuvent être instanciés. La présente Recommandation utilise le mécanisme orienté objet de dérivation de sous-classes à partir des classes existantes, ainsi que le concept associé d'héritage. Toutefois, étant donné que les objets MHEG doivent être considérés comme des données plutôt que comme un code exécutable, aucune interface d'objets MHEG sous forme de signatures de méthodes n'est définie.

Il va sans dire que ce n'est pas parce qu'une approche orientée objet est utilisée pour définir les structures MHEG qu'un système interprétant ces structures doit être basé sur cette approche. La présente Recommandation définit un format de transfert pour les informations multimédias et hypermédias, mais elle ne fait aucune hypothèse sur la représentation interne des structures MHEG ou sur la conception des systèmes, des moteurs, des interpréteurs, des outils ou des applications utilisatrices.

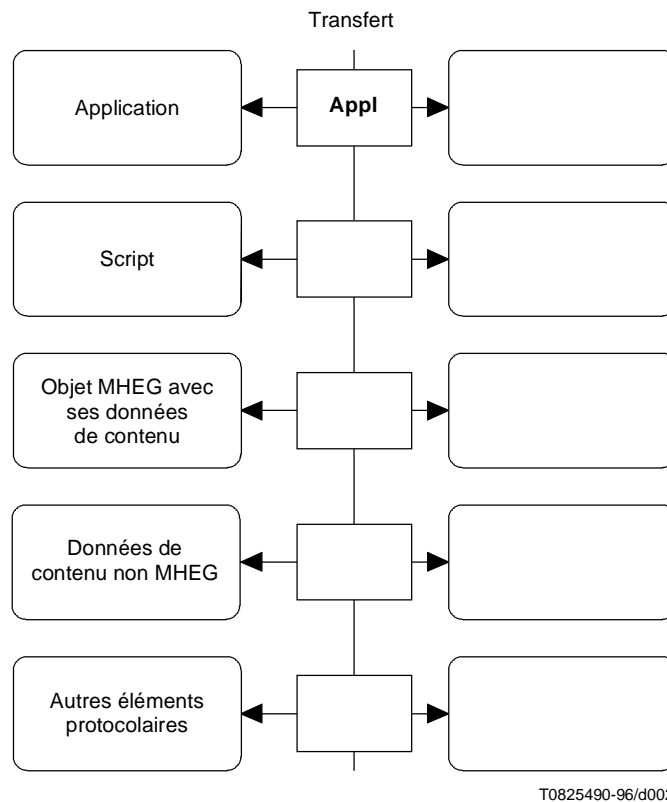
Les classes MHEG présentent un niveau de complexité qui convient bien à leur utilisation en tant que structures génériques dans un large éventail d'applications utilisatrices et de domaines. On suppose que la sémantique associée à l'utilisation des classes MHEG est définie au niveau de l'application et non au niveau de la présente Recommandation. Lorsque les applications utilisatrices se transmettent des informations multimédias et hypermédias, elles s'appuient généralement sur des services de transfert, qui fournissent le support approprié pour la transmission des données. Il existe divers niveaux de complexité pour le transfert de données, comme le montre la Figure 0.2

Sur la Figure 0.2, les applications A et B peuvent être considérées comme une seule application par l'utilisateur.

La Figure 0.2 montre le transfert entre les applications utilisatrices A et B et indique les niveaux auxquels ce transfert a lieu.

- Niveau application: ce transfert (Appl) n'est pas traité dans la présente Recommandation. L'application utilisatrice peut utiliser une norme de transfert de script au niveau de la couche immédiatement inférieure.
- Niveau script: ce transfert (S), traité dans l'ISO/CEI 13522-3 et dans des normes portant sur le transfert d'hyperdocuments et de scripticiel, peut utiliser la norme de transfert d'objet MHEG au niveau de la couche immédiatement inférieure.
- Niveau objet MHEG: ce transfert (M) fait l'objet de la présente Recommandation. Il utilise des Recommandations et des Normes portant sur le transfert de données de contenu.
- Niveau données de contenu non MHEG: ce transfert (C) est traité dans des Recommandations et des Normes monomédias indépendantes.
- Niveau des autres éléments protocolaires: ce transfert (OPE) d'éléments (messages et acquittements par exemple) est requis par l'application mais n'est pas traité dans la présente Recommandation.

Ce modèle montre aussi qu'il est nécessaire d'étudier l'établissement de liens entre le script et les objets MHEG et entre les objets MHEG et les données de contenu.



**Figure 0.2/T.171 – Modèle de transfert d'informations multimédias et hypermédias**

### 0.5.2 Codage

La présente Recommandation définit une représentation codée de chaque classe MHEG. Les instances de ces représentations codées, qui sont des objets MHEG, représentent les données que les applications multimédias et hypermédias doivent transférer et présenter.

La représentation codée de base, fondée sur les Recommandations X.680 «*Notation de syntaxe abstraite numéro un: Spécification de la notation de base*» (ASN.1) et X.690 «*Règles de codage de l'ASN.1: Spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives*», est décrite dans la présente Recommandation.

D'autres représentations codées peuvent être fournies; mais elles seront isomorphes à la représentation codée de base. Ces différentes notations et ces différents schémas de codage servent tous à exprimer la même et unique représentation des objets MHEG.

### 0.5.3 Aperçu des classes MHEG

Les classes définies dans la présente Recommandation peuvent servir à spécifier:

- des objets contenant des informations monomédias;
- une relation entre objets;
- un comportement dynamique d'objet;
- des informations permettant d'optimiser le traitement en temps réel des objets.

La présente Recommandation définit les classes instanciables suivantes:

#### 0.5.3.1 Classe *contenu*

La classe *contenu* est une classe modèle; elle contient ou fait référence à la représentation codée d'informations monomédias ainsi qu'à un ensemble de paramètres comportant les informations nécessaires à la présentation du contenu. Cet ensemble de paramètres comporte un identificateur de la méthode de codage et un champ pour la spécification des paramètres orientés application (polices de caractères et table des couleurs par exemple). La classe *contenu* spécifie aussi la taille, la durée et le volume d'origine des données. Ces valeurs sont exprimées en unités temporelles et spatiales génériques. On peut dériver des objets *contenus multiplexés* à partir de la classe *contenu*; un objet *contenu* qui n'est pas multiplexé est appelé objet *contenu pur* ou objet *contenu non multiplexé*.

#### 0.5.3.2 Classe *contenu multiplexé*

La classe *contenu multiplexé* est une classe modèle qui est une sous-classe de la classe *contenu*. Elle contient ou fait référence à la représentation codée de données monomédias multiplexées ainsi qu'à une description de chaque flux multiplexé.

#### 0.5.3.3 Classe *composite*

La classe *composite* est une classe modèle; elle permet de prendre en charge l'association des objets multimédias et hypermédias. Ce mécanisme fournit une approche cohérente pour l'établissement de liens entre objets et leur synchronisation temporelle et spatiale. Cette classe fournit aussi la structure logique permettant de décrire la liste des interactions possibles offertes à l'utilisateur, mais ne définit pas les fonctionnalités d'interaction assurées par l'interface utilisateur. Une telle interaction peut être réalisée de diverses manières (interface graphique d'utilisateur ou clavier par exemple). La présente Recommandation ne définit pas l'aspect et la convivialité des présentations multimédias interactives, elle ne propose pas non plus de modifier les concepts existants dans les interfaces graphiques d'utilisateur classiques ou d'en ajouter de nouveaux. Etant donné que la présente Recommandation est générique et indépendante de toute plate-forme et de toute réalisation, elle décrit l'interaction à un niveau virtuel. Il appartient à l'application utilisatrice d'appliquer ces mécanismes selon ses propres caractéristiques de reproduction et de convivialité.

#### 0.5.3.4 Classe *action*

La présente Recommandation prévoit un comportement initial pour chaque objet MHEG, objet d'exécution (rt-objet) et canal (par exemple une position par défaut d'un rt-contenu) et fournit un moyen de modifier ce comportement initial par la définition d'une liste d'actions élémentaires à appliquer aux objets. La modification du comportement se fait par le transfert des actions élémentaires correspondantes dans les objets *actions*. Ces objets sont utilisés dans un objet *lien* pour décrire l'effet du lien.

La classe *action* définit une structure qui spécifie un ensemble synchronisé d'actions élémentaires à appliquer à une ou plusieurs cibles.

#### 0.5.3.5 Actions élémentaires

La présente Recommandation définit des actions élémentaires qui peuvent affecter les comportements suivants d'un objet MHEG, d'un rt-objet ou d'un canal.

- **Préparation:** des actions sont prévues pour régir la disponibilité de l'objet MHEG dans le système. Les actions `Prepare` (préparation) et `Destroy` (destruction) permettent par exemple d'ajouter ou de supprimer un objet MHEG dans le système.
- **Création d'objets d'exécution (rt-objets):** l'action `New` (nouveau) permet de créer des rt-objets (rt-script, rt-composant) à partir d'un objet modèle (script, composant); l'action `Delete` (suppression) permet de les détruire.

- **Présentation:** des actions sont prévues pour régir la progression des rt-composants dans le système. On peut par exemple appliquer les actions `Run` (exécution) et `Stop` (arrêt) pour commander la progression d'un rt-composant dans le temps.
- **Reproduction:** des actions sont prévues pour régir la projection du rt-composant sur le système. Ces actions diffèrent selon le type de support. On peut par exemple appliquer l'action `Set GTF` (fixation du facteur GTF) pour modifier la vitesse de présentation des supports à dimension temporelle, ou l'action `Set OVS` (fixation de la taille OVS) pour modifier la taille de présentation des supports visuels.
- **Interaction:** des actions sont prévues pour régir les résultats de l'interaction avec un rt-composant dans le système. Par exemple, l'action `Set Interaction Ability` (fixation de la capacité d'interaction) spécifie s'il est possible de sélectionner et de modifier un rt-composant.
- **Activation:** des actions sont prévues pour régir l'activation des rt-scripts dans le système (actions `Run` (exécution) et `Stop` (arrêt) par exemple).

#### 0.5.3.6 Actions d'extraction

La présente Recommandation définit aussi un moyen permettant de consulter le comportement d'un objet MHEG, d'un rt-objet ou d'un canal. Des actions sont prévues pour extraire l'attribut de comportement ou la valeur d'état des objets MHEG, des rt-objets et des canaux. Ces actions sont appelées «actions d'extraction» (*get*). Le résultat d'une telle action est une valeur générique ou la valeur 'undefined' (indéfini) en cas d'erreur. Les actions d'extraction servent à exprimer la valeur source et la valeur de comparaison de la condition de lien sous forme d'un objet *lien* ou sous forme d'un paramètre quelconque d'une action élémentaire.

#### 0.5.3.7 Classe *lien*

La classe *lien* définit une structure qui spécifie un ensemble de relations. Chaque relation lie une ou plusieurs sources à une ou plusieurs cibles. Chaque relation est composée de conditions associées aux sources (condition de lien) et d'actions à appliquer aux cibles (effet de lien). Les actions, qui sont décrites dans des objets *actions*, doivent être appliquées aux cibles lorsque les conditions sont satisfaites.

Chaque source et chaque cible peut être une instance d'une classe MHEG quelconque, y compris d'une classe *lien* ou d'une classe *action*. Les sources et les cibles peuvent aussi être des objets d'exécution quelconques.

Des instances de la classe *lien* servent à spécifier le séquençage temporel, le positionnement spatial ou l'interaction logique entre objets MHEG et objets d'exécution.

#### 0.5.3.8 Classe *script*

La classe *script* est une classe modèle. Elle définit un conteneur pour la relation complexe qui existe entre objets MHEG, objets d'exécution et canaux, spécifiée dans un langage d'écriture de script. La présente Recommandation ne définit pas ce langage proprement dit, mais définit la classe *script* qui encapsule un script et une indication sur le langage utilisé.

On suppose que le langage d'écriture de script utilisé dans un objet *script* est en mesure de faire référence aux objets MHEG, aux rt-objets et aux canaux, et aussi d'accéder à leurs attributs. A cet égard, la Recommandation T.174 définit une interface avec les entités MHEG et la Recommandation T.173 définit un format de transfert pour les scripts.

#### 0.5.3.9 Classe *descripteur*

La classe *descripteur* définit une structure pour le transfert d'informations relatives aux ressources à propos d'un objet ou d'un ensemble d'autres objets transférés. Les objets décrits sont appelés objets «connexes». Ces informations peuvent servir à faciliter l'adéquation entre les ressources nécessaires pour présenter les objets et les ressources disponibles dans le système, ou à effectuer une négociation entre la source des objets MHEG et le site de présentation.

#### 0.5.3.10 Classe *conteneur*

La classe *conteneur* permet de regrouper les données multimédias et hypermédias afin de les transférer en un seul et même ensemble.

### 0.5.4 Objets d'exécution (rt-objets)

Pour pouvoir réutiliser les objets modèles (objets *scripts*, objets *composants*) dans différentes présentations ou activations, une nette distinction est faite entre l'objet modèle transféré, qui contient la composition ou les données réutilisables, et le rt-objet correspondant à une vue particulière de la composition ou des données modèles.

Les rt-objets (objets rt-scripts, rt-contenus, rt-mux et rt-composites) sont créés par le concepteur d'objet à l'aide de l'action MHEG New (nouveau). La présentation ou l'activation d'un rt-objet n'affecte pas l'objet modèle; cela permet de réutiliser un même objet modèle dans différents rt-objets.

### 0.5.5 Canaux

Un canal est un espace logique dans lequel les rt-composants (objets rt-contenus et rt-composites) sont présentés et perçus par l'utilisateur. Les canaux sont créés par le concepteur d'objet à l'aide d'une action MHEG. Le concepteur d'objet peut fournir des informations dans l'objet *descripteur* pour faciliter la projection par le moteur MHEG des canaux logiques sur le monde réel.

## 0.6 Interface d'application MHEG

La présente Recommandation ne définit pas d'interface de programmation d'application API (*Application Programming Interface*) pour le traitement des objets dans un système. Toutefois, on suppose que le moteur MHEG fournit à l'application utilisatrice des fonctionnalités pour prendre en charge les actions définies dans la présente Recommandation.

## 0.7 Possibilités d'extension T.171

La nature des classes définies dans la présente Recommandation permet de les utiliser dans un large éventail d'applications et de domaines. Mais certaines applications peuvent nécessiter une fonctionnalité particulière qui n'est pas directement fournie par les classes; une application utilisatrice peut par exemple avoir besoin de ressources spécialisées disponibles sur une plate-forme donnée. Dans ce cas, il est possible d'associer aux objets MHEG des ressources supplémentaires qui ne sont pas fournies directement par la présente Recommandation. Cette association est définie à l'aide de l'une des techniques d'extension suivantes. Ces techniques ne modifient pas la représentation codée des fonctionnalités définies par la présente Recommandation (voir paragraphe 15):

- 1) extension des actions élémentaires;
- 2) extension des attributs d'un objet MHEG, d'un rt-objet ou d'un canal;
- 3) extension des types de données, des classifications, des styles, des événements, des types de support des canaux;
- 4) adjonction de nouvelles classes d'objets.

NOTE – Cette fonctionnalité peut par exemple servir à créer et à manipuler de nouveaux attributs, comme le paiement ou la couleur, ou à faire varier un paramètre de modulation de la parole dans un objet audiophonique.

Pour les points 1) à 3), l'ISO/CEI 13522-4 prévoit des catalogues et une procédure d'enregistrement. Cela permet aux utilisateurs de définir des extensions sans réduire de façon importante la portabilité. Ces extensions permettent aux utilisateurs d'utiliser des catalogues privés ou enregistrés (voir paragraphes 14 et 15.1). Un moteur MHEG peut prendre en considération une rubrique de catalogue quelconque puis lancer le processus d'extension New (nouveau); un autre moteur MHEG dépourvu d'extension peut ignorer l'extension.

L'adjonction de nouvelles classes d'objets suppose l'adaptation du moteur MHEG pour pouvoir les interpréter, ce qui limite l'utilisation d'objets MHEG dérivés de ces classes à ce seul moteur MHEG.

## 1 Domaine d'application

La présente Recommandation porte sur la représentation codée des objets informationnels multimédias et hypermédias sous forme finale, qui seront transférés en tant qu'unités dans ou entre les services et applications, par un moyen de transfert quelconque (support mémoire, réseau local, réseau de télécommunication, de radiocommunication ou de télédiffusion par exemple). Ces objets définissent la structure d'une présentation multimédia.

Ces objets, appelés objets MHEG dans la suite, fournissent les fonctionnalités permettant de prendre en charge:

- la représentation sous forme finale,
- les systèmes à ressources minimales,



- l'interactivité et la synchronisation multimédia,
- la présentation en temps réel,
- le transfert en temps réel,

sous forme d'une base commune pour de nombreuses applications multimédias et hypermédias.

La présente Recommandation définit les spécifications des objets MHEG qui sont communes à tous les types de représentation codée, et définit aussi un type particulier de représentation codée, la représentation codée de base.

## 1.1 Spécificité du domaine d'application

Etant donné qu'on prévoit que de nombreuses Recommandations, Normes et applications utiliseront la présente Recommandation, celle-ci s'intéresse essentiellement aux aspects de structuration générique. La présente Recommandation reconnaît la sémantique qui est induite par la spécification des objets MHEG mais n'impose aucune interprétation sémantique à l'application utilisatrice.

## 1.2 Questions sortant du cadre T.171

La normalisation de modèles, de services, de systèmes, de protocoles ou d'applications susceptibles d'utiliser des objets MHEG sort du cadre de la présente Recommandation. L'interfonctionnement et l'intégration d'objets MHEG dans ces modèles, services, systèmes, protocoles ou applications sont définis par d'autres organes de normalisation du JTC1 ou de l'UIT-T. La partie générique de l'interface entre objets MHEG et applications a été définie en liaison et en collaboration avec ces organes.

La représentation codée des données de contenu sort du cadre de la présente Recommandation.

## 2 Conformité

La conformité est évaluée sur les objets transférés. La conformité des systèmes, des moteurs et des processus n'est pas traitée ici.

Un objet transféré est conforme s'il peut être interprété correctement, c'est-à-dire si:

- son codage ASN.1 est correct (conformité à la syntaxe concrète);
- sa notation ASN.1 est conforme à la grammaire spécifiée dans la présente Recommandation (conformité à la syntaxe abstraite).

Un décodeur MHEG conforme reconnaîtra et décodera tous les objets MHEG définis dans la présente Recommandation.

### 2.1 Profils

Des profils peuvent être fournis par les applications utilisatrices.

### 2.2 Syntaxe

La conformité est évaluée à partir de la représentation de base de la présente Recommandation. Une instance d'objet MHEG est représentée à l'aide de la notation ASN.1.

La conformité au niveau de la représentation codée est garantie par l'utilisation de normes internationales sur le codage de la syntaxe.

### 2.3 Sémantique

La présente Recommandation définit une représentation codée permettant d'exprimer le comportement prévu d'objets (relations temporelles et spatiales par exemple), mais ne définit pas les techniques à même de traiter ces relations entre les objets.

### 3 Références normatives

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

- Recommandation UIT-T X.680 (1994) | ISO/CEI 8824-1:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification de la notation de base.*
- Recommandation UIT-T X.681 (1994) | ISO/CEI 8824-2:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des objets informationnels.*
- Recommandation UIT-T X.682 (1994) | ISO/CEI 8824-3:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: spécification des contraintes.*
- Recommandation UIT-T X.683 (1994) | ISO/CEI 8824-4:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: paramétrage des spécifications de la notation de syntaxe abstraite numéro un.*
- Recommandation UIT-T X.690 (1994) | ISO/CEI 8825-1:1995, *Technologies de l'information – Règles de codage de la notation de syntaxe abstraite numéro un: spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives.*
- Recommandation UIT-T X.691 (1995) | ISO/CEI 8825-2:1995, *Technologies de l'information – Règles de codage de la notation de syntaxe abstraite numéro un: spécification des règles de codage compact.*
- ISO/CEI 13522-4:1996, *Technologies de l'information – Représentation codée d'informations multimédia et hypermédia – Partie 4: procédure d'enregistrement MHEG.*

### 4 Définitions

La présente Recommandation définit les termes suivants:

- définitions données dans les Recommandations X.680 et X.690.
- définitions ci-dessous.

**4.1 classe abstraite:** classe MHEG qui ne peut pas être instanciée, c'est-à-dire dont les objets ne peuvent être transférés. Une telle classe encapsule des informations et transmet en héritage des attributs à ses sous-classes; elle agit aussi comme un mécanisme permettant de spécifier les actions MHEG qui peuvent être appliquées à une classe MHEG donnée.

**4.2 action:** action élémentaire ou action d'extraction.

**4.3 effet d'action:** correspond, pour toutes les actions élémentaires, à la concaténation de l'effet MHEG et de l'effet utilisateur. Pour les actions d'extraction, il correspond à l'effet MHEG.

**4.4 objet action:** instance de classe MHEG qui définit un ensemble organisé d'actions élémentaires.

**4.5 activation; actif:** action élémentaire qui fait passer un objet *lien* à l'état actif; seul un lien actif peut être déclenché.

**4.6 alias:** attribut pouvant être associé à une entité MHEG, à des données ou à un flux. Il est utilisé comme identificateur de remplacement.

**4.7 ancre:** objet *contenu* contenant des informations d'ancrage et non des données monomédias. Les informations d'ancrage servent à positionner une ancre par rapport à un document pivot.

**4.8 attribut:** valeur typée représentant des caractéristiques particulières d'un objet MHEG transféré, d'un rt-objet ou d'un canal.

**4.9 auteur; concepteur d'objet:** logiciel ou autre dispositif lorsque les objets sont générés par des moyens mécaniques ou électroniques.

**4.10 canal:** dispositif logique dans lequel des rt-composants sont positionnés pour être présentés sous forme finale. Les canaux sont projetés par un moteur MHEG sur des dispositifs physiques (fenêtres d'écrans ou haut-parleurs par exemple) pour permettre à l'utilisateur de percevoir les rt-objets qu'ils contiennent.

- 4.11 représentation codée:** représentation binaire de la structure et des données figurant dans une instance d'objet.
- 4.12 objet composant:** instance de classe MHEG qui représente un objet *contenu*, un objet *contenu multiplexé* ou un objet *composite*.
- 4.13 objet composite:** instance de classe MHEG qui définit une liste d'éléments de composition regroupés pour la présentation. La présentation d'un objet *composite* correspond à la présentation de ses éléments de composition.
- 4.14 éléments de composition:** attribut de la classe *composite* MHEG qui définit sa présentation et sa sous-hiérarchie.
- 4.15 contrainte:** partie d'une condition de lien décrivant une valeur d'attribut et une valeur d'état. Elle désigne une condition supplémentaire à remplir une fois les conditions de déclenchement du lien satisfaites.
- 4.16 objet conteneur:** instance de classe MHEG qui définit une liste d'objets regroupés pour le transfert. L'objet *conteneur* permet de regrouper des objets sans spécifier de relation particulière.
- 4.17 objet contenu:** instance de classe MHEG qui fournit une structure cohérente pour le transfert d'une valeur générique codée ou de données de présentation quelconques. Il s'agit d'un contenu multiplexé ou non multiplexé.
- 4.18 données:** attribut des objets *contenus* et *scripts*.
- 4.19 canal par défaut:** canal qui est toujours mis à la disposition du moteur MHEG au moyen d'un identificateur réservé (à savoir, «Default-Channel» [canal par défaut]). Les actions élémentaires «New channel» (nouveau canal) et «Delete channel» (suppression de canal) sont ignorées dans le cas du canal par défaut. Lorsqu'un rt-composant racine n'est pas assigné à un canal particulier, il est implicitement assigné au canal par défaut.
- 4.20 suppression, suppression d'un canal, indisponible:** action élémentaire qui supprime un rt-objet ou un canal d'un moteur MHEG. Le rt-objet ou le canal est dit alors indisponible.
- 4.21 démultiplexage:** extraction de deux flux d'information ou plus à partir d'un unique schéma de codage combiné pour les présenter ou les transférer séparément.
- 4.22 descendant:** entité MHEG quelconque de la sous-arborescence ayant l'objet initial pour racine.
- NOTE – Le graphe de la descendance peut contenir des boucles. Une entité peut être son propre descendant.
- 4.23 objet descripteur:** instance de classe MHEG qui définit les informations associées à un objet quelconque. Une application utilisatrice peut employer cet objet pour négocier les caractéristiques des moteurs MHEG.
- 4.24 destruction, pas prêt, indisponible:** action élémentaire qui supprime un objet MHEG d'un moteur MHEG. L'objet est dit alors pas prêt ou indisponible.
- 4.25 action élémentaire:** action servant à modifier le comportement d'une entité MHEG dans le moteur MHEG. Les actions élémentaires sont transférées dans des objets *actions*. Une action élémentaire est typiquement composée d'un ensemble de cibles, de paramètres spécifiques définis pour chaque action élémentaire et d'une durée de transition optionnelle.
- NOTE – Exemples d'actions élémentaires: Prepare [*préparation*] (ensemble de cibles), Run [*exécution*] (ensemble de cibles, nombre d'exécutions).
- 4.26 réceptacle vide:** réceptacle dans lequel aucun rt-composant n'a été enfiché.
- 4.27 événement:** signal généré par certains composants de système ou par le moteur MHEG (frappe d'une touche, déplacement de la souris par exemple).
- 4.28 forme finale:** syntaxe de transfert destinée à la présentation qui ne nécessite pas la modification de la structure des objets.
- 4.29 déclenchement d'un lien:** chaque fois qu'une condition d'un lien actif est satisfaite, le lien est déclenché, c'est-à-dire que son effet est exécuté.

**4.30 génération:** un objet *composite* a plusieurs réceptacles auxquels sont attachés des objets *composants* liés par la relation de fratrie. Par conséquent, on dit qu'un objet *composite* définit une génération d'objets *composants*. Plusieurs générations peuvent être constituées si d'autres objets *composites* sont enfichés dans les réceptacles. Etant donné qu'un rt-composite est créé à partir d'un objet *composite*, les rt-composites définissent aussi des générations.

**4.31 valeur générique:** booléen générique, numérique générique, entier générique, fraction générique, chaîne générique, référence générique ou liste générique. Il peut s'agir d'une constante ou d'une valeur évaluée.

**4.32 action d'extraction (*get*):** appliquée à une entité MHEG, elle spécifie l'attribut ou la valeur d'état à évaluer. L'action d'extraction fournit une valeur générique. Cette valeur, utilisée dans le cadre de l'évaluation, est une valeur locale. Si deux actions d'extraction identiques sont exécutées en parallèle dans deux liens différents, elles conduiront à deux valeurs génériques qui pourront être différentes. Une action d'extraction peut être utilisée et transférée lorsqu'une valeur est autorisée, par exemple, comme paramètre d'une action élémentaire ou d'une action d'extraction, ou comme une valeur source ou une valeur de comparaison dans un objet *lien*.

NOTE – Exemples d'actions d'extraction: `Get Preparation Status` [*extraction de l'état de préparation*] (cible) et `Get Running Status` [*extraction de l'état d'exécution*] (cible).

**4.33 crochet:** attribut de la classe *contenu* et de la classe *script* contenant des informations de codage et de décodage permettant d'utiliser les données. La sémantique de ces paramètres n'est pas définie par la présente Recommandation mais est donnée par la Norme ou Recommandation sur le codage des données.

**4.34 hypermédia:** capacité à accéder aux informations monomédias et multimédias en navigant à travers des liens.

**4.35 attribut de transfert:** attribut d'un objet MHEG transféré.

**4.36 support de transfert:** support utilisé pour transférer les données: il peut s'agir d'un support de stockage, d'un support de transmission ou d'une combinaison des deux.

**4.37 valeur de transfert:** valeur d'un attribut transféré.

**4.38 étiquette:** chaîne générique associée à un élément d'un objet composite. Lorsqu'un rt-composite est créé à partir d'un objet *composite*, un rt-contenu est créé à partir de cette étiquette et est automatiquement enfiché dans le réceptacle correspondant.

**4.39 condition de lien:** conditions à satisfaire pour déclencher un lien. Une condition de lien est composée de conditions de déclenchement et de contraintes, reliées entre elles par des opérateurs logiques.

**4.40 effet de lien:** effets d'un lien lorsqu'il est déclenché. Un effet de lien est composé d'actions élémentaires et/ou d'objets *actions*.

**4.41 objet lien:** instance de classe MHEG qui définit une relation spatio-temporelle ou conditionnelle entre entités MHEG. Un objet *lien* est composé d'une condition de lien et d'un effet de lien.

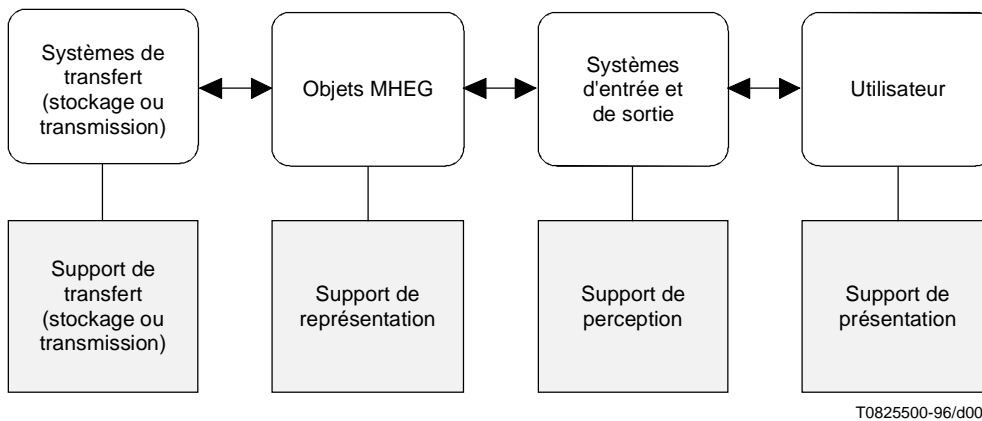
**4.42 macro-action:** objet *action* qui permet de remplacer les paramètres dans les objets *actions* fréquemment utilisés.

**4.43 macro-lien:** objet *lien* qui permet de remplacer les paramètres dans les actions conditionnelles fréquemment utilisées.

**4.44 macro-paramètre:** paramètre d'une action ou attribut de la classe *action*, qui est remplacé par les informations suivantes: un identificateur de définition de macro-instruction et une valeur d'utilisation par défaut de macro.

**4.45 valeur d'utilisation de macro:** dans l'effet de lien d'un objet *lien*, une valeur d'utilisation peut être fournie pour chaque macro-paramètre utilisé dans l'effet de lien. L'affectation est effectuée lorsque le lien est déclenché.

**4.46 support:** moyen par lequel l'information est perçue, exprimée, stockée ou transmise.



**Figure 1/T.171 – Relations entre les différentes significations du terme support**

La Figure 1 vise à clarifier la relation entre les différents sens du terme «support». Il faut distinguer le support de représentation du support de présentation pour des raisons de virtualité et de portabilité (indépendance par rapport au dispositif). Une chaîne de caractères donnée (représentation codée) peut être présentée à l'aide de différents moyens (écran, papier, haut-parleur après synthèse vocale). La même commande peut être lancée par l'utilisateur par des moyens différents mais équivalents (par exemple, sélection dans une liste avec une souris, avec un écran tactile, avec des chiffres + une touche de validation, avec un clavier pour saisir la chaîne de caractères correspondante, ou même via un microphone avec un système de reconnaissance vocale capable de reconnaître un vocabulaire limité comprenant le nom de la commande mentionnée).

NOTE – Cette définition est faible car ce terme a des sens différents selon le contexte. Il faut donc éviter d'utiliser ce terme tout seul. Pour éviter toute ambiguïté, il ne doit être utilisé que dans des expressions comme support de perception, support de représentation, support de présentation, support de stockage, support de transmission.

**4.47 classe d'objets MH:** classe d'objets MHEG qui est la racine de la hiérarchie des classes MHEG. Elle définit des attributs communs à toutes les autres classes MHEG.

**4.48 classe MHEG:** classe d'objets MHEG définie par la présente Recommandation

**4.49 effet MHEG:** effet engendré par l'exécution d'une action élémentaire ou d'une action d'extraction se traduisant par la modification ou par la récupération d'un certain état interne.

**4.50 moteur MHEG:** pseudo-mécanisme qui peut être composé d'un processus ou d'un ensemble de processus qui interprètent les objets MHEG codés conformément aux spécifications de codage de la présente Recommandation.

NOTE – La réalisation du moteur MHEG sort du cadre de la présente Recommandation, qui ne définit aucun élément constitutif du moteur MHEG.

**4.51 entité MHEG; entité:** objet MHEG, rt-objet ou canal quelconque.

**4.52 objet MHEG:** représentation codée d'un objet multimédia ou hypermédia conforme à la présente Recommandation.

**4.53 ressources minimales:** système comportant la capacité minimale de mémoire tampon, le débit minimal de canaux de communication et une puissance de calcul limitée.

**4.54 objet modèle:** instance de classe MHEG qui représente un objet *script* ou un objet *composant*.

**4.55 représentation multimédia:** manipulation de plusieurs types de supports de représentation.

#### NOTES

1 – Le terme «multimédia» est un adjectif; il sera utilisé comme qualificatif d'un nom qui indiquera le contexte (service ou application multimédia, terminal multimédia, réseau multimédia, présentation multimédia par exemple).

2 – Dans la présente Recommandation, le terme «multimédia» est utilisé dans le sens «plusieurs supports de représentation». Par conséquent, un service de télétraitement ou vidéotex utilisant un clavier et un système d'affichage de texte n'est pas multimédia, l'interaction ne faisant intervenir qu'un seul support (texte).

3 – Dans un service ou une application multimédia, un objet MHEG peut ne contenir qu'un seul support de représentation.

**4.56 multiplexage:** combinaison de deux flux d'information ou plus en un schéma de codage unique pour les présenter ou les transférer simultanément.

**4.57 objet contenu multiplexé:** instance de classe MHEG qui fournit une structure cohérente unique pour le transfert de données de présentation multiplexées quelconques.

**4.58 nouveau, nouveau canal, disponible:** action élémentaire qui permet de mettre un rt-objet ou un canal à la disposition du moteur MHEG pour un traitement ultérieur (présentation par exemple). On dit que le rt-objet ou le canal est disponible.

**4.59 objet contenu non multiplexé:** objet *contenu pur*, c'est-à-dire un objet *contenu* où le contenu n'est pas multiplexé.

**4.60 néant:** valeur spécifique représentant l'ensemble vide.

**4.61 données vides:** les données vides sont générées par le moteur MHEG lorsqu'il y est fait référence au moyen d'un identificateur réservé: «Null-Data» (données vides). Les données vides peuvent remplacer des données d'un type quelconque. Les actions exécutées sur un rt-script/rt-contenu dérivé d'un objet *script*/objet *contenu* avec des données vides ne sont pas ignorées. L'effet utilisateur de ces actions est défini par les informations de crochet et de classification données dans l'objet modèle.

**4.62 objet MHEG vide:** un objet vide est généré par le moteur MHEG lorsqu'il y est fait référence au moyen d'un identificateur réservé: «Null-MH» (objet vide). L'objet vide peut remplacer un objet quelconque d'une classe MHEG quelconque. Une action exécutée sur un objet MHEG vide a un effet nul.

**4.63 rt-objet racine vide:** un rt-objet racine vide est généré par le moteur MHEG lorsqu'il y est fait référence au moyen d'un identificateur réservé: «Null-Root-Rt» (rt-objet racine vide). Le rt-objet racine vide peut remplacer un rt-objet racine quelconque. Une action exécutée sur un rt-objet racine vide a un effet nul.

**4.64 objet:** élément d'information fini, indépendant, auto-défini, qui peut être manipulé comme un tout par les moteurs MHEG et transféré sous forme d'une seule unité.

#### NOTES

1 – Dans le cadre de la présente Recommandation, l'objet peut être de différents types (par exemple action, lien, script, contenu, composite, descripteur ou conteneur).

2 – «Auto-défini» signifie que l'objet contient les informations nécessaires à son traitement ou qu'il contient une référence à ces informations.

3 – La définition n'est pas restreinte aux objets à codage numérique, elle peut aussi s'appliquer aux objets analogiques (séquence vidéodisque par exemple).

**4.65 classe d'objets:** catégorie quelconque d'objets qui présentent un modèle propre et homogène (le modèle regroupant des caractéristiques et un comportement relatifs aux informations contenues et à leurs actions modifiant leur comportement).

**4.66 relation parent-enfant:** relation existant lorsque l'objet parent est un objet *composite* qui contient un objet enfant ou qui contient une référence à un objet enfant.

**4.67 support de perception:** nature des informations telles qu'elles sont perçues par l'être humain (parole, bruit, musique, texte, dessins, scènes animées par exemple).

**4.68 enfichage:** action élémentaire qui attache un rt-composant à un réceptacle.

**4.69 préparation; prêt:** action élémentaire qui permet de mettre un objet à la disposition du moteur MHEG pour un traitement ultérieur (présentation par exemple). On dit que l'objet est prêt.

- 4.70 présentation:** reproduction d'un rt-composant pour permettre à l'utilisateur de le percevoir.
- 4.71 support de présentation:** élément d'un dispositif de sortie sur lequel sont reproduites les informations destinées à l'utilisateur (écran, papier, imprimante, haut-parleur, ...) ou élément d'un dispositif d'entrée servant à acquérir des informations de l'utilisateur (clavier, souris, microphone, caméra, ...).
- 4.72 espace de présentation:** espace servant à présenter les rt-composants à l'utilisateur. Il est constitué d'un axe temporel, de trois axes spatiaux et d'une plage de volume sonore.
- 4.73 projection:** l'espace de présentation d'un réceptacle peut être projeté dans l'espace de présentation de son parent ou dans l'espace de présentation d'un canal, avec modification de sa position temporelle, de sa taille ou de son volume. Un rt-composant racine est toujours projeté dans l'espace de présentation d'un canal. Un certain nombre d'actions élémentaires servent à effectuer ces projections.
- 4.74 transfert en temps réel:** capacité à transférer des objets entre systèmes dans les limites de temps requises pour le programme de présentation. Cela suppose un retard limité constant. Plus particulièrement, les données monomédias temporelles et les objets synchronisés dans le temps ont des prescriptions de présentation qui peuvent ne pas être satisfaites par la mise en mémoire tampon.
- 4.75 présentation et interaction en temps réel:** capacité à présenter des objets avec un retard constant. Les objets à dimension temporelle intrinsèque (audio ou vidéo par exemple) et les objets qui ont un comportement temporel ou des contraintes temporelles (synchronisation par exemple) sont concernés.
- 4.76 système en temps réel:** système exécutant les opérations d'une manière telle que l'utilisateur a l'impression qu'elles se produisent à l'instant même. Un système en temps réel correspond donc à un système ayant seulement un retard constant. Dans un sens plus large, on peut accepter un retard limité et une gigue. Dans le cas de certaines informations monomédias (audio ou vidéo par exemple), le «temps réel» impose une présentation qui respecte les prescriptions de base de temps d'origine.
- 4.77 reproduction:** capacité à montrer un objet donné d'une façon particulière à l'utilisateur. Un certain nombre d'actions élémentaires permettent de réaliser de telles reproductions.
- 4.78 support de représentation:** type des données transférées qui définit la nature de l'information telle qu'elle est décrite par sa forme codée.

#### NOTES

1 – Exemples de supports de représentation et de formes codées possibles des informations:

- caractères ou texte: Téléx, ASCII, EBCDIC;
- graphiques: CEPT, vidéotex NAPLPS ou CAPTAIN, CGM;
- audio: Recommandation G.711, MIDI, audio MPEG;
- image fixe: télécopie du groupe 3, JBIG, JPEG;
- séquence audiovisuelle: Recommandation 601 du CCIR plus audio associé: MPEG.

2 – Le support de représentation est défini indépendamment du sens de transfert (c'est-à-dire en direction ou en provenance de l'utilisateur ou entre équipements). Chaque support de représentation peut être utilisé en entrée ou en sortie, par exemple une représentation de type caractère peut servir à la fois pour l'affichage de texte et pour la saisie de texte à partir d'un clavier; une représentation de type graphique peut servir à la fois pour l'affichage de graphiques et la saisie de graphiques (position) à l'aide d'une souris. Les représentations de type audio ou picturale peuvent servir à la fois pour la reproduction et la saisie.

- 4.79 représentation d'un objet:** description de la structure de l'objet et de ses contenus.
- 4.80 rt-composant racine:** rt-composant qui n'est pas enfiché. Un rt-composant racine désigne soit un rt-composite racine soit un rt-contenu racine.
- 4.81 rt-composite racine:** rt-composite qui n'est pas enfiché.
- 4.82 rt-contenu racine:** rt-contenu qui n'est pas enfiché. Un rt-contenu racine désigne soit un rt-non-mux racine soit un rt-mux racine.
- 4.83 rt-mux racine:** rt-mux qui n'est pas enfiché.
- 4.84 rt-non-mux racine:** rt-non-mux qui n'est pas enfiché.

- 4.85 rt-composant:** désigne soit un rt-composant racine soit un réceptacle de rt-composant.
- 4.86 réceptacle de rt-composant:** réceptacle dans lequel un rt-composant est enfiché. Un tel réceptacle désigne soit un réceptacle de rt-composite soit un réceptacle de rt-contenu.
- 4.87 rt-composite:** objet d'exécution créé à partir d'un objet *composite* modèle. Les éléments d'un rt-composite sont appelés réceptacles. Un rt-composite désigne soit un rt-composite racine soit un réceptacle de rt-composite.
- 4.88 réceptacle de rt-composite:** rt-composite qui est enfiché.
- 4.89 rt-contenu:** objet d'exécution créé à partir d'un objet *contenu* modèle. Un rt-contenu désigne soit un rt-contenu racine soit un réceptacle de rt-contenu.
- 4.90 réceptacle de rt-contenu:** rt-contenu qui est enfiché. Un réceptacle de rt-contenu désigne soit un réceptacle de rt-non-mux soit un réceptacle de rt-mux.
- 4.91 rt-mux:** contenu d'exécution créé à partir d'un objet *contenu multiplexé* modèle. Un rt-mux désigne soit un rt-mux racine soit un réceptacle de rt-mux.
- 4.92 réceptacle de rt-mux:** rt-mux qui est enfiché.
- 4.93 rt-non-mux:** contenu d'exécution créé à partir d'un objet *contenu pur* modèle, c'est-à-dire un objet *contenu* où le contenu n'est pas multiplexé. Un rt-non-mux désigne soit un rt-non-mux racine soit un réceptacle de rt-non-mux.
- 4.94 réceptacle de rt-non-mux:** rt-non-mux qui est enfiché.
- 4.95 rt-objet, objet d'exécution:** objet d'exécution créé à partir d'un objet modèle. S'agissant de la réutilisation d'objets modèles dans différents contextes, une nette distinction est faite entre l'objet MHEG modèle de transfert, qui contient la définition d'origine et le rt-objet correspondant à un emploi particulier du modèle. L'utilisation d'un rt-objet n'affecte pas l'objet modèle. Cela permet de réutiliser le même objet modèle dans différents rt-objets. Un objet modèle est considéré comme un modèle. Un nombre quelconque de rt-objets peuvent être créés sur instruction de l'auteur (action New (nouveau)). La présente Recommandation définit un comportement initial pour chaque rt-objet et les actions qui modifieront et consulteront ce comportement. La représentation interne des rt-objets n'est pas définie par la norme. Chaque moteur MHEG peut avoir sa propre technique de représentation interne. Toutefois, la présente Recommandation définit les techniques d'identification permettant de faire référence à un rt-objet afin de modifier ou de consulter son comportement.
- Un rt-objet désigne soit rt-script soit un rt-composant.
- 4.96 rt-script:** objet d'exécution créé à partir d'un objet *script* modèle.
- 4.97 processus d'exécution:** capacité à traiter un code interprété dans un moteur MHEG.
- 4.98 objet script:** instance de classe MHEG définissant une structure pour transférer des données de script sous une forme codée spécifiée.
- 4.99 scripticiel:** processus ou ensemble de processus traitant des scripts.
- 4.100 séquencement:** ordonnancement de la préparation des objets, selon les spécifications du concepteur d'objet, qui permet d'utiliser au mieux les ressources disponibles pour réaliser une présentation acceptable.
- 4.101 relation de fratrie:** relation entre réceptacles d'une génération donnée dans un rt-composite.



**4.102 réceptacle:** élément d'un rt-composite dans lequel des rt-composants sont enfichés. Par conséquent, un réceptacle est équivalent à un rt-composant enfiché. Une fois un rt-composant enfiché, on ne peut y faire référence que par l'identificateur de son réceptacle. Différents types de réceptacles sont définis selon le rt-composant qui y est enfiché: réceptacle vide, réceptacle de rt-contenu (réceptacle de rt-non-mux ou réceptacle de rt-mux) et réceptacle de rt-composite.

**4.103 état:** valeur représentant l'état courant d'un objet dans le moteur MHEG.

NOTE – Les processus composant le moteur MHEG sortent du cadre de la présente Recommandation. Celle-ci nécessite toutefois un mécanisme permettant d'évaluer l'état des différents objets à l'intérieur du moteur MHEG.

**4.104 support de stockage:** support utilisé pour stocker les informations (mémoire électronique, disquette, disque dur, disque optique, bande magnétique, ...).

**4.105 structure d'un objet; structure d'objet MHEG:** description de la façon dont les informations sont organisées dans un objet.

**4.106 synchronisation:** présentation de rt-composants dans le temps et dans l'espace conformément aux contraintes et à la relation définies entre ces objets. Les contraintes et la relation peuvent être définies explicitement dans des objets *liens*, *composites*, *scripts* ou implicitement par la nature de l'application.

**4.107 position temporelle:** position spécifiée sur l'axe temporel d'un rt-composant ou d'un canal.

**4.108 marqueur temporel:** marqueur identifié pour une position temporelle. Ce marqueur peut être utilisé pour déclencher des liens.

**4.109 durée de transition:** paramètre optionnel qui peut être fourni pour certaines actions élémentaires, par exemple *Set Current Volume [fixation du volume courant]* (ensemble de cibles, volume sonore, durée de transition). Une durée de transition est exprimée en GTU. Lorsqu'une durée de transition est spécifiée, l'action élémentaire correspondante doit être exécutée pendant la durée spécifiée; par exemple, le volume sonore variera progressivement de sa valeur précédente à la valeur indiquée pendant la durée de transition spécifiée.

**4.110 support de transmission:** support servant à transmettre les informations (paires torsadées, câbles coaxiaux, fibres optiques, liaisons radioélectriques, ...).

**4.111 condition de déclenchement:** partie d'une condition de lien décrivant la modification d'une valeur d'attribut ou d'une valeur d'état. Si la modification décrite se produit, on dit que la condition de déclenchement est satisfaite.

**4.112 effet utilisateur:** effet entraîné par l'exécution d'une action élémentaire et qui est perçu par l'utilisateur.

**4.113 application utilisatrice:** application utilisant des objets MHEG, y compris des Recommandations UIT-T ou des Normes ISO.

## 5 Symboles et abréviations

La présente Recommandation utilise les symboles et abréviations suivants.

AP	( <i>attachment point</i> ) point de rattachement
ASN.1	notation et règles de codage définies dans les Recommandations X.680 et X.681
AVR	( <i>audible volume range</i> ) plage du volume sonore
BER	( <i>basic encoding rules</i> ) règles de codage de base de l'ASN.1 définies dans la Rec. UIT-T X.690   ISO/CEI 8825-1
CC	( <i>current condition</i> ) condition courante dans une condition de lien
CGSU	( <i>channel original generic spatial unit</i> ) unité spatiale générique d'origine d'un canal
CPS	( <i>channel presentation space</i> ) espace de présentation d'un canal
CRPS	( <i>channel relative presentation space</i> ) espace de présentation relatif d'un canal
CTP	( <i>current temporal position</i> ) position temporelle courante

CV	( <i>current audible volume</i> ) volume sonore courant
EA	( <i>elementary action</i> ) action élémentaire
GF	( <i>generic factor</i> ) facteur générique
GSF	( <i>generic spatial factor</i> ) facteur spatial générique
GSU	( <i>generic spatial unit</i> ) unité spatiale générique
GTF	( <i>generic temporal factor</i> ) facteur temporel générique
GTU	( <i>generic temporal unit</i> ) unité temporelle générique
GU	( <i>generic unit</i> ) unité générique
GUI	( <i>graphical user interface</i> ) interface graphique d'utilisateur
GVF	( <i>generic audible volume factor</i> ) facteur de volume sonore générique
JBIG	norme de codage des images en deux tons définie dans la Recommandation T.82: <i>Technologies de l'information – Représentation codée des images et du son – Compression progressive des images en deux tons</i>
JPEG	norme de codage des images fixes définie dans les Recommandations T.81, T.83, T.84: <i>Technologies de l'information – Compression numérique et codage des images fixes de nature photographique</i>
LC	( <i>link condition</i> ) condition de lien dans un objet <i>lien</i>
LE	( <i>link effect</i> ) effet de lien dans un objet <i>lien</i>
MH	multimédia hypermédia
MHEG	( <i>multimedia hypermedia experts group</i> ) adjectif désignant un schéma de codage d'informations multimédias et hypermédias défini par le GT12/SC29/JTC1 de l'ISO/CEI – groupe d'experts en multimédia hypermédia  NOTE – Si l'abréviation MHEG n'accompagne pas de substantif, elle désigne le groupe de travail au sein du JTC1. Sinon, elle doit être utilisée en tant qu'adjectif avec un substantif pour lever toute ambiguïté en ce qui concerne le sens (objets MHEG, actions MHEG, classes MHEG par exemple).
MPEG	normes de codage audiovisuel définies dans la Recommandation H.262: <i>Technologies de l'information – Codage générique des images cinématographiques et des informations audio associées – Données vidéo</i> (pour des supports de stockage numériques d'une capacité à des débits jusqu'à environ 1,5 Mbit/s) et Recommandation H.222: <i>Technologies de l'information – Codage générique des images cinématographiques et informations audio associés: systèmes</i>
mux	multiplexé
OAP	( <i>original visible size attachment point</i> ) point de rattachement de taille de visualisation d'origine
OD	( <i>original duration</i> ) durée d'origine
OGSU	( <i>original generic spatial unit</i> ) unité spatiale générique d'origine
OGTU	( <i>original generic temporal unit</i> ) unité temporelle générique d'origine
OGU	( <i>original generic unit</i> ) unité générique d'origine
OGVU	( <i>original generic audible volume unit</i> ) unité de volume sonore générique d'origine
OPS	( <i>original presentation space</i> ) espace de présentation d'origine
OS	( <i>original size</i> ) taille d'origine
OV	( <i>original audible volume</i> ) volume sonore d'origine
OVD	( <i>original visible duration</i> ) durée de visualisation d'origine
OVS	( <i>original visible size</i> ) taille de visualisation d'origine
PAP	( <i>projected original visible size attachment point</i> ) point de rattachement de taille de visualisation d'origine projetée
PC	( <i>previous condition</i> ) condition précédente dans une condition de lien
PCV	( <i>projected current audible volume</i> ) volume sonore courant projeté
POD	( <i>projected original duration</i> ) durée d'origine projetée
POS	( <i>projected original size</i> ) taille d'origine projetée
PRPS	( <i>parent relative presentation space</i> ) espace de présentation relatif du parent

PS	( <i>presentation space</i> ) espace de présentation
PTU	( <i>physical temporal unit</i> ) unité temporelle physique
PU	( <i>physical unit</i> ) unité physique
PVD	( <i>projected visible duration</i> ) durée de visualisation projetée
PVS	( <i>projected visible size</i> ) taille de visualisation projetée
PVU	( <i>physical audible volume unit</i> ) unité de volume sonore physique
RGSU	( <i>relative generic spatial unit</i> ) unité spatiale générique relative
RGTU	( <i>relative generic temporal unit</i> ) unité temporelle générique relative
RGU	( <i>relative generic unit</i> ) unité générique relative
RGVU	( <i>relative generic audible volume unit</i> ) unité de volume sonore générique relative
RPS	( <i>relative presentation space</i> ) espace de présentation relatif
rt	( <i>run-time</i> ) exécution
TC	( <i>trigger condition</i> ) condition de déclenchement dans une condition de lien
TD	( <i>transition duration</i> ) durée de transition
VD	( <i>visible duration</i> ) durée visible

## SECTION 1 – APERÇU GÉNÉRAL

### 6 Principales caractéristiques T.171

#### 6.1 Objets multimédias de transfert

Un objet multimédia n'a que peu d'utilité en soi. Il trouve son utilité dans une application multimédia. Une telle application nécessite l'emploi de certaines méthodes, grâce auxquelles il est possible de décrire les objets multimédias et de les représenter d'une manière indépendante de l'application pour pouvoir les transférer entre applications. Dans ce contexte, la présente Recommandation offre des moyens de transfert d'objets multimédias sous forme d'objets MHEG.

#### 6.2 Approche orientée objet

La présente Recommandation se sert de certaines techniques orientées objets comme les classes d'objets et l'héritage. Ce sous-paragraphe décrit l'usage spécifique des concepts orientés objet dans la présente Recommandation.

##### 6.2.1 Classes d'objets

La présente Recommandation n'utilise le terme «classe d'objet» que pour désigner les structures d'objets hyper et multimédias transférables. Ainsi, les classes MHEG spécifient la structure des objets MHEG plutôt que leurs interfaces. En particulier, le groupe MHEG ne spécifie pas les méthodes comme des parties de classes (ou respectivement leurs interfaces).

Cette approche vise à considérer les objets MHEG comme des entités d'information à interpréter par un processus approprié, c'est-à-dire le moteur MHEG, afin d'en réaliser leur comportement.

##### 6.2.2 Sous-classes et héritage

La présente Recommandation se sert du concept de dérivation de nouvelles classes à partir de classes existantes. Les sous-classes héritent de toute la structure de leur hyperclasse (c'est-à-dire des classes dont elles sont les sous-classes). De nouveaux attributs sont ajoutés afin de former de nouveaux comportements, et des attributs hérités sont remplacés afin d'adapter les comportements existants.

### 6.2.3 Polymorphisme

Le moteur MHEG peut exécuter certaines actions sur les objets MHEG, comme les préparer à d'autres utilisations ou positionner certaines de leurs valeurs d'attribut. Le polymorphisme s'applique à ces actions. Il signifie que si l'application d'une action est valide pour les objets d'une certaine classe, elle l'est aussi pour les objets de toutes ses sous-classes, mais que l'effet peut être différent pour différentes classes. Le polymorphisme est un moyen d'indiquer que le comportement du parent est générique (commun) et que celui des enfants peut être spécifique.

Par exemple, la réalisation d'un objet *action* Run (exécution) applicable à un objet *contenu d'exécution* représentant un texte sera différent de la réalisation d'un tel objet applicable à un objet *contenu multiplexé d'exécution* représentant un train de données MPEG. Le comportement commun générique est la présentation d'une instance (*run*) et la partie applicable à la vidéo est de présenter un document vidéo.

### 6.2.4 Objets et cycles de vie d'objets

On distingue trois formes d'objets MHEG à l'intérieur de la présente Recommandation comme le montre la Figure 2:

- 1) information hyper ou multimédia transférable codée conformément à l'Annexe A;
- 2) représentation à l'intérieur du moteur MHEG, à des fins internes (mémoire de l'objet MHEG dans sa représentation interne d'origine, calcul des conditions de lien, par exemple);
- 3) copies des représentations mentionnées au point 2) utilisables pour la présentation par l'utilisateur; ces copies sont désignées comme des rt-objets.

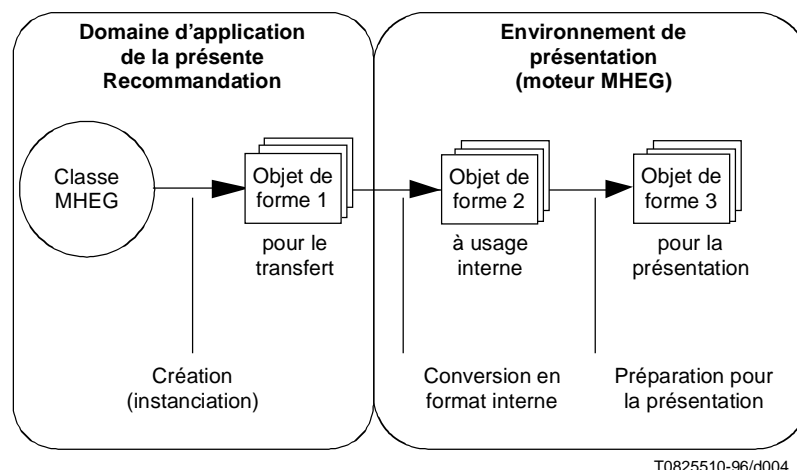


Figure 2/T.171 – Les différentes formes d'objets MHEG

A la base, le cycle de vie des objets de forme 1 dépasse le cadre de la présente Recommandation. Une fois créés par des mécanismes de codage sous forme d'instances de classes MHEG concrètes, ces objets restent en vie (par exemple dans une réserve) jusqu'à ce qu'ils soient détruits par des moyens qui ne relèvent pas de la présente Recommandation. Toutefois, si un objet MHEG de forme 1, qui peut être une copie d'un objet d'origine, est reçu par un moteur MHEG, il est converti en objet de forme 2. Cette forme 2 existe jusqu'à ce qu'il lui soit appliqué l'action *Destroy* (destruction). La forme 3 apparaît chaque fois qu'une action *New* (nouveau) est appliquée à un objet de forme 2 approprié, (c'est-à-dire un objet *modèle*), ce qui aboutit à la copie de cet objet mais en un objet de forme 3 qui peut être présenté et avoir des valeurs d'attributs différentes. Les objets de forme 3 sont supprimés au moyen d'une action *Delete* (suppression).

Il est à noter que les objets de formes 2 et 3 existent à l'intérieur d'un environnement de présentation, c'est-à-dire le moteur MHEG. Cela suppose qu'ils disparaissent avec cet environnement.

### 6.3 Caractéristiques techniques

Ce sous-paragraphe décrit certaines caractéristiques techniques découlant de la présente Recommandation.

En raison de la nature générique de cette Recommandation, rien n'empêche un utilisateur de combiner les fonctionnalités suivantes. Une application utilisatrice peut toutefois imposer des règles de mise en œuvre d'une architecture plus appropriée.

#### 6.3.1 Composition d'objets pour le transfert et la présentation

La présente Recommandation utilise et prévoit deux techniques de composition d'objets. L'une d'elles concerne le concept de *conteneur* qui s'appuie sur la classe *conteneur* qui sert comme outil d'emballage de l'information hyper et multimédia à véhiculer. L'autre traite de la composition de la présentation fournie par la classe *composite* qui sert comme outil de présentation d'information sous le contrôle d'un script hyper et multimédia.

La classe *conteneur* permet la combinaison d'un ensemble d'objets hyper et multimédia à transférer entre les systèmes de mémorisation et de présentation. Cette classe regroupe un ensemble d'objets à transférer comme un tout en prenant en compte les besoins des systèmes à ressources minimales. En utilisant ce mécanisme, un transfert efficace de données multimédias entre des systèmes hétérogènes peut être réalisé selon leurs besoins, par exemple: lorsque des objets de la classe *conteneur* doivent être utilisés dans une présentation, un certain nombre d'actions de présentation sont nécessaires pour qu'ils puissent participer à cette présentation.

La Figure 3 illustre l'utilisation de la présente Recommandation en tant qu'unité de transfert et en tant qu'identificateur d'information multimédia codée conformément à d'autres normes internationales.

La classe *composite* permet la combinaison d'un ensemble d'objets hyper et multimédia à présenter. Elle offre une fonctionnalité de présentation commune pour les objets qu'elle contient et décrit les différentes présentations allant de celle de l'objet monomédia le plus simple aux interactions hyper et multimédias les plus complexes.

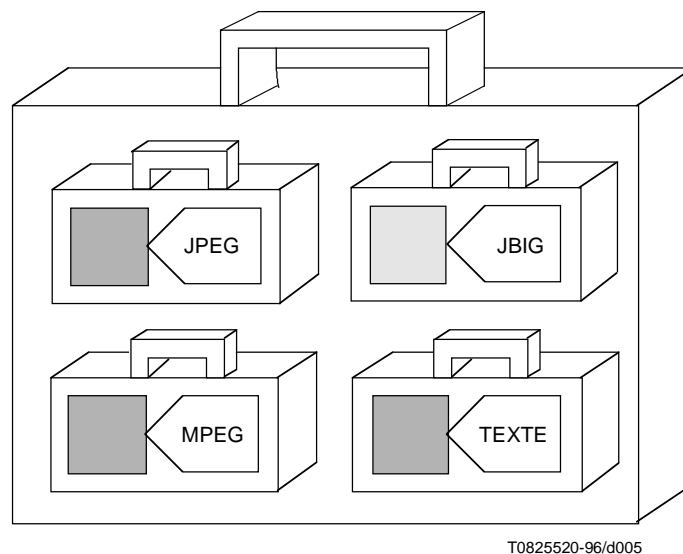


Figure 3/T.171 – Transfert et identification des informations multimédias

### 6.3.2 Objets d'exécution

Pour pouvoir réutiliser les objets dans différents contextes, une distinction nette doit être faite entre un objet de transfert, l'**objet modèle**, et les objets d'exécution, les **rt-objets**. L'objet *modèle* correspond à une information d'origine réutilisable. C'est une instance de classe d'objets contenant des données de script, des informations mono, multi et/ou hypermédia. Un rt-objet représente une utilisation spécifique de l'objet *modèle*. Un tel objet sert à la présentation et il est possible d'en créer un nombre quelconque à partir d'un même objet *modèle*.

La présente Recommandation ne définit pas la représentation interne d'un rt-objet. Chaque moteur MHEG peut disposer de sa propre technique de représentation interne. L'activation et la modification des attributs d'un rt-objet n'affectent pas l'objet *modèle* d'origine. Cela permet de réutiliser le même objet *modèle* d'origine dans différents contextes.

NOTE – Par exemple, afin de disposer d'un jeu de hockey, il est nécessaire d'avoir deux objets *contenus*: B (équipe bleue) et R (équipe rouge). L'objet *contenu* B contient les positions d'origine des joueurs de l'équipe bleue et l'objet *contenu* R contient celles des joueurs de l'équipe rouge. Il est possible dans ces conditions de dériver plusieurs rt-contenus à partir des objets *contenus modèles*:

- création à partir du *contenu* B des rt-contenus 1 à 10 correspondant aux 10 joueurs de l'équipe bleue;
- création à partir du *contenu* R des rt-contenus 11 à 20 correspondant aux 10 joueurs de l'équipe rouge.

Toutes les informations de présentation sont décrites pour chaque rt-contenu. Les objets 1 à 6 (équipe bleue) et 11 à 16 (équipe rouge) ont différentes positions sur le terrain. Les objets 7 à 10 ont des positions sur le banc de l'équipe bleue et les objets 17 à 20 sur le banc de l'équipe rouge.

La taille présentée de chaque rt-contenu peut être différente. Les attributs de présentation peuvent être évalués pour chaque rt-contenu afin de savoir à un moment précis si un joueur (c'est-à-dire un rt-contenu) est à l'hôpital (ne joue pas) ou sur la glace (joue).

### 6.3.3 Comportement individuel commun à tous les rt-objets créés à partir d'un objet modèle donné

La présente Recommandation offre un mécanisme de description de comportements individuels communs à tous les rt-objets, qui peuvent être créés à partir d'un objet *modèle* donné. Les avantages de cette fonctionnalité sont:

- le comportement n'est décrit qu'une fois et il s'applique à tous les rt-objets;
- le concepteur d'objets est capable de décrire un comportement individuel commun sans connaître le nombre de rt-objets à créer à partir de l'objet *modèle* donné;
- une fois préparé le lien s'applique aussi bien à tous les rt-objets créés ou à créer.

Cette fonctionnalité est assurée par les techniques de postambule «?» (voir 11.3).

### 6.3.4 Synchronisation

Les mécanismes de synchronisation des objets multimédias sont fondamentalement répartis en quatre classes:

- synchronisation élémentaire: les deux objets sont synchronisés:
  - par rapport à la même référence temporelle d'origine (mode parallèle); ou
  - l'un par rapport à l'autre (mode séquentiel);
- synchronisation à la chaîne: tous les objets d'un ensemble sont présentés à la chaîne, l'un derrière l'autre;
- synchronisation cyclique: un ou plusieurs objets sont présentés de façon répétitive;
- synchronisation conditionnelle: la présentation d'un objet est liée à la satisfaction d'une condition.

Dans la présente Recommandation, ces quatre classes de synchronisation sont décrites d'une même manière, la synchronisation conditionnelle.

Les niveaux de synchronisation d'objets multimédias sont classés en quatre niveaux:

- 1) synchronisation de script: ce niveau peut contenir une synchronisation complexe qui prend par exemple en compte les réponses de l'utilisateur, les valeurs calculées et les états des ressources du système;
  - la présente Recommandation ne définit pas ce niveau;
- 2) synchronisation conditionnelle: l'état courant de l'objet présenté peut déclencher une action réflexe sur un autre objet, par exemple: «lorsque le son s'arrête, poser une question»;

- 3) synchronisation spatio-temporelle: la position spatiale et temporelle d'un objet peut être définie relativement par rapport à un autre, par exemple: «afficher le nom du produit 2 cm au-dessus de l'image et 2 secondes après la présentation de celle-ci»;
- 4) synchronisation intermédia: synchronisation étroite à l'intérieur d'un objet *contenu multiplexé*, par exemple: «synchronisation de la parole» au cinéma;

la synchronisation intermédia est traitée par d'autres normes comme la norme MPEG.

### 6.3.5 Liens

Les liens MHEG sont associatifs, dynamiques et régis par les événements. Il en existe de nombreux types qu'on peut regrouper en deux catégories:

- 1) lien de synchronisation d'objets: décrit le mécanisme de synchronisation entre objets multimédias;
- 2) hyperlien: décrit les relations d'objets au sens du contexte. Tous les objets hypermédias en correspondance sont liés par ce mécanisme.

La présente Recommandation assure les deux types de liens par un même mécanisme de liaison d'objets à l'intérieur d'un objet *lien*.

L'ensemble des liens MHEG est hautement dynamique. Des liens peuvent être ajoutés ou supprimés à tout moment et peuvent être conditionnés par l'activité de l'utilisateur et les actions précédemment exécutées afin de pouvoir en adapter le comportement pendant toute la durée de leur présentation.

Les liens MHEG sont régis par un changement de comportement de l'entité MHEG, par exemple: passage d'un objet à l'état prêt ou augmentation du volume d'un rt-contenu. Chaque lien décrit un changement qui peut le déclencher ainsi que l'effet qu'il aura.

Les liens MHEG sont entièrement déterminés et ne nécessitent pas d'autres traitement que leur exécution directe.

### 6.3.6 Entrée

Afin de prendre en charge les systèmes hyper et multimédias interactifs, il est essentiel de fournir certains mécanismes d'entrée d'utilisateur. Dans la présente Recommandation, la liste des interactions possibles, qui sont offertes à l'utilisateur, est décrite au moyen d'objets *composites*. L'entrée est considérée comme un comportement et dépend des comportements de sélection et de modification des rt-composants.

La présente Recommandation offre des fonctionnalités pour décrire les résultats de l'interaction de l'utilisateur mais ne traite pas des fonctionnalités d'interaction. De telles interactions peuvent être réalisées de différentes façons, par exemple: interfaces GUI (interface graphique d'utilisateur) ou claviers. La présente Recommandation ne définit pas le rendu et la convivialité des présentations multimédias interactives; elle ne propose pas non plus de modifier des concepts existants ou d'en ajouter de nouveaux par rapport à ceux des interfaces GUI types. Comme la présente Recommandation est générique et indépendante de la plate-forme et de la réalisation, elle décrit l'interaction à un niveau virtuel.

### 6.3.7 Traitement d'événements

Les objets MHEG sont capables d'interagir avec des événements grâce aux mécanismes de traitement d'événements. Un événement est généré par le système extérieur ou le moteur MHEG lui-même.

Lorsqu'un événement est généré, le moteur MHEG positionne l'identificateur de l'événement catalogué correspondant sur l'attribut de l'événement et, si nécessaire, l'information associée sur l'attribut *données de l'événement* du rt-objet ou du canal qui doit accueillir cet événement. L'événement «clicquage de souris sur un bouton» généré par l'interface GUI peut être envoyé à un rt-objet comme un événement numéro 10 associé à une position en x et y.

Le mécanisme de traitement d'événements peut servir à des fins de synchronisation générale et au mécanisme d'interaction.

### 6.3.8 Ancre

Afin de prendre en charge les systèmes hypertexte et hypermédia, la présente Recommandation fournit un mécanisme d'ancrage. Une ancre est un objet *contenu* qui conserve une information d'ancrage plutôt que des données monomédias. Cette information définit une zone à l'intérieur des autres données de l'objet *contenu*. Cette région doit être présentée sur les autres données par l'interface GUI avec l'interprétation spécifiée dans les objets ou à l'aide d'un style spécifié. Comme l'ancre est transférée à l'intérieur d'un objet *contenu*, le rt-contenu peut être créé à partir de l'objet *contenu* et tous les comportements des rt-contenus s'appliquent, par exemple: exécution, sélection.

### 6.3.9 Temps réel

La présente Recommandation n'a pas de contrôle et ne suppose rien sur la qualité ou la capacité des réseaux sous-jacents à travers lesquels les applications utilisatrices peuvent transférer des objets MHEG. Il est possible qu'il existe déjà certains réseaux locaux (LAN, *local area network*) qui peuvent respecter toutes les prescriptions en matière de capacité de transfert d'application en temps réel; de futures normes pourront aussi donner des spécifications sur des réseaux régionaux (WAN, *wide area network*) avec des capacités appropriées. La présente Recommandation traite ces incertitudes de deux façons:

- la présente Recommandation autorise l'auteur à définir un ensemble de spécifications qui devront être satisfaites pour assurer l'existence d'applications en temps réel indépendamment des capacités du support de transfert. De plus, la spécification des objets MHEG fournit des mécanismes qui facilitent le transfert en temps réel indépendamment des capacités des réseaux sous-jacents;
- un objet *descripteur* MHEG fournit les informations suivantes:
  - description des objets à prendre en charge par le système;
  - description du codage monomédia;
  - document lisible par le système et mécanisme d'auto-assistance «lisez-moi» grâce auxquels les applications en communication peuvent négocier une session de transfert optimale. L'objet *descripteur* fournit à cet effet un pointeur vers les références aux autres normes nécessaires à l'allocation des ressources du réseau parmi les applications complémentaires ou en concurrence;
  - information de décalage des objets.

Les prescriptions en matière de capacité en temps réel sont satisfaites de la manière suivante:

- l'accès progressif aux objets et l'optimisation de leur positionnement est de la responsabilité de l'application utilisatrice. L'information de décalage des objets peut être utilisée pour effectuer une récupération rapide de certains objets codés se trouvant dans d'autres objets;
- la récupération partielle d'objets, le séquençement des objets et la récupération de la description d'un objet et de son contenu indépendamment l'un de l'autre sont des caractéristiques concernant le traitement et l'utilisation des actions *Prepare* (préparation) et *Destroy* (destruction);
- les prescriptions en matière de ressources sont prises en charge par l'objet *descripteur* MHEG;
- l'entrelacement d'objets est pris en charge par l'objet *contenu multiplexé*; la présente Recommandation n'offre pas de fonctionnalités d'entrelacement mais prend en charge l'indication d'objets entrelacés;
- la récupération groupée est prise en charge par l'objet *conteneur* MHEG.

### 6.3.10 Gestion d'objet

#### 6.3.10.1 Identification d'objet

L'identification générale des objets est nécessaire aux droits de reproduction, à l'identification du propriétaire et l'identification unique. La présente Recommandation fournit un mécanisme de description de cette information dans un objet MHEG. Au sein de ce mécanisme, le mécanisme d'identification unique est une clé du traitement des objets MHEG dans le moteur MHEG.

Un identificateur interne et un identificateur externe sont prévus afin de prendre en charge l'identification unique. Ces identificateurs servent d'objets de référence. L'identificateur interne est un identificateur codé à l'intérieur d'un objet MHEG et comporte un identificateur ou un index. L'identificateur externe agit comme un connecteur entre le domaine de l'objet MHEG et les entités situées à l'extérieur de ce domaine et comporte un identificateur public et un identificateur de système.

La présente Recommandation prévoit aussi une identification symbolique à l'aide d'un alias. Cette identification peut remplacer l'une quelconque des autres.

#### 6.3.10.2 Référence à des objets

La référence à des objets locaux ou distants en tant qu'alternative à l'inclusion physique est nécessaire. La présente Recommandation fournit un mécanisme de référence générique à toute entité qui y est définie.



### 6.3.10.3 Contenu d'objet

Une interface unifiée pour les données de contenu est nécessaire. La présente Recommandation fournit ce mécanisme dans la description de la classe *contenu*.

### 6.3.10.4 Objets uniformes

Il est nécessaire que les applications utilisatrices aient une vue uniforme des objets. La présente Recommandation fournit cette vue uniforme au moyen d'une arborescence d'objets MHEG.

### 6.3.11 Ressources minimales

La présente Recommandation traite de l'extraction des données des composantes au moment voulu pour les systèmes à ressources minimales. Ce problème comporte plusieurs aspects:

- 1) **entrelacement des objets:** cet aspect dépasse le cadre de la présente Recommandation parce que c'est une question qui relève du système. Le groupe MHEG ne fournit pas de fonctionnalités d'entrelacement mais prend en charge l'indication d'objets entrelacés à l'aide d'objets *contenu multiplexé*.
- 2) **séquencement:** cet aspect est défini dans la présente Recommandation au moyen de l'utilisation particulière de l'action `Prepare` (préparation).
- 3) **dégradation et négociation des ressources nécessaires:** les objets *descripteurs* fournissent des informations complémentaires sur un ensemble d'objets transférés, ces informations étant utiles pour les prescriptions en matière de dégradation de certaines présentation et pour la négociation des ressources minimales.

### 6.3.12 Dynamique de présentation et de structuration

Une fois qu'un *composite* d'exécution, le *rt-composite*, est créé à partir de l'objet *composite* d'origine, les *rt-composants* enfichés dans le réceptacle peuvent être retirés ou remplacés à l'aide de l'action `plug` (enficher). Cela permet une dynamique de présentation si les *rt-contenus* enfichés sont changés et une dynamique de structure du *rt-composite* si les *rt-composites* enfichés sont changés.

### 6.3.13 Macro-actions et macro-liens

Les macro-instructions offrent une technique générale de codage efficace des objets *actions* fréquemment utilisés dans lesquels seules quelques valeurs changent d'un lien à l'autre. De même, il est possible d'utiliser cette technique sur les conditions de liens fréquentes dans lesquelles seules quelques valeurs changent d'un lien à l'autre. Cela permet le partage et la réutilisation de comportements complexes. Un auteur peut créer un catalogue de squelettes prédéfinis d'objets *actions* et d'objets *liens*.

### 6.3.14 Affectation statique et dynamique de valeur générique

Une valeur générique peut être spécifiée comme une constante ou un résultat évalué à partir d'une action `get` (extraction). Les valeurs génériques sont des valeurs essentielles traitées dans un moteur MHEG. Elles peuvent être mémorisées et récupérées à partir des données d'objet *contenu*. Une valeur évaluée peut être mémorisée une fois évaluée, c'est-à-dire sous forme de constante; par exemple: données de contenu = «hello». Une valeur évaluée peut aussi être mémorisée sans traitement, dans ce cas l'action `get` est mémorisée, par exemple: données de contenu = `Get Preparation Status` (cible) – extraction de l'état de préparation –; par la suite, l'évaluation sera réalisée dynamiquement à chaque récupération de valeur générique.

## 7 Hypothèses relatives au moteur MHEG

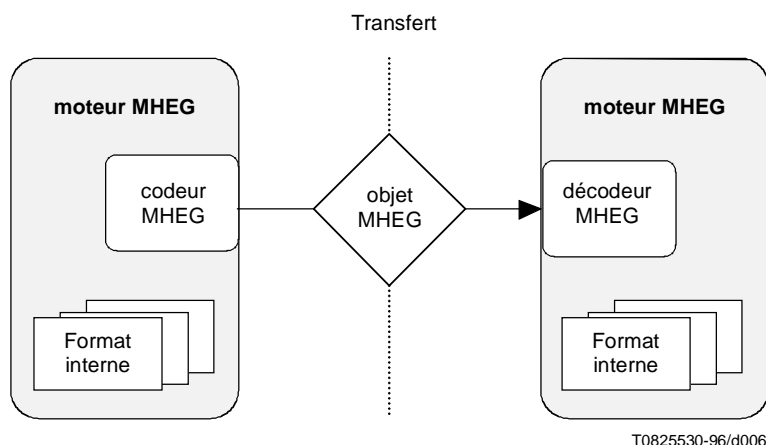
### 7.1 Traitement et transfert des objets

La présente Recommandation ne définit pas la manière dont le moteur MHEG traite les objets, les objets MHEG transférés ou les *rt-objets*. Mais elle fait certaines hypothèses quant aux caractéristiques des moteurs MHEG afin d'être capable de définir plus précisément les actions sur les objets.

- 1) La présente Recommandation suppose que l'objet MHEG, une fois transféré, est utilisé par une application utilisatrice et est traité par le moteur MHEG.

- 2) La présente Recommandation ne fait pas d'hypothèses sur la structure et l'organisation internes du moteur MHEG à l'exception des points suivants:
- il est censé interpréter correctement l'effet MHEG d'une action MHEG;
  - il est capable d'évaluer les valeurs d'états et d'attributs des entités MHEG qui se trouvent sous son contrôle. Les modifications apportées aux valeurs d'états ou d'attributs peuvent servir à exprimer les conditions de liens;
  - lorsqu'un moteur MHEG utilise une entrée de catalogue, il est censé l'interpréter correctement conformément à la description du catalogue;
  - le moteur MHEG est capable de travailler avec un système support d'interface utilisateur, s'il existe, et d'obtenir de ce système les résultats de l'interaction avec l'utilisateur. Ces résultats sont formalisés par le moteur MHEG afin de modifier les états de l'interaction correspondante. Le système support d'interface utilisateur indiquera par exemple au moteur qu'un élément particulier d'un objet donné a été sélectionné; le moteur MHEG sera alors capable de modifier les états d'interaction et de sélection correspondants. Ce changement peut déclencher un lien et exécuter une action conformément aux spécifications d'un objet MHEG donné.

Pour mieux comprendre le rôle joué par la présente Recommandation, ce paragraphe détaille le rôle des objets dans le transfert entre deux systèmes; ces rôles sont illustrés par la Figure 4.



**Figure 4/T.171 – Transfert d'objet MHEG**

L'objet MHEG n'est défini qu'au point de transfert entre les applications utilisatrices A et B. Lorsque le système A souhaite envoyer un objet MHEG au système B, il appelle un **codeur** MHEG, qui convertit le format interne utilisé par A dans le format défini par la présente Recommandation.

Lorsque B reçoit un objet MHEG, l'objet est décodé par un **décodeur** MHEG. Les valeurs à l'intérieur de l'objet sont passées par le décodeur, qui peut les convertir dans le format interne utilisé par B.

#### NOTES

1 – Les formats internes des applications utilisatrices A et B peuvent être les mêmes que le format MHEG, mais la présente Recommandation ne le prescrit pas.

2 – Si le transfert entre A et B est bidirectionnel, les systèmes peuvent tous deux disposer d'un codeur et d'un décodeur.

## 7.2 Interface d'application MHEG

Le moteur MHEG est censé fournir une interface à l'application utilisatrice. Cette interface offrira des fonctionnalités de commande du moteur MHEG; elle constitue le seul point d'accès de l'application aux objets MHEG à l'intérieur du moteur MHEG. La présente Recommandation ne définit ni le domaine d'application ni la forme de l'interface du moteur MHEG; elle ne définit pas non plus la structure de passage des données à travers l'interface.

### 7.3 Traitement des exceptions

La présente Recommandation décrit les réactions recommandées d'un moteur MHEG dans des situations anormales ou exceptionnelles susceptibles d'être rencontrées en cours d'exécution. Ces réactions ne sont considérées ni comme obligatoires ni comme limitatives, ce qui signifie qu'un moteur MHEG peut en faire plus (écrire dans un fichier de journalisation ou informer l'application, par exemple) ou même moins (arrêt du traitement, par exemple). Le comportement décrit est recommandé mais pas prescrit pour la mise en conformité.

NOTE – La section *comportement* de l'ISO 13522-1 indique les réactions spécifiques aux conditions d'erreurs pour chaque situation particulière.

## 8 Méthodologie

Ce paragraphe, qui décrit la méthodologie adoptée pour spécifier la présente Recommandation, est destiné à aider l'utilisateur à comprendre cette Recommandation. Cela n'implique pas qu'un système ou un produit conforme doive adopter cette méthodologie au niveau de l'analyse ou de la réalisation. C'est en particulier le cas pour le concept général d'orientation objet qui a été adopté dans la conception de la présente Recommandation.

Dans le contexte de la présente Recommandation, une classe d'objets désigne une structure d'information hyper et multimédia transférable, à partir de laquelle des objets MHEG peuvent être instanciés. La présente Recommandation restreint l'utilisation des concepts orientés objet à la spécialisation, l'héritage et le polymorphisme.

### 8.1 Modularité

Afin de permettre à une application utilisatrice de n'utiliser que certains des objets prévus par la présente Recommandation, la représentation elle-même est donnée sous forme d'un ensemble de modules organisés de la manière suivante:

- un module par classe d'objets transférés;
- un module contenant toutes les définitions utiles et les classes abstraites;
- un module contenant toutes les actions élémentaires.

#### NOTES

1 – Avec la structure ci-dessus, un processus d'application imposera l'en-tête minimal correspondant aux objets non utilisés.

2 – Dans toutes les futures extensions de la présente Recommandation, il est prévu que les représentations soient aussi isomorphes de la représentation de base en ce qui concerne la modularité.

### 8.2 Méthodologie de représentation des objets MHEG

La présente Recommandation donne une description d'un objet MHEG, une définition précise de sa structure et une représentation codée de base de l'objet. La représentation définie dans la présente Recommandation utilise l'ASN.1.

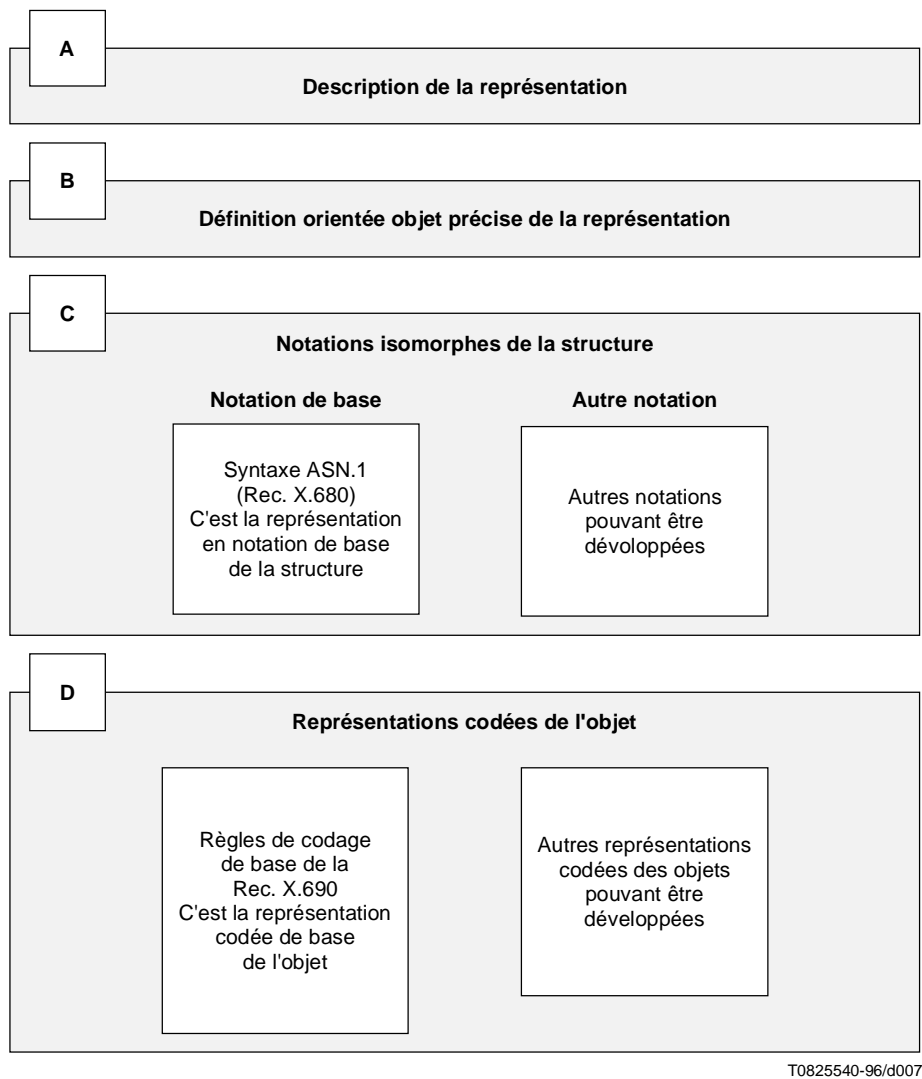
D'autres représentations de structure peuvent être fournies. Elles sont censées être isomorphes et équivalentes dans le cadre de la présente Recommandation (voir Figure 5).

NOTE – Il est prévu que la syntaxe de transfert de base de la présente Recommandation et les autres syntaxes définies dans les autres parties de la présente Recommandation soient compatibles entre elles, et qu'il soit possible de convertir une instance d'objet MHEG en une autre. Toutefois la présente Recommandation ne définit pas ce processus.

La représentation d'un objet MHEG est spécifiée sur quatre niveaux:

- description de la représentation;
- définition orientée objet (structure et sémantique);
- notation pour la structure de représentation;
- représentation codée définie par l'application des règles de codage à la représentation (voir Figure 5).

La description de la représentation fournie par la présente Recommandation comporte la structure de représentation et la sémantique suffisante pour permettre l'interprétation concrète de la signification d'un objet MHEG. La présente Recommandation ne fournit pas la sémantique des processus d'application conformes utilisant cet objet.



**Figure 5/T.171 – Niveaux de méthodologie utilisée par la présente Recommandation**

### 8.2.1 Niveau A – Description de la représentation

Le niveau A fournit la description de la représentation des définitions utiles (voir section 2), des objets MHEG (voir section 3) et des comportements d'entités MHEG (voir sections 4, 5, 6, 7 et 8).

### 8.2.2 Niveau B – Définition orientée objet précise

Pour chaque classe d'objets MHEG, la définition orientée objet précise est donnée par les éléments suivants:

- arborescence de classe: la classe dont elle hérite et la ou les classes qui héritent d'elle;
- utilisation: la ou les classes qu'elle utilise et la ou les classes qui l'utilisent;
- structure: les attributs qui lui sont propres en plus de ceux dont elle hérite;
- description non formelle: brève explication de la sémantique et du comportement.

La structure des classes est décrite au moyen d'une grammaire indépendante du contexte avec les conventions ci-dessous:

- **X ::= I, J, K** X est défini comme la séquence formée par l'attribut I suivi de J puis de K
- **X?** zéro ou une instance de X
- **X+** une ou plusieurs instances de X
- **X\*** nombre quelconque d'instances de X
- **(I | J | K)** un et un seul élément de l'ensemble (I, J, K)
- **(I, J, K)?** zéro ou une instance de la séquence (I, J, K) complète
- **(I, J, K)+** une ou plusieurs instances de la séquence (I, J, K) complète
- **(I, J, K)\*** zéro, une ou plusieurs instances de la séquence (I, J, K) complète

La structure et la sémantique de chaque attribut de représentation de l'objet (B en Figure 5) sont décrites à la suite à l'aide de la structure illustrée sur la Figure 6.

#### Nom d'attribut

- sémantique
- type OU
- séquence de sous-attributs OU
- choix entre attributs OU
- liste de sous-attributs

**Figure 6/T171 – Structure du sous-paragraphe attribut**

#### NOTE

- Chaque sous-attribut est décrit comme un attribut.
- Séquence (A, B): signifie que le sous-attribut A est suivi du sous-attribut B.
- Liste (A, B): signifie qu'une ou plusieurs instances de la séquence (A, B) peuvent exister.

### 8.2.3 Niveau C – Notations isomorphes pour la structure d'objets MHEG

Les normes fournissent un ensemble de notations équivalentes pour la description du niveau B de la structure d'un objet MHEG. Par équivalence, on entend l'isomorphisme des notations par rapport à la structure d'objets MHEG décrite au niveau B. L'une des notations est toutefois la notation de base, à laquelle on se rapporte en cas de questions sur la mise en conformité d'un produit ou d'un processus d'application.

La notation de base de la représentation de la structure des objets MHEG est en ASN.1. Il s'agit de la représentation de base (C en Figure 5).

NOTE – Dans les futures extensions de la présente Recommandation, les notations utilisées pour la représentation des objets MHEG seront isomorphes de celles utilisées pour la structure de ces objets décrite au niveau B.

### 8.2.3.1 Techniques ASN.1 utilisées au niveau C

Les notations ASN.1 types de la structure de chaque classe MHEG sont représentées sur la Figure 7.

```
-- MODULE DE CLASSES D'OBJETS MHEG

© Organisation Internationale de Normalisation, 1995. Il est permis de reproduire les
déclarations ci-dessous sous quelque forme que ce soit afin de les utiliser avec les
moteurs et les applications MHEG selon les spécifications de la Recommandation T.171,
sous réserve d'inclure cette notice dans toutes les copies.

ISOMHEG-cl {joint-iso-itu-t(2) mheg (19), version (1), class-identif (xx)},
DEFINITIONS:=BEGIN
EXPORTS MHEG-Object-Class;
IMPORTS MH-Object-Class FROM ISOMHEG-MH {joint-iso-itu-t(2) mheg (19), version (1),
MH-object (1000)};
MHEG-Object-Class:= SEQUENCE
{
class-identification    OBJECT IDENTIFIER    {joint-iso-itu-t(2),           -- racine
                                                mheg (19),                -- arc1
                                                version (1),              -- arc2
                                                class-identif(xx)        -- arc3
                                                },
COMPONENTS OF MH-object,
-- attributs de classes d'objets
}
END
```

Figure 7/T.171 – Exemple de module ASN.1 utilisé pour la représentation d'objets MHEG

Les techniques ASN.1 suivantes servent à représenter les objets MHEG:

- La technique des identificateurs d'objets ASN.1 est utilisée. Les identificateurs d'objets définis dans la présente Recommandation sont résumés dans l'Annexe A. La sémantique des identificateurs est définie par rapport à l'arborescence des identificateurs d'objets suivante.
- **racine**: origine de la norme. Cette racine a la valeur joint-iso-itu-t(2).
- **arc1**: identification de la présente Recommandation. Sa valeur est **19**. L'identification a été attribuée par l'autorité de dénomination conjointe.
- **arc2**: version MHEG normalisée. Sa valeur est **1** pour cette présente version de la Recommandation.
- **arc3**: identificateur de classe. C'est un nombre assigné à chaque classe MHEG.
- Les modules ASN.1 sont utilisés dans la présente Recommandation pour représenter les classes d'objets MHEG et leurs hyperclasses ainsi que pour décrire les définitions utiles. Chaque module a un identificateur du type identificateur d'objet ASN.1. Cet identificateur accompagne la définition du module et ne fait pas partie de la définition de la classe MHEG.

Ce module d'identificateur d'objet n'est pas codé à l'intérieur de l'objet MHEG. Il est utilisé par d'autres modules par les fonctionnalités d'importation et d'exportation pour identifier formellement un module.

- Les fonctionnalités ASN.1 d'importation ou d'exportation et COMPONENT OF (composante de) servent à représenter l'héritage des attributs des classes d'objets MHEG. Chaque hyperclasse définit une SEQUENCE d'attributs à exporter. Cette SEQUENCE est importée par les sous-classes et la déclaration COMPONENT OF sert à hériter de ces attributs.
- Les fonctionnalités ASN.1 d'importation et d'exportation servent aussi à représenter les définitions MHEG utiles et les actions élémentaires. Le module de définitions utiles exporte les définitions et les classes abstraites. Le module d'actions élémentaires exporte les actions élémentaires. Chaque module de classe d'objets MHEG importe les définitions correspondantes dont il a besoin.

- Chaque objet MHEG est représenté comme une séquence d'attributs utilisant les fonctionnalités de la SEQUENCE ASN.1. Chaque attribut a une structure arbitraire.
- La technique d'identificateur d'objets ASN.1 est aussi utilisée pour identifier de manière unique chaque classe d'objets MHEG. L'identificateur d'objets correspondant fait partie de la définition de la classe et sera codé dans chaque instance d'objet.

### 8.2.3.2 Droits de reproduction

Les définitions formelles ASN.1 font partie du texte de la présente Recommandation; elles sont protégées par des droits de reproduction. Afin de faciliter la mise en conformité aux normes MHEG, les définitions qui figurent dans le corps de la présente Recommandation et dans l'Annexe A peuvent être recopiées de la manière spécifiée dans la notice de droits de reproduction suivante:

- © Organisation internationale de normalisation, 1995. Il est permis de reproduire les définitions formelles ASN.1 sous quelque forme que ce soit afin de les utiliser avec les moteurs et les applications MHEG conformes selon les spécifications de la présente Recommandation, sous réserve d'inclure cette notice dans toutes les copies.

L'autorisation de reproduction ne s'applique pas aux autres documents de la présente Recommandation.

NOTE – La présente Recommandation définit un attribut servant à spécifier la notice de droits de reproduction qui accompagne un objet MHEG.

### 8.2.4 Niveau D – Représentation codée des objets MHEG

La présente Recommandation offre un ensemble de représentations codées équivalentes des objets MHEG. Elles sont équivalentes parce qu'elles sont isomorphes de celles utilisées au niveau B. Toutefois, l'une d'entre elles est la représentation codée de base.

Cette représentation est fournie par les règles de codage de base (BER) ASN.1.

NOTE – Dans toute future extension de la présente Recommandation la notation des représentations codées des objets MHEG sera isomorphe de celle utilisée au niveau B.

## SECTION 2 – MÉCANISMES DES FONCTIONNALITÉS GÉNÉRIQUES ET DES DÉFINITIONS UTILES

La présente Recommandation définit les mécanismes suivants:

- mécanisme de présentation (paragraphe 9);
- mécanisme d'identification générique (paragraphe 10);
- mécanisme de référence générique (paragraphe 11);
- valeur générique (paragraphe 12);
- mécanisme de macro-instruction (paragraphe 13);
- crochets (paragraphe 14);
- extensibilité (paragraphe 15).

Ces mécanismes et définitions génériques sont utilisés dans les paragraphes de la présente Recommandation.

Leur représentation codée est regroupée dans un module ASN.1 séparé, le «module des définitions utiles».

Lorsqu'un objet MHEG nécessite une définition particulière, sa représentation de classe ASN.1 importe la définition correspondante du module des définitions utiles.

## 9 Mécanisme de présentation

Ce paragraphe décrit le mécanisme général utilisé dans un espace de présentation (PS) (paragraphe 9.1) pour présenter les rt-composants à l'utilisateur. Ce mécanisme comporte un repère générique d'espace et de temps, ainsi qu'un repère sonore (AVR) pour reproduire le son.

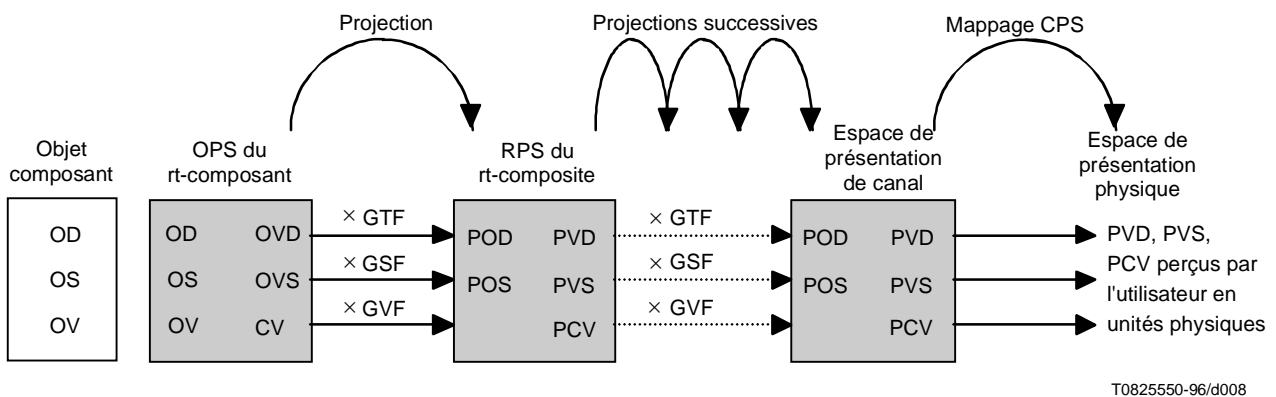
Les valeurs utilisées pour définir le comportement spatial, temporel et sonore des rt-composants sont exprimées en unités génériques (GU), par exemple la position, la taille, la durée et le volume. Ce mécanisme permet d'interpréter les unités par rapport à l'espace de présentation associé.

NOTE – Cela permet la cohabitation de données provenant d'outils de création différents avec des systèmes de coordonnées différents et un repère sonore différent.

A chaque rt-composant est associé un espace de présentation: l'espace de présentation d'origine (OPS) (paragraphe 9.2). De même, un espace de présentation, appelé espace de présentation de canal (CPS) est associé à chaque canal (paragraphe 9.3).

Afin de permettre le positionnement des enfants par rapport à leurs parents, chaque rt-composant peut être projeté dans un espace de présentation parent. Pour un enfant, son espace de présentation parent est appelé espace de présentation relatif (RPS) (paragraphe 9.4). Cet ascendant peut aussi être l'enfant d'un autre ascendant. Ainsi, il y a une chaîne de projections qui se termine par l'espace de présentation de canal: OPS à RPS ... à RPS à CPS. A chaque projection est appliqué un facteur d'échelle, appelé facteur générique (GF). Celui-ci est défini comme un rapport et spécifie le nombre d'unités génériques relatives (RGU) devant correspondre à une unité générique d'origine (OGU). Finalement afin d'être perçu par l'utilisateur, chaque espace de présentation de canal est projeté par le moteur MHEG sur un espace physique (paragraphe 11.1).

La Figure 8 résume les techniques de présentation.



**Figure 8/T.171 – Aperçu général des techniques de présentation**

## 9.1 Espace de présentation

Chaque espace de présentation est défini de la manière suivante:

- un repère temporel T;
- un repère spatial X, Y, Z;
- un repère sonore.

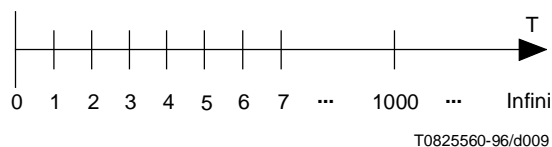
### 9.1.1 Repère temporel

Chaque espace de présentation possède un repère temporel T.

Ce repère est constitué d'un axe T qui est un intervalle  $[0, T\text{EndPoint}]$ , où le point TEndPoint sera toujours supérieur ou égal à zéro. Le point TEndPoint peut être égal à l'infini. Le point 0 est défini comme l'origine du repère.

Le nombre de points adressables dans le repère temporel est  $(T\text{EndPoint} + 1)$ .

L'unité générique utilisée le long de cet axe est appelée unité temporelle générique (GTU).



**Figure 9/T.171 – Repère temporel T**

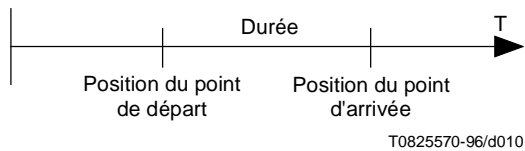


Le comportement temporel du rt-composant utilise des positions temporelles et des durées temporelles, exprimées en unités temporelles génériques dans l'espace de présentation:

- Position temporelle:
  - point de l'axe T;
  - appartenant à l'intervalle [0, TEndPoint].
- Durée:
  - distance entre deux positions temporelles;
  - définie comme la longueur d'un intervalle [StartPoint, EndPoint];
  - appartenant toujours à l'intervalle [0, TEndPoint];
  - EndPoint est plus grand ou égal à StartPoint;
  - le nombre de points adressables d'une durée est (EndPoint – StartPoint + 1).

La Figure 10 illustre les attributs de position temporelle et de durée.

Le facteur d'échelle temporel est appelé facteur temporel générique (GTF). Sa valeur n spécifie qu'une unité temporelle générique d'origine (OGTU) est projetée en n unités temporelles génériques relatives (RGTU).



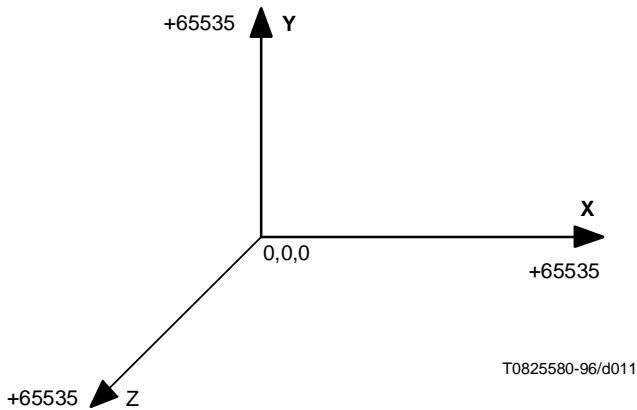
**Figure 10/T.171 – Position temporelle et durée**

**9.1.2 Repère spatial**

Chaque espace de présentation possède un repère spatial tridimensionnel orthogonal direct (X, Y, Z).

La Figure 11 indique le positionnement des axes par rapport à l'utilisateur: X va de gauche à droite, Y de bas en haut et Z du dispositif d'affichage vers l'utilisateur.

NOTE – Des extensions peuvent être fournies pour permettre un autre positionnement, par exemple, au moyen d'attributs supplémentaires de canal.



**Figure 11/T.171 – Les axes X, Y, Z de l'espace de présentation**

Toutes les instances d'un espace de présentation seront colinéaires, c'est à dire que tous les repères seront parallèles.

Chaque axe du repère spatial est un intervalle  $[0, S\text{EndPoint}]$  de longueur finie; le point  $S\text{EndPoint}$ , toujours supérieur ou égal à zéro a une valeur par défaut de +65535. Le point 0 est défini comme l'origine de l'axe. Le point  $(0,0,0)$  est appelé l'origine du repère spatial.

Le nombre de points adressables d'un axe est  $(S\text{EndPoint} + 1)$ .

L'unité GU utilisée pour chaque repère spatial est appelée unité spatiale générique (GSU).

Le comportement spatial du rt-composant utilise des positions spatiales et des tailles, exprimées en unité GSU dans l'espace de présentation:

- Une position spatiale est:
  - un triplet de coordonnées  $(x, y, z)$ , par rapport aux trois axes spatiaux;
  - définie chacune sur l'intervalle  $[0, S\text{EndPoint}]$ .
- Une taille est:
  - un triplet de distances  $(l_x, l_y, l_z)$  prises le long de chacun des axes du repère spatial;
  - chaque distance est définie comme une longueur d'intervalle  $[StartPoint, EndPoint]$ ;
  - appartenant toujours à l'intervalle  $[0, S\text{EndPoint}]$  de l'axe correspondant;
  - chaque distance est égale à  $(EndPoint - StartPoint)$ . Le point  $EndPoint$  est supérieur ou égal au point  $StartPoint$ ;
  - le nombre de points adressables de chaque distance est  $(EndPoint - StartPoint + 1)$ .

La Figure 12 illustre les attributs de position spatiale et de taille. Sur cette figure, l'axe Z est omis par soucis de clarté.

Un facteur d'échelle spatial, appelé facteur spatial générique (GSF), est défini pour chaque axe du repère spatial. Sa valeur  $n$  spécifie qu'une unité spatiale générique d'origine (OGSU) est projetée en  $n$  unités spatiales génériques relatives (RGSU).

Le facteur GSF peut différer d'un axe spatial à l'autre de l'espace de présentation.

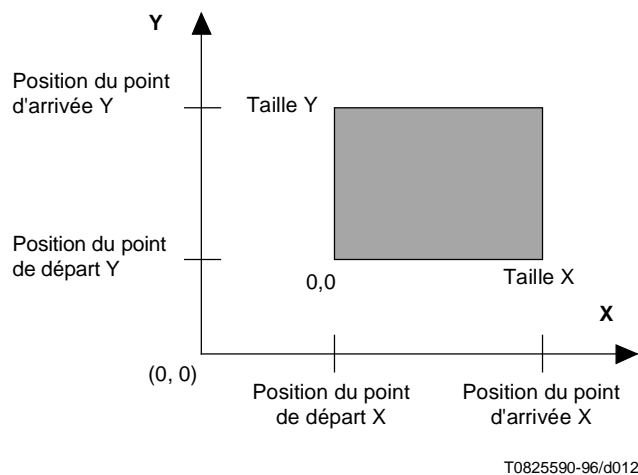


Figure 12/T.171 – Position spatiale et taille

### 9.1.3 Repère sonore

La présente Recommandation définit un repère sonore (AVR) où chaque son est décrit sur un axe appartenant à l'intervalle  $[0, +255]$ .

Le nombre de points adressables sur l'axe est 256.

L'unité utilisée le long de l'axe est l'unité de volume sonore générique (GVU).

Le comportement sonore du *rt-composant* utilise des volumes sonores, exprimés en unité GVU dans l'espace de présentation. Un volume sonore est toujours un point dans l'intervalle sonore défini par la présente Recommandation.

Le facteur d'échelle de volume sonore est appelé facteur de volume sonore générique (GVF). Une valeur *n* spécifie qu'une unité OGVU doit être projetée en *n* unités RGVU.

## 9.2 Espace de présentation d'origine

### 9.2.1 Initialisation de la taille d'origine et de la durée d'origine

Lorsqu'un objet *rt-composant* devient disponible, un espace de présentation, appelé l'espace de présentation d'origine (OPS), lui est associé. Les repères, spatial et temporel, sont initialisés de la manière suivante:

- a) l'extrémité TEndPoint est initialisée avec la valeur de durée d'origine (OD), obtenue de la manière suivante:
  - 1) si la durée OD est codée dans l'objet *composant*, cette valeur est utilisée;
  - 2) pour un objet *rt-contenu*, si la durée OD n'est pas codée dans l'objet *contenu*, la valeur de la durée OD peut être extraite de l'information contenue dans le crochet ou dans les données elles-mêmes;
  - 3) autrement, la valeur de la durée OD est positionnée à l'infini;
- b) chaque extrémité SEndPoint sera initialisée avec la valeur de la longueur de la taille d'origine (OS) sur l'axe correspondant, obtenue de la manière suivante:
  - 1) si la taille OS est codée dans l'objet *composant*:
    - i) si une longueur de la taille OS est fournie pour un axe spatial, cette valeur sera utilisée pour cet axe;
    - ii) autrement, la longueur de la taille OS est positionnée à 0;
  - 2) pour un objet *rt-contenu*, si la taille OS n'est pas codée dans l'objet *contenu*, la valeur de la taille OS peut être extraite de l'information contenue dans le crochet ou dans les données elles-mêmes;  
NOTE – Pour un objet *contenu multiplexé*, on utilise la taille OS des données globales multiplexées. La taille OS peut être codée individuellement pour chaque train binaire à l'intérieur d'un objet *contenu multiplexé*. Cette information ne doit pas être utilisée pour fixer la valeur de la taille OS de l'objet *rt-mux*, mais uniquement pour aider un moteur MHEG à manipuler et présenter un train binaire.
  - 3) sinon, la valeur de la longueur de la taille OS est fixée à 0 sur chaque axe.

Les unités génériques d'origine (OGU) suivantes, sont définies le long de chaque axe de l'espace de présentation d'origine (OPS):

- les unités GTU, le long de l'axe du repère temporel de l'espace OPS, sont appelées unités temporelles génériques d'origine (OGTU);
- les unités GSU, le long des axes du repère spatial de l'espace OPS, sont appelées unités spatiales génériques d'origine (OGSU);
- les unités GVU, le long du repère sonore de l'espace OPS, sont appelées unités GVU d'origine (OGVU).

### 9.2.2 Initialisation des facteurs génériques (GF)

Les facteurs d'échelle associés aux axes de l'espace OPS sont initialisés de la manière suivante:

- le facteur temporel générique (GTF) sera initialisé à 1. Sa valeur peut être changée au moyen de l'action `Set GTF` (positionnement du facteur GTF).

NOTE 1 – Par exemple, si le facteur GTF passe de 1 à 10, une unité RGTU durera 10 fois plus longtemps. Si le facteur GTF passe de 1 à 1/10, une unité RGTU durera 10 fois moins longtemps. Une valeur du facteur GTF de 0 a pour effet de réduire la durée du *rt-composant* à une durée infiniment courte dans l'espace OPS, parce que la durée complète du *rt-composant* est perçue dans 0 unité RGTU;

- le facteur GSF de chaque axe du repère spatial est initialisé à 1. Sa valeur peut être changée au moyen de l'action `Set GSF` (positionnement du facteur GSF).

NOTE 2 – Par exemple, si le facteur GSF passe de 1 à 2 sur un axe, une unité OGSU est projetée en deux unités RGVU sur cet axe. Le *rt-composant* aura alors une dimension double sur cet axe. Une valeur du facteur GSF de 0 a pour effet de réduire la perception du *rt-composant* à une taille infiniment petite dans l'espace OPS, parce que le *rt-composant*, en entier, sera perçu dans 0 unité RGSU.

- Le facteur GVF est initialisé à 1. Sa valeur peut être changée au moyen de l'action `Set GVF` (positionnement du facteur GVF);

NOTE 3 – Par exemple, si le facteur GVF passe de 1 à 2, une unité OGVU est projetée en deux unités RGVU. Le *rt-composant* est alors perçu deux fois plus fort. Une valeur GVF de 0 a pour effet de réduire la perception du *rt-composant* en un volume infiniment faible dans l'espace RPS, parce que le volume de l'objet *rt-composant* est perçu dans 0 unité RGVU.

### 9.2.3 Initialisation des attributs pour les *rt-components*

Lorsque l'espace OPS est créée, les attributs ci-dessous sont définis pour chaque *rt-composant*:

- OD: définit la durée d'origine du *rt-composant*. Cela est expliqué ci-dessus;
- OVD (durée visible d'origine): c'est un sous ensemble de la durée OD, définissant la partie de la durée OD qui sera perçue par l'utilisateur. Elle est initialisée à la valeur de la durée OD et peut être modifiée par l'action `Set OVD` (positionnement de l'attribut OVD);
- OS: définit la taille d'origine du *rt-composant*. Sa valeur est obtenue selon les indications ci-dessus;
- OVS (taille visible d'origine): est définie comme une taille et se positionne par rapport à OS. La portion de la taille OS qui tient dans la taille OVS sera perçue par l'utilisateur. La portion de la taille OS qui est à l'extérieur de la taille OVS est tronquée. Si la taille OVS dépasse les limites de la taille OS, la portion supplémentaire est remplie par le fond du *rt-composant*. La taille OVS sera initialisée à la valeur de la taille OS et sera positionnée à l'origine de la taille OS. La taille peut être modifiée au moyen de l'action `Set OVS`. La position peut être modifiée au moyen de l'action `Set OVS Position`.

NOTE – le fond d'un *rt-composant* n'est pas défini dans l'ISO/CEI 13522-1. Il peut être extrait du crochet ou des données de l'objet *composant*. L'auteur peut aussi utiliser une action `Set Style` (désignation de style) pour désigner un style de fond catalogué.

### 9.2.4 Initialisation des attributs pour les *rt-contenus*

Lorsque l'espace OPS est créée, les attributs suivants sont définis pour chaque *rt-contenu*:

- a) OV: définit le volume sonore d'origine de l'objet *rt-contenu*. Sa valeur est obtenue de la manière suivante:
  - 1) si le volume OV est codé dans l'objet *contenu*, cette valeur est utilisée;
  - 2) sinon, la valeur du volume OV peut être extraite de l'information contenue dans le crochet ou dans les données elles-mêmes;
  - 3) autrement, la valeur du volume OV est positionnée à la valeur minimale du repère sonore.
- b) CV (volume sonore courant): correspond au volume sonore qui sera perçu par l'utilisateur. Il est initialisé au volume sonore OV et peut être changé avec l'action `Set CV` (positionnement du volume CV).

NOTE – Lorsqu'un objet *rt-mux* est chargé de la présentation de plusieurs trains binaires, c'est au processus de démultiplexage et de l'interface GUI de conserver une cohérence entre la durée, la taille, le volume de l'objet *rt-mux* et les valeurs correspondantes de chaque train binaire sélectionné pour cet objet.

Pour assurer la présentation finale, la durée et le volume, codés dans chaque objet *rt-contenu*, seront exprimés respectivement en milliseconde (PTU) et décibel (PVU). Si la durée et le volume sont extraits du crochet, leurs valeurs sont traduites et exprimées respectivement en milliseconde (PTU) et décibel (PVU).

## 9.3 Espace de présentation d'un canal

L'espace de présentation d'un canal (CPS) est initialisé de la manière suivante:

- 1) L'extrémité TEndPoint est initialisé à la valeur de la durée CPS, obtenue de la manière suivante:
  - a) si l'action `Set CPS` (positionnement de l'attribut CPS) est applicable à ce canal et contient un paramètre de durée CPS, cette valeur est adoptée;
  - b) sinon, la valeur de durée CPS prend la valeur infini.
- 2) Chaque extrémité SEndPoint est initialisé à la valeur de la longueur de la taille CPS sur l'axe correspondant, obtenue de la manière suivante:
  - a) si l'action `Set CPS` (positionnement de l'attribut CPS) est appliqué à ce canal et contient un paramètre de taille CPS, les conditions suivantes s'appliquent:
    - i) si la longueur de la taille CPS est fournie pour un axe, sa valeur est utilisée pour cet axe;
    - ii) sinon, la longueur de la taille CPS sur cet axe prend la valeur 0;

- b) sinon, la valeur de la longueur de la taille CPS sur chaque axe est positionnée par défaut à la valeur `SEndPoint`, fournie dans la présente Recommandation.

Lorsque différents auteurs génèrent différents objets MHEG dans le cadre d'une même application, et que chaque auteur utilise le canal par défaut, il est recommandé, que les auteurs soient tous informés, de la modification éventuelle de la définition du CPS courant du canal par défaut.

- 3) La valeur dans le repère sonore est positionnée à la valeur fournie par la présente Recommandation.

Il n'y a pas de facteur d'échelle associé aux axes du repère CPS.

## 9.4 Espace de présentation relatif

L'espace de présentation relatif (RPS) d'un *rt-composant* est l'espace de présentation auquel le *rt-composant* est affecté et dans lequel il est projeté. A chaque instant, chaque *rt-composant* est affecté à un seul espace RPS. Les espaces RPS suivants sont définis:

- 1) PRPS (RPS parent): un réceptacle peut être projeté dans l'espace OPS de son *rt-composite* parent, afin de décrire une composition temporelle, spatiale et sonore. Ceci peut être répété de façon récurrente si le *rt-composite* parent est lui même un réceptacle *rt-composite*;
- 2) CRPS (espace de présentation relatif à un canal):
  - a) un réceptacle peut aussi être projeté directement dans un espace CPS lorsqu'il ne participe pas à la composition temporelle, spatiale et sonore décrite par son parent;
  - b) chaque *rt-composite* racine est projeté dans un espace CPS.

Initialement, chaque *rt-composant* racine est affecté au canal par défaut, et chaque réceptacle est affecté à l'espace de présentation du *rt-contenu* parent. Cette affectation à un espace RPS peut être modifiée au moyen de l'action `Set RPS` (affectation de l'attribut RPS).

Sur chaque axe de l'espace RPS, les unités génériques relatives (RGU) suivantes, sont définies:

- l'unité GTU sur l'axe temporel de l'espace RPS est appelée unité temporelle générique relative (RGTU);
- l'unités GSU sur chacun des axes spatiaux de l'espace RPS est appelée unité spatiale générique relative (RGSU);
- l'unité GVU sur le repère sonore de l'espace RPS est appelée unité de volume sonore générique relative (RGVU).

Les attributs définis pour chaque *rt-composant* dans l'espace OPS sont projetés de la manière suivante dans l'espace RPS auquel l'objet est affecté:

- POD (durée d'origine projetée): projection de la durée OD sur l'espace RPS avec:  $OD \times GTF = POD$ ;
- PVD (durée visible projetée): projection de la durée VD avec:  $OVD \times GTF = PVD$ ;
- POS (taille d'origine projetée): projection de la taille OS dans l'espace RPS avec:  $OS \times GSF = POS$ ;
- PVS (taille visible projetée): projection de la taille OVS avec:  $OVS \times GSF = PVS$ . La valeur de la taille PVS peut aussi être directement positionnée par l'utilisation de l'action `Set OVS` (positionnement de l'attribut OVS) et pourra recevoir la valeur 'calculated' (calculée) de l'attribut `OVS Proj Strategy` (projection stratégique de la taille visible d'origine).

Les attributs définis pour chaque *rt-contenu* dans l'espace OPS sont projetés de la manière suivante dans l'espace RPS auquel l'objet est affecté:

- PCV (volume sonore courant projeté): projection du volume courant dans l'espace RPS avec:  $CV \times GVF = PCV$ .

## 9.5 Mappage CPS

Chaque espace CPS est un espace virtuel. Il appartient au moteur MHEG d'affecter un espace de présentation physique à un espace virtuel et d'en convertir les coordonnées virtuelles GU en coordonnées physiques.

Le moteur MHEG affecte à chaque espace CPS un espace de présentation physique. L'objet descripteur peut fournir de l'information permettant de faciliter le mappage de telle manière que:

- un mappage sur un type de support prévu, peut être fourni pour faciliter l'affectation à un espace de présentation physique. Le moteur MHEG associe l'espace CPS à des dispositifs tels que téléviseurs, stations de travail, caméras, fenêtres, haut-parleurs et microphones;

- une résolution conseillée pour les axes de l'espace de présentation physique peut être fournie pour un rendu satisfaisant: résolution en x, y, z, et t;

Les auteurs doivent prendre en compte les possibilités limitées de l'équipement à ressource minimale, et ne pas exiger inutilement de fines résolutions;

- Il est possible de définir une plage audio dynamique avec une fréquence minimale et une fréquence maximale recommandées, pour présenter des données sonores avec une amplitude appropriée.

Les coordonnées virtuelles sont converties en coordonnées physiques de la manière suivante:

- le repère temporel CPS est projeté sur un repère temporel physique. Les unités temporelles physiques sont exprimées en milliseconde. Une unité CGTU est projetée en une unité PTU;
- le repère spatial CPS est projeté sur un repère spatial physique. Les unités spatiales physiques dépendent de chaque espace de présentation physique, par exemple, le pixel ou le millimètre. Une unité CGSU est projetée en une unité PSU;

il appartient au moteur MHEG de s'assurer de la direction de projection de chaque axe du repère spatial (Figure 11) dans l'espace physique de manière à ce qu'il soit correctement perçu par l'utilisateur. Ainsi, si on prend par exemple, un axe Y de l'espace CPS défini sur un intervalle [0, 65535] dirigé vers le haut, il peut être requis de projeter cet axe Y sur l'intervalle [0, 479] dirigé vers le bas à l'écran;

- l'espace CPS du repère sonore est projeté sur un repère sonore physique. Les unités de volume sonore physique sont exprimées en décibels [dB] en respectant les limites du repère sonore physique. Une unité CGVU est projetée en une unité PVU.

## 10 Mécanisme d'identification générique

Le système MHEG prévoit des mécanismes génériques pour identifier n'importe quelle entité, donnée ou train binaire MHEG (voir la Figure 13).

Les mécanismes d'identification sont de trois types:

- **Identification externe:** l'identification n'est pas codée à l'intérieur de l'objet MHEG. Les identifications externes peuvent être décodées pour des besoins de référence, sans avoir à décoder l'objet MHEG (voir 10.1).
- **Identification interne:** l'identification est codée dans l'objet MHEG. L'identification ne peut pas être lue tant que cet objet n'est pas décodé (voir 10.2).
- **Identification symbolique:** l'identification peut remplacer une identification interne ou externe (voir 10.3).

NOTE – Il ne faut pas confondre une identification interne ou externe avec la notion d'objet respectivement *distant* ou *local*. Un objet, qui soit *local* ou *distant* peut avoir une identification externe ou interne.

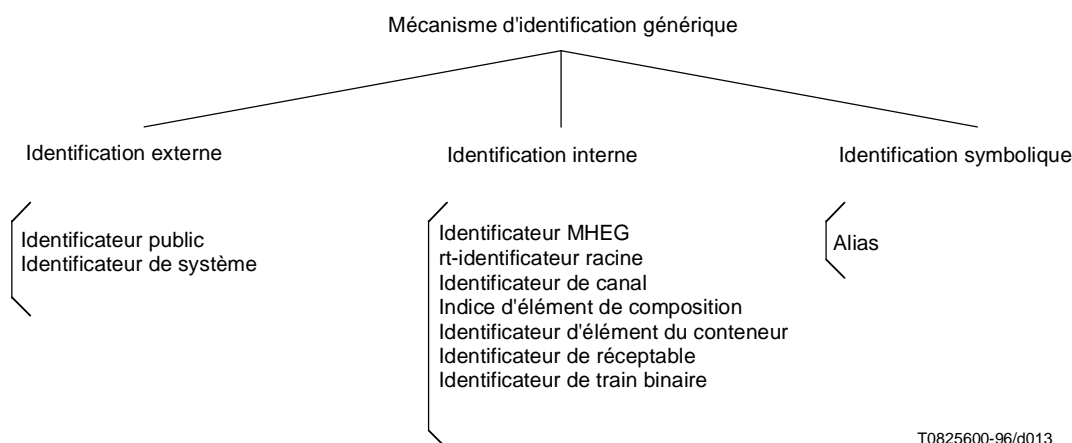


Figure 13/T.171 – Aperçu général du mécanisme d'identification générique

## 10.1 Identification externe

Ce mécanisme n'est pas défini par la présente Recommandation et n'est pas codé dans un objet MHEG.

NOTE – Il faut distinguer clairement l'identification et les références. Contrairement à l'identification externe qui n'est pas codée dans l'objet MHEG, la référence à un autre objet l'est, même si elle utilise une référence externe.

Ce mécanisme peut servir à identifier un objet ou une donnée quelconque qui n'est pas incluse dans un objet *contenu* ou un objet *script*.

Cette technique est définie dans la partie «Identificateurs publics formels» de l'ISO 8879 et dans l'ISO/CEI 9070 «Procédure d'enregistrement pour identificateurs de propriétaire de texte public», qui fournissent à la fois les notations ASN.1 et SGML pour la représentation des identificateurs de propriétaires enregistrés. Les identificateurs externes se composent de:

- un identificateur public (voir 10.1.1);
- un identificateur de système (voir 10.1.2).

### 10.1.1 Identificateur public

Un identificateur public est une chaîne de caractères dont la syntaxe est fixée par le 10.2/ISO 8879. Dans la présente Recommandation, un identificateur public est codé comme une chaîne imprimable ASN.1.

NOTE – La méthode prescrite par la présente Recommandation pour coder les identificateurs publics apporte des simplifications à celle qui est définie dans l'Annexe D de l'ISO/CEI 9070 afin d'en faciliter l'implémentation.

L'exemple d'identificateur public, donné à la Figure 14, illustre l'identification de l'image de Lena, la dame au chapeau, tel qu'il est stocké dans le système. La Figure 15 illustre l'exemple d'un identificateur public, identifiant un objet *distant*, nécessitant un accès par télécommunication. Un exemple complet d'identificateur public est donné à la Figure 16.

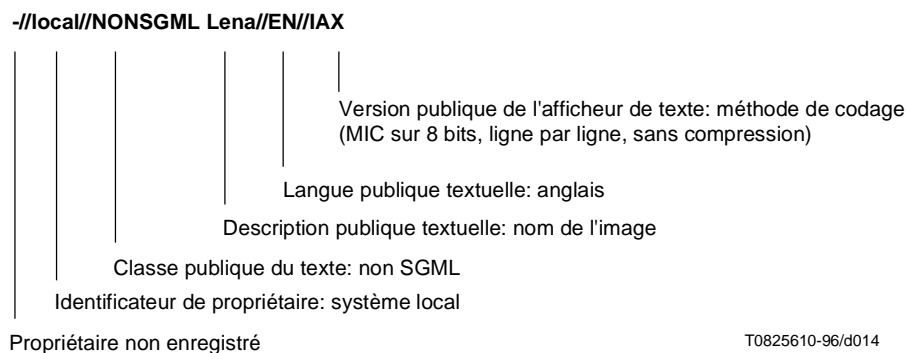


Figure 14/T.171 – Exemple d'identificateur public pour un propriétaire non enregistré

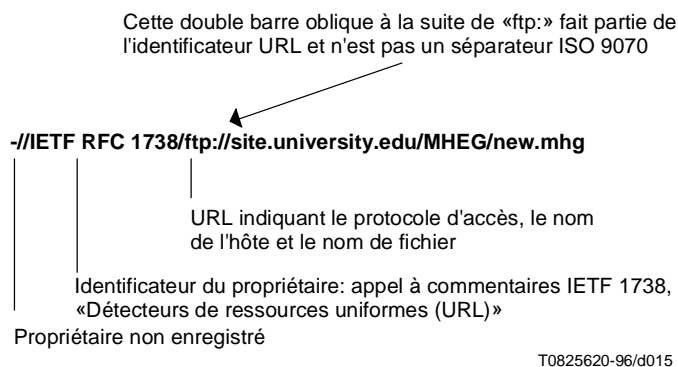
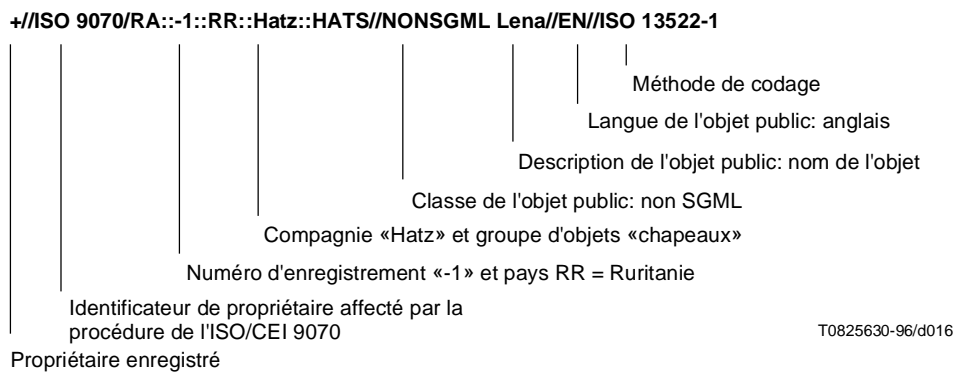


Figure 15/T.171 – Exemple d'identificateur public pour un identificateur de ressource uniforme URL



**Figure 16/T.171 – Exemple d'identificateur public pour un propriétaire enregistré**

La Figure 16 illustre le cas imaginaire d'un fabricant de chapeaux ruritanien appelé Hatz, qui s'inscrit auprès d'une autorité d'enregistrement de l'ISO/CEI 9070 et qui se voit affecter le numéro d'enregistrement «-1». La compagnie suit les conventions recommandées et décide de commencer les composantes nom du propriétaire avec «/RR:Hatz».

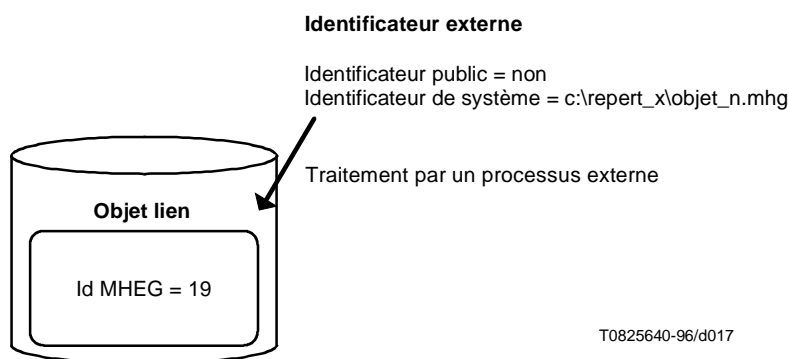
La compagnie Hatz décide de former un groupe d'objets MHEG publics contenant la description des chapeaux. La composante nom du propriétaire supplémentaire est HATS (chapeaux en ruritanien) portant ainsi le nom complet du propriétaire: ISO/CEI 9070/RA::-1::RR::Hatz::HATS.

### 10.1.2 Identificateur de système

Un identificateur de système fournit des moyens propres au système pour identifier les entités d'information du système. Il peut être aussi utilisé pour indiquer le système qui fournit l'information. Le jeu de caractères et la syntaxe de l'identificateur de système ne sont pas définis dans la présente Recommandation et dépendent du système. La présente Recommandation code l'identificateur de système comme une chaîne d'octets ASN.1.

NOTE – L'identificateur de système peut être utilisé pour décrire une requête de base de données ou de télécommunication pour un objet.

La Figure 17 illustre le fait que les mécanismes d'identification peuvent être combinés. L'objet *lien* a un identificateur interne codé intérieurement au moyen d'un identificateur MHEG, et d'un identificateur externe codé extérieurement au moyen d'un identificateur externe.



**Figure 17/T.171 – Exemple d'identificateur de système sans un identificateur public**



## 10.2 Identification interne

Ce mécanisme d'identification est codé dans les objets MHEG.

L'ISO/CEI 13522-1 prévoit les identificateurs internes suivants pour les entités externes:

- **identificateur MHEG**: identifie un objet MHEG quelconque (voir 17.2);
- **rt-identificateur racine**: identifie un rt-objet racine quelconque (voir 10.2.2);
- **identificateur de canal**: identifie un canal quelconque (voir 10.2.3).

L'identification interne est aussi utilisée pour identifier des éléments appartenant à une structure. Pour cela, on utilise les définitions suivantes:

- **indice**: identifie une génération donnée dans une structure. Les éléments d'une même génération reçoivent pour indice la suite des entiers commençant par un. La notation suivante est utilisée dans la présente Recommandation pour exprimer l'indice: `id_object_externe.i`, où *i* indique l'indice dans l'objet *externe*, par exemple, le troisième élément d'un objet *conteneur* portant l'identificateur MHEG 1000 est désigné par `1000.3`;
- **postambule**: identifie un élément dans une hiérarchie de structure. Un postambule est une liste d'indice et elle désigne un chemin partant d'une entité extérieure jusqu'à un élément donné. Chaque indice correspond à l'indice d'un élément d'une structure. Une suite d'indices correspond à des éléments imbriqués. Le dernier indice de la liste est l'indice de l'élément à identifier. Le postambule s'exprime par une notation de la forme `id_object_externe.i1.i2...in`, où «*i<sub>1</sub>, i<sub>2</sub>, ..., i<sub>n</sub>*» est la liste d'indices indiquant le chemin entre l'objet *externe* et l'élément.

L'ISO/CEI 13522-1 prévoit aussi les identificateurs internes suivants pour les éléments:

- 1) **indice d'élément de composition**: identifie un élément quelconque de composite (voir 10.2.4);
- 2) **identificateur d'élément conteneur**: identifie un élément quelconque de conteneur (voir 10.2.5);
- 3) **identificateur de réceptacle**: identifie un réceptacle quelconque (voir 10.2.6);
- 4) **identificateur de train binaire**: identifie un train binaire quelconque dans un objet *contenu multiplexé* (voir 10.2.7).

### 10.2.1 Identificateur MHEG

Il est possible d'affecter un identificateur MHEG à chaque objet MHEG. Lorsqu'il est utilisé, l'identificateur MHEG sera codé à l'intérieur de l'objet MHEG et comportera les éléments suivants:

- un **identificateur d'application** optionnel, fourni par le concepteur de l'application. Il sera codé selon la présente Recommandation sous la forme d'une chaîne d'octets ASN.1.
- un **numéro MH** (multimédia/hypermédia), qui identifie de manière univoque un objet MHEG dans une application. Le numéro MH sera codé sous la forme d'un entier. Il est obligatoire lorsque l'identificateur MHEG est présent.

Un identificateur MHEG réservé: «Null-MH» sert à identifier un objet MHEG vide.

Il appartient au concepteur de l'objet de s'assurer de l'unicité des identificateurs MHEG des objets MHEG.

Les identificateurs MHEG utilisent une notation de la forme  $a_1.a_2.a_3\dots a_{n-m}$ , où « $a_1, a_2, \dots, a_n$ » représente l'identificateur de l'application et  $m$  représente le numéro MH. Lorsque l'identificateur de l'application est omis, les identificateurs MHEG sont exprimés au moyen du seul numéro  $m$ .

**Exemple:** un objet *contenu* peut recevoir l'identificateur MHEG suivant: 129.63.24.2-49 tandis qu'un autre objet *composite* peut recevoir l'identificateur MHEG: 700, ce qui signifie que le *composite* n'a pas d'identificateur d'application.

### 10.2.2 Rt-identificateur racine

Un rt-identificateur racine est affecté à chaque rt-objet racine. Cet identificateur obligatoire comporte les éléments suivants:

- **identificateur de l'objet modèle**, qui identifie l'objet modèle. Cet objet modèle peut être identifié par l'un quelconque des mécanismes d'identification prévus par la présente Recommandation pour les objets MHEG: identificateur MHEG, indice d'élément conteneur, identificateur externe, alias...;
- **rt-numéro**, un entier choisi par l'auteur.

Un rt-identificateur racine réservé «Null-Root-Rt» est prévu pour identifier les rt-objets vides.

Il appartient au concepteur de l'objet de s'assurer de l'unicité des rt-identificateurs racine.

L'unicité des rt-numéros est requise à un instant donné. Un rt-numéro peut être réutilisé une fois que ce numéro est à nouveau disponible, c'est à dire, une fois le rt-objet qui l'utilisait, supprimé. Il n'est pas possible d'avoir, à un instant donné, deux rt-objets portant le même rt-numéro.

Les rt-identificateurs racine utilisent la notation *id\_objet\_modele:r*, où *id\_objet\_modele* peut être l'un quelconque des identificateurs prévus par la présente Recommandation pour identifier un objet MHEG et où «r» est le rt-numéro. Par exemple, le premier objet *rt-composite* créée à partir de l'objet *composite* portant l'identificateur MHEG 700, se verra affecter le rt-identificateur racine 700:1. Le troisième *rt-contenu* créée à partir de l'objet *contenu*, portant l'identificateur MHEG 129.63.24.2-49, se verra affecter le rt-identificateur 129.63.24.2-49:3.

### 10.2.3 Identificateur de canal

Un identificateur de canal est affecté à chaque canal. Cet identificateur, obligatoire, est un entier.

Un identificateur de canal réservé «canal par défaut» sert à identifier le canal par défaut.

Il appartient au concepteur de s'assurer de l'unicité des identificateurs de canal.

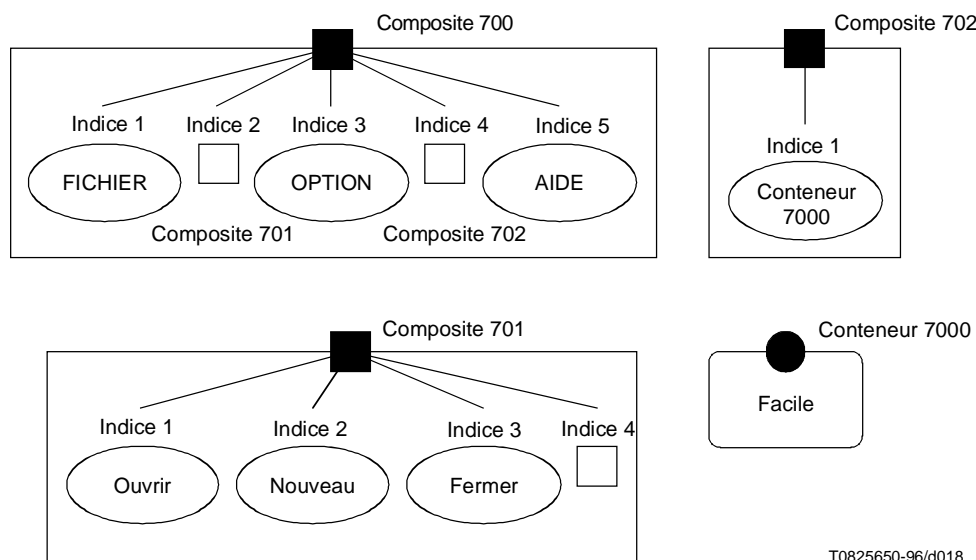
### 10.2.4 Indice d'élément de composition

La fonctionnalité *indice* sert dans un objet *composite* à identifier chacun des éléments qui le compose.

Chaque objet *composite* définit une génération. Les indices des éléments dont les modèles associés ne sont pas vides, sont fournis par l'auteur. Les indices manquants sont automatiquement fournis par le moteur MHEG. De tels éléments sont considérés comme des éléments avec un modèle associé vide.

Dans la Figure 18, les cinq éléments de l'objet *composite* 700 sont identifiés avec les indices 1, 2, 3, 4, et 5. Les quatre éléments de l'objet *composite* 701 sont identifiés avec les indices 1, 2, 3, et 4. L'élément unique de l'objet *composite* 702 est identifié par l'indice 1. Les objets *composite* 701 et 702, qui sont des modèles associés à l'objet *composite* 700, et le contenu 7000, qui est un modèle associé à l'objet *composite* 702, sont pointés par référence plutôt qu'inclus. Ils sont identifiés au moyen soit de leur propre identificateur ou de l'indice des éléments *composite* dans lesquels ils sont associés. La Figure 18 ne montre pas le comportement de composition dans les objets *composite*.

NOTE – L'indice d'élément est typiquement utilisé pour la construction d'un objet *rt-contenu*.



T0825650-96/d018

Figure 18/T.171 – Indexation d'éléments dans un objet *composite*

### 10.2.5 Identificateur d'élément conteneur

La fonctionnalité indice sert dans un objet *conteneur* à identifier chaque élément dans cet objet. Lorsqu'un objet *conteneur* contient une hiérarchie d'objets *conteneur* imbriqués, la fonctionnalité postambule sert à identifier les éléments imbriqués de la hiérarchie des objets *conteneur*.

Chaque objet *conteneur* définit une génération. Les indices des éléments sont automatiquement prévus par le moteur MHEG et sont indexés dans chaque génération, séquentiellement en partant de un.

Un élément d'un objet *conteneur* est toujours un objet MHEG. Il est possible qu'un élément conteneur, donc un objet MHEG, peut avoir déjà un autre identificateur différent, par exemple, un identificateur externe ou un identificateur MHEG. Dans ce cas, cet élément a deux identifications différentes, soit comme un élément d'un conteneur ou directement comme un objet MHEG.

NOTE – L'indice ou le postambule dans un conteneur est typiquement utilisé pour accéder rapidement aux éléments d'un conteneur. Lorsqu'une action *Prepare* (préparation) est appliquée à un objet MHEG au moyen de son identification dans un conteneur (par exemple 1000.3, élément 3 du conteneur 1000), le moteur MHEG pourra récupérer l'objet du conteneur avec un minimum de traitement. Si un identificateur MHEG est codé dans cet objet, il est disponible pour le moteur MHEG dès que l'objet sera préparé (résultat de l'action *Prepare*). Il est recommandé mais non requis, que les actions ultérieures appliquées sur cet objet, utilisent l'identificateur MHEG plutôt que l'indexage de conteneur.

Dans la Figure 19, les éléments définis dans l'objet *conteneur* sont des objets MHEG. Ils sont identifiés par un indice ou un postambule dans le conteneur; par exemple, l'indice 1 du conteneur 800 ou les postambules 1000.8.1 et 800.1 identifient l'objet *lien* portant l'identificateur MHEG égale à 810. Un identificateur MHEG et un identificateur externe peuvent également être affectés à chaque objet.

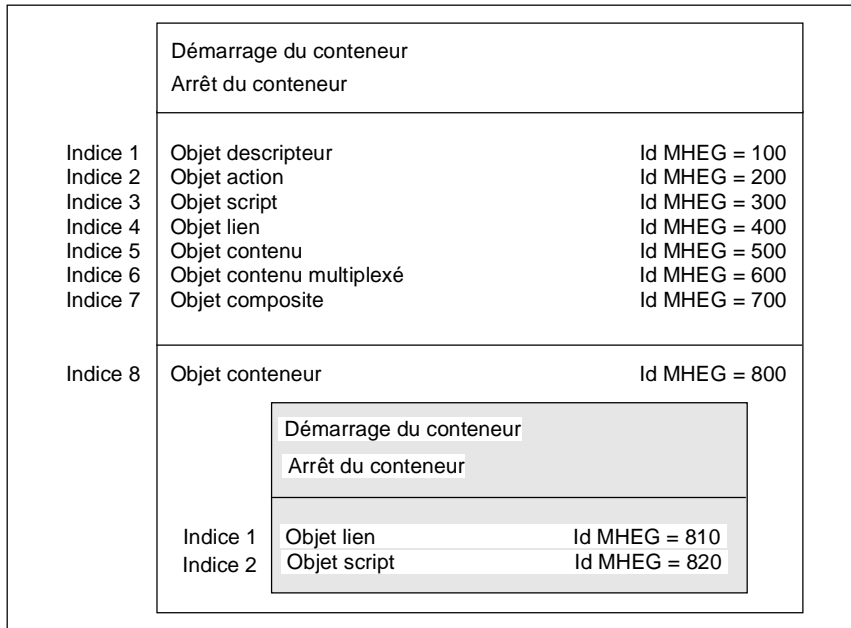
### 10.2.6 Identificateur de réceptacle

La fonctionnalité indice sert à identifier chaque élément d'une génération donnée de *rt-composite*. La fonctionnalité postambule est utilisée dans l'objet *rt-composite* racine pour identifier chaque élément du *rt-composite*.

Chaque *rt-composite* racine définit une hiérarchie complète. Dans chaque génération d'un *rt-composite*, les réceptacles sont indexés séquentiellement en partant de un, et déduits de l'objet *composite* modèle. Le postambule est utilisé pour identifier chaque réceptacle en indiquant le chemin depuis le *rt-composite* racine et jusqu'au réceptacle.

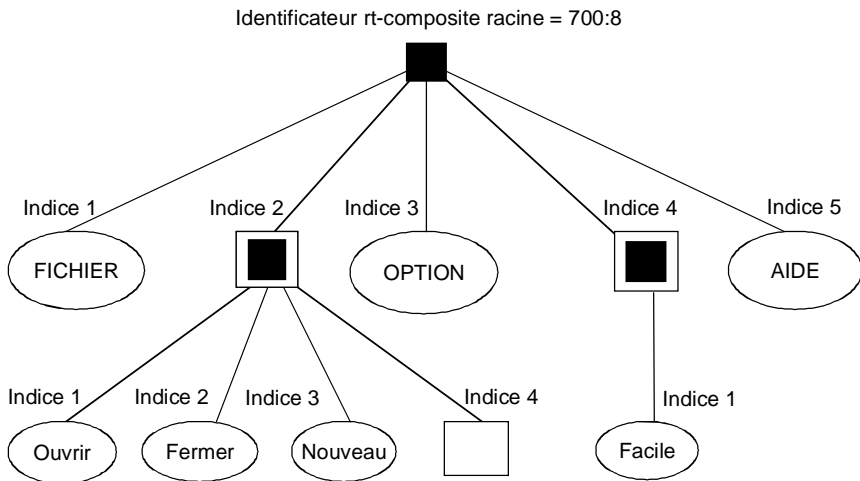
La Figure 20 illustre un *rt-composite* qui est le résultat d'une action *New* (nouveau) appliquée à un *rt-composite* créé à partir de l'objet modèle *composite* 700 représenté sur la Figure 18. Le *rt-numéro* est 8. Ce *rt-composite* est créé soit directement à partir de l'objet *composite* 700, ou à partir du même objet considéré comme élément 7 du conteneur 1000.

Objet conteneur: Id MHEG: 1000



T0825660-96/d019

Figure 19/T.171 – Indexation d'éléments dans un objet composite



T0825670-96/d020

Figure 20/T.171 – Indexation des réceptacles

Les réceptacles de la première génération sont identifiés comme les éléments du composite 700 par les indices 1, 2, 3, 4 et 5, ou par les postambules 700:8.1, 700:8.2, 700:8.3, 700:8.4 et 700:8.5.

NOTE – Ces objets peuvent aussi être identifiés par les postambules: 1000.7:8.1, 1000.7:8.2, 1000.7:8.3, 1000.7:8.4 et 1000.7:8.5, car 700:8 peut être identifié par 1000.7:8 (voir la note dans le paragraphe 10.2.5: Identification d'élément conteneur).

Dans la Figure 20, les trois réceptacles de la deuxième génération, sous l'indice 2 de la première génération, sont identifiés par les indices 1, 2, 3 comme les éléments du composite 701 ou par les postambules 700:8.2.1, 700:8.2.2, 700:8.3.2.

Le réceptacle de la seconde génération sous l'indice 4 de la première génération est identifié comme l'élément d'indice 1 du composite 702 ou par le postambule 700:8.4.1.

### 10.2.7 Identificateur de train binaire

Un identificateur de train binaire sert à identifier chaque train binaire d'un objet *contenu multiplexé*. Cet identificateur, qui s'exprime sous la forme d'un postambule, désigne un chemin partant d'un train binaire externe jusqu'à un train binaire interne dans les données multiplexées.

Il appartient au concepteur de s'assurer de l'unicité des identificateurs de train binaire dans chaque ensemble de données multiplexées.

## 10.3 Identification symbolique

La présente Recommandation fournit un moyen d'affecter un identificateur symbolique, qui peut servir à remplacer un identificateur interne ou externe quelconque. Cet identificateur symbolique est appelé alias ou pseudonyme.

Un alias offre les caractéristiques suivantes:

- adressage d'entités sous un autre nom;
- regroupement d'entités auxquelles s'appliquent les mêmes actions au même moment;
- évaluation multiple d'objets utilisant une affectation retardée d'alias. Une application complète peut être décrite par un auteur aux moyens d'un alias. L'affectation réelle de l'entité peut être faite dynamiquement plus tard.

Ceci est particulièrement utile pour un traitement multilingue/international, un format de support de résolution multiple, des objets à version multiple et pour un format de support à codage multiple. Par exemple, un auteur fournit un objet pour les différentes langues, et les actions sont décrites par l'utilisation d'alias. Ensuite, les actions `Set Alias` (positionnement de l'identificateur alias) sont utilisées pour affecter à chaque entité la langue correspondante.

Un alias est une chaîne de caractères qui est utilisée comme une alternative pratique à d'autres formats d'identification. Cet alias est affecté par un auteur au moyen de l'action `Set Alias` (positionnement de l'identificateur alias).

#### NOTES

1 – L'alias sera différent du nom de l'objet de la classe d'objet MH.

2 – Par exemple: `Set Alias(START, «e:\appli\start-up.mhg»)`: l'alias «START» peut alors remplacer l'identificateur externe: «e:\appli\start-up.mhg».

3 – Par exemple, `Set Alias(OPTION,700:8.2)`: l'alias «OPTION» peut alors remplacer le réceptacle 2 du rt-composite racine numéro 8 créée à partir du composite modèle 700 (Figure 20).

Une entité donnée peut avoir plusieurs alias simultanément. Un alias donné peut être affecté à plusieurs entités simultanément.

## 11 Mécanisme de référence générique

De nombreuses entités de données, trains binaire MHEG, doivent être pointés soit à partir d'un autre objet MHEG ou comme paramètres d'une action. Les entités MHEG sont aussi désignées comme cible des actions par leur référence.

### 11.1 Référence générique au moyen des identifications génériques

Les références génériques suivantes sont définies:

- 1) **Référence par identificateur externe**: référence à un objet ou à une donnée MHEG par son identificateur externe (voir 10.1).
- 2) **Référence par identificateur MHEG**: référence à un objet MHEG par son identificateur MHEG (voir 17.2).

La référence à un identificateur MHEG n'est résolue qu'une fois l'objet MHEG, auquel il est fait référence, est décodé. Une application utilisatrice peut disposer d'un mécanisme pour résoudre cette référence sans décodé l'objet, par exemple, au moyen d'une table de correspondance entre une localisation d'un objet MHEG et son identificateur MHEG.

- 3) **Référence par rt-identificateur racine:** utilisée pour faire référence à un rt-objet racine. Elle est définie comme une référence à l'objet modèle suivie d'une référence à un rt-numéro. Pour désigner l'objet modèle, on utilise une des références définies pour les objets MHEG. La référence d'un objet *rt-numéro*, peut se faire d'une des manières suivantes:
- un rt-numéro: utilisé pour faire référence à un rt-objet racine;
  - la constante \*: utilisée au moment de résoudre la référence pour désigner tous les rt-objets racine créés à partir de l'objet modèle spécifié;
  - la constante ?: utilisée pour faire référence à un seul rt-numéro déterminé dynamiquement (voir 11.3).  
Par exemple, si on considère l'objet *composite* défini dans les Figures 18 et 19:
    - 700:\* représente une référence à tous les objets *rt-composite* racine créés à partir de cet objet modèle *composite* au moment de résoudre cette référence;
    - 700:? représente une référence à un objet *rt-composite* racine créée à partir de l'objet modèle *composite* qui sera déterminé dynamiquement.
- 4) **Référence par identificateur de canal:** référence à un canal au moyen de son identificateur de canal (voir 10.2.3).
- 5) **Référence par postambule:** cette référence ne peut être utilisée directement par l'auteur. Elle fait partie de la référence de l'élément conteneur et de la référence identificateur de réceptacle. Elle prend l'une des formes suivantes:
- postambule: utilisé pour faire référence à un élément donné au moyen de son identification postambule (voir 11.1);
  - complément de postambule: fait référence aux mots **enfants** ou **descendants**. Lorsque ce complément est utilisé seul, il fait référence aux enfants (une génération ou toutes les générations: descendants, c'est-à-dire, descendants de l'entité externe elle-même);
  - postambule suivi par un complément de postambule: le complément de postambule, dans ce cas, s'applique à un élément désigné au moyen de l'identification postambule. Cet élément sera lui-même une entité construite, c'est-à-dire un autre objet *conteneur* ou un réceptacle *rt-composite*.
- 6) **Référence par élément conteneur:** référence à un objet MHEG par son élément conteneur. Il est défini comme une référence à un objet *conteneur* suivi par une **référence postambule**. L'objet *conteneur* est désigné par une des références définies pour les objets MHEG. La référence postambule permet à l'auteur de désigner:
- un seul objet MHEG, ou un élément du conteneur;
  - les enfants d'un objet *conteneur* externe;
  - les enfants d'un objet *conteneur*, lui-même un élément d'un objet *conteneur* externe;
  - les descendants de l'objet *conteneur* externe;
  - les descendants d'un objet *conteneur*, lui-même un élément d'un objet *conteneur* externe;
- 7) NOTE 1 – Dans la Figure 19, 1000.8.children ou 800.children représente une référence aux deux éléments de ce conteneur imbriqué, c'est-à-dire l'objet *lien* 810 et l'objet *contenu multiplexé* 820.
- 8) NOTE 2 – 1000.descendants représente une référence à tous les éléments de l'objet *conteneur* 1000 et de l'objet *conteneur* imbriqué 800.
- 9) NOTE 3 – La référence à un élément conteneur ne peut pas être résolue avant que l'objet *conteneur* désigné soit décodé. Cependant, si l'auteur fournit une information de décalage dans le descripteur (voir 0.5.3.9) pour un élément conteneur, le moteur MHEG peut utiliser cette information pour décoder uniquement l'élément spécifié du conteneur, sans décoder le reste de l'objet *conteneur*.
- 10) NOTE 4 – Si un auteur fournit une identification supplémentaire à l'élément d'un conteneur, par exemple un identificateur MHEG, cet élément du conteneur peut être aussi désigné par l'une des références définies pour un objet MHEG.
- 11) **Référence par identificateur de réceptacle:** utilisée pour désigner un réceptacle. Elle est définie comme une référence à un *rt-composite* racine suivie par une **référence postambule** de réceptacle. L'objet *rt-composite* racine est désigné au moyen d'une référence *rt-identificateur* racine. La référence postambule de réceptacle permet à l'auteur de désigner:
- un seul réceptacle;
  - les enfants du *rt-composite* racine;

- c) les enfants d'un réceptacle rt-composite du rt-composite racine;
  - d) les descendants du rt-composite racine;
  - e) les descendants d'un réceptacle rt-composite du rt-composite racine;
- 12) **Référence par postambule de réceptacle:** cette référence ne peut pas être utilisée directement par l'auteur. Elle fait partie de la référence par identificateur de réceptacle. Elle peut prendre les formes suivantes:
- a) un postambule: utilisé pour désigner un réceptacle donné à partir de son identification de postambule (voir ci-dessus);
  - b) un complément postambule réceptacle qui peut prendre les formes suivantes:
    - i) les constantes fournies par un complément postambule, c'est-à-dire, enfants ou descendants (voir ci-dessus). Utilisé seul, le complément postambule s'applique au rt-composite racine lui-même;
    - ii) la constante **?child** utilisée pour désigner un seul réceptacle du rt-composite racine lui-même, et dont la détermination s'effectuera dynamiquement;
    - iii) la constante **?descendant** utilisée pour désigner un seul réceptacle parmi les réceptacles descendants du rt-composite racine lui-même et dont la détermination s'effectuera dynamiquement. Lorsque l'on utilise le mécanisme ?child ou ?descendant, la technique pour déterminer le réceptacle est la même que celle pour le mécanisme ? (voir 11.3). La phase d'individualisation, dans ce cas, est effectuée sur chaque réceptacle enfant ou descendant du rt-composite racine;

les fonctionnalités ?child et ?descendant peuvent être utilisées de la manière suivante: soit par exemple, un rt-composite racine décrivant un menu, chaque réceptacle représentant une rubrique. Une des rubriques est sélectionnée, le même comportement est spécifié pour les différentes rubriques. Ce comportement est traité à chaque fois qu'une rubrique est sélectionnée. On peut traduire ce mécanisme par un seul lien au moyen des fonctionnalités ?descendant ou ?child. Le nombre des réceptacles enfants ou descendants pourra varier au cours de la vie du menu, le lien continuera à s'appliquer;
  - c) un postambule suivi par un complément postambule de réceptacle: le complément postambule s'applique, dans ce cas, au réceptacle désigné par l'identification de postambule. Ce réceptacle sera un réceptacle rt-composite;
- 13) NOTE 5 – Dans la Figure 20, 700:8.children représente une référence à tous les éléments enfants du rt-composite racine (indices 1 à 5, première génération).
- 14) NOTE 6 – 700:8.2.children représente une référence à tous les réceptacles enfants du réceptacle rt-composite 700:8.2 (indices 1 à 4, seconde génération).
- 15) NOTE 7 – 700:8.descendants représente une référence à tous les réceptacles descendant du rt-composite racine (toutes les générations).
- 16) NOTE 8 – 700:?.1 représente une référence au réceptacle rt-contenu (fichier) d'un rt-composite racine, créée à partir du modèle *composite* 700 et déterminé dynamiquement.
- 17) NOTE 9 – 700:?.descendants représente une référence à tous les réceptacles descendant d'un rt-composite racine, créée à partir du modèle *composite* 700, et déterminé dynamiquement.
- 18) NOTE 10 – 700:\*.1 représente une référence au réceptacle 1 du rt-contenu (fichier) de tous les rt-composite racine créés à partir de ce modèle *composite* au moment de résoudre cette référence.
- 19) NOTE 11 – 700:8.?descendant représente une référence à un seul réceptacle, déterminé dynamiquement, situé entre les réceptacles descendant (toutes les générations) du rt-composite racine.
- 20) NOTE 12 – 700:8.2.?child représente une référence à un seul réceptacle déterminé dynamiquement, situé entre les réceptacles enfant (ouvert, fermer, nouveau, indice) du réceptacle rt-composite 700:8.2.
- 21) NOTE 13 – 700:?.?child représente une référence à un seul réceptacle déterminé dynamiquement et situé entre les réceptacle enfant (une génération) d'un rt-composite déterminé dynamiquement créée à partir de l'objet modèle *composite* 700.
- 22) **Référence par identificateur de train binaire:** référence à un train binaire par son identificateur de train binaire (voir 10.2.7).
- 23) **Référence par alias:** référence à une entité, donnée ou train binaire MHEG par un des alias qui lui sont affectés (voir 10.3). La référence à un alias ne peut être résolue jusqu'à ce que l'alias soit affecté.

## 11.2 Références prédéfinies

Les valeurs constantes des références prédéfinies sont:

- 1) **Null-Data**: référence à une donnée vide.
- 2) **Null-MH**: référence à un objet MHEG vide.
- 3) **Null-Root-Rt**: référence à un rt-objet racine vide.
- 4) **Default-Channel**: référence à un canal par défaut.
- 5) **«This»**: référence locale à un composite ou un conteneur à partir de lui-même. Il est possible de faire référence à un objet *composite* ou un objet *conteneur* par l'une des techniques de référence définies pour un objet MHEG (par exemple, référence par identificateur MHEG, référence par élément du conteneur, référence par alias, référence identificateur externe,...) ou par la fonctionnalité «This» prévue à des fins de références locales depuis l'intérieur de l'objet *composite* ou à l'objet *conteneur* lui-même. «This» sera utilisé seulement à l'intérieur d'un objet *lien* ou d'un objet *action*. Il ne sera utilisé ni dans une référence d'élément conteneur ou dans une référence d'objet *composant* d'un élément de composition. Les conditions suivantes s'appliquent:
  - a) lorsque l'objet *composite* ou l'objet *conteneur* est prêt, «This» prend la valeur d'identificateur de l'objet *conteneur* ou *composite*. Lorsque le lien démarrage, le rt-démarrage et le démarrage du conteneur sont activés, la valeur de «This» leur est communiquée;
  - b) la propagation de la valeur de «This» à un objet *lien*, a pour effet de remplacer chaque occurrence du symbole «This» dans la condition de lien par sa valeur;
  - c) lorsque un tel lien est déclenché, la valeur de «This» est communiquée à son effet de lien. On applique alors les propositions suivantes:
    - i) chaque occurrence du symbole «This» dans la valeur d'utilisation de la macro-résolution et dans l'objet *action* est remplacée par la valeur de «This». Si l'objet *action* est un objet *action* imbriqué, toute occurrence du symbole «This», dans les objets *action* imbriqués, n'est alors remplacée par la valeur de «This»;
    - ii) Si le lien est dans un objet *lien* imbriqué, la valeur de «This» est alors communiquée dans les objets lien imbriqués.
- 6) NOTE 1 – Le symbole «This» a le rôle d'une variable locale dans le domaine de visibilité d'un composite ou d'un conteneur lui-même.
- 7) NOTE 2 – Dans la Figure 18, le démarrage de la disponibilité de l'objet *composite* a l'effet de lien suivant: New this:1, Active link L1. L'objet *lien* L1 est défini comme suit: TC = this:1.3 becomes selected, LE = Run (this:1.5). Lorsque le composite 700 est prêt, l'action de démarrage est activée, le numéro 1 du rt-composite racine est créée et la valeur this = 700 est propagée à l'objet *lien* L1.

## 11.3 référence ?

La référence ? est utilisée dans les objets *lien* et les objets *action* pour décrire un comportement individuel commun à tous les rt-objets racine créés à partir d'un objet modèle donné. Une telle fonctionnalité a les avantages suivants:

- le comportement est décrit seulement une fois et s'applique à tous les rt-objets racine créés à partir du même objet modèle;
- le concepteur de l'objet est capable de décrire un comportement individuel pour chaque rt-objet racine sans connaître le numéro des rt-objets racines qui sera créée à partir de l'objet modèle;
- une fois activé, le lien s'applique à tous les rt-objets racine existants ou futurs;
- cette fonctionnalité permet une indépendance totale entre les rt-objets racine créés à partir du même modèle, c'est-à-dire, qu'ils peuvent être présentés indépendamment, le comportement étant un comportement individuel attaché à chaque rt-objet racine.

La référence <id\_objet\_modèle:?> peut être utilisée dans un objet *lien* ou dans un objet *action* lorsqu'il est permis d'utiliser une référence rt-objet. Cependant, il faudra respecter les conditions suivantes:

- 1) si l'objet *lien* n'est pas imbriqué, la référence <id\_objet\_modèle:?> sera utilisée dans la valeur source, dans une condition de déclenchement au moins, afin d'affecter une valeur d'utilisation à la référence ?;
- le lien de démarrage du rt-disponibilité composite répond à cette prescription, sa condition de lien est «This:? prêt». Tous les liens imbriqués activés par la mise en œuvre de rt-disponibilité sont activés pour chaque nouveau rt-composite racine créé.



- 2) dans un objet *lien* imbriqué, la référence ? peut être utilisée pour différents objets modèles, par exemple, les références suivantes, «id\_objet\_modèle1:?», «id\_objet\_modèle2:?» peuvent être utilisées dans le même objet *lien* imbriqué;

il est déconseillé d'utiliser la référence ? avec plusieurs identificateurs d'objet modèle dans un objet *lien* non imbriqué. Ceci s'explique par le fait que la valeur de la référence ? est résolue lorsqu'une et une seule condition de déclenchement est vérifiée à un moment donné. Alors dans ce cas seulement, la référence ? est résolue et toutes les autres sont considérées comme 'undefined' (non définies).

La détermination du rt-numéro, pris en compte dans la position de chaque référence ?, est fortement dynamique. Lorsqu'un objet *lien* non imbriqué est activé, on applique les phases suivantes:

- 1) Chaque objet *lien* contenant une condition de déclenchement avec une valeur d'origine codé avec la référence «id\_objet\_modèle:?», doit être fixée en parallèle pour chaque rt-objet racine créé à partir de la valeur de référence «id\_objet\_modèle» indiquée. Pour cela, toutes les instances de références ? portant le même «id\_objet\_modèle» dans la condition de lien sont remplacées par une instance de rt-numéro de rt-objet racine dans chaque évaluation parallèle.

Pendant l'activation d'un tel objet *lien*, les rt-objets racine créés à partir de l'id\_objet\_modèle, peuvent varier, c'est-à-dire, de nouveaux rt-objets racine peuvent être créés et d'autres peuvent être détruits.

- 2) Si une condition de déclenchement est vérifiée pour un rt-objet racine donné, et si une condition de lien est satisfaite, son rt-numéro prend la valeur '? usage value' (valeur d'utilisation de ?). Chaque instance de référence ? avec le même id\_objet\_modèle sera remplacé par la valeur '? usage value' (valeur d'utilisation de ?) aux endroits suivants:

- a) dans l'effet de lien. Si l'objet *action* définissant l'effet de lien, est un objet *action* imbriqué, la valeur '? usage value' (valeur d'utilisation de ?) est propagée à chaque instance de référence ? avec le même id\_objet\_modèle dans les objets *action* imbriqués;

- b) lorsqu'un objet *lien* imbriqué est activé par l'objet *lien*, les dispositions suivantes s'appliquent:

- i) une instance particulière de l'objet *lien* imbriqué traite la valeur: '? usage value' (valeur d'utilisation de ?);
- ii) la valeur '? usage value' (valeur d'utilisation de ?) est communiquée à chaque instance de référence ? avec le même id\_objet\_modèle de l'instance du lien imbriqué;
- iii) si cette instance de l'objet *lien* imbriqué contient quelques références ? non résolus dans une condition de déclenchement, les phases 1 et 2 ci-dessus s'appliquent;
- iv) si quelques références ? restent non résolus après les phases 1 et 2, les références correspondantes doivent être considérée comme 'undefined' (non définies);

- 3) La phase 1 est traitée en parallèle sur tous les rt-objets racine créés à partir de l'objet modèle. Si la condition de lien d'un objet *lien* imbriqué est vérifiée pour deux rt-objets racine en parallèle dans la phase 1, par exemple 700:3 et 700:4, la phase 2 doit être traitée pour chaque valeur '? usage value' (valeur d'utilisation de ?) en parallèle. Ainsi, si l'effet de lien déclenche un lien imbriqué, deux instances de ce lien imbriqué sont traités en parallèle pour chaque valeur '? usage value' (valeur d'utilisation de ?) des objets 700:3 et 700:4.

Si le lien n'est pas déclenché, il ne pourra pas l'être pour n'importe quelle valeur '? usage value' (valeur d'utilisation de ?).

L'exemple suivant tient compte de l'objet *composite* défini dans la Figure 18:

- son démarrage peut avoir les effets de lien suivants: new 700:3, new 700:9;
- son effet de lien de démarrage de rt-disponibilité est: activate link object L1;
- le lien L1 est un objet *lien* imbriqué, il est défini de la manière suivante: TC = 700:?.3 becomes selected, LE = run 700:?.5. Ce lien est utilisé pour décrire un comportement individuel commun à tous les rt-composite racine qui peuvent être créés à partir de l'objet modèle 700, c'est-à-dire pour 700:3, 700:9 mais aussi pour n'importe quel futur rt-composite racine créée à partir du modèle 700.

Les applications suivantes sont exécutées:

- 1) lorsque le démarrage de rt-disponibilité est activé, c'est-à-dire, à chaque moment un nouveau rt-composite racine est créé, une instance de L1 traite ce nouveau rt-composite racine. Le rt-numéro de ce nouveau rt-composite racine est considéré comme la valeur '? usage value' (valeur d'utilisation de ?) et est transmis à cette instance de L1;

- 2) il y a une instance de lien L1 pour le rt-composite racine déjà créée à partir du modèle 700. Ainsi, l'on peut considérer que le lien L1 est activé pour tous les rt-composite racine du modèle 700 et que son déclenchement est fixé, chaque fois que l'état de sélection du réceptacle 3 pour n'importe quel rt-composite racine créée à partir du modèle 700, change;
- 3) lorsque le déclenchement d'une instance de lien L1 est vérifié pour un rt-composite racine, c'est-à-dire 700:2, la condition de lien est vérifiée et la valeur de ce rt-numéro rt-composite est considérée égale à la valeur '? usage value' (valeur d'utilisation de ?). Cette dernière est transmise à l'effet de lien de cette instance lien L1. Ainsi le réceptacle 5 du rt-composite racine 700:2 devient actif;
- 4) lorsque les deux 700:2.3 et 700:3.3 sont sélectionnés en parallèle, deux instances du lien L1 sont satisfaites en parallèle, leur effet de lien doit être exécuté en parallèle. C'est le réceptacle 5 du rt-composite 700:2 et le réceptacle 5 du rt-composite 700:3 qui deviennent en exécution parallèlement.

Ce qui suit est un autre exemple:

- l'objet *lien* est défini de la manière suivante: TC = 700:?.3 becomes selected, LE = run (700:?.5), activate (link object L2, link object L3);
- le lien L2 est un objet *lien* imbriqué défini comme suit: LC:900:3.1 becomes selected AND (700:?.2) is running, LE: stop 700:?.5 run 700:?.2;
- le lien L3 est un objet *lien* imbriqué plus complexe, il est défini de la manière suivante: LC: 999:?.1 becomes selected AND (700:?.5) is running, LE: stop (999:?.1, 700:?.5) run (999:?.2, 700:?.2).

Les applications suivantes sont exécutées:

- 1) les instances du lien L1 sont traités comme dans l'exemple précédent;
- 2) on considère maintenant qu'une instance de condition de lien L1 est satisfaite pour le rt-composite racine 700:7, 7 devient la valeur '? usage value' (valeur d'utilisation de ?) pour le composite 700. Cette valeur est transmise à l'effet de lien de cette instance L1. C'est le réceptacle 5 du rt-composite racine 700:7 qui devient actif comme dans l'exemple précédent, mais en plus les liens L2 et L3 sont activés;
- 3) lorsque L2 est activé, une instance de L2 traite la valeur '? usage value' (valeur d'utilisation de ?) et toutes les représentations de 700:? sont remplacés par 700:7 dans l'effet de lien de cette instance. Lorsque cette instance de lien L2 est traitée et lorsque sa condition de lien est vérifiée, c'est-à-dire 900:3.1 est sélectionné et 700:7.2 est en exécution, alors 700:7.5 est arrêté et 700:7.2 devient en exécution;
- 4) lorsque L3 est activé, l'instance de L3 traite la valeur '? usage value' (valeur d'utilisation de ?) et toutes les représentations de 700:? sont remplacées par 700:7 dans l'effet de lien de cette instance. Si une référence ? est restée dans une condition de déclenchement, l'instance du lien L3 est fixée en parallèle pour tous les rt-composites racine créés depuis 999. Si la condition de déclenchement de cette instance de L3 est valable pour 999.9, la valeur '? usage value' (valeur d'utilisation de ?) 9 est remplacée dans cette instance, 999:9.1 et 700:7.5 sont aussi arrêtés, 999:7.2 et 700:7.2 deviennent en exécution.

## 12 Valeur générique

Ce paragraphe décrit les mécanismes prévus par la présente Recommandation pour spécifier les valeurs de manière générique. Les valeurs génériques sont utilisées pour exprimer:

- des paramètres d'action élémentaire;
- des paramètres d'action `get` (extraction);
- des conditions;
- des attributs d'objet MHEG.

Une valeur générique peut être définie:

- 1) **constante**: une constante particulière est aussi prévue pour identifier une valeur 'unspecified' (non spécifiée).

Cette valeur est utilisée dans la condition de lien comme valeur de comparaison;

- 2) **valeurs évaluées**: une valeur évaluée est le résultat du traitement d'une action `get` (extraction) par le moteur MHEG. Lorsque le moteur MHEG n'est pas capable de traiter une action `get` (extraction), la valeur 'undefined' (non définie) est adoptée comme résultat de l'évaluation;

les valeurs génériques peuvent être stockées dans le champ de données d'un objet *contenu* puis extraites de celui-ci.

Une valeur générique appartient à l'un des types suivants:

- booléen générique (voir 12.1);
- numérique générique (voir 12.2);
- entier générique (voir 12.3);
- fraction générique (voir 12.4);
- chaîne générique (voir 12.5);
- référence générique (voir 12.6);
- liste générique (voir 12.7).

Il appartient au concepteur de l'objet de s'assurer que les valeurs génériques données comme paramètres d'action, sont conformes en type et en nombre aux paramètres tels qu'ils sont spécifiés par la présente Recommandation. Si les paramètres effectifs ne sont pas conformes en type ou en nombre à ce qui est prévu, alors il y a erreur et l'action est ignorée.

Le type d'une valeur générique détermine:

- l'ensemble des valeurs qu'il est possible de prendre;
- la sémantique commune à ces valeurs.

### 12.1 Booléen générique

Un booléen générique a une valeur égale à: Vrai, Faux. Elle peut être spécifiée de la manière suivante:

- constante booléenne;
- résultat de l'évaluation d'une action `get` (extraction);
- résultat d'une opération logique d'une condition de lien;
- résultat d'une opération de comparaison d'une condition de lien.

### 12.2 Numérique générique

Un numérique générique a une valeur numérique. La présente Recommandation n'impose aucune limite à ces valeurs numériques. Les valeurs peuvent être entières, réelles ou complexes. Toutefois, les représentations codées peuvent imposer des limites. La syntaxe ASN.1 définie par la présente Recommandation ne prend en charge que les valeurs entières.

NOTE – Il est recommandé que l'auteur tienne compte de ces limitations.

Une valeur numériques génériques peut être spécifiée de la manière suivante:

- constante numérique;
- résultat de l'évaluation d'une action `get` (extraction).

### 12.3 Entier générique

Un entier générique a une valeur entière.

Elle peut être spécifiée de la façon suivante:

- constante entière;
- résultat de l'évaluation d'une action `get` (extraction).

### 12.4 Fraction générique

Une fraction générique a pour valeur un couple d'entiers (m, n) avec m le numérateur, n le dénominateur et n étant supérieur ou égale à un. Si n est omis, on lui donne, par défaut, la valeur 100, interprétant m comme un pourcentage.

Une valeur de fraction générique peut être spécifiée de la manière suivante:

- constante fractionnaire;
- résultat de l'évaluation d'une action `get` (extraction).

## 12.5 Chaîne générique

Une chaîne générique a pour valeur un nombre quelconque de caractères. La présente Recommandation ne limite d'aucune manière, le schéma de codage des valeurs de type chaîne. Ces valeurs peuvent être représentées dans un jeu de code quelconque. Cependant, les représentations codées peuvent imposer des limitations. La syntaxe ASN.1 définie par la présente Recommandation ne prend en charge uniquement les chaînes graphiques.

NOTE 1 – Il est recommandé que l'auteur tienne compte de ces limitations.

Les chaînes graphiques contiennent tous les jeux de caractères G et le caractère d'espace comme définies dans la syntaxe ASN.1. Une telle spécification prévoit la possibilité d'utiliser des chaînes internationales.

La chaîne est considérée comme un tout, c'est-à-dire, les caractères composants ne peuvent pas être adressés individuellement.

NOTE 2 – Typiquement, la chaîne de caractère est utilisée comme un label, par exemple, 'Aide'.

La valeur des chaînes génériques peut être définie comme suit:

- constante de type chaîne;
- résultat de l'évaluation d'une action `get` (extraction).

Les minuscules et majuscules sont considérées comme différentes.

## 12.6 Référence générique

Une référence générique est une des références définies dans la présente Recommandation (voir paragraphe 11).

La valeur des références génériques peut être définie de la façon suivante:

- constante de type référence;
- résultat de l'évaluation d'une action `get` (extraction).

## 12.7 Liste générique

La liste générique a pour valeur un ensemble ordonné, éventuellement vide, de valeurs génériques de type quelconque. Les éléments d'une liste générique peuvent être d'un type différent des autres éléments de la liste. Chaque élément d'une liste générique possède un indice implicite.

NOTES

1 – Une liste générique peut constituer un élément.

2 – Chaque élément d'une liste ou d'une sous liste peut être extrait séparément par la fonctionnalité de l'action `Get Data` (extraction de données).

3 – Le type de liste générique peut servir à construire des vecteurs (éléments tous du même type) et pour construire des ensembles composés (ensembles d'éléments de types différents).

une valeur de liste générique peut être spécifiée de la manière suivante:

- liste de constantes;
- résultat de l'évaluation d'une action `get` (extraction).

## 13 Mécanisme de macro

Le mécanisme de macro-instruction fournit une technique générale pour coder efficacement des objets *action* et *lien* fréquemment utilisés et dans lesquels seules quelques valeurs varient d'un objet à l'autre. Ceci permet de partager des comportements complexes et de les réutiliser. Un auteur peut créer un catalogue de modèles prédéfinis pour les objets *action* et *lien*, appelées objet *macro-action* (voir 18.4) et objet *macro-lien* (voir 19.5). Dans une *macro-action* ou un *macro-lien*, un attribut, au moins, est codé comme une macro-valeur paramétrique.

Une macro-valeur paramétrique se compose de:

- un identificateur de macro-définition: c'est soit une chaîne ou un entier. Il est utilisé comme un symbole représentant le macro-paramètre. Un identificateur de macro-définition donné peut être utilisé par plusieurs macro-valeurs paramétriques;
- une valeur d'utilisation par défaut: c'est une valeur, qui doit être affectée à l'attribut codé comme une macro-valeur paramétrique, lorsque l'affectation d'une macro-valeur d'utilisation est omise.

Chaque objet *lien* contient un ensemble facultatif d'attribut appelés 'macro parameter resolution' (résolution de macro-paramètre) et qui sert à résoudre la macro-valeur paramétrique contenue dans son effet de lien. Chaque résolution de macro-paramètre est composée d'un identificateur de macro-définition et d'une macro-valeur d'utilisation, affectée à l'identificateur de macro-définition.

Lorsqu'un objet *lien* est déclenché, l'effet de lien est traité. La première étape de ce traitement est appelé phase de résolution des macro-mécanismes; cette phase se limite à ce processus d'effet de lien.

NOTE 1 – Ce même lien peut être déclenché plusieurs fois en parallèle, par conséquent, la phase de résolution de macro-instruction se limite à un traitement d'effet de lien donné.

Le but de cette phase est d'affecter une macro-valeur d'utilisation à chaque macro-valeur paramétrique contenue dans l'effet de lien. Chaque résolution de macro-paramètre est communiquée à l'objet *action* décrivant l'effet de lien. Les conditions suivantes s'appliquent:

- si l'action est un objet *action* imbriqué, la résolution de macro-paramètre est communiquée à l'objet *action* imbriqué;
- si un objet *lien* imbriqué est activé par l'effet de lien, la résolution de macro-paramètre est communiquée à son objet *lien* imbriqué, excepté si cet objet *lien* imbriqué lui-même contient une résolution macro-paramètre qui utilise le même identificateur de macro-définition.

Durant la phase de propagation, pour chaque macro-valeur paramétrique, on applique ceci:

- L'identificateur de macro-définition de la macro-valeur paramétrique est extrait d'une des résolutions de macro-paramètre communiquées. Cette macro-valeur paramétrique est remplacée par la macro-valeur d'utilisation de cette résolution de macro-paramètre communiquée.
- NOTE 2 – Même si la valeur d'utilisation n'est pas compatible avec le type espéré de l'attribut codé comme macro-valeur paramétrique, le moteur MHEG est requis pour l'affecter. Il appartient au concepteur de l'objet de s'assurer de la compatibilité.
- NOTE 3 – Si la valeur d'utilisation est une valeur évaluée, la macro-valeur paramétrique est remplacée par cette action `get` (extraction) sans la traiter.
- Sinon, cette macro-valeur paramétrique est remplacée par la valeur d'utilisation par défaut correspondante, codée dans la macro-valeur paramétrique. Si il n'y a pas de valeur d'utilisation par défaut qui est fournie, cette macro-valeur paramétrique est remplacée par la valeur 'undefined' (non définie).

Dans les exemples ci-dessous, on utilise les abréviations suivantes:

- MRP (*macro parameter resolution*) résolution de macro-paramètre;
- MDI (*macro def id*) identificateur de macro-définition;
- MUV (*macro usage value*) valeur d'utilisation macro;
- DUV (*default usage value*) valeur par défaut (lorsque DUV est omis aucune valeur par défaut n'est fournie);
- ALE (*action in link effect*) action dans un effet de lien.

**Exemple 1** – Effet de lien simple:

L1 a les spécifications suivantes:

- LC1: sélection du bouton 1;
- MPR1: (MDI = TARGET1, MUV = 700:1), (MDI = TARGET2, MUV = 700:2).
- ALE1: New(MDI = TARGET1), New(MDI = TARGET2), Run(MDI = TARGET1), Run(MDI = TARGET2).

Si LC1 est satisfait, la phase de macro-résolution est traitée et l'action ALE1 est traité de la manière suivante:

- ALE1: New(700:1), New(700:2), Run(700:1), Run(700:2).

**Exemple 2** – Action imbriquée de l'effet de lien:

L2 est spécifié de la manière suivante:

- LC2: sélection du bouton 1;
- MPR2: (MDI = SYNC, MUV = serial), (MDI = TARGET1, MUV = 700:1), (MDI = TARGET2, MUV = 700:2);
- ALE2: ActionObject1.

L'objet ActionObject1 est défini ci-dessous:

- indicateur de synchronisation: SYNC;
- ensemble d'actions: New(TARGET1), Run(TARGET1), Stop(TARGET2).

Si LC2 est satisfait, la phase de macro-résolution est traitée et chaque valeur MUV est communiquée à l'objet ActionObject1 et ce dernier est traité de la manière suivante:

- indicateur de synchronisation: série;
- ensemble d'actions: New(700:1), Run(700:1), Stop(700:2).

Si l'indicateur de synchronisation est résolue comme 'undefined' (non défini), l'action n'est pas traitée.

**Exemple 3** – Liens imbriqués dans l'effet de lien:

L3 a les spécifications suivantes:

- LC3: sélection du bouton 1;
- MPR3: (MDI = TARGET1, MUV = 700:1);
- ALE3: New(TARGET1), Activate(L10), Activate(L20).

L10 est défini ci-dessous:

- LC10: si TARGET1 est 'not running' (pas en fonctionnement);
- MPR10: aucun;
- LE10: Run(TARGET1).

L20 est défini ci-dessous:

- LC20: si TARGET1 est 'not running' (pas en fonctionnement);
- MPR20: (MDI = TARGET1, MUV = 999:1);
- ALE20: New(TARGET1), Run(TARGET1).

Si LC3 est exécuté, MPR3 est communiqué à ALE3, L10, L20 et L10 est traité de la manière suivante:

- LC10: si 700:1 est 'not running' (pas en fonctionnement);
- LE10: Run(700:1).

Sinon, L20 est traitée de la façon suivante:

- LC20: si 700:1 est 'not running' (pas en fonctionnement);
- ALE20: New(999:1), Run(999:1).

## 14 Crochet

Ce paragraphe décrit la prise en charge du traitement des crochets, qui sont utilisés dans les objets *contenu*, *script* et *descripteur*. Les crochets fournissent l'information pour l'identification de type de données codées. Un crochet se compose de l'information de codage qui identifie la méthode de codage ainsi qu'une description de codage facultative qui peut être utilisées pour des paramètres particuliers de la méthode de codage.

Deux genres de crochets sont prévus, l'un pour les données contenues et l'autre pour les données de script:

- 1) **Crochet de contenu:** composé d'un champ d'identification pour le codage de données standards et d'un champ descriptif:
  - a) l'information du codage de contenu contient une identification de la norme de codage. Une liste comportant un large éventail de normes de codage existantes – MPEG, G711, JPEG, ... – sont tenus à jour dans le catalogue d'enregistrement par la procédure décrite dans l'ISO/CEI 13522-4. Des formats de codage privé peuvent également être appliqués par l'utilisation d'un catalogue privé;
  - b) la description du codage de contenu donne une description plus précise des caractéristiques nécessaires pour décoder le contenu. La sémantique de ces paramètres n'est pas définie par la norme MHEG mais est donnée par la sémantique de la norme du codage des données ou Recommandation qui se rapporte à l'application utilisatrice. De telles informations peuvent aussi être fournies dans les catalogues de codage de contenu.

- 2) **Crochet de script:** composé d'un champ d'identification du langage de script et d'un champ descriptif:
  - a) l'information du codage de script contient une identification de la norme de codage de script. Une liste comportant un large éventail de normes de codages existantes – C, C++, Lisp,... – sont tenues à jour dans le catalogue d'enregistrement par la procédure décrite dans l'ISO/CEI 13522-4. Des formats de codage privés peuvent également être appliqués par l'utilisation d'un catalogue privé;
  - b) la description du langage de script donne une description plus précise des caractéristiques nécessaires pour décoder le script. La sémantique de ces paramètres n'est pas définie par la norme MHEG mais est donnée par la sémantique des normes de codages de données ou Recommandations qui se rapporte à l'application utilisatrice. De telles informations peuvent aussi être fournies dans les catalogues de codage de script.

## 15 Extensibilité de T.171

La présente Recommandation prévoit les possibilités d'extension suivantes:

- 1) catalogues (voir 15.1);
- 2) ajout de nouvelles classes d'objet MHEG (voir 15.2);
- 3) disposition d'extensibilité (voir 15.3).

### 15.1 Catalogues

L'ISO/CEI 13522-1 fournit un ensemble de catalogues pour assurer la compatibilité des moteurs MHEG entre eux et pour permettre des extensions futures. De nouveaux concepts peuvent être mis à la disposition de tous les systèmes MHEG si on enregistre une référence à un identificateur de format dans un catalogue d'enregistrement. Cette possibilité est limitée à un groupe restreint d'application si l'enregistrement est effectué dans un catalogue privé:

- 1) les catalogues d'enregistrement sont tenus à jour conformément à l'ISO/CEI 13522-4. Des concepts normalisés, bien établis sont énumérés dans le catalogue d'enregistrement. Il est possible, dans ce catalogue, d'enregistrer des concepts en cours d'élaboration;
- 2) des concepts propres à une application se trouvent dans le catalogue privé qui n'est utilisé que par un groupe restreint d'applications. Seul des moteurs MHEG particuliers pourront faire référence à de tels catalogues. La tenue à jour du catalogue privé est assurée par l'application utilisatrice.

Dans chaque catalogue, une description facultative peut être associée à chaque entrée pour en clarifier l'usage.

Un moteur MHEG peut prendre en compte une entrée de catalogue quelconque et introduire alors le concept correspondant dans son processus. Lorsqu'un auteur fournit un objet ou une action MHEG, utilisant une telle entrée, le moteur MHEG sera capable de les traiter. Si l'auteur utilise une entrée qui n'est pas tenue en compte par le moteur MHEG, celui-ci ignorera l'objet ou l'action MHEG correspondant.

Un catalogue d'enregistrement est prévu, et une application utilisatrice peut tenir à jour un catalogue privé pour une utilisation privée. Pour chacun des attributs ci-dessous, utilisés par la présente Recommandation, il est prévu un catalogue d'enregistrement; une application utilisatrice pourra également tenir à jour un catalogue pour une utilisation privée:

- **codage de contenu:** contient une description précise des méthodes de codage de contenu qui peut être indiquée dans le crochet de contenu, par exemple: ISO/CEI 11172-2 (MPEG2 vidéo). Une description de codage de contenu peut être associée à chaque entrée de ce catalogue;
- **classification de contenu:** prévue comme une aide facultative pour la détermination du type des données de contenu. Elle fournit une information sur le type des données. Pour les données monomédia, elle sert à indiquer le support de perception – texte, graphique, son,... –. Cette information peut servir dans un traitement de négociation, une base de données; elle peut également être utilisée par un moteur MHEG pour choisir un décodeur de script;
- **codage de script:** contient le langage de script, il peut être indiqué dans le crochet de script – C++, Smalltalk, SMSL, ... –. Une description de codage de script peut être associée à chaque entrée de ce catalogue;
- **classification de script:** prévue comme une aide facultative dans la détermination du type de langage de script. Cette information peut servir dans un traitement de négociation ou une base de données; elle peut également être utilisée par un moteur MHEG pour choisir un décodeur de script;
- **type de support:** indique le type du support et peut être utilisé dans l'objet *descripteur* pour faciliter le mappage du canal sur un dispositif de reproduction physique – texte, image fixe, vidéo, ... –;

- **style:** contient les styles qui peuvent servir dans une action `set style` (désignation du style), par exemple, pour l'interaction utilisateur: bouton, curseur, champ d'entrée. Les styles peuvent dépendre de l'interface graphique, de la plate-forme et de sa personnalisation par l'utilisateur. Des informations supplémentaires peuvent être associées à chaque entrée de ce catalogue afin de définir le style de manière plus précise – un intervalle de valeur pour le curseur, une orientation pour le menu, ... –;
- **événement:** contient une liste d'identificateurs d'événements catalogués – cliquement sur la souris, enfoncement d'une touche, télécommande, ... – qui peut servir dans l'objet *descripteur* pour indiquer le mappage prévu des identificateurs d'événement déterminés par l'auteur avec les identificateurs d'événements catalogués. Des informations supplémentaires peuvent être associées à chaque entrée de ce catalogue afin de définir l'événement de manière plus précise, par exemple, la position du click;
- **action élémentaire étendue:** contient une liste des actions élémentaires cataloguées, utilisée dans l'action `catalogued elementary` (élémentaire cataloguée). Ces actions sont utilisées pour compléter le nombre des actions élémentaires définies par la présente Recommandation – tracé d'une ligne, opération arithmétique,... –. Des informations supplémentaires peuvent être associées à chaque entrée de ce catalogue afin de définir le comportement d'une action de manière plus précise – les paramètres, l'effet MHEG, les périodes permises, les conditions d'erreur supplémentaires, ... –;
- **attribut étendu:** contient une liste d'attributs catalogués utilisés dans l'action `set catalogued attribut` (positionnement d'attribut catalogué). Ces attributs sont utilisés pour compléter le nombre d'attributs d'entités MHEG définies par la présente Recommandation – couleur, police de caractères, ... –. Des informations supplémentaires peuvent être associées à chaque entrée de ce catalogue afin de définir l'attribut de manière plus précise, par exemple, le type.

## 15.2 Ajout de nouvelles classes d'objet MHEG

La présente Recommandation réserve les numéros 0 à 9999 à l'arc 3 ASN.1 OBJECT IDENTIFIER (identificateur d'objet ASN1). Les numéros 0 à 9999 ne seront pas utilisés à titre privé. Cependant, une application utilisatrice peut utiliser d'autres numéros pour créer de nouvelles classes d'objet, qui peuvent être des modifications de classes d'objets complètement nouvelles ou des modifications d'objets de certaines classes existantes.

## 15.3 Disposition d'extensibilité

Afin de permettre de future extension d'attributs, une disposition d'extension a été prévue; elle est codée sous la forme de points de suspensions («...») en ASN1:

- attribut `elementary action` (action élémentaire);
- attribut `evaluated value` (valeur évaluée);
- attribut de classe `mh-objet description` (description);
- chaque classe feuille d'objet MHEG.

Pour chaque attribut de disposition d'extension, une liste d'étiquettes numérotées de 0 à 9999 est réservée afin de servir dans les autres Recommandations de la série T.170. Les étiquettes réservées ne seront pas utilisées à titre privé.

## SECTION 3 – APERÇU GÉNÉRAL DES CLASSES MHEG

Les paragraphes suivants présentent la structure fournie par le groupe MHEG pour transférer les informations hyper et multimédia. Le moteur MHEG procède à l'analyse de cette structure.

## 16 Aperçu général des classes d'objets MHEG

L'approche orientée objet a été adoptée pour concevoir la norme parce qu'elle s'adapte aux prescriptions en matière d'objets actifs, autonomes et réutilisables. Les classes d'objets transférables ont été classées comme le montre la Figure 21. Chaque objet transféré appartient à l'une des classes **imprimées en gras**. Des classes abstraites ont été ajoutées pour décrire les attributs communs ou pour grouper les classes traitant de sujets similaires. Le groupe MHEG ne définit pas de méthodes sur ses classes. L'utilisation du paradigme orienté objet se alors limite à l'héritage des attributs.



Dans la Figure 21, le signe *plus grand que* (>) signifie «est la base des sous-classes suivantes». Seules les instances de classes imprimées en gras sont transférables. Les définitions utiles ne sont pas considérées comme une classe distincte. Elles constituent un simple groupement commode d'attributs utiles.

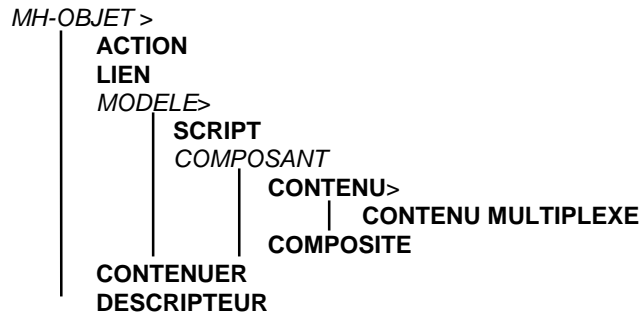


Figure 21/T.171 – Arborescence d'héritage MHEG

## 17 Structure de la classe *MH-objet*

Ce paragraphe décrit la structure commune pour transférer toutes les instances de classes MHEG, c'est à dire les objets MHEG que définit la présente Recommandation. Ce mécanisme donne une approche cohérente de l'identification et du transfert des objets MHEG.

La classe *MH-objet* fournit l'information suivante pour identifier les objets MHEG.

### 17.1 identification de classe

L'identification de la classe comporte l'identification de la norme, sa version et un identificateur spécifique unique pour chaque classe d'objet MHEG. L'attribut *identification de classe* est codé au niveau C à l'aide de la technique ASN.1 d'identificateurs d'objets (voir 8.2.3.1). Pour faciliter le codage, cet attribut n'est pas codé dans la classe *MH-objet* au niveau C. Il est codé individuellement pour chaque classe transférée afin d'attribuer un numéro unique à chaque classe au lieu de l'hériter directement de la classe *MH-objet*.

### 17.2 Identificateur MHEG

Un identificateur MHEG peut être fourni pour identifier de façon unique un objet transféré à l'intérieur d'un domaine de visibilité de l'application.

### 17.3 Information générale sur les objets

La classe *MH-objet* fournit également les informations supplémentaires suivantes sur les objets MHEG:

- 1) nom
- 2) propriétaire
- 3) version
- 4) date de la dernière modification;
- 5) liste des mots clé qualifiant l'objet MHEG;
- 6) information en clair de droit de reproduction de l'objet MHEG;
- 7) identificateur de droit de reproduction utilisé par certains organismes autorisées pour identifier le type d'ouvrage protégé, par exemple:
  - a) ISBN (numéro d'identification international d'ouvrage publié),
  - b) ISSN (numéro d'identification international de publication périodique),
  - c) ISRC (numéro d'identification international attribué d'enregistrement),
  - d) ISAN (numéro d'identification international attribué de produit audiovisuel);

- 8) numéro d'identification utilisé par certaines organisations autorisées pour identifier de façon unique l'information relative au droit de reproduction d'un objet MHEG dans le cadre réservé à l'identificateur: 2-11072557-5 est par exemple un numéro d'identification ISBN;
- 9) information en clair de droit de licence relative à l'objet MHEG;
- 10) priorité d'antémémoire: peut être utilisée par le moteur MHEG comme une indication de gestion des objets MHEG. Il s'agit d'un entier appartenant à l'intervalle 0 à 255. La valeur 0 signifie que le moteur MHEG doit entièrement effacer l'objet de sa mémoire s'il reçoit une action `Destroy` (destruction). La valeur 255 signifie que le moteur MHEG est fortement incité à conserver l'objet MHEG en antémémoire au moment de l'application de l'action `Destroy`. Les autres valeurs seront interprétées selon chaque moteur MHEG en tenant compte des diverses possibilités. Le moteur MHEG n'est toutefois pas obligé de suivre cette instruction. Elle ne lui est fournie que comme indication;
- 11) commentaire en clair sur l'objet MHEG;
- 12) réserve d'extension utilisée par d'autres normes qui décrivent une extension de la présente Recommandation.

## 18 Structure de la classe *action*

La classe *action* définit une combinaison réutilisable d'actions élémentaires. Les objets *actions*, qui sont des instances de la classe *action*, sont utilisés comme des objets *liens* afin de décrire leurs effets de lien. Un objet *action* donné peut être pointé par de nombreux objets *liens*. Plusieurs conditions de liens dans différents liens peuvent être satisfaites au même instant et sont supposées être déclenchées et traitées en parallèle. Le traitement de ces actions a donc un effet local limité au lien lui-même.

La classe *action* MHEG fournit les informations suivantes:

- 1) un indicateur de synchronisation spécifie le type de traitement des actions synchronisées. Deux valeurs sont définies: «parallèle» et «série». Quand cet indicateur est positionné sur «série», toutes les actions de «l'ensemble d'actions synchronisées» sont traitées séquentiellement. Quand il est positionné sur «parallèle», toutes les actions sont traitées en parallèle (voir 31.3). Puisque les actions peuvent être traitées en parallèle, il appartient à l'auteur de tenir compte de tout effet de bord dû au parallélisme. La spécification d'actions contradictoires est déconseillée.

### NOTES

1 – un objet *action* qui applique en parallèle les actions `Run` (exécution) et `stop` (arrêt) à un rt-contenu introduit une ambiguïté puisque l'état final n'est pas défini.

2 – un objet *action* qui applique en série à un rt-contenu les actions `run` puis `stop` avec un délai d'une seconde n'introduit pas d'ambiguïté. Le rt-contenu doit être présenté pendant une seconde.

3 – un moteur MHEG n'est pas tenu d'exécuter effectivement le traitement en parallèle, mais les auteurs doivent organiser le groupe d'actions comme si tel était le cas;

- 2) un ensemble d'actions synchronisées est un ensemble d'objets *actions élémentaires* ou d'autres objets *actions imbriquées*.

En utilisant cette structure, un auteur est capable de décrire les objets *actions* suivants:

- objet *action de base* (voir 18.2);
- objet *action enveloppante* (voir 18.3);
- objet *macroaction* (voir 18.4).

### 18.1 Actions élémentaires

La présente Recommandation définit une liste d'actions élémentaires qui peuvent être incluses dans un objet *action* pour modifier le comportement des entités MHEG (voir la section 4 intitulée *comportement commun aux entités MHEG*), par exemple: `Prepare` (préparation), `Run` (exécution). La structure de chaque action élémentaire est la suivante:

- 1) un ensemble de cibles est défini comme une liste de références génériques ou comme une référence générique unique. Chaque action élémentaire s'appliquera cet ensemble. Si l'ensemble comporte plus d'une cible, l'action élémentaire s'appliquera en parallèle à toutes les cibles de l'ensemble quelle que soit la valeur de l'indicateur de synchronisation. Les auteurs doivent supposer que le moteur MHEG applique l'action élémentaire aux différentes cibles en parallèle.

Un ensemble de cibles commun peut être spécifié pour plusieurs actions élémentaires à l'aide d'un alias (voir 10.3) ou d'un macroparamètre (voir paragraphe 13);

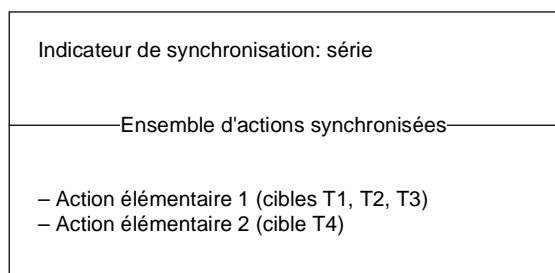
- 2) une durée de transition optionnelle peut être fournie pour certaines actions élémentaires. Elle s'exprime en GTU (unité temporelle générique). Quand une telle durée est spécifiée, l'action élémentaire correspondante doit être exécutée pendant cette durée. Lorsqu'il n'est spécifié aucune durée, cette dernière est considérée comme nulle;
- 3) des paramètres d'actions spécifiques sont définis pour chaque action élémentaire (sections 4, 5, 6, 7 et 8).

## 18.2 Objet *action de base*

Un objet *action de base* ne contient que des actions élémentaires. Il ne contient pas de macroparamètres ou d'objets *actions enveloppantes* à l'intérieur de son ensemble d'actions synchronisées.

NOTE – Il appartient à l'application utilisatrice de déterminer le nombre approprié d'actions élémentaires.

Sur la Figure 22, le moteur MHEG applique en parallèle l'effet de l'action élémentaire MHEG 1 aux cibles T1, T2 et T3. Quand il a terminé, il applique l'effet de l'action élémentaire MHEG 2 à la cible T4.



T0825680-96/d021

Figure 22/T.21 – Exemple d'objet *action de base*

## 18.3 Objet *action enveloppante*

Afin de décrire un comportement plus complexe, un objet *action imbriquée* peut être contenu dans l'ensemble d'actions synchronisées. L'objet *action imbriquée* peut être lui-même un objet *action enveloppante*. Une hiérarchisation complète de ces objets *actions* peut être établie avec ce mécanisme.

NOTE – Il appartient à l'application utilisatrice de déterminer la profondeur d'imbrication appropriée.

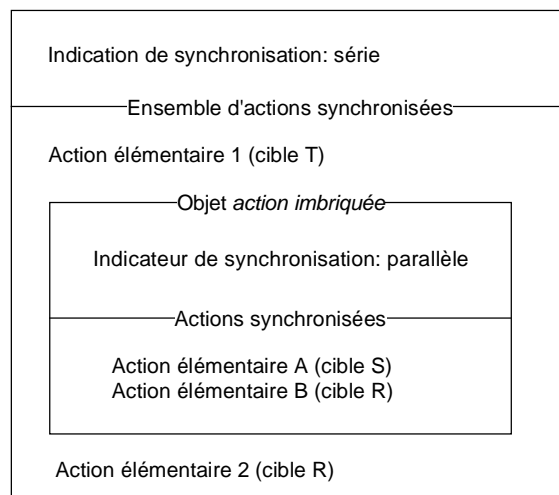
Dans l'exemple de la Figure 23, le moteur MHEG applique l'effet de l'action élémentaire 1 à la cible T. Quand il a terminé, il applique en parallèle l'effet de l'action élémentaire A à la cible S et l'effet de l'action élémentaire B à la cible R. Quand il a terminé, l'effet de l'action élémentaire 2 est appliqué sur la cible R.

## 18.4 Objet *macroaction*

Afin de produire un codage efficace des objets *actions* qui sont fréquemment utilisés et pour lesquels seules quelques valeurs changent d'un lien à l'autre, il est possible de spécifier l'attribut *indicateur de synchronisation* d'un objet *action* et tous les paramètres d'une action élémentaire comme les valeurs d'un macroparamètre. Un objet *macroaction* contient au moins une valeur de macroparamètre. Chacune de ses valeurs est résolue quand l'objet *lien* qui utilise cette action est déclenché (voir 19.3 pour plus de détails).

## 19 Structure de la classe *lien*

Une classe *lien* MHEG est définie pour spécifier des relations spatiales, temporelles et conditionnelles entre les entités MHEG et les actions qu'elles déclenchent. Le lien contient une condition, qui lorsqu'elle est satisfaite est évaluée à Vrai et entraîne l'application d'actions aux cibles qui produisent l'effet requis. On dit que le lien a été déclenché.



T0825690-96/d022

**Figure 23/T.171 – Exemple d'objet *action enveloppante***

Seuls les objets *liens* qui ont été activés sont déclenchables (voir 29.2). Plusieurs conditions de liens dans différents liens peuvent être satisfaites au même moment; elles sont supposées être déclenchées et exécutées en parallèle par le moteur MHEG.

L'objet *lien* est distinct de tous les objets qui produisent les contextes de déclenchement. Le comportement de base de chaque entité MHEG (volume, marche/arrêt d'un canal, par exemple) fait partie de l'entité elle-même. Durant la vie des entités, un changement de comportement peut servir à déclencher un objet *lien* distinct qui commande les actions à exécuter. Ces actions, à leur tour, modifient le comportement de base d'une autre entité MHEG, par exemple, en augmentant le volume. Cette modification peut à son tour générer d'autres changements de comportement et déclencher d'autres liens.

Chaque instance de classe *lien* contient une condition de lien et un effet de lien.

En utilisant cette structure, un auteur peut décrire les types d'objets *liens* suivants:

- objet *lien de base* (voir 19.3);
- objet *lien enveloppant* (voir 19.4);
- objet *macrolien* (voir 19.5).

## 19.1 Condition de lien

La condition de lien est définie comme une condition de déclenchement (voir 19.1.1) ou une combinaison logique de conditions de déclenchements et de contraintes (voir paragraphe 0).

Elle est évaluée de la manière suivante:

- 1) quand un objet *lien* devient actif, sa condition de lien est initialisée à faux;
- 2) si la condition de lien est une condition de déclenchement simple, elle prend la valeur de cette condition. S'il s'agit d'une combinaison logique de conditions, elle prend la valeur du nœud sommital de l'arbre logique combinant les conditions (voir Figure 25);
- 3) si la condition de lien est évaluée à Vrai, le lien est déclenché, c'est à dire que son effet est appliqué par le moteur MHEG (voir 31.2). Si elle est évaluée à Faux ou à 'undefined' (non définie), le lien n'est pas déclenché, c'est à dire que son effet n'est pas appliqué.

Tant qu'il est actif, un lien peut être déclenché plusieurs fois: il le sera en effet chaque fois que sa condition de lien passe à Vrai.

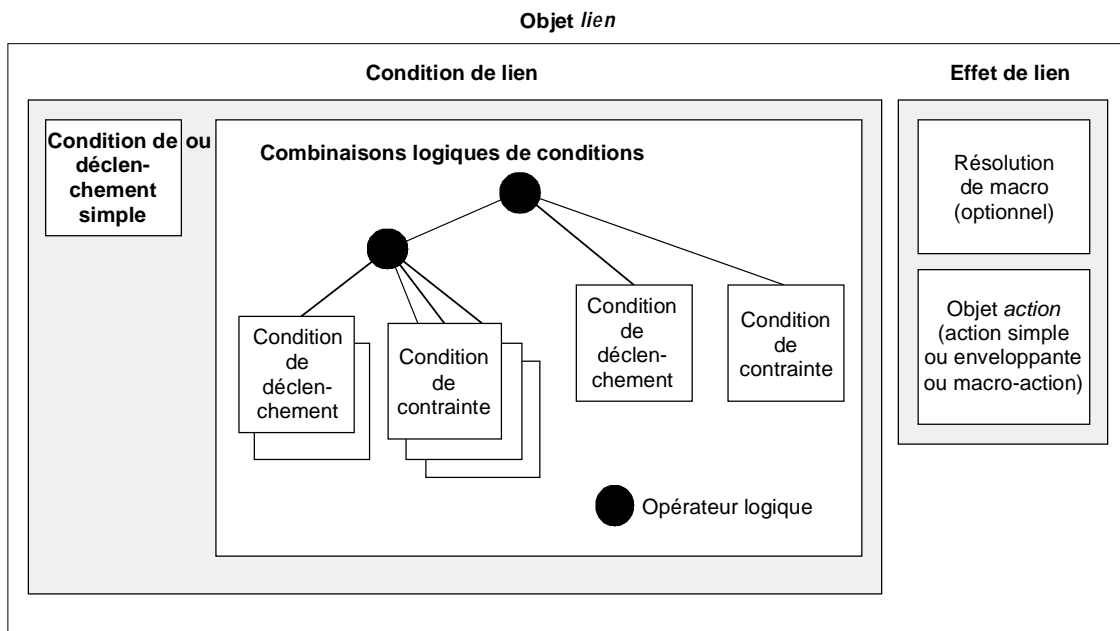


Figure 24/T.171 – Structure d'objet *lien*

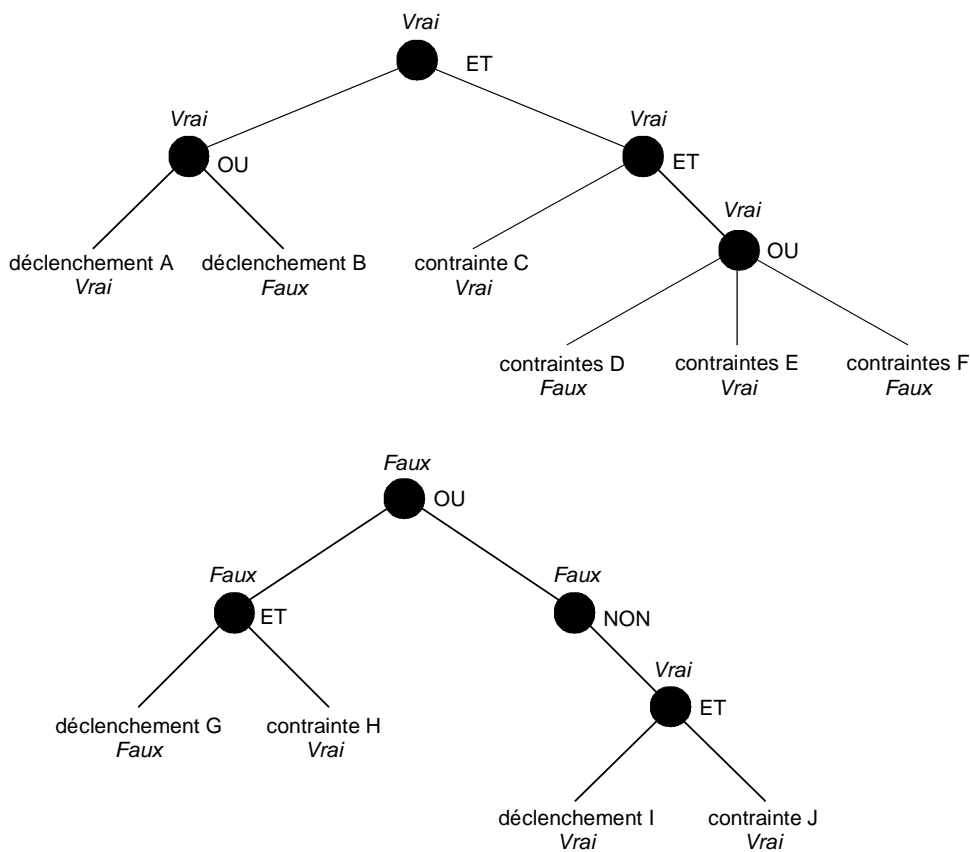


Figure 25/T.171 – Exemples d'arbre logique décrivant une combinaison logique de conditions

### 19.1.1 Condition de déclenchement

Une condition de déclenchement décrit la fréquence de changement de comportement des entités MHEG déclenchant le lien. Le déclenchement est décrit comme un changement de valeur d'attribut ou d'état d'une entité MHEG. Les changements de comportement suivants sont déclenchables:

- 1) changements temporels provoqués par les marqueurs temporels ou le délai;
- 2) changements d'action: conséquences de toute action élémentaire;
- 3) changements d'interaction: surviennent par exemple lorsque l'utilisateur choisit un menu ou clique sur un bouton;
- 4) tout changement d'événement catalogué: survient par exemple sous forme d'un événement système.

La condition de déclenchement s'exprime comme une transition c'est à dire comme une condition double sur la même source. Elle comporte:

- une valeur source;
- une condition précédente qui exprime la condition à satisfaire avant l'apparition de la transition;
- une condition courante qui exprime la condition qui devrait être satisfaite après l'apparition de la transition;

Une condition de déclenchement est évaluée à une valeur booléenne ou à 'undefined'. Lors de l'évaluation, il est procédé de la manière suivante:

- 1) quand un objet *lien* devient actif, chaque condition de déclenchement est évaluée à Faux;
- 2) évaluation de la condition précédente de chaque condition de déclenchement;
- 3) la valeur source de chaque condition de déclenchement est une valeur évaluée sur un attribut ou un état de cible, obtenue par exemple par l'application de l'action d'extraction de l'état de préparation `get preparation status(target)` en plusieurs étapes:
  - a) assignation de cette valeur à la valeur de comparaison courante et évaluation de la condition courante;
  - b) évaluation de la condition de déclenchement par l'exécution de l'opération logique suivante: «condition précédente» ET «condition courante»:
    - i) si le résultat est Vrai et si la condition de déclenchement est définie comme une condition de déclenchement simple, la condition de lien est évaluée à vrai;
    - ii) si le résultat est Vrai et si cette condition de déclenchement fait partie d'une combinaison logique de conditions, cette combinaison doit être évaluée (voir 19.1.8).
  - c) évaluation de la condition de déclenchement à faux;
  - d) passage au point 2.

NOTE – Les étapes 3) a) à 3) d) doivent être considérées comme une même opération atomique de lien, c'est-à-dire que si une valeur testée par une condition de déclenchement change alors que le moteur MHEG exécute les étapes 3) a) à 3) d), ce changement est différé jusqu'au prochain passage à l'étape 3).

Le Tableau 1 montre certains exemples de conditions de déclenchement. Dans ces exemples, la valeur de déclenchement source `Get CV (rt-contenu N)` signifie que l'action `Get CV` – extraction du volume sonore courant (CV) – est appliquée au rt-contenu N afin d'en extraire le volume, l'action `Get Event (rt-contenu N)` signifie que l'action `Get Event` – extraction du numéro d'événement – est appliquée au rt-contenu N afin d'extraire le numéro du dernier événement, et l'action `Get Data (contenu A)` signifie que l'action `Get Data` – extraction de données – est appliquée à l'objet *contenu A* afin d'extraire la valeur générique contenue.

**Tableau 1/T.171 – Exemple de conditions de déclenchement**

Valeur de déclenchement source	Condition précédente		Condition courante		Sémantique	Nature du changement
	Opérateur	Valeur précédente	Opérateur	Valeur courante		
Get CV (rt-contenu N) – extraction du volume sonore courant (CV) du rt-contenu N –	==	10	<	5	le volume CV du rt-contenu N est égal à 10, et décroît à moins de 5	passage d'une valeur spécifiée précédente à une valeur spécifiée courante
Get CV (rt-contenu N)	==	Get Data (contenu A) – extraction des données du contenu A –	<	Get Data (contenu B)	la valeur du volume CV du rt-contenu N est égale à celle spécifiée dans les données de l'objet contenu A, et passe à une valeur inférieure à celle de l'objet contenu B	
Get CV (rt-contenu N)	quelconque	'unspecified'	quelconque	'unspecified'	le volume CV du rt-contenu N a changé. Sa valeur n'est pas significative	passage d'une valeur 'unspecified' (non spécifiée) à une valeur 'unspecified'. Le déclenchement se fait sur changement de la valeur source quelle que soit ses valeurs précédente et courante
Get CV (rt-contenu N)	==	10	quelconque	'unspecified'	le volume CV du rt-contenu N était à 10 et a changé. La nouvelle valeur n'est pas significative	passage d'une valeur spécifiée à une valeur 'unspecified'. Le déclenchement se fait sur changement de la valeur source quelle que soit sa valeur courante
Get CV (rt-contenu N)	quelconque	'unspecified'	>	10	le volume CV du rt-contenu N passe à une valeur supérieure à 10. La valeur précédente n'est pas significative. A noter que la valeur précédente pouvait être supérieure à 10	passage d'une valeur précédente quelconque à un intervalle de valeurs courantes spécifié
Get CV (rt-contenu N)	quelconque	'unspecified'	==	10	le volume CV du rt-contenu N passe à 10. La valeur précédente n'est pas significative. A noter que la valeur précédente pouvait être égale à 10	passage d'une valeur 'unspecified' à une valeur courante spécifiée. Le déclenchement se fait sur changement de la valeur source quelle que soit sa valeur précédente
Get CV (rt-contenu N)	<	Get CV (rt-contenu P)	>	Get CV (rt-contenu P)	le volume CV du rt-contenu N était inférieur à celui du rt-contenu P et il est passé à une valeur supérieure	changement par rapport à d'autres rt-contenus
Get Event (rt-contenu N)	quelconque	'unspecified'	==	10	l'événement 10 est survenu sur le rt-contenu N. La valeur de l'événement précédent n'est pas significative	devient Vrai dès que l'événement 10 survient
Get Event (rt-contenu N)	==	10	==	10	l'événement précédent survenu sur le rt-contenu N était 10. Le même événement survient encore.	devient Vrai à chaque fois que l'événement 10 se reproduit
Get CV (rt-contenu N)	la précédente condition est omise: elle est alors la négation de la condition courante		>=	5	le volume CV du rt-contenu N était inférieur à 5 et il passe à une valeur supérieure ou égale à 5. La condition précédente est interprétée comme $CV < 5$ . L'intervalle de valeur précédente est $(-\infty, 4)$ , l'intervalle de valeurs courantes est $(5, +\infty)$ . Ces deux intervalles de valeurs ne se chevauchent pas et leur réunion est l'ensemble des entiers	passage d'un intervalle de valeurs précédentes à un intervalle de valeurs courantes où les deux intervalles ne se chevauchent pas et où la réunion de ces deux intervalles est l'intervalle le plus large possible. Dans ce cas, la condition précédente est omise et est interprétée comme la négation de la valeur courante

### 19.1.2 Condition de contrainte

Une condition de contrainte peut apparaître dans une combinaison logique de conditions. Elle exprime un état contextuel requis au moment de la satisfaction des conditions de déclenchement. Elle permet de spécifier plus précisément le contexte dans lequel la condition de lien sera satisfaite.

Une condition de contrainte comporte une valeur source et une condition courante qui exprime la condition que l'on devrait satisfaire.

NOTE – Les contraintes ne nécessitent pas de condition précédente parce qu'elles expriment des états conceptuels existants au moment du déclenchement.

Une condition de contrainte est évaluée à une valeur booléenne ou à la valeur 'undefined'. Elle est satisfaite lorsque la condition courante spécifiée est satisfaite pour la valeur source. Les conditions de contraintes doivent être évaluées à l'instant où l'une des conditions de déclenchement devient Vraie.

Le Tableau 2 montre certains exemples de conditions de contraintes.

Tableau 2/T.171 – Exemples de conditions de contraintes

Valeur source	Condition courante		Sémantique
	Opérateur	Valeur courante	
Get CV (rt-contenu N) – extraction du volume sonore courant du rt-contenu N –	<	5	la valeur du volume CV du rt-contenu N est inférieure à 5
Get CV (rt-contenu N)	<	Get Data (contenu A) – extraction des données du contenu A –	la valeur du volume CV du rt-contenu N est inférieure à celle de l'objet contenu A
Get CV (rt-contenu N)	==	5	la valeur du volume CV du rt-contenu N est égale à 5

### 19.1.3 Valeur source

La valeur source est utilisée comme base de la comparaison décrite dans la condition précédente et la condition courante. La présence de cette valeur dans la condition est obligatoire.

La valeur source est toujours fournie par une valeur évaluée, c'est à dire qu'elle s'exprime comme un résultat d'action `get` (extraction).

### 19.1.4 Valeur de comparaison

La valeur de comparaison est la valeur qui doit être comparée à la valeur source à l'aide de l'opérateur relationnel de condition précédente et de la condition courante. La présence de cette valeur dans la condition est obligatoire.

La valeur de comparaison est spécifiée de la manière suivante:

- 1) valeur évaluée: exprimée comme un résultat d'une action `get` (extraction);
- 2) valeur constante;
- 3) une des constantes de comparaison que la présente Recommandation définit, par exemple 'ready' (prêt) et 'not ready' (non prêt) pour l'état de préparation;
- 4) valeur 'unspecified' pour indiquer que la valeur à comparer avec la source est sans importance; ce qui est important c'est que la valeur a changé. Les conditions suivantes s'appliquent:
  - a) à l'intérieur d'une condition de déclenchement:
    - i) à l'intérieur de la condition précédente: la condition de déclenchement se fait sur changement de la valeur source quelle que soit sa valeur précédente;
    - ii) à l'intérieur de la condition courante: la condition de déclenchement se fait sur changement de la valeur source quelle que soit sa nouvelle valeur courante;



- iii) à l'intérieur à la fois de la condition précédente et de la condition courante: la condition de déclenchement se fait sur un changement de valeur quelle que soient sa valeur précédente et sa nouvelle valeur courante.

La valeur courante peut être la même que la valeur précédente, mais le moteur a «volontairement» positionner la même valeur sur un attribut ou un état. Cette caractéristique est particulièrement utile lorsqu'apparaît à plusieurs reprises le même événement (voir 6.3.7) ou marqueur temporel (voir 72.16.14), c'est à dire que le même événement peut survenir plusieurs fois et à chaque fois, l'identificateur d'événement est positionné sur la valeur correspondante;

- b) à l'intérieur d'une condition de contrainte: cela signifie que la condition de contrainte est toujours évaluée à Vrai.

### 19.1.5 Condition précédente

La condition précédente n'est utilisée que dans la condition de déclenchement. Elle comporte un opérateur relationnel et une valeur de comparaison précédente.

La valeur de comparaison précédente est une valeur générique (voir paragraphe 11).

Quand la condition précédente est omise, elle doit être interprétée comme la négation de la condition courante indiquée, et les conditions suivantes s'appliquent:

- l'opérateur relationnel de la condition précédente est la négation de l'opérateur relationnel de la condition courante;
- la valeur de comparaison de la condition précédente est la même que celle de la condition courante.

La condition précédente est satisfaite quand le résultat de l'opération de comparaison, avec l'opérateur relationnel spécifié, entre la valeur source et la valeur de comparaison précédente est évaluée à Vrai.

### 19.1.6 Condition courante

La condition courante est obligatoire dans une condition générique et comporte un opérateur relationnel et la valeur de comparaison courante.

La valeur de comparaison courante est une valeur générique (voir paragraphe 11).

La condition courante est satisfaite lorsque le résultat de la comparaison, avec l'opérateur relationnel spécifié, entre la valeur source et la valeur de comparaison courante est évalué à Vrai.

### 19.1.7 Opérateur relationnel

L'opérateur relationnel utilisé dans les conditions précédente et courante est l'un des suivants:

- == test d'égalité
- != test de non égalité
- < test d'infériorité stricte
- <= test d'infériorité
- > test de supériorité stricte
- >= test de supériorité

NOTE 1 – Une application utilisatrice peut fournir d'autres opérateurs relationnels.

Un test relationnel compare la valeur source à une valeur de comparaison. Le résultat de la comparaison est défini dans le Tableau 3.

**Tableau 3/T.171 – Les comparaisons**

Opérateur relationnel	Valeur source (SV)	Valeur de comparaison (CV)	Valeur de résultat (Booléen ou 'undefined')	Remarques
==	* Booléen générique * Terme numérique générique * Entier générique * Fraction générique * Chaîne générique * Référence générique * Liste générique	* Booléen générique * Terme numérique générique * Entier générique * Fraction générique * Chaîne générique * Référence générique * Liste générique	Vrai: si SV == CV Faux: autrement	voir 1) voir 2) voir 3) voir 4)
!=	* Booléen générique * Terme numérique générique * Entier générique * Fraction générique * Chaîne générique * Référence générique * Liste générique	* Booléen générique * Terme numérique générique * Entier générique * Fraction générique * Chaîne générique * Référence générique * Liste générique	Vrai: si SV != CV Faux: autrement	voir 1) voir 2) voir 3) voir 4)
<	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	Vrai: si SV < CV Faux: autrement	voir 1) voir 4)
<=	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	Vrai: si SV <= CV Faux: autrement	voir 1) voir 4)
>	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	Vrai: si SV > CV Faux: autrement	voir 1) voir 4)
>=	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	* Terme numérique générique * Entier générique * Fraction générique * Liste générique de termes numériques / entiers / fractions génériques	Vrai: si SV >= CV Faux: autrement	voir 1) voir 4)
quelconque	* quelconque	* Le type CV est différent du type SV.	Faux	voir 5)
quelconque	* type SV quelconque * 'undefined' * 'undefined' * 'undefined'	* 'undefined' * type CV quelconque * 'unspecified' * 'undefined'	'undefined'	voir 6)
quelconque	* type SV quelconque	* 'unspecified'	Vrai	voir 7)

La présente Recommandation ne définit pas la sémantique des opérations appliquées à des types de valeurs différents de ceux présentés dans le Tableau 3. Si une telle opération doit être traitée par le moteur MHEG et si aucune sémantique n'a été prévue par l'application utilisatrice, cette opération est évaluée à la valeur 'undefined'.

NOTE 2 – True > False and «Hello» <= «Bye!» sont évalués à 'undefined'.

Afin d'évaluer le résultat de certaines opérations, les conditions suivantes s'appliquent:

- 1) deux fractions génériques seront comparées de la manière suivante:
  - a) si le dénominateur est omis, il est supposé être égal à 100. Par exemple: (10/100) == (10/omission) est Vrai;

- b) toute fraction générique sera réduite si elle n'est pas irréductible. Cette réduction ne doit être faite qu'à des fins de comparaison. La valeur de la fraction générique reste la même qu'avant la réduction. Par exemple  $(10/100) == (1/10)$  pour la comparaison;
- c) toute fraction générique est considérée comme une liste contenant deux entiers génériques. Ainsi, la méthode de comparaison de deux listes s'applique (voir ci-dessous).
- 2) La comparaison entre chaînes est effectuée au niveau de chaque caractère. Les majuscules se distinguent des minuscules.
- NOTE 3 – Cette comparaison est similaire la comparaison de deux listes (voir ci-dessous). Il y a égalité quand les chaînes sont de même longueurs et que les caractères de même rang sont identiques.
- NOTE 4 – Exemples: Le résultat de la comparaison des chaînes «HELLO» == «HELLO» est Vrai. Celui des chaînes «hello» == «HELLO» est Faux. Celui des chaînes «Hello» != «Hello» est Faux.
- 3) La comparaison de deux références s'effectue sur les entités adressées. Deux références différentes peuvent pointer la même entité. Dans ce cas, le test d'égalité est évalué à Vrai. Une comparaison peut être faite entre les références adressant un groupe d'entités. Dans ce cas, la comparaison est faite au niveau des entités adressées. Le test d'égalité entre deux groupes n'ayant pas le même nombre d'entités est évalué à Faux. Lorsque le moteur MHEG est incapable de comparer deux références, le résultat de la comparaison est évalué à 'undefined'.
- NOTE 5 – Cette situation peut apparaître lorsque deux références ne sont pas encore résolues par le moteur MHEG ou quand une référence adresse un objet inconnu.
- NOTE 6 – Par exemple: l'objet *contenu A* a un identificateur MHEG de valeur 1000 et son identificateur externe est: «e:\mydir\contenu.mhg».  $1000 == \langle e:\mydir\contenu.mhg \rangle$  est Vrai. Si un alias, par exemple *contenu A*, est affecté à ce contenu, *Contenu A* == 1000 est Vrai, *Contenu A* != «e:\mydir\contenu.mhg» est Faux.
- 4) Deux listes seront comparées de la manière suivante:
- a) si elles ont des longueurs différentes (c'est à dire un nombre différent d'éléments) le résultat de comparaison est évalué à Faux;
- b) si elles ont la même longueur (c'est à dire le même nombre d'éléments), la comparaison est effectuée élément par élément. Deux éléments de même rang dans les deux listes sont comparés à l'aide de l'opérateur relationnel:
- i) si une des comparaisons est évaluée à Faux au niveau des éléments, la comparaison au niveau de la liste est évaluée à Faux;
- ii) si une des comparaisons au niveau des éléments est évaluée à 'undefined', la comparaison au niveau des éléments est évaluée à 'undefined'.
- iii) Autrement la comparaison au niveau de la liste est évaluée à Vrai; Un ET logique est exécuté entre le résultat de chaque comparaison.
- c) deux listes de longueur zéro sont considérées comme strictement égales. Par exemple: comparaison de listes génériques de termes numériques génériques  $(2, 3, 5) == (2, 3)$  est Fausse,  $(2, 3, 5) == (2, 3, 5)$  est Vraie,  $(2, 3, 5) == (2, 4, 5)$  est Fausse et  $(2, 3, 5) > (0, 1, 6)$  est Fausse parce que  $6 > 5$ ,  $() <= ()$  est Vrai.
- 5) Le résultat de la comparaison entre deux types de valeurs non compatibles est toujours évalué à Faux. Par exemple: comparaison de valeurs non compatibles:  $3 != \langle a \rangle$  est fausse et  $5 != \langle a \rangle$  est fausse.
- 6) La comparaison de deux valeurs donne toujours la valeur 'undefined' si une des valeurs est égale à 'undefined'. Par exemple:  $(2, 3, 5) == (2, \text{undefined}, 5)$  est 'undefined',  $(2, 3, 5) > (0, 1, \text{undefined})$  est 'undefined',  $\text{'undefined'} == \text{'undefined'}$  est 'undefined' et  $\text{'undefined'} == \text{'unspecified'}$  est 'undefined'.
- 7) Une valeur de comparaison donnée peut être spécifiée par l'auteur comme 'unspecified' à la place de toute valeur de comparaison. La comparaison d'une valeur avec une valeur 'unspecified' est toujours Vrai excepté si la valeur à spécifier est 'undefined'.
- NOTE 7 – Par exemple:  $(2, 5, 7) <= (2, \text{unspecified}, \text{unspecified})$  est Vrai,  $(2, 5, 7) >= (2, \text{unspecified}, \text{unspecified})$  est Vrai, «Hello» == unspecified est Vrai et 'undefined' <= 'unspecified' est 'undefined'.
- NOTE 8 – Bien distinguer entre les valeurs 'undefined' et 'unspecified'. La valeur 'undefined' n'est pas fournie par un auteur. Il s'agit toujours d'un résultat d'une action `get` (extraction) avec une erreur ou d'un résultat de comparaison contenant une valeur source ou une valeur de comparaison évaluée à 'undefined'. La valeur 'unspecified' est une omission volontaire de l'auteur.

### 19.1.8 Combinaison logique

Une combinaison de conditions logiques est définie de la manière suivante:

- 1) opérateur logique: à appliquer à la liste de conditions;
- 2) liste de conditions: chaque condition est une condition de déclenchement, une condition de contrainte ou une autre combinaison logique de conditions. Il s'agit d'une structure récursive. Si l'opérateur NON est utilisé, la liste ne contiendra qu'une condition. Si d'autres opérateurs sont utilisés, la liste contiendra au moins deux conditions.

Une combinaison logique de conditions a la forme d'un arbre logique, dont les feuilles sont des conditions de déclenchement ou de contrainte. Un opérateur logique est associé à chaque nœud.

Les deux arbres décrits sur la Figure 25 montrent différentes combinaisons de conditions. Tous les types de combinaisons sont possibles. Certaines combinaisons n'ont pas de signification, par exemple OU (condition de déclenchement K, condition de déclenchement L, condition de contrainte M). Dans ce cas, la condition de contrainte est inutile.

L'évaluation de la combinaison de conditions logique est la suivante:

- 1) chaque condition de déclenchement est évaluée comme le spécifie le 19.1.1;
- 2) chaque fois que l'une des conditions de déclenchement devient vraie, les conditions suivantes s'appliquent:
  - a) chaque condition de contrainte est évaluée conformément aux spécifications du paragraphe 0
  - b) puis chaque nœud est évalué de bas en haut de la manière suivante:
    - i) l'opérateur logique associé au nœud s'applique sur les conditions évaluées associées à ce nœud, comme le spécifie le 19.1.9;
    - ii) chaque nœud évalue le résultat de l'opération logique précédente, c'est à dire valeur booléenne ou 'undefined';
- 3) la condition de lien évalue la valeur du nœud sommital, c'est à dire la racine de l'arbre logique de la Figure 25.

### 19.1.9 Opérateur logique

L'opérateur logique sert à l'intérieur d'une opération logique. On définit les opérateurs suivants:

- 1) ET: intersection logique,
- 2) OU: réunion logique,
- 3) XOR: ou logique exclusif,
- 4) NON: négation logique

Une opération logique se compose d'un opérateur logique et d'une liste d'opérandes. Son résultat s'évalue de la manière suivante lorsque l'opérateur NON est utilisé:

- 1) si l'opérateur NON est appliqué à un opérande, le résultat est le suivant:

a) NON Vrai	évalué à: Faux
b) NON Faux	évalué à: Vrai
c) NON 'undefined'	évalué à: 'undefined'
- 2) dans les autres cas, le résultat est 'undefined'

Pour les autres opérateurs, le résultat de l'opération logique est évalué de la manière suivante:

- 1) si l'opérateur ET, OU ou XOR est appliqué à n opérandes, n supérieur à 2, l'opération logique se décompose en n-1 opérations, qui sont évalués de gauche à droite.  
Exemple: ET(c1, c2, c3, ..., cn-1, cn) se décompose en ET(...ET(ET(ET(c1, c2), c3), ...), cn);
- 2) si l'opérateur ET est appliqué à deux opérandes, le résultat est le suivant:

a) Vrai ET Vrai	évalué à: Vrai
b) Faux ET Faux	évalué à: Faux
c) 'undefined' ET 'undefined'	évalué à: 'undefined'
d) Vrai ET Faux	évalué à: Faux
e) Vrai ET 'undefined'	évalué à: 'undefined'
f) Faux ET 'undefined'	évalué à: 'undefined'

- 3) si l'opérateur OU est appliqué à deux opérandes, le résultat est le suivant:
  - a) Vrai OU Vrai évalué à: Vrai
  - b) Faux OU Faux évalué à: Faux
  - c) 'undefined' OU 'undefined' évalué à: 'undefined'
  - d) Vrai OU Faux évalué à: Vrai
  - e) Vrai OU 'undefined' évalué à: Vrai
  - f) Faux OU 'undefined' évalué à: Faux
- 4) si l'opérateur XOR est appliqué à deux opérandes, le résultat est le suivant:
  - a) Vrai XOR Vrai évalué à: Faux
  - b) Faux XOR Faux évalué à: Faux
  - c) 'undefined' XOR 'undefined' évalué à: 'undefined'
  - d) Vrai XOR Faux évalué à: Vrai
  - e) Vrai XOR 'undefined' évalué à: 'undefined'
  - f) Faux XOR 'undefined' évalué à: 'undefined'
- 5) Si l'opérateur ET, OU ou XOR est appliqué à un opérande, le résultat est 'undefined'.

## 19.2 Effet de lien

L'effet de lien contient l'information suivante:

- 1) une macrorésolution optionnelle utilisée pour affecter une valeur d'utilisation à chaque macroparamètre. La macrorésolution est définie comme une liste de résolutions de macroparamètre. Chaque résolution est définie de la manière suivante:
  - a) un identificateur de définition de macroinstruction identifie un des macroparamètres;
  - b) la macrovaleur utilisée généralement est spécifiée comme valeur générique.
- 2) une action: objet *action de base* (voir 18.2), objet *action enveloppante* (voir 18.3) ou un objet *macroaction* (voir 18.4).

L'effet de lien est traité conformément au 31.2 quand la condition de lien est satisfaite, c'est-à-dire qu'elle est évaluée à Vrai.

## 19.3 Objet lien de base

Un objet *lien de base* est défini comme une combinaison de conditions de lien de base et d'un effet de lien de base. Les combinaisons peuvent former un objet *lien* simple ou plus complexe.

Une condition de lien de base ne contient pas des macroparamètres. Les conditions de liens de base suivantes peuvent être définies (avec TC = condition de déclenchement, CC = condition courante):

- 1) TC;
- 2) combinaison logique simple de conditions; exemple: TC ET CC, TC1 OU TC2;
- 3) combinaison logique complexe de conditions TC et CC; exemple:  
NON(((TC1 ET CC1) OU (TC2 ET NON (CC3 ET CC4 ET CC5)) XOR (TC3 ET CC6));

Un effet lien de base ne contient pas d'actions Activate (activation) (appliquées aux objets *liens*). Les effets lien de base suivants peuvent être définis:

- 1) pas de résolution de macroparamètre et un objet *action de base*;
- 2) pas de résolution de macroparamètre et un objet *action enveloppante*;
- 3) résolution d'un ensemble de macroparamètres et une action *macro*.

## 19.4 Objet lien enveloppant

Pour décrire des comportements plus complexes, l'auteur peut définir un objet *lien enveloppant*. Un tel objet contient une action élémentaire Activate à l'intérieur de son effet de *lien*. La cible de cette action élémentaire est appelée objet *lien imbriqué*. Quand le *lien enveloppant* est déclenché, le *lien imbriqué* est activé.

L'objet *lien imbriqué* peut être lui-même un objet *lien enveloppant*. Un tel mécanisme permet de réaliser une arborescence complète d'objets *liens*.

NOTE – Considérons par exemple les objets *liens* L1, L2 et L3:

- L1 a l'effet LE1 défini par (exécuter Image1, activer L2);
- L2 a l'effet LE2 défini par (exécuter Image2, activer L3);
- L3 a l'effet LE3 défini par (exécuter Contenu6).

Si la condition de lien de L1 est satisfaite, la présentation de l'image commence et l'objet imbriqué L2 est activé par l'effet LE1. Si la condition de lien de L2 est satisfaite après l'activation de L2, la présentation de l'image2 commence et l'objet imbriqué L3 est activé par l'effet LE2. Et ainsi de suite.

Les liens L2 et L3 sont dits imbriqués du point de vue de L1. Le lien L3 est dit imbriqué du point de vue de L2.

## 19.5 Objet *macrolien*

Pour produire un codage efficace des objets *liens* fréquemment utilisés et pour lesquels seules quelques valeurs changent d'une condition de lien à une autre, il est possible de spécifier un objet *macrolien*. Un tel objet est un objet *lien imbriqué* qui contient au moins une valeur de macroparamètre à l'intérieur de sa condition de lien. Chaque valeur du macroparamètre est résolue quand elle est activée à partir d'un objet *lien enveloppant* (voir 6.3.13).

NOTE – Il est nécessaire que l'objet *macrolien* soit un objet *lien imbriqué*, autrement la valeur du macroparamètre ne sera jamais résolue, et la condition de lien décrite ne sera jamais satisfaite.

## 20 Structure des classes *modèles*

Les objets *modèles*, instances des classes *modèles*, peuvent être transférés à l'intérieur des applications utilisatrices ou entre elles. Un objet *modèle* est considéré comme un objet squelette. A partir de ce modèle, les rt-objets peuvent être créés en fonction d'instructions données par l'auteur.

La classe *modèle* est une classe abstraite dont héritent les classes *script* et *composant*.

Un nombre quelconque de rt-objets peut être créés à partir d'un même objet *modèle*. L'activation d'un rt-objet n'affecte pas l'objet *modèle*. Cela autorise la réutilisation du même objet *modèle* dans différents contextes, c'est à dire, différents rt-objets.

La présente Recommandation ne définit pas la représentation interne des rt-objets. Chaque moteur MHEG aura sa propre technique de représentation interne.

NOTE – La classe *modèle* ne contient pas d'attribut de transfert.

## 21 Structure de la classe *script*

Une classe *script* MHEG est définie pour spécifier des actions conditionnelles complexes sur les entités MHEG. Les objets *script*, instances de la classe *script*, peuvent être transférés à l'intérieur des applications utilisatrices ou entre elles. Un objet *script* est comme un objet *modèle*. Il est possible de créer plusieurs rt-scripts à partir de ce modèle conformément aux instructions de l'auteur.

La classe *script* MHEG fournit les informations suivantes pour le transfert des scripts:

- classification optionnelle des scripts (voir le paragraphe 14): aide à déterminer le type des données *script*;
- crochet de script (voir le paragraphe 14): identifie le langage d'écriture de script et décrit l'information de codage et de décodage permettant l'utilisation des données *script*. Il comporte une identification et une description du langage d'écriture de script;
- données script: inclusion ou référence à un script codé ou des données vides 'Null-data'.

## 22 Structure de la classe *composant*

La classe *composant* est une classe abstraite dont les classes *contenu* et *composite* héritent.

Les objets *composants*, instances de la classe *composant*, peuvent être transférés à l'intérieur des applications utilisatrices ou entre elles.

Un objet *composant* est comme un objet *modèle*. Il est possible de créer plusieurs rt-composants à partir de ce modèle conformément aux instructions de l'auteur.

En plus des attributs hérités de la classe *modèle*, la classe *composant* définit un attribut optionnel «d'initialisation de l'espace de présentation d'origine (OPS)»; celui-ci comporte une durée d'origine (OD) et une taille d'origine (OS). Il sert à initialiser l'espace OPS de chaque rt-composant créé à partir de cet objet *composant*. La durée OD et la taille OS servent à initialiser les longueurs des axes temporel et spatiaux de l'espace OPS. La plage du volume sonore AVR est positionnée sur l'intervalle de valeurs défini par la présente Recommandation. La durée OD et la taille OS sont fournies au moteur MHEG comme aide optionnelle. Il appartient au concepteur de l'objet de s'assurer que cette information est compatible avec les informations similaires susceptibles d'être contenues dans le crochet ou les données.

## 23 Structure de la classe *contenu*

Ce paragraphe décrit comment le traitement des données codées figurant dans la présente Recommandation est pris en charge. Ce mécanisme est commun à tous les types de données codées et prévoit une approche cohérente pour transférer les informations suivantes et en identifier le type:

- données qui peuvent être présentées à l'utilisateur et perçues par ce dernier, par exemple: données graphiques ou audiophoniques;
- données qui ne peuvent pas être présentées à l'utilisateur d'une manière directe;
- données vides qui peuvent ou non être présentées à l'utilisateur en fonction de l'information donnée dans le crochet, les attributs de classification ou les styles appliqués: zone transparente encadrée d'un rectangle par exemple;
- valeur générique mémorisée dans les données. Cette valeur peut être présentée à l'utilisateur en fonction de l'information donnée dans le crochet, les attributs de classification ou les styles appliqués: un terme numérique peut par exemple être présenté comme un nombre pur ou un curseur, une chaîne littérale comme une chaîne ou un bouton. Cette valeur peut aussi être utilisée pour mémoriser une valeur à extraire ultérieurement, pour une comparaison dans une condition de lien ou pour une autre assignation d'attribut par exemple. Des actions `get data` (extraction de données) et `set data` (écriture de données) spécifiques permettent d'extraire et de modifier cette valeur, par exemple `set CV – [ensemble de cibles, Get Data (cible contenu) -désignation du volume sonore courant (ensemble de cibles, extraction des données de la cible contenu)]`.

Un objet *contenu* est un objet *modèle*. Il est possible de créer plusieurs rt-contenus à partir de ce modèle conformément aux instructions de l'auteur.

La classe *contenu* MHEG est définie et elle fournit les informations suivantes pour transférer les données codées et en identifier le type:

- classification optionnelle des données: aide à déterminer le type des données. La classification est hors norme ou enregistrée: valeur générique, graphique ou vidéo (voir le paragraphe 14).
- crochet de contenu: décrit l'information de codage et de décodage permettant l'utilisation des données codées (voir le paragraphe 14).
- volume sonore d'origine (OV) optionnel: aide à déterminer le volume sonore courant codé dans les données. Une fois fourni, ce volume sera défini à l'intérieur de la plage de volume sonore (AVR) (voir 9.1.3).
- données de contenu: inclusion ou référence à valeur générique, des données codées fournies par d'autres Recommandations ou normes et par la présente Recommandation, des données vides «null-Data». La valeur générique de la présente Recommandation est codée en ASN.1 avec la syntaxe de valeur générique définie dans le module de définitions utiles.

## 24 Structure de la classe *contenu multiplexé*

Ce paragraphe décrit comment le traitement des données multiplexées est pris en charge par la présente Recommandation. Ce mécanisme est une extension aux mécanismes de la classe *contenu*.

Une classe *contenu multiplexé* MHEG fournit une liste de trains de données ordonnée en plus de l'information de la classe *contenu*. Elle décrit les trains contenus dans les données multiplexées.

Chaque train comporte les informations suivantes:

- 1) identificateur de train de données: chaque train a un identificateur unique dans un multiplex;
- 2) classification *contenu* catalogué optionnelle: aide à déterminer le type de train de données (voir le paragraphe 14);
- 3) crochet de *contenu*: décrit l'information de codage et de décodage permettant l'utilisation de ce train de données (voir le paragraphe 14).

NOTE – Exemple d'objet *contenu multiplexé*:

- *crochet de données*:  
système MPEG
- *données*:  
données multiplexées
- *train de données*:  
identificateur de train 2.1, crochet de train de données MPEG2-Vidéo  
identificateur de train 2.4, crochet de train de données MPEG2-Audio  
identificateur de train 2.5, crochet de train de données MPEG2-Audio

## 25 Structure de la classe *composite*

Cette classe décrit comment l'association des objets hyper et multimédias est prise en charge. Le mécanisme décrit donne une approche cohérente de la synchronisation temporelle et spatiale et de la liaison d'un ensemble d'objets.

Cette classe prévoit aussi la structure logique permettant de décrire la liste des interactions possibles offertes à l'utilisateur mais elle ne traite pas des fonctionnalités d'interaction fournies par l'interface utilisateur. Une telle interaction peut être réalisée de différentes façons: interfaces graphiques d'utilisateur (GUI) ou clavier etc. La présente Recommandation ne définit pas le rendu et la convivialité des présentations multimédia interactives; elle ne propose pas non plus de modifier des concepts existants ou d'en ajouter de nouveaux par rapport à ceux des interfaces GUI types. Comme la présente Recommandation est générique et indépendante de la plate-forme et de la mise en œuvre, elle décrit l'interaction à un niveau virtuel. Il appartient à l'application utilisatrice de mettre en œuvre ces mécanismes en ayant recours à ses propres caractéristiques de rendu et de convivialité.

NOTE 1 – De cette manière, la présente Recommandation permet de réaliser une interface homogène entre un moteur MHEG et les services d'interaction d'utilisateur ou d'autres applications, tout en permettant encore à l'auteur de conserver le rendu et la convivialité.

Chaque objet *composite* fournit le moyen de décrire une génération d'information et son comportement. Chaque génération comporte une liste d'éléments. L'élément est la brique élémentaire de la classe *composite*. Chaque élément fournit également un moyen de pointer vers une autre génération définie par un autre objet *composite*. Une structure récursive est alors construite en utilisant de façon répétitive l'objet *composite* et les pointeurs figurant dans ces éléments.

Chaque génération représente une unité dans la logique de présentation voulue par l'auteur.

Il existe une relation intrinsèque type entre les éléments de la fratrie d'une même génération.

Il existe également une relation intrinsèque de propagation type entre générations.

NOTE 2 – Une utilisation typique d'un objet *composite* est de définir un élément pour chaque choix à offrir à l'utilisateur dans le menu principal. Un deuxième *composite* sert à décrire les choix des sous-menus.

NOTE 3 – Un objet *composite* sert aussi à définir les éléments à présenter simultanément.

La classe *composite* MHEG fournit les informations suivantes en plus d'une identification générique de la composition:

- 1) comportement de composition: cet ensemble de comportements peut décrire le séquençement des éléments de ce *composite* et leurs interrelations ainsi que les hyperliens qui existent avec d'autres types d'informations. Le comportement de composition contient les informations suivantes:
  - a) comportement prédéfini servant à spécifier les comportements de départ. Le démarrage et l'arrêt sont définis de la manière suivante:
    - i) démarrage de disponibilité (*availability start-up*): lorsque l'objet *composite* est prêt;
    - ii) arrêt de disponibilité (*availability close-down*): lorsque l'objet *composite* est détruit;



- iii) démarrage de rt-disponibilité (`rt-availability start-up`): quand un `rt-composite` est créé à partir de l'objet `composite`;
- iv) arrêt de rt-disponibilité (`rt-availability close-down`): quand un `rt-composite` est supprimé;
- b) ensemble de liens pour la spécification des actions conditionnelles;
- c) ensemble d'actions: les actions et les liens servent à décrire le comportement de chaque élément et les relations parents-enfants;
- 2) nombre d'éléments contenus dans la composition définie par cet objet composite;
- 3) éléments de composition. Chaque composite définit une génération. Une génération comporte une liste de tous les éléments de la fratrie.

Les éléments vides ne sont pas représentés sur la Figure 26.

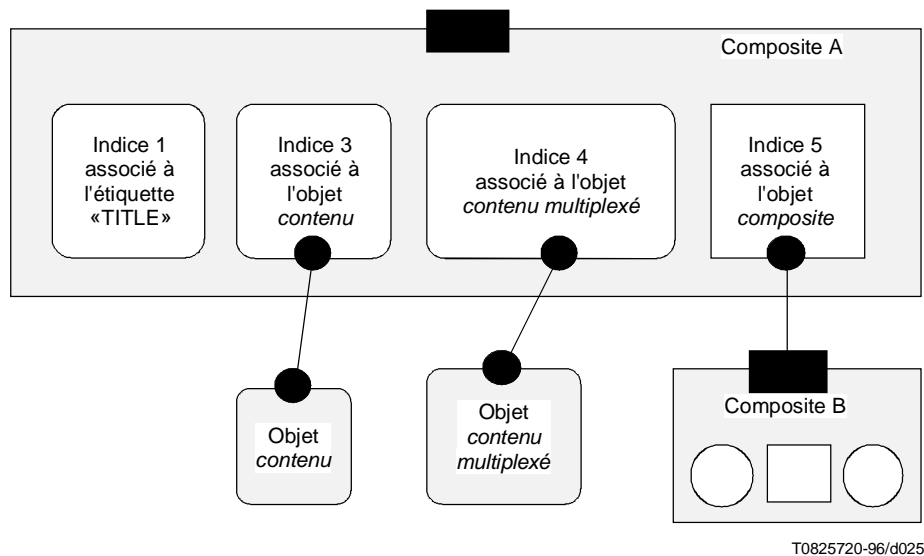


Figure 26/T.171 – Exemple de structure d'objet composite

## 25.1 Démarrage de disponibilité (`availability start-up`)

Le démarrage de disponibilité est un objet *lien* qui a toujours la même condition de lien: passage de l'objet `composite` à l'état de préparation 'ready' – prêt – [cela signifie que l'action `Get Preparation Status` – extraction de l'état de préparation («cet-objet») – donne le résultat prêt]. L'effet de lien peut servir pour des effets de préparation supplémentaires du composite. Il s'agit d'un effet de lien personnalisé fourni par l'utilisateur ou de l'un des effets de lien par défaut suivants:

- démarrage automatique 1: active tous les liens inclus ou pointés par référence dans un comportement donné;
- démarrage automatique 2: prépare tous les objets modèles associés pointés par référence;
- démarrage automatique 3: prépare tous les objets *actions* et *liens* pointés par référence dans un comportement donné;
- démarrage automatique 4: active tous les liens inclus ou pointés dans un comportement donné et prépare tous les objets *modèles* associés pointés par référence;
- démarrage automatique 5: prépare tous les objets *modèles* associés pointés par référence et prépare tous les objets *actions* et *liens* pointés dans un comportement donné.

Quand le démarrage de disponibilité est omis, il n'y a pas d'effet de démarrage, c'est-à-dire que le démarrage de disponibilité est inhibé.

NOTE – Le démarrage de disponibilité est un lien. Il ne doit pas être confondu avec le constructeur. Il n'est exécuté que quand l'état de préparation des objets passe à «prêt».

## 25.2 Arrêt de disponibilité (Availability close-down)

L'arrêt de disponibilité est un objet *lien* qui a toujours la même condition de lien: passage de l'objet *composite* à l'état de préparation non prêt, cela signifie que l'action `Get Preparation Status` («cet-objet») donne le résultat non prêt. L'effet de lien peut servir pour des effets de destruction supplémentaires du composite. Il s'agit d'un effet de lien personnalisé fourni par l'utilisateur ou de l'un des effets de lien par défaut suivants:

- *arrêt automatique 1:*
  - destruction d'objets *lien* et *action* pointés par référence dans le comportement donné;
  - destruction des objets *actions* pointés par référence décrivant l'effet de lien dans l'un des objets *liens*;
  - destruction des objets *actions enveloppantes* pointés par référence dans les objets *actions* inclus ou pointés dans le comportement de composition;
  - une fois terminé, destruction du lien d'arrêt de disponibilité lui-même;
- *arrêt automatique 2:*
  - destruction des composantes associées pointées par référence à l'intérieur des éléments de composition;
  - une fois terminé, destruction du lien d'arrêt de disponibilité lui-même;
- *arrêt automatique 3:*
  - destruction des objets *lien* et *action* pointés par référence dans le comportement donné;
  - destruction des objets *actions* pointés par référence décrivant l'effet de lien dans l'un des objets *liens*;
  - destruction des objets *actions enveloppantes* pointés par référence dans les objets qui sont eux-mêmes inclus ou pointés par référence dans le comportement de composition;
  - destruction des composants associés pointés par référence dans les éléments de composition;
  - une fois terminé, destruction du lien d'arrêt de disponibilité.

Quand l'arrêt de disponibilité est omis, il n'existe pas d'effet d'arrêt, c'est-à-dire que cet arrêt de disponibilité est inhibé.

NOTE – L'arrêt de disponibilité est un lien. Il ne doit pas être confondu avec un destructeur. Il n'est exécuté que lorsque l'état de préparation de l'objet composite passe à 'not ready'.

## 25.3 Démarrage de rt-disponibilité (rt-Availability Start-up)

Le démarrage de rt-disponibilité est un objet *lien* qui a toujours la même condition de lien: création d'un rt-composite est créée à partir de ce composite, cela signifie que l'action `Get Rt-Availability Status` («cet-objet»:?) donne le résultat disponible. L'effet de lien peut servir pour des effets de présentation supplémentaires du rt-composite. Il s'agit d'un effet de lien personnalisé fourni par l'auteur ou de l'effet de lien par défaut suivants:

- rt-démarrage automatique: application de l'action `rUn` (exécution) à ce rt-composite.

Quand le démarrage de rt-disponibilité est omis, il n'existe pas d'effet de démarrage, c'est-à-dire que le démarrage de rt-disponibilité est inhibé.

NOTE – Le démarrage de disponibilité est un lien. Il ne doit pas être confondu avec un constructeur de rt-composant. Il n'est exécuté que quand l'état de rt-disponibilité de l'objet composite passe à 'available' pour chaque rt-composite créé à partir de cet objet composite.

## 25.4 Arrêt de rt-disponibilité (rt-Availability Close-down)

L'arrêt de rt-disponibilité est un objet *lien* qui a toujours la même condition de lien: création d'un rt-composite créé à partir de ce composite est supprimé, c'est-à-dire que l'action `Get Rt-Availability Status` («cet-objet»:?) donne le résultat non disponible. L'effet de lien peut servir pour des effets de destruction supplémentaires du rt-composite. Il s'agit toujours d'un effet de lien personnalisé fourni par l'auteur. Quand l'arrêt de rt-disponibilité est omis, il n'existe pas d'effet d'arrêt, c'est-à-dire que l'arrêt de rt-disponibilité est inhibé.

NOTE – L'arrêt de rt-disponibilité est un lien. Il ne doit pas être confondu avec le destructeur d'un rt-composant. Il n'est exécuté que quand l'état de rt-disponibilité du rt-composite passe à 'not available' (non disponible) à chaque rt-composite de cet objet composite.

## 25.5 Elément de composition

Chaque élément est composé de la manière suivante:

- 1) **indice d'élément** de composition: chaque élément à un identificateur unique à l'intérieur de l'objet *composite*.
- 2) **modèle associé**: peut servir à associer l'information à présenter à l'utilisateur. Le modèle est une étiquette, un contenu, un composite ou un modèle vide:
  - a) étiquette: chaîne générique présentée à l'utilisateur comme un élément d'interaction selon le style appliqué au rt-composite. Elle peut par exemple correspondre au titre d'un menu ou à la pression d'un bouton;
  - b) contenu: représente un élément terminal qui est un objet *contenu* ou un objet *contenu multiplexé*;
  - c) composite: permet de construire une relation parent-enfant à travers les différentes générations. Il appartient à l'auteur d'éviter la formation de boucle dans l'objet *composite*, c'est-à-dire d'éviter qu'un élément soit son propre descendant;
  - d) modèle vide: pas de modèle associé à l'élément.

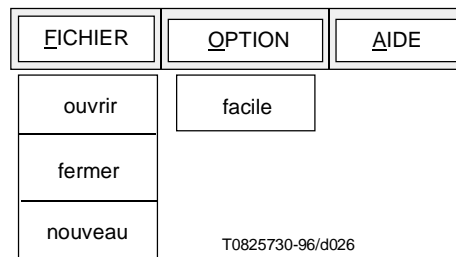
Quand un rt-composite est créé à partir de l'objet composite, le moteur MHEG crée des rt-objets à partir des modèles associés qui sont décrits dans la composition (un rt-contenu à partir d'une étiquette associée ou d'un contenu associé, un rt-composite à partir du composite associé, un objet «Null Root-Rt» (Rt-racine-vide) à partir du modèle vide associé). Ces rt-objets sont enfichés dans les réceptacles correspondants à l'intérieur du rt-composite.

## 25.6 Exemple de composition

La Figure 27 montre un exemple de menu utilisant la racine rt-composite 700:8, représenté Figure 20, qui est créée à partir du composite 700 de la Figure 18. Le composite 700 définit une structure logique de composition indépendamment de sa présentation. Dans cet exemple, un menu de styles a été appliqué à la racine 700:8 afin de représenter différentes générations de menus et de sous-menus. Les conditions suivantes s'appliquent:

- réceptacles rt-contenu: «FICHIER», «OPTION» et «AIDE» sont présentés dans le menu principal;
- réceptacles rt-composite 700:8:2 et 700:8:4 représentent les sous-menus;
- réceptacles rt-contenu: «ouvrir», «fermer» et «nouveau» sont présentés lorsque le premier menu est activé;
- réceptacle rt-contenu: «facile» est présenté quand le second menu est activé.

La présentation de ce menu n'est pas définie par la présente Recommandation; elle fait partie du rendu et de la convivialité du système de présentation.



**Figure 27/T.171 – Exemple de structure de menu avec sous-menus**

## 26 Structure de la classe *conteneur*

Ce paragraphe montre comment le regroupement des objets MHEG est pris en charge. Ce regroupement vise à faciliter le transfert afin d'obtenir un ensemble de transfert unique. Une fois transféré, il s'assure également que tous les objets inclus ont été transférés.

Cette classe peut aussi être utilisée par le concepteur d'objets pour regrouper l'information en jeu dans une même application.

L'objet *conteneur* offre les fonctionnalités suivantes:

- comportement optionnel prédéfini composé d'un démarrage de conteneur (voir 26.1) et d'un arrêt de conteneur (voir 26.2);
- une liste d'éléments de *conteneur* (voir 26.3).

### 26.1 Démarrage de conteneur (*container start-up*)

Le démarrage du conteneur est un objet *lien* qui a toujours la même condition: passage de l'objet *conteneur* à l'état de préparation prêt, c'est-à-dire quand l'action `Get Preparation Status` («cet-objet») donne le résultat prêt. L'effet de lien peut être utilisé pour des effets de préparation supplémentaires du conteneur. Il s'agit d'un effet de lien personnalisé fourni par l'auteur ou de l'un des effets de lien par défaut suivants:

- démarrage automatique du conteneur 1: l'action `Prepare` est appliquée aux objets inclus;
- démarrage automatique du conteneur 2: l'action `Prepare` est appliquée aux objets inclus et pointés par référence;
- démarrage automatique du conteneur 3: activation des objets liens inclus;
- démarrage automatique du conteneur 4: activation des objets liens inclus et pointés par référence;
- démarrage automatique du conteneur 5: l'action `Prepare` est appliquée aux objets inclus; activation des objets liens inclus;
- démarrage automatique du conteneur 6: l'action `Prepare` est appliquée aux objets inclus; activation des objets liens inclus et pointés par référence;
- démarrage automatique du conteneur 7: l'action `Prepare` est appliquée aux objets inclus et pointés par référence; activation des objets liens inclus;
- démarrage automatique du conteneur 8: l'action `Prepare` est appliquée aux objets inclus et pointés par référence; activation des objets liens inclus et pointés par référence.

Quand le démarrage du conteneur est omis il n'existe pas d'effet de démarrage, c'est-à-dire que le démarrage du conteneur est inhibé.

NOTE – Le démarrage du conteneur est un lien. Il ne doit pas être confondu avec un constructeur. Il n'est exécuté que quand l'état de préparation de l'objet *conteneur* passe à prêt.

### 26.2 Arrêt du conteneur (*container close-down*)

L'arrêt du conteneur est un objet *lien* qui a toujours la même condition de lien: passage de l'objet *conteneur* à l'état non prêt, cela signifie que l'action `Get Preparation Status` («cet-objet») donne le résultat non prêt. L'effet de lien peut servir à un effet de destruction supplémentaire du conteneur. Il s'agit d'un effet de lien personnalisé fourni par l'auteur ou de l'un des effets de lien par défaut suivants:

- arrêt automatique du conteneur;
- destruction des objets pointés par référence;
- une fois terminé, destruction du lien d'arrêt de disponibilité lui-même.

Quand l'arrêt du conteneur est omis, il n'existe pas d'effet d'arrêt c'est-à-dire que l'arrêt du conteneur est inhibé.

NOTE – L'arrêt de disponibilité est un lien. Il ne doit pas être confondu avec un destructeur. Il n'est exécuté que quand l'état de préparation de l'objet *contenant* passe à non prêt.

### 26.3 Élément de conteneur

Il s'agit d'une liste ordonnée d'éléments de conteneurs. Chaque élément peut être une instance d'autres classes MHEG susceptible d'être incluse ou pointée par référence, par exemple:

- objet *action*
- objet *lien*

- objet *script*
- objet *contenu*
- objet *contenu multiplexé*
- objet *composite*
- objet *contenant*
- objet *descripteur*

Un indice d'élément de conteneur est affecté à chaque élément de conteneur. Chaque élément a un identificateur unique dans un conteneur: le moteur MHEG indexe les éléments de 1 à n.

## 27 Structure de la classe *descripteur*

Ce paragraphe décrit comment la description des autres objets à transférer est prise en charge. Cette description vise à faciliter la négociation, l'installation, le fonctionnement et la gestion des applications utilisant les objets MHEG.

On sait que différents systèmes de présentation peuvent avoir des capacités de présentation extrêmement différentes. En transférant et en interprétant les objets *descripteurs*, il est possible d'adapter les ressources du système de présentation aux prescriptions des objets décrits. Les objets *descripteurs* permettent aux moteurs MHEG de déterminer s'ils peuvent ou non exécuter une présentation et autorisent les applications utilisatrices à déterminer les ressources nécessaires à la présentation d'un ensemble d'objets.

La présente Recommandation n'impose pas à tous les objets transférés d'être décrits par un objet *descripteur*. Si aucun descripteur n'est associé à un objet; c'est à l'application utilisatrice de définir la description de l'objet. Il appartient à l'auteur d'assurer la compatibilité entre les objets *descripteurs* et les objets décrits. La présente Recommandation n'offre pas de mécanisme de traitement des erreurs ou des incohérences des objets *descripteurs*.

L'objet *descripteur* peut fournir un certain nombre d'information sous forme plus ou moins formalisées:

- ensemble d'objets connexes (voir 27.1)
- autre description (voir 27.2)
- auto-assistance «lisez-moi» (voir 27.3)
- information lisible par le système (voir 27.4)
- ensemble d'informations sur les canaux (voir 27.5)
- ensemble d'informations sur les styles catalogués (voir 27.6)
- ensemble d'informations sur les actions élémentaires étendues cataloguées (voir 27.7)
- ensemble d'informations sur les attributs étendus catalogués (voir 27.8)

### 27.1 Objet connexe

Un ensemble d'objets connexes peut être fourni pour spécifier le domaine d'application de l'objet *descripteur*. Si cet ensemble est omis, le domaine est défini par l'application utilisatrice. Dans ce cas, le domaine inclut généralement tous les objets appartenant à l'application. Chaque attribut d'objet connexe fournit une information sur un seul objet connexe; il est spécifié comme une liste contenant les attributs suivants:

- référence d'objet : identifie l'objet connexe auquel cette information s'applique;
- informations sur l'objet: optionnel, définit plus précisément l'objet connexe avec les attributs suivants:
  - taille d'objet: indique en octets la taille de l'objet connexe codé (optionnel);
  - identificateur de classe: identifie la classe de l'objet connexe;
  - ensemble d'attributs propres à la classe: choix d'ensembles d'attributs, qui véhiculent une information propre à l'objet sur l'objet connexe:
    - attributs spécifiques à un objet *script* connexe (voir 27.1.1);
    - attributs spécifiques à un objet *contenu* connexe (voir 27.1.2);
    - attributs spécifiques à un objet *contenu multiplexé* connexe (voir 27.1.3);
  - décalage: fournit une position relative en octet d'un objet codé qui est inclus dans un autre objet codé (voir 27.1.5).

La présente Recommandation ne définit pas la manière de traiter les descriptions contradictoires, si un objet connexe donné est décrit plus d'une fois.

### 27.1.1 Information sur la classe *script*

Ce paragraphe fournit un ensemble d'attributs décrivant l'objet *script* connexe. Les informations suivantes peuvent être fournies:

- classification de *script*: informe le système utilisateur sur la classification des objets *script* connexes (voir 15.1);
- crochet de *script*: informe le système utilisateur sur les langages d'écriture de *script* utilisé par l'objet *script* connexe. Ce système peut utiliser cette information pour préparer l'interpréteur *script* correspondant ou pour rejeter l'objet *script* si cet interpréteur n'est pas disponible (voir le paragraphe 14).

### 27.1.2 Information sur la classe *contenu*

Ce paragraphe fournit un ensemble d'attributs décrivant l'objet *contenu* connexe. Les informations suivantes peuvent être fournies:

- classification *contenu*: informe le système utilisateur sur la classification des objets *contenu* connexe (voir 15.1);
- crochet de contenu: informe le système utilisateur sur le codage utilisé pour les données du *contenu* connexe. Ce système peut utiliser cette information pour préparer le décodeur correspondant (voir le paragraphe 14).
- objets alternatifs: offre une fonctionnalité pour indiquer les objets *contenu* alternatifs qui peuvent être utilisés à la place de ce *contenu* connexe (voir 27.1.4).

### 27.1.3 Information sur la classe *contenu multiplexé*

Ce paragraphe fournit un ensemble d'attributs décrivant l'objet *contenu multiplexé* connexe. L'information suivante peut être fournie:

- information de classe *contenu*: informe le système utilisateur sur la totalité des données multiplexées de l'objet *contenu multiplexé* connexe, c'est-à-dire classification, crochet, objets alternatifs (voir 27.1.2);
- nombre de trains de données: reflète le nombre de trains de données contenus dans les données multiplexées de l'objet connexe;
- ensemble d'informations de trains de données: chaque information de trains de données comporte les attributs suivants:
  - identificateur de trains de données: identifie un train de données dans l'objet *contenu multiplexé* connexe;
  - information sur la classe *contenu*: un train de données simple peut être considéré comme un objet *contenu* simple; on utilise par conséquent l'ensemble d'attributs qui décrit un contenu connexe, c'est-à-dire: classification, crochet, objets alternatifs (voir 27.1.2).

### 27.1.4 Objets alternatifs (Alternative object)

Ce paragraphe fournit des informations sur des objets *contenu* alternatifs qui peuvent remplacer l'objet connexe.

Les objets *contenu* alternatifs peuvent avoir un codage différent ou des attributs différents de ceux de l'objet connexe. L'application utilisatrice peut utiliser cette fonctionnalité pour adapter les données à présenter à l'environnement local. Il appartient à l'auteur du descripteur d'assurer l'intégrité des objets alternatifs dans le contexte de l'objet connexe.

Les objets alternatifs sont décrits comme une liste d'objets alternatifs. Chacun d'entre eux comporte les attributs suivants:

- référence à un objet *contenu* alternatif;
- crochet d'alternatif : spécifiant le crochet de l'objet alternatif;
- objet descripteur d'alternatif: pointage optionnel de l'objet descripteur décrivant l'objet alternatif;
- «lisez-moi» d'alternatif: optionnel et spécifié comme une chaîne de format libre. Cette auto-assistance peut être utilisée par l'application utilisatrice pour identifier la nature de l'objet alternatif. Par exemple, dans une application multilingue, l'information «lisez-moi» peut véhiculer une identification de la langue d'un objet alternatif. Il appartient à l'application utilisatrice d'interpréter la signification de l'information «lisez-moi».

### 27.1.5 Décalage

Un objet MHEG codé peut être inclus à l'intérieur d'un autre objet MHEG codé. Afin d'extraire cet objet inclus sans décoder le tout, une information de décalage peut être fournie. Ce décalage indique le premier octet de l'objet MHEG inclus au niveau D de la représentation.

### 27.2 Autre descripteur

Un ensemble de pointeurs vers d'autres objets *descripteurs* peut être fourni et utilisé par l'auteur pour connecter différents objets *descripteurs*. Ceux-ci peuvent décrire différents objets et les domaines de visibilité d'application des descripteurs peuvent être différents. La présente Recommandation ne définit pas comment traiter les descriptions contradictoires si le domaine de deux ou plusieurs descripteurs se chevauchent.

Cette fonctionnalité fournit un mécanisme de création d'un ensemble structurés de descripteurs. Les structures possibles sont des listes liées ou des arbres liés. Il appartient à l'auteur de définir une structure utile.

### 27.3 «lisez-moi»

Ce texte informel fournit aux utilisateurs l'information et les conseils nécessaires à l'installation et au fonctionnement d'un ensemble d'objets MHEG connexes. Le choix des mots est laissé à l'auteur, qui peut utiliser l'information «lisez-moi» pour pointer vers d'autres sources d'informations et d'autres supports.

### 27.4 Information lisibles par le système

Une application utilisatrice peut placer de l'information qu'elle trouve utile dans l'objet descripteur. La présente Recommandation ne définit ni cette information ni le codage utilisé par l'application. Un auteur ou une application utilisatrice peut inclure l'information dans le descripteur à l'intention du moteur MHEG. L'auteur doit toutefois savoir que la portabilité de cette information vers d'autres systèmes MHEG n'est pas assurée.

### 27.5 Informations de canal

Un ensemble d'informations de canal peut être fourni par l'auteur afin que le moteur MHEG mappe chaque canal plus facilement sur l'environnement réel. Chaque canal peut être décrit de la manière suivante:

- identificateur de canal: identifie le canal à décrire;
- x-min, x-max, y-min, y-max, z-min, z-max: décrit pour chaque axe du référentiel spatial les valeurs minimales et maximales requises pour le dispositif associé à ce canal;
- x-resolution, y-resolution, z-resolution: décrit pour chaque axe du référentiel spatial la résolution du dispositif associé à ce canal. Il s'agit du nombre d'unités physiques spatiales adressables;
- t-resolution: décrit la résolution temporelle pour le dispositif associé à ce canal. Il s'agit du nombre d'unités physiques temporelles adressables pendant une seconde;
- f-min, f-max: définit en «hertz» les fréquences présentables minimale et maximale qui sont requises pour ce canal si des documents sonores y sont présentés;
- audio-dynamique: définit la dynamique audio, en dB, qui est requise pour ce canal si des documents sonores y sont présentés;
- types de support: spécifie les types de support des *rt-composants* affectés à ce canal. Cela autorise le moteur MHEG à affecter des canaux aux dispositifs physiques. Il s'agit d'une des valeurs cataloguées qui peuvent être enregistrées ou hors norme. Si le système de présentation ne prend pas en charge le dispositif requis, le moteur peut ignorer le type de support;
- mappage d'événements: spécifie les événements prévus susceptibles d'arriver sur ce canal ou sur les *rt-composants* affectés à ce canal. Il est également possible d'affecter un événement catalogué à chaque événement. Cela autorise un moteur MHEG à mapper l'événement défini par l'auteur sur un événement réel susceptible de survenir dans son environnement. L'événement catalogué peut être enregistré ou hors norme. Si aucun mappage sur les événements catalogués n'est fourni, il est considéré que l'identificateur d'événement fourni par l'auteur mappe directement sur l'identificateur d'événement catalogué. Si ce n'est pas le cas, il appartient à l'application utilisatrice de définir comment mapper l'identificateur d'événement sur un événement réel. La présente Recommandation ne définit pas comment traiter les descriptions contradictoires si un identificateur d'événement donné est mappé plus d'une fois.

La présente Recommandation ne définit pas comment traiter les descriptions contradictoires si un canal est décrit plus d'une fois.

## 27.6 Informations sur les styles catalogués

Un ensemble de styles catalogués peut être fourni afin d'informer le système utilisateur sur les styles de catalogues utilisés dans les actions `Set Style` (désignation du style) transférés dans les objets *actions* connexes. Les styles catalogués peuvent être enregistrés ou hors norme.

## 27.7 Informations sur les actions élémentaires étendues cataloguées

Un ensemble d'actions élémentaires étendues cataloguées peut être fourni afin d'informer le système utilisateur sur les actions de ce type qui sont utilisées dans l'action élémentaire cataloguée transférée dans des objets *actions* connexes. Les actions élémentaires étendues cataloguées peuvent être enregistrées ou hors norme.

## 27.8 Informations sur les attributs étendus catalogués

Un ensemble d'attributs étendus catalogués peut être fourni afin d'informer le système d'exécution sur les attributs de ce type qui sont utilisés dans les actions `Set Catalogued Attributes` – (positionnement des attributs catalogués) transférées dans les objets *actions* et *liens* connexes. Ces attributs peuvent être enregistrés ou hors norme.

# SECTION 4 – COMPORTEMENT COMMUN AUX ENTITÉS MHEG

La présente Recommandation définit le comportement prévu et le comportement initial des objets, rt-objets et canaux MHEG en utilisant les attributs de comportement et leurs valeurs d'états. Elle définit aussi les actions servant à modifier et à extraire chacun des comportements prévus. Les effets de l'application de ces actions perceptibles à l'utilisateur sont appelés «effets utilisateur»; ceux qui ne sont accessibles que pour le moteur MHEG sont appelés «effets MHEG». La présente Recommandation ne traite que des effets MHEG. Les effets utilisateurs sont perçus par ce dernier via une interface GUI qui décide du rendu et de la convivialité des entités MHEG.

## 28 Comportement de l'entité MHEG

Le comportement initial de chaque entité MHEG que décrit la présente Recommandation peut être modifié en transférant les instances des classes MHEG suivantes:

- classe *action*: permet de transférer un ensemble d'actions avec des références vers des objets, des rt-objets ou des canaux MHEG;
- classe *lien*: permet de transférer un ensemble d'actions et d'en spécifier les conditions;
- classe *script*: permet de transférer des scripts décrivant des comportements complexes.

## 29 Définition des états des entités MHEG

Les diagrammes suivants définissent la vie d'un objet, d'un rt-objet ou d'un canal MHEG. Comme certaines actions ne sont valides que lors d'une phase donnée d'une entité MHEG, la définition de la vie de cette entité sert à vérifier la validité des actions. Deux méthodes de descriptions sont utilisées:

- 1) diagrammes temporels: ils décrivent le comportement principal en fonction du temps comme le montre la Figure 28, qui illustre un changement d'état d'une entité MHEG donnée. L'entité est d'abord située en phase i. Puis en lui appliquant une action donnée, il apparaît certains effets MHEG causés par le moteur MHEG ou par une interface GUI, mais l'entité reste encore en phase i. Cela signifie qu'une action `Get` (extraction) appliquée à cette entité durant l'exécution de l'effet MHEG extrait toujours la valeur associée à cette phase (valeur associée à la phase i). A la fin de l'effet MHEG, l'entité passe en phase j. Si cette phase a un effet utilisateur, l'apparence de l'entité est modifiée et ce changement est perçu par l'utilisateur. La Figure 28 illustre la durée de l'effet MHEG et les effets utilisateurs susceptibles d'apparaître;
- 2) diagrammes d'états finis: ils décrivent les transitions d'états possibles en fonction des actions qui sont illustrées sur la Figure 29. Sur cette figure, deux ellipses (Etat 1, Etat 2) représentent les états, et 3 rectangles (action x, action y, action z) représentent les actions susceptibles de modifier les états. L'état initial est l'état 1. Si l'action x est appliquée, le système reste à l'état 1. Si l'action y est appliquée, le système passe à l'état 2. L'action z provoque un retour à l'état 1. Si le système est à l'état 2, les actions x et y, non valides, sont ignorées; s'il est à l'état 1, l'action z, non valide, est ignorée.



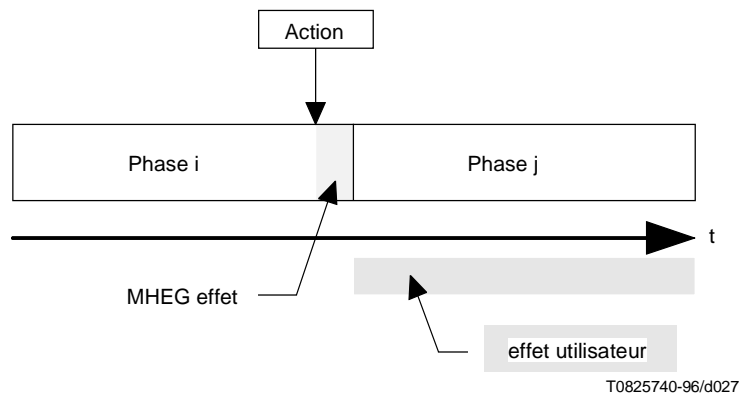


Figure 28/T.171 – Exemple de diagramme temporel

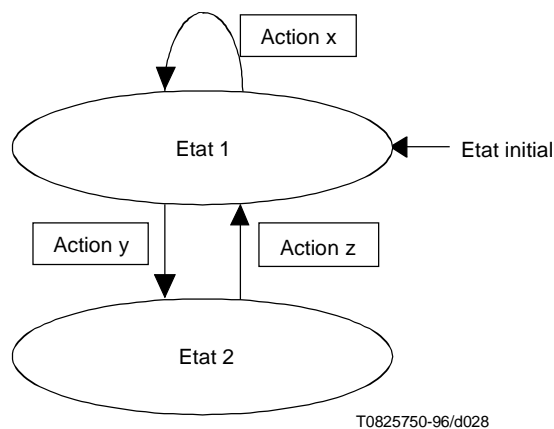


Figure 29/T.171 – Exemple de diagramme d'états finis

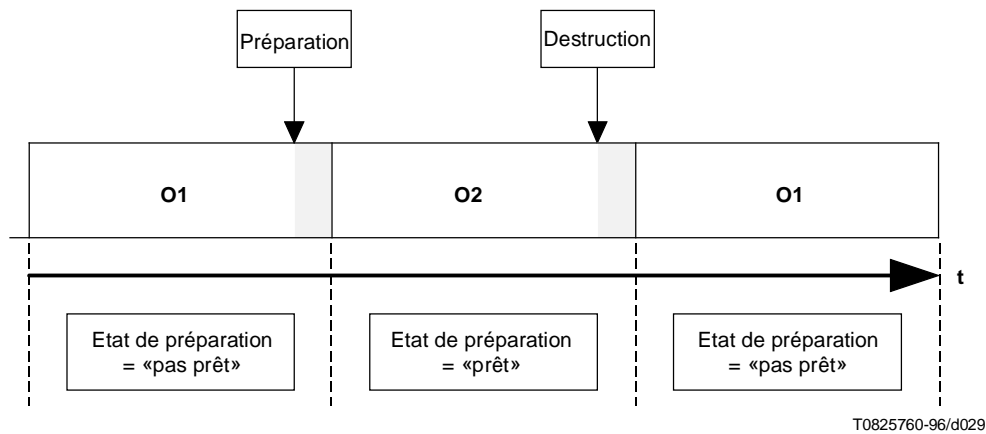
Plusieurs phases sont définies pour les comportements suivants des entités MHEG:

- disponibilité d'objets MHEG (voir 29.1)
- activation d'objets lien (voir 29.2)
- disponibilité de canal (voir 29.3)
- disponibilité de rt-objet (voir 29.4)
- comportement de fonctionnement d'un Rt-Composant (voir 29.5)
- comportement de présentation d'un Rt-composant (voir 29.6).

### 29.1 Disponibilité d'objet MHEG

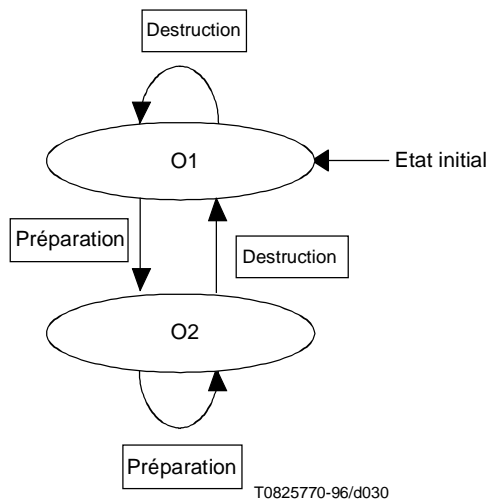
Les Figures 30 et 31 illustrent la disponibilité d'un objet MHEG. Il peut se trouver dans les deux états suivants:

- **O1:** l'objet MHEG n'est pas connu du moteur MHEG. Même s'il n'est pas créé, cet objet est dit en phase O1. Durant cette phase, tous les objets MHEG, sauf les objets *liens*, n'ont qu'un seul attribut: l'état de préparation qui a la valeur «pas prêt». Les objets *liens* ont en plus l'attribut état d'activation qui a la valeur «inactif». Les autres attributs décrits dans la présente Recommandation n'existent pas. Toute action *Get* appliquée à ces attributs inexistants les évalue à «non défini».
- **O2:** l'objet MHEG est disponible pour le moteur MHEG. L'état de préparation de cet objet est «prêt». L'objet a été décodé. Tous les autres attributs d'objets ont été créés et positionnés sur leurs valeurs de transfert correspondantes.



T0825760-96/d029

Figure 30/T.171 – Diagramme temporel de disponibilité d'un objet MHEG



T0825770-96/d030

Figure 31/T.171 – Diagramme d'états finis de disponibilité d'un objet MHEG

## 29.2 Activation de liens

Les Figures 32 et 33 illustrent l'activation de lien. En tant qu'objets MHEG, les objets liens possèdent les états de disponibilité. Ils possèdent de plus les deux états d'activation suivants. Les deux figures montrent la relation entre les états de disponibilité et d'activation:

- **L1:** l'état d'activation de l'objet *lien* est «inactif». Le lien ne peut pas être déclenché;
- **L2:** l'état d'activation de l'objet *lien* est «actif». L'objet *lien* a été activé. Durant l'état L2, chaque fois que la condition de lien passe à Vrai, le lien est déclenché. L'action interruption de lien n'affecte pas ces états.

Une action `Activate` (activation) appliquée à un lien dont l'état de préparation est «pas prêt» exécute implicitement une action `Prepare` (préparation).

Une action `Destroy` (destruction) peut aussi être appliquée à l'état L2. L'objet *lien* passe alors directement en phase L1/O1.

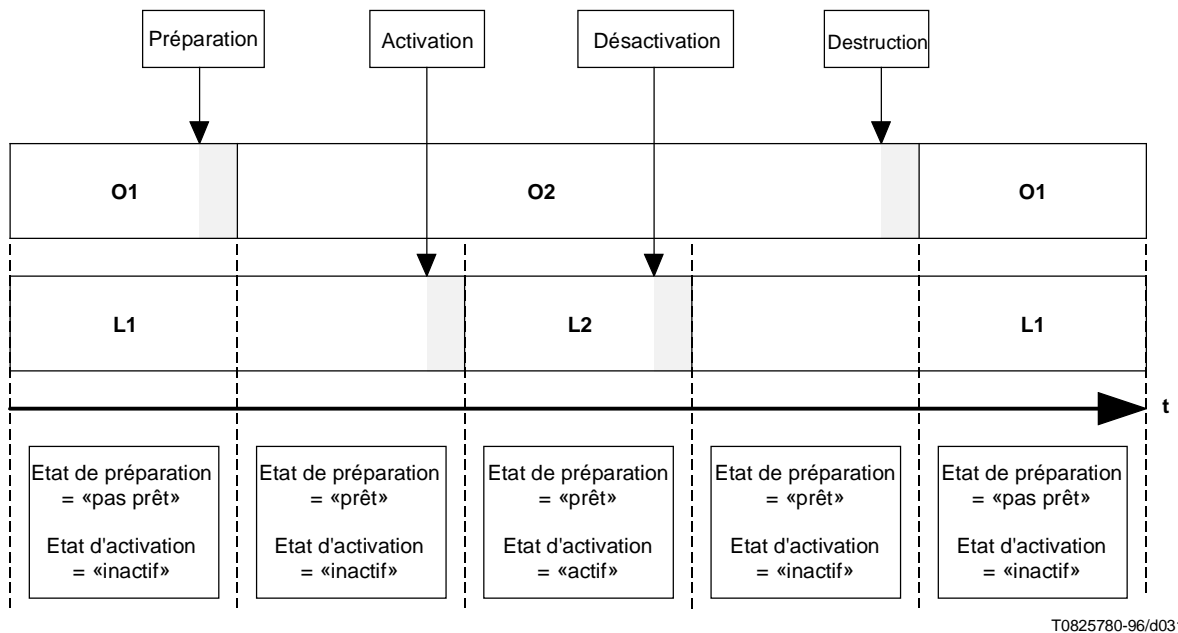


Figure 32/T.171 – Diagramme temporel de disponibilité et d'activation d'objets liens

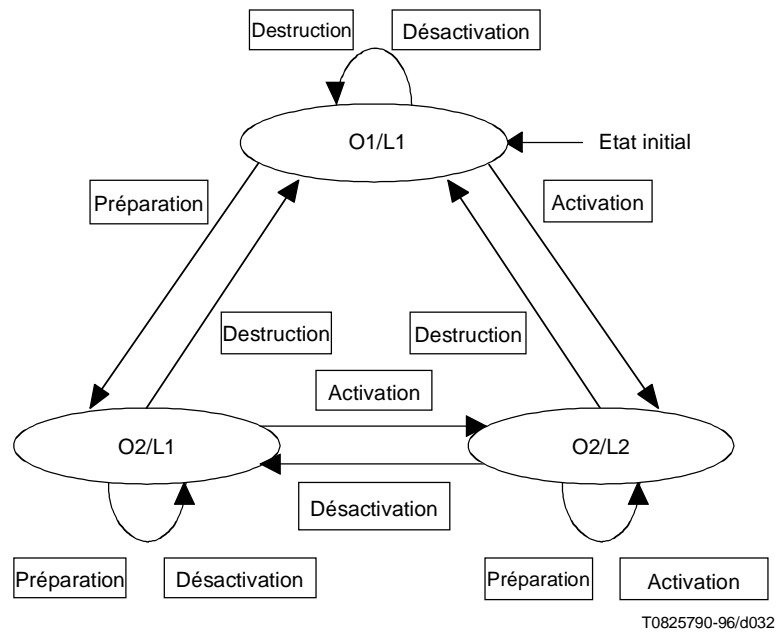


Figure 33/T.171 – Diagramme d'états finis de disponibilité et d'activation d'objets liens

### 29.3 Disponibilité de canal

Les Figures 34 et 35 illustrent la disponibilité de canal. Celui-ci peut se trouver dans l'un des états suivants:

- C1: le canal n'est pas encore créé et donc pas disponible pour le moteur MHEG;
- C2: le canal est disponible pour le moteur MHEG.

Le canal par défaut ne peut être ni créé ni détruit. Il est toujours disponible pour le moteur MHEG (c'est-à-dire qu'il est toujours à l'état C2). Il ignore les actions New (nouveau) et Delete (suppression).

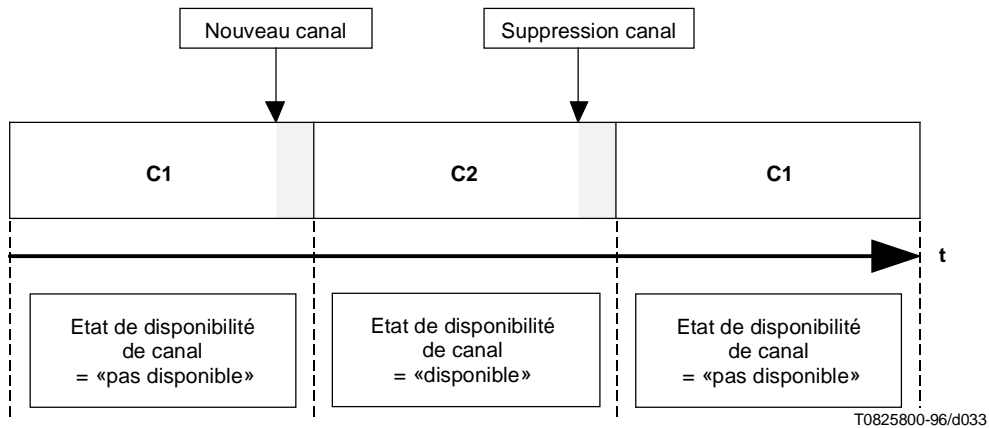


Figure 34/T.171 – Diagramme temporel de disponibilité de canal

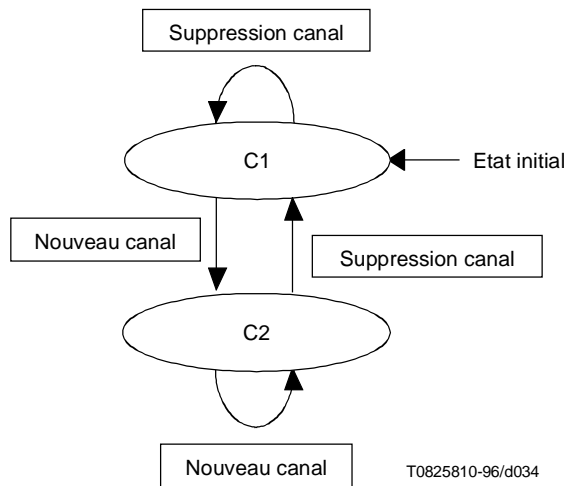


Figure 35/T.171 – Diagramme d'états finis de disponibilité de canal

## 29.4 Disponibilité de rt-objet

Les Figures 36 et 37 illustrent la disponibilité d'un rt-objet. Elles montrent aussi la relation entre l'état de disponibilité d'un rt-objet et celui de l'objet *modèle* duquel le rt-objet est dérivé. Le rt-objet peut se trouver dans les deux états suivants:

- **R1:** l'objet modèle est prêt (état O2), le rt-objet n'est pas créé, il n'est pas disponible;
- **R2:** le rt-objet est disponible pour le moteur MHEG. Durant cette phase, le rt-objet peut être présenté à l'aide d'une action Run (exécution).

Une action New appliquée en phase O1/R1 prépare implicitement l'objet modèle. Une action Destroy appliquée en phase O2/R2 supprime tous les rt-objets créés implicitement à partir de ce modèle.

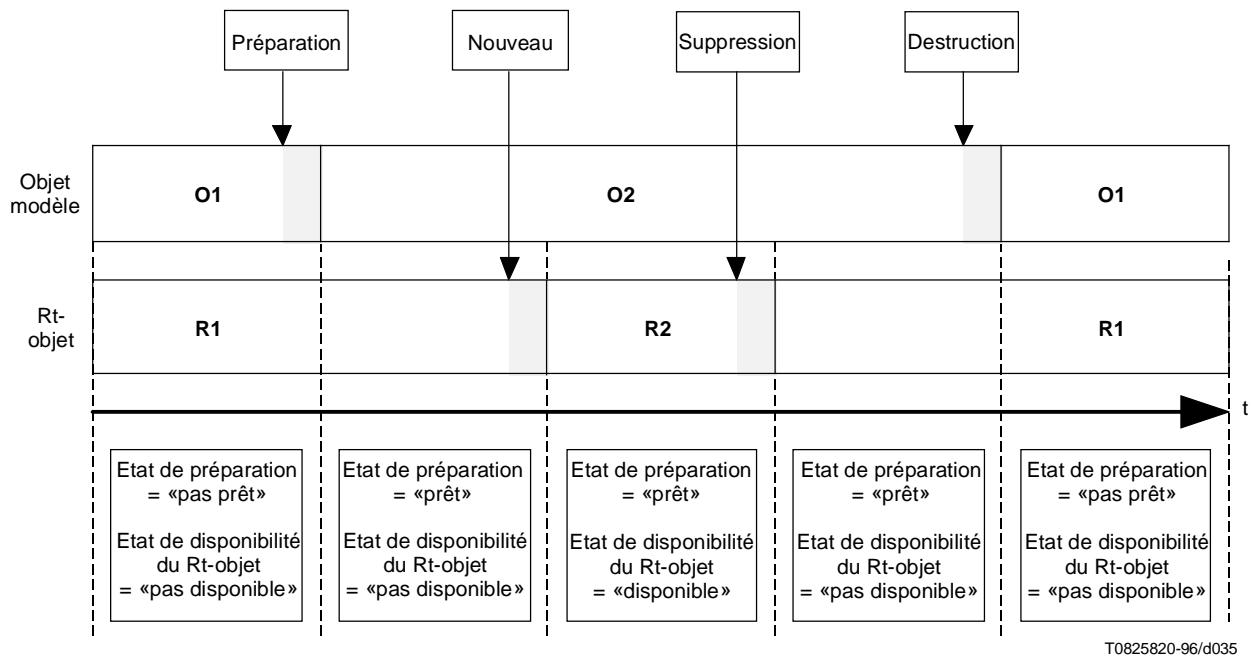


Figure 36/T.171 – Diagramme temporel de disponibilité d'un rt-objet

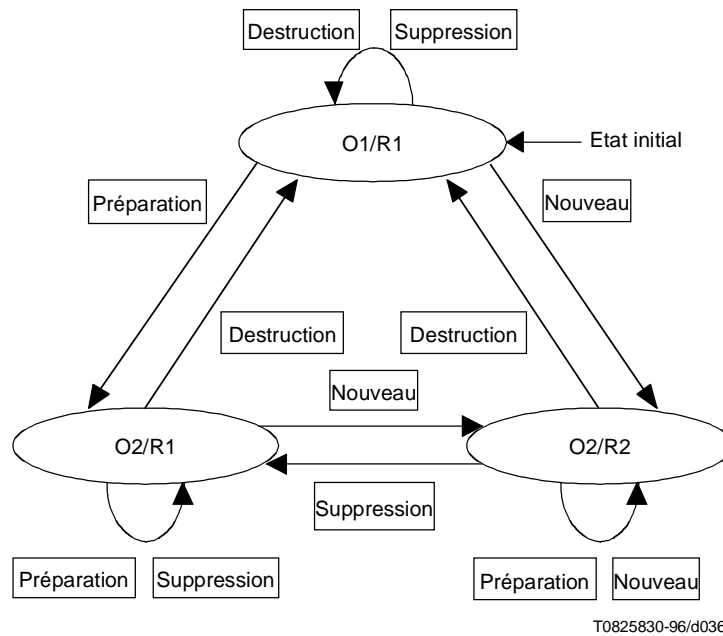


Figure 37/T.171 – Diagramme temporel de disponibilité d'un rt-objet

## 29.5 Comportement de fonctionnement des rt-composants

Les Figures 38 et 39 illustrent le comportement des rt-composants en fonctionnement. Une fois qu'un rt-composant est «disponible» (c'est-à-dire à l'état R2), il est prêt pour la présentation. Il peut se trouver dans les deux états suivants:

- **R2:** le rt-composant n'est pas en fonctionnement. Il n'est pas présenté à l'utilisateur;
- **R3:** le rt-composant est en fonctionnement. Il peut être présenté à l'utilisateur. Durant cette phase, il peut apparaître un effet utilisateur résultant de la mise en fonctionnement par l'utilisateur du rt-composant avec ses attributs de comportement spécifiés. L'effet utilisateur peut dépendre du moteur MHEG ou de l'interface GUI.

Si une action Run est appliquée à un rt-composant inexistant (c'est-à-dire à l'état R1), une action New implicite est appliquée à ce rt-composant avant l'exécution de l'action Run.

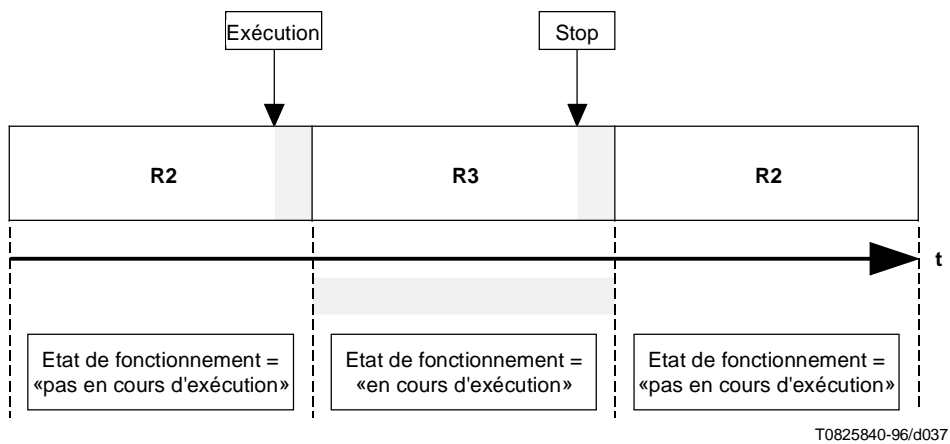


Figure 38/T.171 – Diagramme temporel d'un rt-composant en fonctionnement

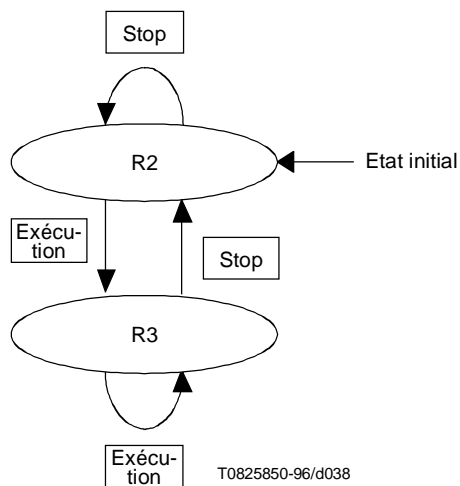


Figure 39/T.171 – Diagramme d'états finis d'un comportement en fonctionnement

## 29.6 Comportement de présentation du rt-composant

Pendant qu'un rt-composant est en fonctionnement (c'est-à-dire à l'état R3), il est possible de changer les attributs de comportement de présentation en appliquant certaines actions `Set`.

La Figure 40 donne un exemple type d'attribut de présentation. Sur cette figure, un attribut «A» de comportement de présentation associé à un rt-composant donné passe de la valeur V0 à V1 pendant que le rt-composant est à l'état R3. Un effet utilisateur peut être associé à ce changement de valeur. Une fois ce rt-composant stoppé, l'attribut A reste à la valeur V1 qui est indépendante de l'état de fonctionnement et des autres attributs.

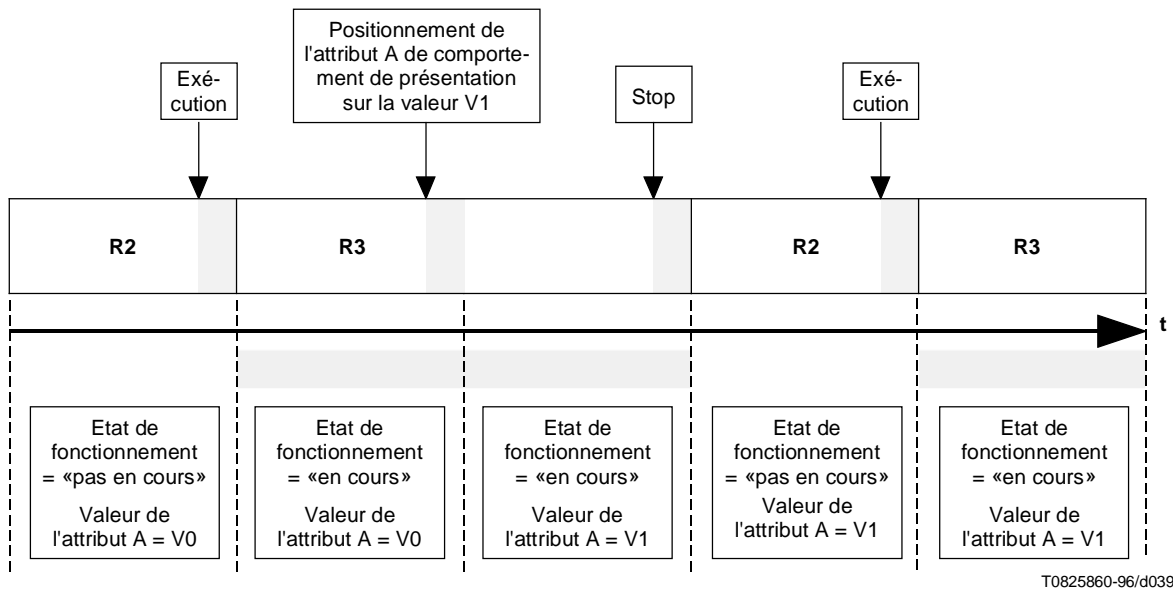


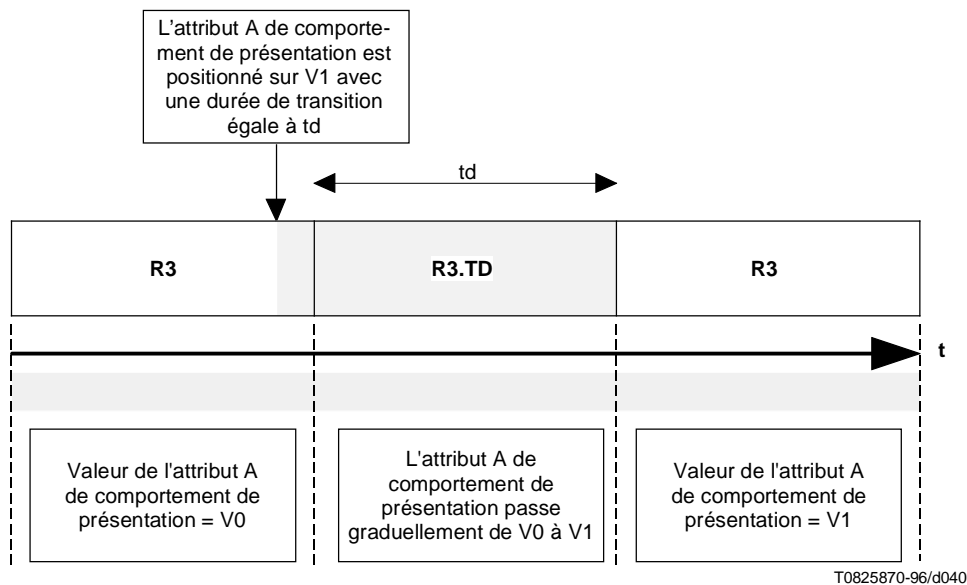
Figure 40/T.171 – Modification des attributs de comportement de présentation

La Figure 41 montre une modification d'un attribut de présentation avec une certaine durée de transition. Quand un rt-composant est en fonctionnement, une action `Set` lui est appliquée afin d'en modifier graduellement l'attribut A. Dans ce cas, une phase d'état supplémentaire, R3.TD, est définie de la manière suivante:

- **R3.TD:** l'état de fonctionnement du rt-composant est «en cours d'exécution». Le moteur MHEG modifie graduellement l'attribut: l'application d'une action `Set` donnée provoque l'apparition d'une durée de transition. Durant la phase R3.TD, la valeur de l'attribut spécifié est modifiée graduellement. Si l'action appliquée fait partie d'un groupe d'actions synchronisées à appliquer en série, l'action suivante de la liste doit être exécutée en fin de phase R3.TD. Un effet utilisateur peut être associé à la phase R3.TD. Il montre l'influence de la durée de transition sur la perception de la valeur de l'attribut. L'utilisateur peut percevoir graduellement le changement de valeur de l'attribut de comportement de présentation de sa valeur précédente à sa valeur spécifiée. L'effet utilisateur peut dépendre du moteur MHEG et de l'interface GUI.

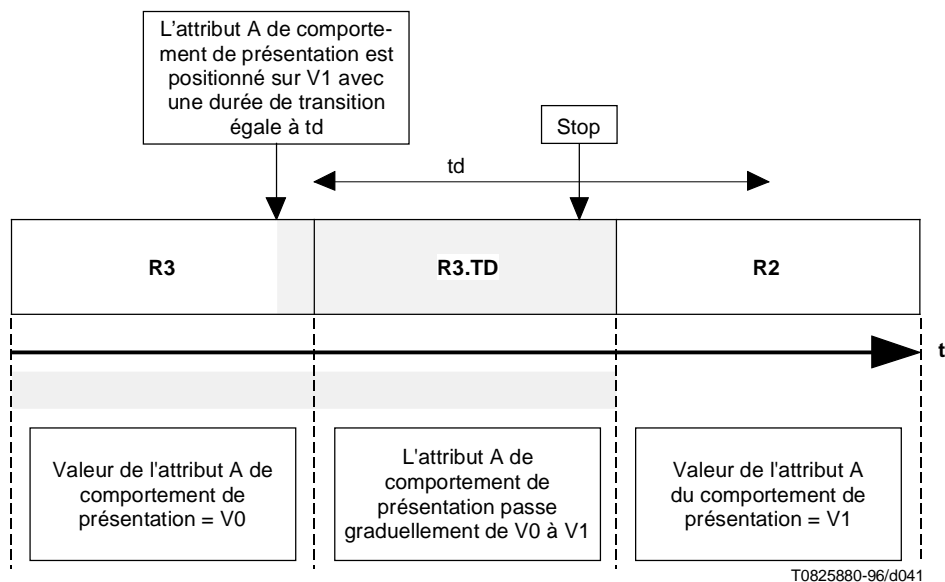
L'effet MHEG d'une action `Set` avec durée de transition comporte deux parties:

- l'application initiale de l'action `Set` à l'état R3
- modification graduelle de l'attribut en phase R3.TD.



**Figure 41/T.171 – Diagramme temporel d'une action Set avec durée de transition**

Quand l'effet MHEG d'une durée de transition est en cours d'application sur un rt-composant et qu'on applique une action Stop (arrêt) sur ce rt-composant, il est mis fin à l'effet de transition après l'achèvement de l'effet MHEG de l'action Stop. L'attribut de comportement de présentation prend la valeur spécifiée dans l'action Set comme si aucune durée de transition n'avait été spécifiée. La Figure 42 illustre cette situation.



**Figure 42/T.171 – Diagramme temporel d'une action Set avec durée de transition dans le cas où une action Stop est appliquée**

Quand l'effet MHEG d'une durée de transition est appliqué à un attribut et que simultanément on lui applique une action Set, il est mis fin à l'effet MHEG de transition. La valeur de comportement de présentation prend la valeur qui est spécifiée par la dernière action Set. Si cette dernière action n'a pas de durée de transition; l'effet MHEG est appliqué de la manière illustrée sur la Figure 40. Si elle a une durée de transition, l'effet MHEG est appliqué comme le montre la Figure 41. La Figure 43 est un exemple du cas précédent.



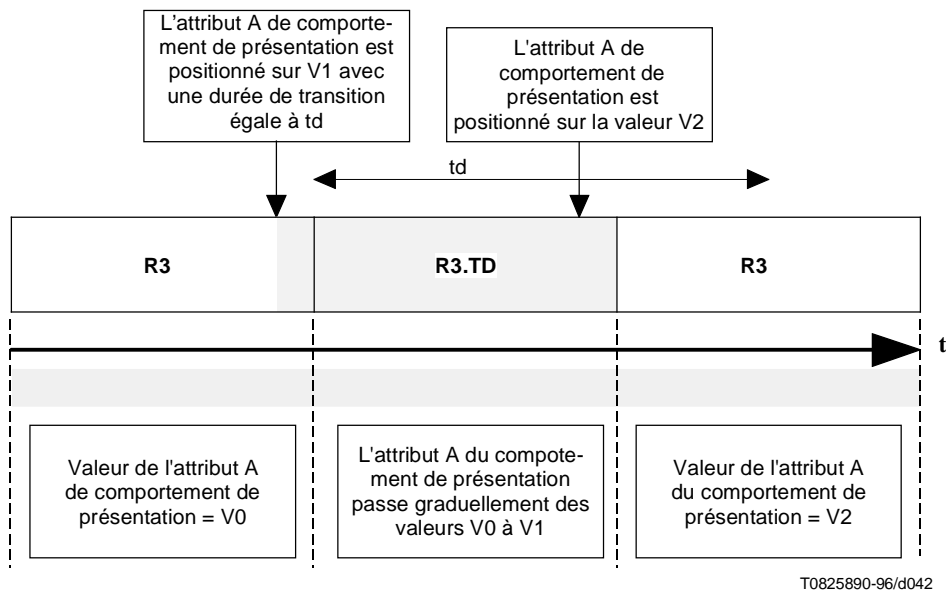


Figure 43/T.171 – Diagramme temporel d'une action Set avec durée de transition interrompue

La Figure 44 montre une modification de l'attribut du comportement de présentation alors qu'un rt-composant est dans l'état «pas en cours d'exécution». L'attribut A de comportement de présentation du rt-composant a la valeur V0 pendant que le rt-composant est à l'état R2. Quand une action Set servant à modifier l'attribut A est appliquée à ce rt-composant, le moteur MHEG applique l'effet MHEG de cette action Set. Si une durée de transition est spécifiée, elle est positionnée sur 0 signifiant par là qu'elle doit être ignorée. Après l'application de l'effet MHEG, la valeur de l'attribut A est égale à V1. Si une action Run est appliquée ensuite, le rt-composant est présenté avec A égal à V1.

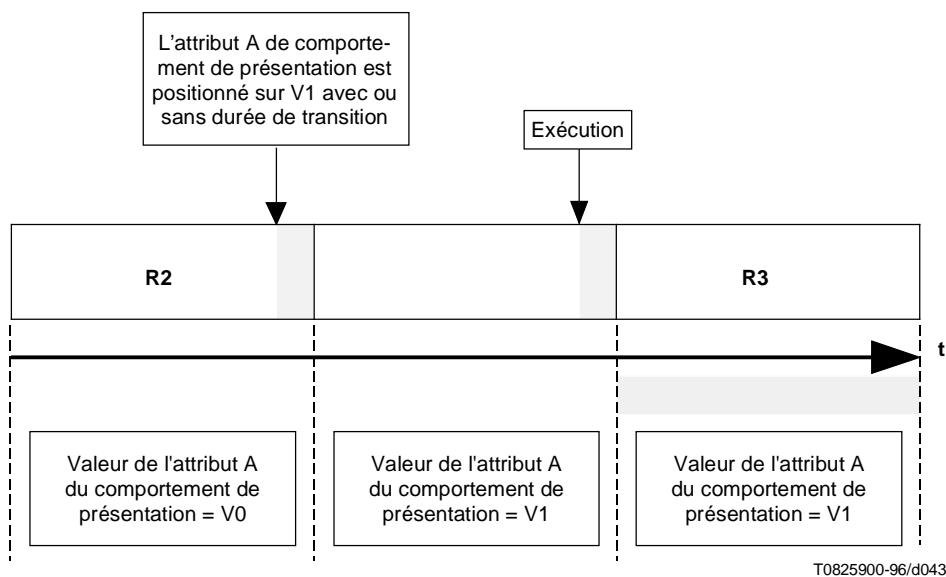


Figure 44/T.171 – Diagramme temporel d'un ensemble d'action appliquée à un rt-composant à l'état de «pas en cours d'exécution»

## 30 Cycle de vie des entités MHEG

Une entité MHEG est créée et détruite à l'aide d'actions élémentaires. L'intervalle de temps séparant ces deux actions correspond à la vie de l'entité. Durant son existence, on utilise certains attributs qui lui sont associés.

Principales règles liant la vie d'une entité aux attributs qui lui sont associés:

- même si une telle entité MHEG n'existe pas, l'état de préparation (voir 29.1), de rt-disponibilité (voir 29.4) ou de disponibilité de canal (voir 29.3) existe et est toujours disponible. L'état de préparation est positionné sur «pas prêt», l'état de disponibilité sur «pas prêt» et l'état de disponibilité de canal sur «pas disponible»;
- si une action `Prepare` (voir 29.1), `New` (voir 29.4) ou `New Channel` (nouveau canal) (voir 29.3) est appliquée, il est demandé au moteur MHEG d'instancier l'entité MHEG spécifiée avec ses attributs correspondants qui sont décrits dans les différents comportements. L'état de préparation est positionné sur «prêt», l'état de rt-disponibilité sur «prêt» et l'état de disponibilité de canal sur «disponible».
- Si une action `Destroy` (voir 29.1), `Delete` (voir 29.4) ou `Delete Channel` (suppression de canal) (voir 29.3) est appliquée, il est demandé au moteur MHEG de retirer l'entité MHEG spécifiée avec les attributs associés qui lui correspondent à l'exception des états de préparation, de rt-disponibilité et de disponibilité de canal qui sont respectivement positionnés sur «pas prêt», «pas prêt» et «pas disponible».

Par conséquent, si une entité MHEG est détruite ou supprimée, seul l'état de préparation, de rt-disponibilité ou de disponibilité de canal est modifié. S'il existe alors un lien dont la condition est «lorsque l'état de préparation passe à pas prêt», «lorsque l'état de disponibilité passe à pas prêt» ou «lorsque l'état de disponibilité de canal passe à l'état pas disponible», il sera déclenché.

Par ailleurs, d'autres attributs associés à l'entité détruite ou supprimée sont entièrement retirés sans modification de leur valeur d'état. Par exemple, s'il existe un rt-composant à l'état «en cours d'exécution», son état est supprimé sans modification de valeur. Cela signifie que le rt-composant a été supprimé alors qu'il fonctionnait. L'état de fonctionnement n'est plus disponible après la suppression. S'il existe un lien dont la condition est «lorsque l'état de fonctionnement passe à pas en fonctionnement», il ne sera pas déclenché au moment de la suppression.

## 31 Mécanismes généraux des actions

Ce paragraphe récapitule les mécanismes généraux des actions et le traitement que leur applique le moteur MHEG. Ce mécanisme s'applique tant aux comportements communs qu'aux autres comportements.

### 31.1 Application des actions

Lors de l'application des actions, on respectera les conditions suivantes:

- si l'effet MHEG d'une action contient une liste de sous-effets, ceux-ci seront exécutés par le moteur MHEG comme s'ils constituaient une même opération indivisible et ininterrompue;
- l'effet de traitement d'une action sur un ensemble de deux ou plusieurs cibles est de propager l'action à chaque cible à l'intérieur de l'ensemble des cibles. Ces actions sont exécutées en parallèle. L'effet MHEG de l'action prend fin lorsque les effets MHEG de toutes les actions propagées prennent fin;
- dans certains cas, afin d'exécuter une action, une action de traitement intermédiaire est implicitement exécutée. Cette action de traitement intermédiaire peut modifier l'état. Un lien déclenché sur ce changement d'état est déclenché avant l'arrêt du traitement de l'action spécifiée.

Par exemple, lorsque qu'un rt-objet est créé à partir d'un objet *modèle* qui n'est pas prêt, la présente Recommandation spécifie qu'une action `Prepare` implicite doit être appliquée à l'objet modèle avant l'action `New`. Un lien dont la condition est «lorsque l'état de préparation de l'objet *modèle* passe à prêt» doit être déclenché quand l'objet *modèle* passe à l'état O2 et non quand le rt-objet passe à l'état R2.

## 31.2 Marche à suivre lors de l'application d'un effet de lien

Lorsque le lien est déclenché, c'est-à-dire lorsqu'il est activé et que la condition de lien est satisfaite, le moteur MHEG applique l'effet de lien. A chaque déclenchement de lien, l'effet de lien est exécuté en deux phases:

- phase de macrorésolution: si certaines macroinstructions sont dans ce lien (voir le paragraphe 13);
- phase d'exécution de l'action: exécution de l'objet *action* inclus ou pointé par référence dans l'effet de lien; cet objet *action* peut être une action de base, une action enveloppante ou une macroaction;

La Figure 45 récapitule la marche à suivre lors de l'application d'un effet de lien.

L'exécution de l'action commence lorsque la condition de lien est satisfaite; la macrorésolution est effectuée lorsque l'action est une macroaction. Chaque objet *action* est une construction récursive contenant un groupe d'actions synchronisées associées à un indicateur de synchronisation.

Cet indicateur spécifie si les actions synchronisées du groupe doivent être appliquées en parallèle ou en série. L'exécution du groupe d'actions dépend de l'indicateur de synchronisation. Si un indicateur est spécifié comme une macroinstruction et si celle-ci est résolue par la valeur «non défini», l'ensemble d'actions synchronisées n'est pas appliqué.

```
COMMENCER EXÉCUTION ACTION de l'objet action à l'intérieur de l'effet de lien
  SI indicateur de synchronisation == série
    POUR J = 1 A nombre_d'actions_synchronisées
      FAIRE
        SI action_synchronisée J == action_élémentaire
          EXÉCUTER résolution d'une cible de l'action_élémentaire J
          EXÉCUTER l'effet de l'action_élémentaire J sur l'ensemble de cibles
        SINON SI action_synchronisée J == objet_action
          EXÉCUTER ACTION RECURSIVE de l'objet_action J
        FIN DE SI
      ATTENDRE la fin de l'effet MHEG de l'action_synchronisée J
    SINON SI indicateur_de_synchronisation == parallèle
      POUR J = 1 A nombre_d'actions_synchronisées
        FAIRE SIMULTANÉMENT
          SI action_synchronisée J == action_élémentaire
            EXÉCUTER résolution de cible d'une action élémentaire J
            EXÉCUTER l'effet de l'action_élémentaire J sur l'ensemble de cibles
          SINON SI action_synchronisée J == objet_action
            EXÉCUTER ACTION RECURSIVE sur l'objet_action J
          FIN DE SI
        ATTENDRE la fin de l'effet MHEG de chaque action de chaque groupe «d'actions synchronisées»
      ATTENDRE la fin de l'effet utilisateur de chaque action Run de chaque groupe «d'actions synchronisées»
      DURANT ce temps SI une cible est dans l'état «en cours d'exécution» ET
        SI sa position temporelle courante == sa position terminale courante
          EXÉCUTER une action Stop sur cette cible
    FIN EXÉCUTION ACTION de l'objet action dans l'effet lien
```

Figure 45/T.171 – Phases d'exécution de l'action

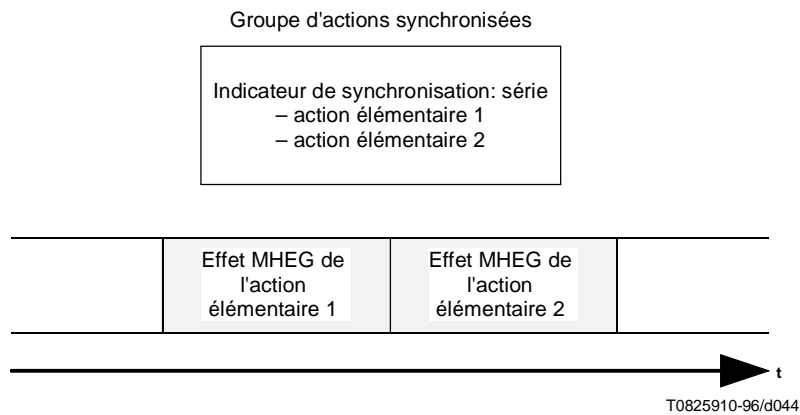
### 31.2.1 Exécution des actions en série

S'il est spécifié que toutes les actions élémentaires d'un groupe d'actions «série» doivent être exécutées, chaque action élémentaire est exécutée de manière séquentielle à la fin de l'effet MHEG de l'action élémentaire précédente.

La Figure 46 donne un exemple où deux actions élémentaires sont incluses dans un effet de lien. A la fin de l'effet MHEG de l'action élémentaire 1, l'action élémentaire 2 est exécutée.

La durée d'un effet MHEG correspondant à une liste d'actions série est la somme des durées des effets MHEG associés à chacune des actions série.

Si le groupe contient une action *Run*, l'effet utilisateur de cette action continue même après la fin des effets MHEG du groupe d'actions, sauf si une action *Stop* est appliquée.



**Figure 46/T.171 – Exemple d'exécution d'actions en série**

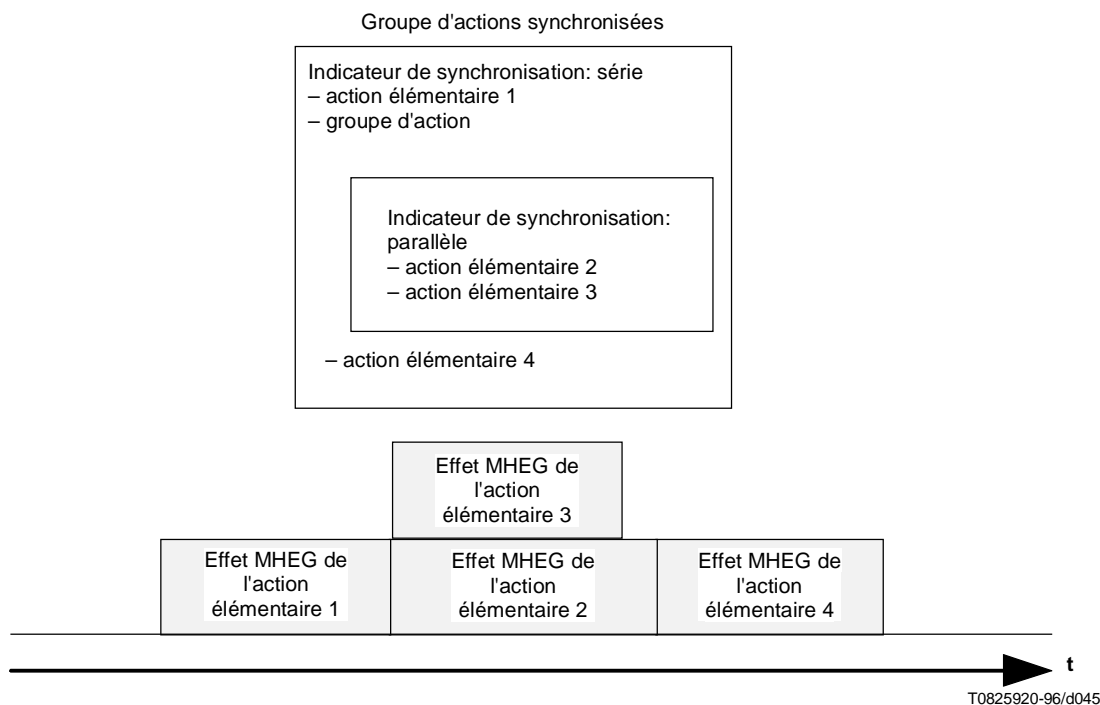
### 31.2.2 Exécution des actions en parallèle

S'il est spécifié que toutes les actions d'un groupe d'actions «parallèle» doivent être exécutées, chaque action élémentaire est appliquée simultanément. L'effet MHEG du groupe prend fin quand les effets de toutes les actions parallèles prennent fin.

Lorsqu'un groupe parallèle est imbriqué dans un groupe d'actions série, il est demandé au moteur MHEG d'attendre la fin de l'effet MHEG correspondant à l'ensemble des actions parallèles avant d'appliquer la prochaine action du groupe d'actions série.

La Figure 47 donne un exemple d'exécution en parallèle. Avant l'application de l'action élémentaire 4, le moteur MHEG attend la fin de l'effet MHEG correspondant au groupe d'actions, c'est-à-dire la fin de l'effet MHEG de l'action élémentaire 2.

La durée de l'effet MHEG d'un ensemble d'actions parallèles est celui de l'action parallèle à l'effet MHEG le plus long.



**Figure 47/T.171 – Exemple d'exécution d'actions en parallèle**

### 31.3 Traitement de base d'une action élémentaire

Lorsque le moteur MHEG exécute une action élémentaire sur une cible simple, on obtient les effets suivants:

- **Effet MHEG sans durée de transition:** commence au moment où l'action est appliquée et finit lorsque l'action a été prise en compte par le moteur MHEG; il conduit à un changement d'état ou de valeur d'attribut de comportement. Les Figures 30, 32, 34, 36, 38, 40 et 44 en donnent des exemples.
- **Effet utilisateur sans durée de transition:** n'existe que si l'état de fonctionnement est «en cours d'exécution». Cet effet commence au moment où l'effet MHEG de l'action `run` prend fin. Cet effet utilisateur dure jusqu'à la fin de l'effet MHEG associé à l'action `stop`. L'effet utilisateur est perceptible par l'utilisateur durant le fonctionnement. A chaque changement d'attribut de présentation d'une cible, si la cible est à l'état «en cours d'exécution», l'effet utilisateur reflète immédiatement ce changement. Les Figures 38 et 40 en donnent quelques exemples.
- **Effet MHEG avec durée de transition:** commence au moment où une action avec durée de transition non nulle est appliquée à une entité. Il prend fin lorsque l'action a été prise en compte par le moteur MHEG. Si la cible n'est pas en fonctionnement, la durée de transition est positionnée sur zéro et ignorée. Les Figures 41, 42 et 43 en donnent quelques exemples.
- **Effet utilisateur avec durée de transition:** n'existe que lorsqu'une cible est en fonctionnement et qu'une action `set` avec une transition non nulle lui est appliquée. Cet effet commence lors de l'instanciation quand l'effet MHEG de l'objet avec durée de transition commence. L'effet de l'action est perçu par l'utilisateur de façon graduelle. Une fois que l'effet MHEG avec durée de transition prend fin, l'effet utilisateur avec durée de transition prend également fin. Les Figures 41, 42 et 43 en donnent quelques exemples.

### 31.4 Résolution d'un ensemble de cibles

Il est possible de spécifier pour chaque action élémentaire une cible simple (un objet MHEG, un élément conteneur, un rt-objet, un élément réceptacle, un canal, par exemple); ou multiple (plusieurs cibles, un alias représentant plusieurs entités, tous les rt-objets créés à partir d'un objet *modèle*, tous les enfants d'une entité spécifiée, par exemple). Les cibles multiples sont spécifiées sous forme d'une liste. L'effet d'action à appliquer aux différentes cibles d'une liste doit être exécuté en parallèle.

#### NOTES

1 – Les auteurs ne tiendront pas compte de l'ordre d'exécution des actions à appliquer en parallèle.

2 – Le moteur MHEG n'est pas tenu de réaliser une exécution véritablement parallèle.

L'exécution d'une action sur une cible non définie n'a pas d'effet.

### 31.5 Précision arithmétique

Des pourcentages et des fractions peuvent être spécifiés dans certains paramètres d'actions élémentaires, par exemple les actions `Set GTF` (positionnement du facteur temporel générique) et `Set OVD` (fixation de la durée de visualisation d'origine). De plus, les fractions servent aussi à projeter l'espace de présentation d'origine sur l'espace de présentation relatif.

Les pourcentages et les fractions se calculent sans arrondi jusqu'à la présentation finale des objets. L'arrondi ne sera fait que pour la présentation finale, c'est-à-dire, au niveau des canaux, dans la chaîne finale de projection de l'espace OPS sur l'espace de présentation de canal (CPS).

#### NOTES

1 – Si un rt-objet de volume CV égal à 17 est connecté à un rt-composite A avec un facteur de volume sonore générique (GVF) égal à 1/3 et que A est lui-même connecté au rt-composite B avec un facteur GTF égal à 3, B étant enfin connecté à un canal, le volume final à mapper sur le dispositif physique est égal à  $17 = (\text{arrondi}(17 \times 1/3 \times 3))$  et non à  $18 = (\text{arrondi}(\text{arrondi}(17 \times 1/3) \times 3))$ .

2 – Si l'exemple précédent où le facteur GTF du rt-composite B est égal à 2, le volume final est égal à  $11 = (\text{arrondi}(17 \times 1/3 \times 2))$ .

3 – Un profil de la présente Recommandation ou une application utilisatrice peut adopter une autre précision arithmétique et une autre politique d'arrondi.

## 32 Effets et traitements communs aux actions

Ce paragraphe décrit les effets et les traitements des actions qui sont communs à toutes les actions élémentaires et actions `get` (extraction) appliquées à tous les comportements définis dans la présente Recommandation. Le traitement effectif des actions élémentaires et des actions `get` dépasse le cadre de la présente Recommandation et doit être prévu par le moteur MHEG.

### 32.1 Actions élémentaires

Les actions élémentaires possèdent les effets et traitements communs suivants:

- si le moteur MHEG est incapable de réaliser l'effet MHEG requis, il est recommandé mais pas imposé qu'il le notifie à l'application utilisatrice;
- si la présente Recommandation ne décrit pas l'application de l'effet d'une action élémentaire à un objet, un rt-objet ou un canal MHEG, cette action est ignorée par le moteur MHEG;
- si l'ensemble de cibles n'est plus disponible durant l'exécution d'une action élémentaire (c'est-à-dire que toutes les cibles de l'ensemble sont détruites ou supprimées), l'exécution de cette action est abandonnée;
- si une référence à une cible d'une action élémentaire est non valide l'action élémentaire est ignorée (c'est-à-dire que son effet est nul et s'arrête immédiatement), sauf lorsqu'il s'agit des actions `Prepare`, `Activate`, `New` et `New Channel`;
- si une cible appartenant à un ensemble de cibles d'une action élémentaire n'existe pas, l'action n'est ignorée que pour cette cible;
- si une référence spécifie un ensemble d'objets, de rt-objets ou de canaux MHEG vide, la référence est ignorée
- si la référence est ignorée, le moteur MHEG se comporte comme si aucune entité n'est adressée
- si, à un moment donné, une action `Prepare`, `New` ou `New Channel` s'applique à une entité MHEG dont l'identificateur est déjà affecté à une autre entité connue du moteur MHEG, cette action est ignorée;
- si une action élémentaire est appliquée à un rt-objet en phase R2 avec un paramètre de durée de transition, la valeur du paramètre de transition spécifié est ignorée, et l'action élémentaire est exécutée comme si la durée de transition indiquée était égale à 0;
- si une action élémentaire est appliquée à un rt-objet en phase R3 avec un paramètre de durée de transition, la valeur du paramètre de transition spécifiée est interprétée par l'unité temporelle générique affectée à la cible. La présentation de la cible change graduellement en fonction de l'action élémentaire appliquée et des modifications apportées à l'attribut de présentation.
- si le paramètre de durée de transition a une valeur négative, il est alors mis à 0.
- la présente Recommandation ne définit pas le comportement du moteur MHEG lors de l'échec d'une ou plusieurs actions au cours de l'exécution d'un groupe d'actions synchronisées en mode séquentiel.

Si un effet utilisateur est associé à une action élémentaire, cela est expliqué dans le paragraphe qui la décrit. Si le paragraphe ne donne aucune explication sur cet effet, alors aucun effet n'est associé à l'action élémentaire.

### 32.2 Actions Get

Les actions `get` possèdent les effets et traitements communs suivants:

- si une référence à une cible d'une action `get` est non valide, le résultat doit être évalué à «non défini» (excepté pour les actions `Get Preparation Status` – extraction de l'état de préparation –, `Get Activation Status` – extraction de l'état d'activation –, `Get Rt-availability Status` – extraction de l'état de rt-disponibilité – et `Get Channel Availability` – extraction de la disponibilité de canal –);
- si un moteur MHEG ne peut pas évaluer une action `get`, cette action est évaluée à «non défini»;
- si plusieurs entités sont adressées par une seule référence (postambule d'enfant, postambule de descendant, par exemple), alors cette référence est ignorée. La cible d'une action `get` ne doit adresser qu'une seule entité;
- si la référence est ignorée, le moteur MHEG se comporte comme si aucune entité n'était pointée.

### 32.3 Traitement recommandé pour les exceptions

La présente Recommandation ne définit aucun traitement d'exception dans le moteur MHEG, mais recommande les traitements suivants dans certaines situations:

- 1) si un objet dont le codage n'est pas conforme à la présente Recommandation est fourni au moteur MHEG, il est recommandé d'ignorer ce message;
- 2) si des entités inexistantes sont pointées par référence, il est recommandé de procéder de la manière suivante en fonction de la référence non valide à l'exception des cibles des actions élémentaires et des actions `get`:
  - a) dans une condition de lien, toutes les conditions sont évaluées à «non défini»
  - b) dans les autres cas (par exemple dans un élément de composition), elles sont remplacées par des objets nuls.

Ces comportements de remplacement sont retirés dès l'instanciation de l'objet pointé par référence.

- 3) si un objet reçoit une action dont le comportement n'est pas défini dans la présente Recommandation, il est recommandé d'ignorer cette action (action `Delete` appliquée à un objet *contenu* ou action `Set` avec une valeur non prise en charge appliquée à un rt-objet,...).

Le comportement peut être traité dans une extension de la présente Recommandation définie par une autre norme ou une application utilisatrice.

- 4) s'il apparaît d'autres exceptions, par exemple: demande d'un aspect de présentation non pris en charge par l'environnement de présentation ou défaillance d'exécution due par exemple à une insuffisance de mémoire, il est souhaitable de prévoir une possibilité de repli sur des modes de fonctionnement dégradés progressifs.

## 33 Comportement différé

Ce comportement diffère l'exécution d'actions quelconque. Initialement, les actions sont exécutées immédiatement.

### 33.1 Attributs et états du comportement

Aucun.

### 33.2 Actions permettant de modifier le comportement

La présente Recommandation définit les moyens de modifier ce comportement par l'utilisation des actions suivantes:

- action `Delay` (voir 33.2.1)

#### 33.2.1 Action `Delay`

Cette action insère un délai entre deux actions d'un ensemble d'actions synchronisées «série» dans un objet *action*.

Elle comporte les paramètres suivants:

- paramètre de référence d'unité temporelle;
- paramètre de durée.

Cette action peut être appliquée au cours d'une phase quelconque.

##### 33.2.1.1 Effet de l'action `Delay`

L'effet MHEG est le suivant:

- 1) si la référence d'unité temporelle est spécifiée comme un rt-composant, le délai s'exprime dans l'unité temporelle générique d'origine (OGTU) du rt-composant;  
Lorsque la référence d'unité temporelle indique un rt-contenu, le délai est interprété selon les règles de la chaîne des espaces OPS-RPS-CPS exactement comme tous les autres attributs temporels.
- 2) dans les autres cas (la référence d'unité temporelle est spécifiée comme le facteur `GF` par défaut), le délai s'exprime en unité temporelle physique;
- 3) si l'action `Delay` fait partie d'un ensemble d'actions synchronisées «série», le moteur MHEG attend la durée spécifiée avant d'exécuter la prochaine action;
- 4) l'effet MHEG prend fin quand le délai expire.

### 33.2.1.2 Autres conditions d'erreurs liées à l'application de l'action Delay

- Si la valeur du paramètre de durée est négative, le paramètre est mis à 0.
- Si le rt-composant spécifié n'existe pas, le facteur GTF extrait est interprétée comme la valeur GF par défaut.

## 34 Comportement de renvoi

Ce comportement renvoie une information à une application utilisatrice ou à des entités externes situées hors du moteur MHEG, par exemple un moteur script, une interface GUI ou un serveur.

### 34.1 Attributs et états de comportement

Aucun.

### 34.2 Actions servant à modifier le comportement

La présente Recommandation définit les moyens de modifier ce comportement par l'action suivante:

- action Return (retour) (voir 34.2.1).

#### 34.2.1 Action Return

Cette action permet de renvoyer une information à une application utilisatrice ou à des entités externes situées hors du moteur MHEG. Il est possible de faire passer des valeurs génériques ou une information contenue dans les objets *contenu*.

Cette action comporte les paramètres suivants:

- ensemble de paramètres de cibles de retour
- paramètre d'indicateur de retour
- ensemble de paramètres de valeur générique retournée
- ensemble de paramètres de référence d'objet *contenu*

Cette action peut être appliquée au cours d'une phase quelconque.

#### 34.2.1.1 Effet de l'action Return

L'effet MHEG de l'action Return est la suivante:

passage à l'entité spécifiée par les paramètres de cibles de retour des paramètres d'indicateur de retour, de valeur générique retournée et de référence d'objet *contenu*. La présente Recommandation ne définit pas le format de codage de l'information retournée.

#### 34.2.1.2 Autres conditions d'erreurs liées à l'application de l'action Return

Aucune.

## 35 Comportement de synonymie

Ce comportement identifie de façon symbolique toutes les références génériques. Initialement, il n'est pas donné d'alias aux références.

### 35.1 Attributs et états de comportement

Aucun.

### 35.2 Actions servant à modifier le comportement

La présente Recommandation définit les moyens de modifier ce comportement par l'action suivante:

- action set alias (création d'un alias) (voir 35.2.1)



### 35.2.1 Action set alias

Cette action permet d'affecter des alias à toutes les références génériques quelconques.

Elle comporte les paramètres suivants:

- ensemble de paramètres de cibles;
- ensemble de paramètres de spécification d'alias.

Cette action peut être appliquée au cours d'une phase quelconque.

#### 35.2.1.1 Effet de l'action Set Alias

L'effet de cette action est le suivant:

- 1) la commande Update (mise à jour), permet d'ajouter, de retirer ou de remplacer un ou plusieurs alias;
- 2) si une cible n'est pas accessible au moment de l'application de cette action, l'alias est affecté à la référence comme si la cible était accessible.

#### 35.2.1.2 Autres conditions d'erreurs liées à l'application de l'action Set Alias

Aucune.

## 36 Comportement d'extensibilité

Ce comportement permet l'exécution d'actions élémentaires cataloguées, ou la modification ou l'extraction d'une valeur d'attribut catalogué (voir le paragraphe 15).

Les actions élémentaire cataloguées visent à réaliser principalement les comportements dans lesquels, contrairement aux attributs, un contrôle de progressivité n'est pas possible (tracé d'une ligne, exécution d'opérations arithmétiques, arrêt du moteur MHEG,...)

Par ailleurs, les attributs catalogués visent à traiter principalement les informations susceptibles d'être exprimé par des attributs, par exemple: les attributs de présentation comme la taille des polices ou les attributs de canal comme le nombre courant de rt-composants affectés.

L'auteur doit être en mesure de distinguer ces deux situations.

Le comportement de styles est aussi un genre d'extension consacré à la présentation et à l'interaction. Par exemple, la conversion du format de contenu (données codées) est une caractéristique des actions élémentaires cataloguées, tandis que la présentation d'un contenu de différentes manières est une caractéristique de comportement de styles (code musical exprimé de façon graphique ou sonore).

### 36.1 Attributs et états de comportement

Ce comportement s'exprime au moyen des attributs et états suivants:

- attributs catalogués (voir 36.2);

### 36.2 Attribut catalogué

Entrée de catalogue représentant une liste d'informations qui peuvent être, par exemple, des attributs de présentation, des tailles de polices, des attributs de canal, .... Les attributs catalogués peuvent servir comme une forme de postambule pour naviguer dans l'arborescence des catalogues.

### 36.3 Actions servant à modifier le comportement

La présente Recommandation définit les moyens de modifier ce comportement par les actions suivantes:

- action élémentaire cataloguée (voir 36.3.1);
- action Set Catalogued Attribute (positionnement d'attribut catalogué) (voir 36.3.2).

### **36.3.1 Action élémentaire cataloguée**

Cette action permet d'utiliser une action élémentaire étendue enregistrée ou hors norme.

Elle comporte les paramètres suivants:

- ensemble de paramètres de cibles;
- paramètres d'action élémentaire étendue cataloguée;
- ensemble de paramètres d'actions élémentaires.

Cette phase est décrite dans le catalogue dont est extraite l'action élémentaire étendue.

#### **36.3.1.1 Effet de l'action élémentaire cataloguée**

L'effet de cette action est décrit dans le catalogue dont est extraite l'action élémentaire.

#### **36.3.1.2 Autres conditions d'erreurs liées à l'application d'une action élémentaire cataloguée**

- Les autres erreurs sont aussi décrites dans le catalogue dont est extraite l'action élémentaire étendue.

### **36.3.2 Action Set Catalogued Attribute (positionnement de l'attribut catalogué)**

Cette action permet de positionner une valeur d'attribut catalogué pour la cible.

Elle comporte les paramètres suivants:

- ensemble de paramètres de cible;
- paramètre d'attribut étendu catalogué;
- paramètre de valeur d'attribut étendu: valeur générique à affecter à l'attribut étendu catalogué;
- paramètre de durée de transition: valeur générique à affecter à un attribut étendu catalogué.

Cette phase est décrite dans le catalogue dont l'action élémentaire étendue est extraite.

#### **36.3.2.1 Effet de l'action Set Catalogued Attribute**

L'effet de cette action dépend des attributs sélectionnés, il est décrit dans le catalogue correspondant.

#### **36.3.2.2 Autres conditions d'erreurs liées à l'application de l'action set catalogued Attribute**

- Les autres erreurs sont aussi décrites dans le catalogue dont l'action élémentaire étendue est extraite.

## **36.4 Actions servant à extraire le comportement**

La présente Recommandation définit le moyen d'extraire ce comportement par l'action suivante:

- action `Get Catalogued Attribute` – extraction d'attribut catalogué – (voir 36.4.1)

### **36.4.1 Action Get Catalogued Attribute**

Cette action permet d'extraire une valeur de l'attribut catalogué pour la cible.

Elle comporte les paramètres suivants:

- ensemble de paramètre de cibles,
- paramètre d'attribut étendu catalogué: valeur générique à affecter à un attribut étendu catalogué

Cette phase est décrite dans le catalogue dont l'action élémentaire étendue est extraite.

#### **36.4.1.1 Effet de l'action Get catalogued Attribute**

Extraction de l'attribut catalogué.

#### **36.4.1.2 Autres conditions d'erreurs liées à l'application de l'action Get Catalogued Attribute**

- Les autres erreurs sont aussi décrites dans le catalogue dont est extraite l'action élémentaire.

## SECTION 5 – COMPORTEMENT DES OBJETS MHEG

### 37 Comportement de disponibilité des objets MHEG

Ce comportement sert à exprimer la disponibilité d'un objet MHEG pour le moteur MHEG. Ce comportement est commun à tous les objets MHEG.

#### 37.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Preparation Status` (état de préparation) (voir 37.2).

#### 37.2 `Preparation Status` (état de préparation)

Il est possible de déterminer la disponibilité de chaque objet MHEG pour le moteur MHEG. Le résultat de cette détermination est mis dans l'état de préparation. On applique ce qui suit (voir la Figure 31):

- 1) Lorsqu'un objet MHEG n'est pas à la disposition du moteur MHEG, son état de préparation vaut «pas prêt», c'est-à-dire que l'objet MHEG est en phase O1. Initialement tout objet MHEG est dans cet état.

NOTE 1 – Par exemple dans cet état, l'objet n'a pas été transféré au système de réception, un pré-traitement est nécessaire afin de préparer l'objet pour une présentation, etc.

- 2) Sinon, son état de préparation vaut «prêt», c'est-à-dire que l'objet MHEG est en phase O2.

NOTE 2 – Un moteur MHEG peut offrir à l'application utilisatrice des messages plus détaillés de diagnostics et d'informations, par exemple, pour expliquer qu'il est impossible de situer un objet ou pour indiquer les conditions d'erreur. Ces messages ne sont pas définis dans la présente Recommandation mais peuvent faire l'objet d'autres normes.

#### 37.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Prepare` (préparation) (voir 37.3.1),
- action `Destroy` (destruction) (voir 37.3.2).

##### 37.3.1 Action `Prepare` (préparation)

Cette action met l'objet MHEG à la disposition du moteur MHEG.

###### NOTES

1 – Par exemple, la récupération d'une séquence audiovisuelle à partir d'un disque peut prendre tellement de temps qu'il peut être efficace de commencer à charger cette séquence avant qu'elle soit nécessaire.

2 – Une application utilisatrice ou un profil particulier définira un comportement concret de cette action. Par exemple, un objet *contenu* qui comporte un important volume de données peut être chargé soit pendant l'effet MHEG de l'action `Prepare` soit en dehors de cet effet, selon la stratégie adoptée.

Cette action a le paramètre suivant:

- ensemble de paramètres `MH-Target Param` (paramètre de MH cible).

Cette action peut être appliquée pendant la phase O1.

##### 37.3.1.1 Effet de l'action `Prepare`

Si l'état de préparation de la cible vaut «pas prêt», on applique ce qui suit:

- 1) Mettre la cible à la disposition du moteur MHEG.
- 2) Exécuter les effets particuliers de l'action `Prepare` sur les cibles suivantes:
  - a) Objet *action*: des actions `Prepare` implicites sont appliquées aux objets *actions* imbriqués (inclus ou cités en référence) dans un objet *action* enveloppant.
  - b) Objet *lien*: des actions `Prepare` implicites sont appliquées aux objets *actions* (inclus ou cités en référence) décrivant l'effet de lien.

Par préparation d'un objet *lien*, on ne sous-entend pas la préparation des cibles des actions d'extraction et des actions élémentaires figurant dans ce lien.

- c) Objet *script*: les données de script (incluses ou citées en référence) sont mises à la disposition du moteur MHEG.

La façon dont l'objet est mis à la disposition du moteur MHEG n'est pas définie dans la présente Recommandation. Par exemple, les données de script peuvent être compilées en code exécutable et éditées pour le moteur MHEG ou elles peuvent être sous une forme interprétée nécessitant un processus distinct.

- d) Objet *contenu*: les données de contenu (incluses ou citées en référence) sont mises à la disposition du moteur MHEG.

La façon dont l'objet est mis à la disposition du moteur MHEG n'est pas définie dans la présente Recommandation. Par exemple, les données de contenu peuvent se trouver dans la mémoire du moteur MHEG ou être mises à sa disposition lorsque c'est nécessaire via un réseau de télécommunication ou de radiodiffusion.

- e) Objet *composite*:

i) Des actions `Prepare` implicites sont appliquées aux objets *liens* et *actions* inclus dans le comportement donné.

ii) Des actions `Prepare` implicites sont appliquées aux objets modèles associés inclus dans les éléments de composition.

iii) Des actions `Activate` (activation) implicites sont appliquées aux liens de démarrage de la disponibilité, de démarrage de la rt-disponibilité, d'arrêt de la disponibilité et d'arrêt de la rt-disponibilité.

- 3) NOTE 1 – Le concepteur d'objet ne peut spécifier une préparation et une activation explicites des objets *liens* et *actions* que lorsqu'il est nécessaire d'inclure ce comportement dans le moteur MHEG ou afin de tenir compte des besoins des systèmes à ressources minimales.

- 4) NOTE 2 – Le concepteur d'objet ne peut spécifier une préparation explicite des objets modèles associés cités en référence que lorsqu'il est nécessaire de créer un rt-composite. Cela tient compte des besoins des systèmes à ressources minimales.

- a) Objet *conteneur*: des actions `Activate` implicites sont appliquées aux liens de démarrage de conteneur et d'arrêt de conteneur.

Le concepteur d'objet peut spécifier, dans ces liens, un séquençement explicite de préparation des objets, et par-là même mettre ces objets à la disposition du moteur MHEG lorsqu'ils sont nécessaires, afin de tenir compte des besoins des systèmes à ressources minimales.

- b) Objet *descripteur*: les informations du descripteur (incluses ou citées en référence) sont mises à la disposition du moteur MHEG.

Le moteur MHEG doit évaluer les informations fournies par le descripteur pendant l'exécution de l'action `Prepare`. La façon dont les informations sont traitées dépend de la réalisation du moteur et du contexte d'exécution.

- 5) Mettre l'état de préparation de la cible à «prêt». La cible entre en phase O2. Des effets supplémentaires doivent être exécutés sur les cibles suivantes:

- a) Objet *contenu*: les comportements suivants sont disponibles:

i) Comportement de stockage de valeur générique;

ii) Comportement de copie (voir 72.9.1).

- b) Objet *composite*: le lien de démarrage de disponibilité est déclenché, c'est-à-dire que son effet de lien est exécuté.

- c) Objet *conteneur*: le lien de démarrage de conteneur est déclenché, c'est-à-dire que son effet de lien est exécuté.

### 37.3.1.2 Conditions d'erreur supplémentaires de l'action `Prepare`

- Si l'une des cibles est un élément indexé à l'intérieur d'un objet *conteneur*, il faut préparer l'objet *conteneur* extérieur. S'il n'est pas prêt, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un objet MHEG en phase O1, cette action est ignorée pour cette cible.

### 37.3.2 Action Destroy (destruction)

Cette action supprime un objet MHEG du moteur MHEG afin de libérer des ressources (par exemple de la mémoire, une certaine largeur de bande ou une certaine puissance de calcul) dans le moteur MHEG.

Cette action a le paramètre suivant:

- ensemble de paramètres MH-Target Param (paramètre de MH cible).

Cette action peut être appliquée pendant la phase O2.

#### 37.3.2.1 Effet de l'action Destroy

Si l'état de préparation de la cible vaut «prêt»; on applique ce qui suit:

- 1) exécuter les effets spécifiés de l'action Destroy sur les cibles suivantes:

- a) objet *action*:

- i) des actions Destroy implicites sont appliquées aux objets *actions* imbriqués (inclus ou cités en référence) dans un objet *action* enveloppant.

NOTE 1 – Les concepteurs d'objet doivent savoir que des effets secondaires peuvent se produire si ces objets sont cités en référence dans d'autres objets;

- ii) si un objet *action* à détruire est en cours d'exécution, l'exécution est interrompue.

La procédure d'interruption de l'exécution n'est pas définie dans la présente Recommandation mais relève du moteur MHEG;

- iii) si un objet *action* enveloppant est détruit dans un objet *action*, cet objet *action* enveloppant est remplacé par un objet vide.

NOTE 2 – L'exécution d'un objet vide a un effet nul.

- b) objet *lien*:

- i) si l'effet de lien de la cible est en cours d'exécution, une action Link Abort (abandon de lien) implicite est appliquée au lien.

Un lien pouvant être déclenché plusieurs fois au cours de sa vie, plusieurs effets de lien peuvent être en cours d'exécution au même moment. L'ensemble de ces effets de lien sont concernés.

- ii) une action Destroy implicite est appliquée à chaque objet *action* inclus ou cité en référence.

NOTE 3 – Les concepteurs d'objet doivent savoir que des effets secondaires peuvent se produire si cet objet *action* est cité en référence dans d'autres objets.

NOTE 4 – La destruction d'un lien n'implique pas la destruction des cibles de ce lien.

- c) objet modèle: une action Delete (suppression) implicite est appliquée à tous les rt-objets existants créés à partir de cet objet modèle. L'état de rt-disponibilité de chaque rt-objet devient 'not-available' (pas disponible). Les autres comportements sont détruits.

NOTE 5 – Les concepteurs d'objet doivent savoir que les rt-objets disparaissent si l'objet modèle d'origine est détruit.

NOTE 6 – Le traitement de la disparition des rt-objets n'est pas défini dans présente Recommandation mais relève du moteur MHEG.

- d) objet *script*: les données de script (incluses ou citées en référence) sont détruites.

Le concepteur d'objet doit savoir que des effets secondaires peuvent se produire si ces données de script sont citées en référence dans un autre objet *script*.

- e) objet *contenu*: les données de contenu (incluses ou citées en référence) sont détruites.

Les concepteurs d'objet doivent savoir que des effets secondaires peuvent se produire si ces données d'objet *contenu* sont citées en référence dans un autre objet *contenu*.

- f) objet *composite*:

- i) des actions Destroy implicites sont appliquées aux liens de démarrage de disponibilité, de démarrage de rt-disponibilité et d'arrêt de rt-disponibilité;

- ii) des actions Destroy implicites sont appliquées aux objets *liens* et *actions* inclus dans le comportement donné;

- iii) des actions `Destroy` implicites sont appliquées aux objets *composants* inclus associés aux éléments de composition;
  - iv) les étiquettes et les objets modèles associés sont aussi détruits.
  - g) objet *conteneur*: une action `Destroy` implicite est appliquée au lien de démarrage de disponibilité.
- 2) mettre l'état de préparation de la cible à «pas prêt». La cible entre en phase O1. Des effets supplémentaires doivent être exécutés sur les cibles suivantes:
    - a) objet *lien*: l'état d'activation passe à «inactif».
    - b) objet modèle: l'état de disponibilité passe à 'not-available' (pas disponible).
    - c) objet *composite*: le lien d'arrêt de disponibilité est déclenché, c'est-à-dire que son effet de lien est exécuté. Après l'effet de lien, le lien d'arrêt de disponibilité est détruit.
    - d) objet *conteneur*: le lien d'arrêt de disponibilité est déclenché, c'est-à-dire que son effet de lien est exécuté. Après l'effet de lien, le lien d'arrêt de disponibilité est détruit.
  - 3) Détruire la cible.

### 37.3.2.2 Conditions d'erreur supplémentaires de l'action `Destroy`

- Si l'une des cibles n'est pas un objet MHEG en phase O2, cette action est ignorée pour cette cible.

## 37.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Preparation Status` (extraction de l'état de préparation) (voir 37.4.1).

### 37.4.1 Action `Get Preparation Status` (extraction de l'état de préparation)

Cette action extrait l'état de préparation de chaque objet MHEG.

Cette action a le paramètre suivant:

- `MH-Target Param` (paramètre de MH cible).

Cette action peut être appliquée pendant les phases: O1 et O2.

#### 37.4.1.1 Effet MHEG

Cette action détermine l'état de préparation de la cible. On applique ce qui suit:

- 1) si la cible est en phase O1, l'état de préparation vaut «pas prêt»;
- 2) sinon (la cible est en phase O2), l'état de préparation vaut «prêt».

#### 37.4.1.2 Conditions d'erreur supplémentaires de l'action `Get Preparation Status`

Aucune.

## 38 Comportement d'activation des objets *liens*

Ce comportement commande le traitement d'un objet *lien* dans le moteur MHEG. Tous les objets *liens* possèdent ce comportement. Les objets *liens* actifs sont surveillés par le moteur MHEG. Chaque fois qu'une condition de lien devient vraie, l'effet de lien correspondant est exécuté.

### 38.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Activation Status` (état d'activation) (voir 38.2).

## 38.2 Activation Status (état d'activation)

Après la préparation, il est possible de déterminer l'activité de chaque objet *lien* pour le moteur MHEG. Le résultat de cette détermination est mis dans l'état d'activation.

Lorsqu'un objet *lien* peut être déclenché, son état d'activation est mis à «actif», c'est-à-dire que l'objet *lien* est en phase L2.

Sinon, son état d'activation est mis à «inactif», c'est-à-dire que l'objet *lien* est en phase L1.

## 38.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Activate` (activation) (voir 38.3.1),
- action `Deactivate` (désactivation) (voir 38.3.2).

### 38.3.1 Action `Activate` (activation)

Cette action active les objets *liens*, c'est-à-dire que ces objets peuvent alors être déclenchés.

Cette action a le paramètre suivant:

- ensemble de paramètres `Link Target Param` (paramètre de lien cible).

Cette action peut être appliquée pendant la phase L1.

#### 38.3.1.1 Action `Activate` appliquée à un seul objet *lien*

L'effet MHEG de cette action est le suivant:

- si la cible est en phase O2/L1, l'état d'activation de la cible est mis à «actif»;
- si la cible est en phase O1, une action `Prepare` implicite est appliquée à cette cible. Après la préparation, l'état d'activation de la cible est mis à «actif».

Un objet *lien* ne peut être déclenché que si son état de préparation vaut «prêt» et si son état d'activation vaut «actif», c'est-à-dire si l'objet *lien* est en phase O2/L2. Une fois l'objet *lien* à l'état actif, l'effet de lien peut être exécuté plusieurs fois (c'est-à-dire chaque fois que la condition de lien devient Vrai) jusqu'à ce que l'objet *lien* soit mis à «inactif».

#### 38.3.1.2 Conditions d'erreur supplémentaires de l'action `Activate`

- Si l'une des cibles n'est pas un objet *lien* en phase O1 ou O2/L1, cette action est ignorée pour cette cible.

### 38.3.2 Action `Deactivate` (désactivation)

Cette action désactive les objets *liens*, c'est-à-dire que ces objets ne peuvent plus être déclenchés.

Cette action a le paramètre suivant:

- ensemble de paramètres `Link Target Param` (paramètre de lien cible).

Cette action peut être appliquée pendant la phase L2.

#### 38.3.2.1 Effet de l'action `Deactivate`

L'effet MHEG de cette action est le suivant:

- Si la cible est en phase L2, l'état d'activation de la cible est mis à «inactif».

#### 38.3.2.2 Conditions d'erreur supplémentaires de l'action `Deactivate`

- Si l'une des cibles n'est pas un objet *lien* en phase L2, cette action est ignorée pour cette cible.

## 38.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Activation Status` (extraction de l'état d'activation) (voir 38.4.1).

### 38.4.1 Action Get Activation Status (extraction de l'état d'activation)

Cette action récupère la valeur de l'état d'activation d'un objet lien.

Cette action a le paramètre suivant:

- Link Target Param (paramètre de lien cible).

Cette action peut être appliquée pendant les phases L1 et L2.

#### 38.4.1.1 Action Get Activation Status appliquée à un objet lien

L'effet MHEG de cette action consiste à récupérer l'état d'activation de la cible. On applique ce qui suit:

- 1) si la cible est en phase L1, l'état d'activation vaut «inactif»;
- 2) si la cible est en phase L2, l'état d'activation vaut «actif».

#### 38.4.1.2 Conditions d'erreur supplémentaires de l'action Get Activation Status

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un objet *lien*, cette action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

## 39 Comportement d'abandon des objets liens

Ce comportement entraîne l'abandon de l'exécution de toutes les actions figurant dans un objet *lien*.

### 39.1 Attributs et états du comportement

Aucun.

### 39.2 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action Link Abort (abandon de lien) (voir 39.2.1).

#### 39.2.1 Action Link Abort (abandon de lien)

Cette action abandonne l'exécution de toutes les actions qui figurent dans un objet *lien*.

Cette action a le paramètre suivant:

- ensemble de paramètres Link Target Param (paramètre de lien cible).

Cette action peut être appliquée pendant la phase L2.

##### 39.2.1.1 Effet de l'action Link Abort

L'effet de cette action est le suivant:

- 1) les actions élémentaires en cours d'exécution dans l'effet de lien de la cible sont abandonnées. Ces actions sont directement contenues dans l'effet de lien ou dans un objet *action* enveloppant. Les autres actions élémentaires contenues dans l'effet de lien ou dans des objets *actions* enveloppants ne sont pas exécutées;
- 2) si l'action élémentaire abandonnée procède à l'exécution de l'effet MHEG modifiant une valeur d'attribut, on applique ce qui suit:
  - a) si aucune durée de transition n'est spécifiée, la valeur n'est pas modifiée;
  - b) sinon (une durée de transition est spécifiée), la valeur passe à la valeur courante en cours de modification.



L'abandon porte sur le processus de la durée de transition. C'est pourquoi, la valeur d'attribut de présentation sur laquelle est appliquée la modification avec durée de transition s'arrête sur sa valeur courante. L'effet est différent de celui de l'action `Stop` appliquée pendant ce temps (voir Figure 42);

- 3) si l'action élémentaire abandonnée est l'action `Run` (exécution), une action `Stop` implicite est appliquée à la cible de l'action `Run`.

Si l'action élémentaire abandonnée a un ensemble de cibles, l'effet dépend de l'état d'exécution de cette action élémentaire pour chaque cible.

#### NOTES

1 – Supposons qu'une action `Prepare` soit appliquée aux objets A et B à l'intérieur du lien L1, que l'effet MHEG de l'action `Prepare` pour l'objet A soit déjà terminé (c'est-à-dire que l'objet A est en phase O2) et que l'effet MHEG de cette action pour l'objet B soit encore en cours d'exécution (c'est-à-dire que l'objet B est en phase O1). Dans ce cas, si une action `Link Abort` est appliquée au lien L1, l'état de préparation de l'objet A vaut «prêt» et celui de l'objet B vaut «pas prêt».

2 – L'action `Link Abort` n'a pas d'incidence sur l'état d'activation d'un lien. Un lien peut être déclenché même si une action `Link Abort` qui lui est appliquée est en cours d'exécution. L'action `Link Abort` n'a d'incidence que sur les actions engendrées par un lien avant que l'action `Link Abort` soit appliquée à ce lien.

3 – Il est conseillé de limiter dans la mesure du possible l'utilisation de l'action `Link Abort`. Son principal rôle est de mettre fin à des actions qui exécutent des effets MHEG avec des durées de transition situées dans la phase R3.TD.

#### 39.2.1.2 Conditions d'erreur supplémentaires de l'action `Link Abort`

- Si l'une des cibles n'est pas un objet *lien* en phase L2, cette action est ignorée pour cette cible.

## 40 Comportement de stockage de valeur générique de classe *contenu*

Ce comportement consiste à stocker et à récupérer une valeur générique dans les données d'un objet *contenu*.

### 40.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Data` (données) (voir 40.2).

### 40.2 `Data` (données)

Une valeur générique peut être stockée dans un objet *contenu* et récupérée à partir d'un tel objet pour un traitement ultérieur. Lorsqu'on utilise le comportement de stockage de valeur générique, un objet *contenu* peut servir à contenir une variable.

### 40.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Set Data` (fixation des données) (voir 40.3.1),
- action `Add` (ajout) (voir 40.3.2),
- action `Substract` (soustraction) (voir 40.3.3).

#### 40.3.1 Action `Set Data` (fixation des données)

Cette action permet de stocker ou de modifier une valeur générique dans le champ de données d'un objet *contenu*.

Cette action a les paramètres suivants:

- ensemble de paramètres `Content Target Param` (paramètre de contenu cible),
- `Substitution Indicator Param` (paramètre d'indication de substitution),
- ensemble de paramètres `Data Element Param` (paramètre d'élément de données).

Cette action peut être appliquée pendant la phase O2.

#### 40.3.1.1 Action Set Data appliquée à un seul objet contenu

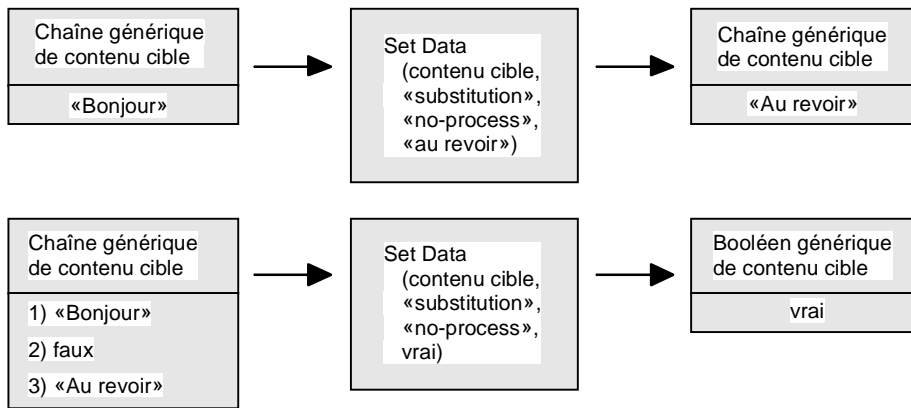
Si un ensemble de paramètres d'élément de données est fourni, les éléments de cet ensemble sont interprétés séquentiellement. Pour chaque paramètre d'élément de données, on applique l'effet suivant:

- 1) si le paramètre d'identification d'élément de liste générique est omis:
  - a) et si le paramètre d'indication de substitution vaut 'substitution', la valeur générique précédemment enregistrée est entièrement remplacée par la valeur générique spécifiée (voir Figure 48);
  - b) sinon (le paramètre d'indication de substitution vaut 'no-substitution'):
    - i) si la cible contient une liste générique, la valeur générique spécifiée est ajoutée comme nouvel élément dans la liste générique de la cible. Si la cible contient n éléments, la nouvelle valeur a pour indice n + 1 (voir Figure 49);
    - ii) sinon, cette action est ignorée;
- 2) sinon (le paramètre d'identification d'élément de liste générique est spécifié):
  - a) si la cible ne comporte pas de liste générique:
    - i) si le paramètre d'indication de substitution vaut 'substitution', la valeur précédemment stockée est rejetée et il faut créer une liste générique. La valeur générique spécifiée est stockée avec comme identificateur l'identificateur d'élément de liste générique spécifié, et les autres éléments dont l'identificateur se situe avant l'identificateur d'élément de liste générique spécifié doivent être remplis avec la valeur 'unspecified' (pas spécifié) (voir Figure 50);
    - ii) sinon, cette action est ignorée;
  - b) sinon (la cible comporte une liste générique):
    - i) si l'identificateur d'élément de liste générique spécifié existe dans la cible:
      - si l'indicateur de substitution vaut 'substitution', la valeur générique spécifiée est stockée avec comme identificateur l'identificateur d'élément spécifié, à la place de la valeur précédente (voir Figure 51);
      - sinon (l'indicateur de substitution vaut 'no-substitution'), cette action est ignorée;
    - ii) si l'identificateur d'élément de liste générique spécifié n'existe pas dans la cible et s'il est possible d'étendre la liste générique jusqu'à l'identificateur spécifié, la liste générique de la cible est étendue afin de stocker la valeur générique spécifiée. Les éléments vides créés par cette extension sont remplis avec la valeur 'unspecified' (pas spécifié). Dans ce cas, l'indicateur de substitution est ignoré (voir Figure 52);
    - iii) sinon (l'identificateur d'élément de liste générique spécifié n'existe pas et il est impossible d'étendre la liste), cette action est ignorée (voir Figure 53).

L'ordre des paramètres d'élément de données est important, en particulier si aucun identificateur d'élément de liste générique n'est spécifié pour la valeur générique dans le cas 'no-substitution' et si la cible possède une liste générique.

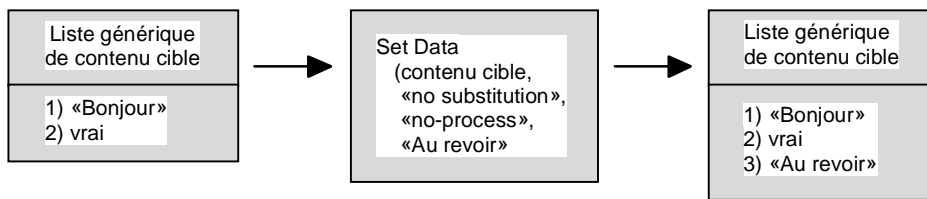
#### 40.3.1.2 Conditions d'erreur supplémentaires de l'action Set Data

- Si l'une des cibles n'est pas un objet *contenu* en phase O2, l'action est ignorée pour cette cible.
- Si l'ancre de l'objet *contenu* cible ne spécifie pas une valeur générique, l'action est ignorée.
- Si l'action Set Data contient une valeur d'élément qui est spécifiée comme une action Get Data (extraction des données) appliquée au même objet *contenu* cible, la valeur de cet élément est remplacée par la valeur 'unspecified' (pas spécifié).
- Si la cible n'est pas un contenu non multiplexé, l'action est ignorée pour cette cible.



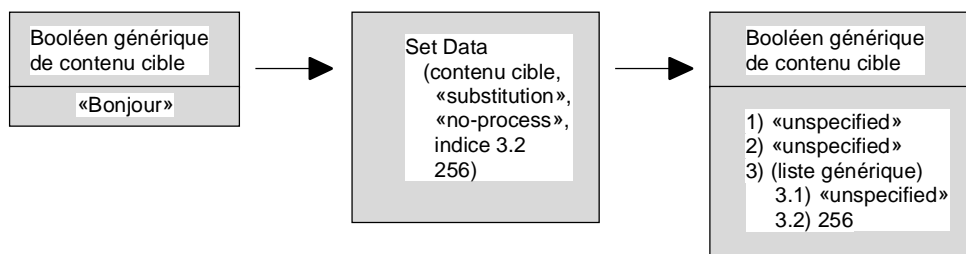
T0825930-96/d046

**Figure 48/T.171 – Remplacement par une valeur générique**



T0825940-96/d047

**Figure 49/T.171 – Ajout d'un élément à une liste générique**



T0825950-96/d048

**Figure 50/T.171 – Création de liste**

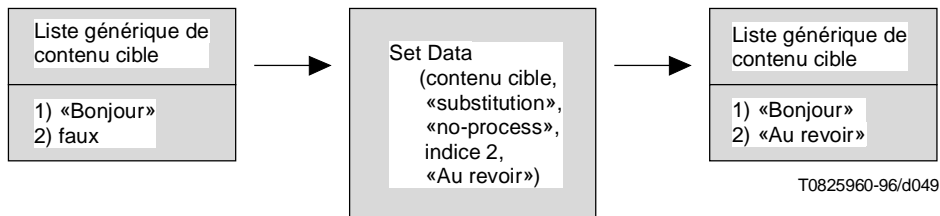


Figure 51/T.171 – Substitution d'un élément dans une liste générique

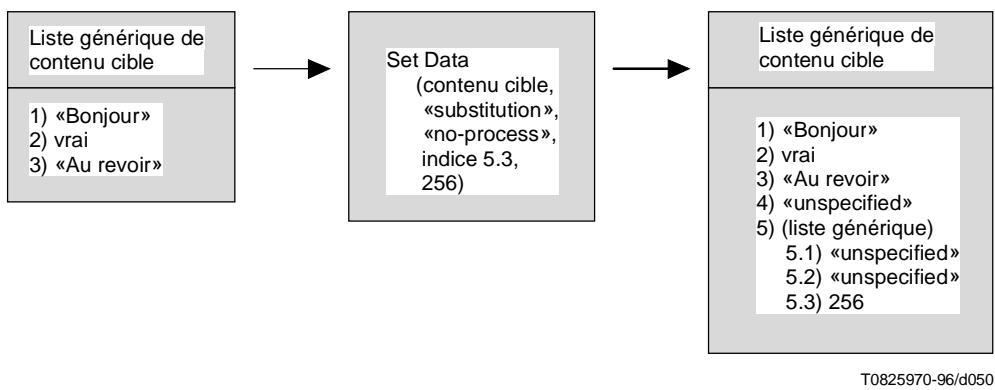


Figure 52/T.171 – Extension d'une liste

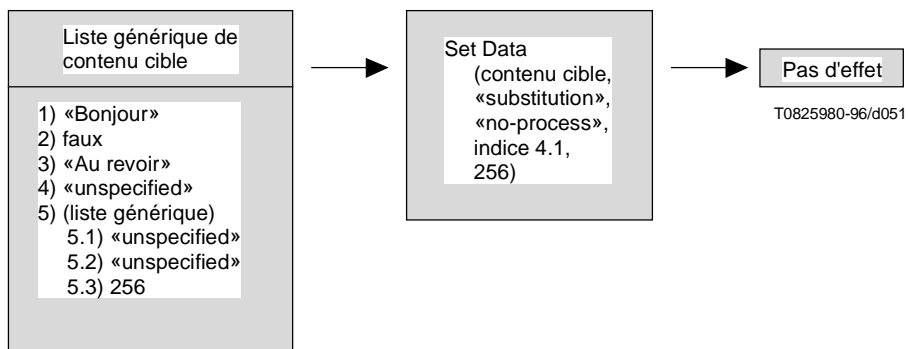


Figure 53/T.171 – Identificateur d'élément de liste générique incompatible

### 40.3.2 Action Add (ajout)

Cette action est une opération arithmétique qui ajoute un numérique générique, un entier générique ou une fraction générique à la valeur générique figurant dans le champ de données de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres Content Target Param (paramètre de contenu cible),
- Generic List Elt ID Param (paramètre d'identification d'élément de liste générique),
- Generic Value Param (paramètre de valeur générique).

Cette action peut être appliquée pendant la phase O2.

#### 40.3.2.1 Effet de l'action Add

Si le paramètre d'identification d'élément de liste générique est omis, cette action doit être appliquée à un objet *contenu* qui ne contient pas de liste générique. Sinon, cette action doit être appliquée à un objet *contenu* qui contient une liste générique et l'identificateur d'élément de liste générique spécifié doit exister.

Le type de valeur générique des données du contenu cible et le type de la valeur générique spécifiée doivent être identiques, et doivent être des numériques génériques, des entiers génériques ou des fractions génériques.

Cette action a les effets suivants:

- 1) si le paramètre de valeur générique est fourni, on ajoute le paramètre de valeur générique aux données de contenu ou aux données d'élément spécifiées par le paramètre d'identification d'élément de liste générique;
- 2) sinon, on incrémente de 1 les données de contenu ou les données d'élément spécifiées par le paramètre d'identification d'élément de liste générique (numérique générique valant 1, entier générique valant 1 ou fraction générique valant 1 selon le type de valeur générique de l'objet *contenu* cible).

#### 40.3.2.2 Conditions d'erreur supplémentaires de l'action Add

- Si l'une des cibles n'est pas un objet *contenu* en phase O2, l'action est ignorée pour cette cible.
- Si le paramètre d'identification d'élément de liste générique est omis et si la cible possède une liste générique, l'action est ignorée.
- Si le paramètre d'identification d'élément de liste générique est fourni et si la cible ne possède pas de liste générique, l'action est ignorée.
- Si le paramètre d'identification d'élément de liste générique est fourni et si la cible n'a pas l'élément spécifié, l'action est ignorée.
- Si le type de la valeur générique des données du contenu cible et si celui de la valeur générique spécifiée sont différents, l'action est ignorée.
- Si le type de la valeur générique des données du contenu cible n'est ni numérique générique, ni entier générique, ni fraction générique, l'action est ignorée.
- Si la cible n'est pas un contenu non multiplexé, l'action est ignorée pour cette cible.

### 40.3.3 Action Subtract (soustraction)

Cette action est une opération arithmétique qui soustrait un numérique générique, un entier générique ou une fraction générique de la valeur générique figurant dans le champ de données de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres Content Target Param (paramètre de contenu cible),
- Generic List Elt ID Param (paramètre d'identification d'élément de liste générique),
- Generic Value Param (paramètre de valeur générique).

Cette action peut être appliquée pendant la phase O2.

#### 40.3.3.1 Action Subtract appliquée à un seul objet *contenu*

Si le paramètre d'identification d'élément de liste générique est omis, cette action doit être appliquée à un objet *contenu* qui ne contient pas de liste générique. Sinon, cette action doit être appliquée à un objet *contenu* qui contient une liste générique et l'identificateur d'élément de liste générique spécifié doit exister.

Le type de valeur générique des données du contenu cible et le type de la valeur générique spécifiée doivent être identiques, et doivent être des numériques génériques, des entiers génériques ou des fractions génériques.

Cette action a les effets suivants:

- 1) si le paramètre de valeur générique est fourni, on retranche le paramètre de valeur générique aux données de contenu ou aux données d'élément spécifiées par le paramètre d'identification d'élément de liste générique;
- 2) sinon, on décrémente de 1 les données de contenu ou les données d'élément spécifiées par le paramètre d'identification d'élément de liste générique (numérique générique valant 1, entier générique valant 1 ou fraction générique valant 1 selon le type de valeur générique de l'objet *contenu* cible).

#### 40.3.3.2 Conditions d'erreur supplémentaires de l'action `Subtract`

- Si l'une des cibles n'est pas un objet *contenu* en phase O2, l'action est ignorée pour cette cible.
- Si le paramètre d'identification d'élément de liste générique est omis et si la cible possède une liste générique, l'action est ignorée.
- Si le paramètre d'identification d'élément de liste générique est fourni et si la cible ne possède pas de liste générique, l'action est ignorée.
- Si le paramètre d'identification d'élément de liste générique est fourni et si la cible n'a pas l'élément spécifié, l'action est ignorée.
- Si le type de la valeur générique des données du contenu cible et si celui de la valeur générique spécifiée sont différents, l'action est ignorée.
- Si le type de la valeur générique des données du contenu cible n'est ni numérique générique, ni entier générique, ni fraction générique, l'action est ignorée.
- Si la cible n'est pas un contenu non multiplexé, l'action est ignorée pour cette cible.

## 40.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Data` (extraction des données) (voir 40.4.1).

### 40.4.1 Action `Get Data` (extraction des données)

Cette action récupère une valeur générique ou un élément de liste générique stocké dans un objet *contenu*.

Cette action a les paramètres suivants:

- `Content Target Param` (paramètre de contenu cible),
- `Generic List Elt ID Param` (paramètre d'identification d'élément de liste générique).

Cette action peut être appliquée pendant la phase O2.

#### 40.4.1.1 Action `Get Data` appliquée à un objet *contenu*

L'effet MHEG de cette action consiste à récupérer une valeur générique ou un élément de liste générique stocké dans un objet *contenu*.

Cette action détermine une valeur générique comme suit:

- 1) si aucun paramètre d'identification d'élément de liste générique n'est spécifié, l'action détermine la valeur générique entière stockée dans l'objet *contenu* cible;
- 2) sinon, l'action détermine la valeur générique stockée à l'endroit spécifié par le paramètre d'identification d'élément de liste générique dans l'objet *contenu* cible.

#### 40.4.1.2 Conditions d'erreur supplémentaires de l'action `Get Data`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un objet *contenu* en phase O2, l'action est ignorée pour cette cible.
- Si la cible n'est pas un objet *contenu* qui ne contient pas de valeur générique, mais qui contient par exemple une image fixe ou des données vidéo, l'action est ignorée pour cette cible.

- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'indice d'élément de liste est incompatible, c'est-à-dire s'il indique un chemin non existant dans les données, l'action renvoie «non défini».
- Si la cible n'est pas un contenu non multiplexé, l'action renvoie «non défini».

## 41 Comportement de copie de classe *contenu*

Ce comportement génère de nouveaux objets *contenus* à partir d'un objet *contenu* existant.

Ce comportement permet de copier un objet *contenu* modèle pour donner d'autres objets *contenus* indépendants. Les objets *contenus* copiés ont des identificateurs indépendants fournis par le concepteur d'objet. Ils sont complètement indépendants de l'objet *contenu* d'origine, c'est-à-dire qu'une modification quelconque de l'objet d'origine n'a pas d'incidence sur les objets *contenus* copiés.

### 41.1 Attributs et états du comportement

Aucun.

### 41.2 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Copy` (copie) (voir 41.2.1).

#### 41.2.1 Action `Copy` (copie)

Cette action spécifie un objet *contenu* cible comme source de l'opération de copie et un ensemble d'objets *contenus* comme destinations de cette même opération.

Cette action a les paramètres suivants:

- `Content Target Param` (paramètre de contenu cible),
- ensemble de paramètres `Destination Param` (paramètre de destination).

Cette action peut être appliquée pendant la phase O2.

##### 41.2.1.1 Effet de l'action `Copy`

L'effet de cette action est le suivant:

- 1) si l'objet *contenu* spécifié comme paramètre de destination n'existe pas, il est créé. Si la destination est spécifiée par un identificateur MHEG, un objet *contenu* comportant l'identificateur MHEG est créé;
- 2) tous les attributs de l'objet *contenu* cible à l'exception de l'identificateur MHEG sont copiés dans chaque objet *contenu* de destination.

Un objet *contenu* cible peut être spécifié par un élément de conteneur qui est l'objet *contenu*. Mais un objet *contenu* de destination ne sera pas spécifié par un élément de conteneur même si c'est un objet *contenu*.

Aucun objet *contenu* de destination ne sera un rt-contenu au moment de l'action `Copy`.

##### 41.2.1.2 Conditions d'erreur supplémentaires de l'action `Copy`

- Si l'une des cibles n'est pas un objet *contenu* en phase O2, l'action est ignorée pour cette cible.
- Si des rt-contenus sont créés à partir d'un objet *contenu* cité en référence dans un paramètre de destination, l'action est ignorée pour cette destination.
- Si la référence à un objet *contenu* citée dans un paramètre de destination est spécifiée comme une référence à un élément de conteneur, l'action est ignorée pour cette destination.
- Si la cible n'est pas un contenu non multiplexé, l'action est ignorée pour cette cible.

## SECTION 6 – COMPORTEMENT DES RT-OBJETS

### 42 Comportement de disponibilité des rt-objets

Ce comportement indique si les rt-objets sont à la disposition du moteur MHEG pour être exécutés. Tous les rt-objets possèdent ce comportement.

#### 42.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Rt-Availability Status` (état de rt-disponibilité) (voir 42.2).

#### 42.2 `Rt-Availability Status` (état de rt-disponibilité)

L'état de rt-disponibilité d'un rt-objet vaut «disponible» si ce rt-objet est à la disposition du moteur MHEG pour être exécuté, c'est-à-dire s'il est créé à partir d'un objet modèle et s'il est en phase R2. L'état de rt-disponibilité d'un rt-objet vaut «pas disponible» si ce rt-objet n'est pas créé ou s'il est supprimé, c'est-à-dire s'il est en phase R1.

Initialement, l'état de rt-disponibilité de chaque rt-objet est à «pas disponible».

#### 42.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `New` (nouveau) (voir 42.3.1),
- action `Delete` (suppression) (voir 42.3.2).

##### 42.3.1 Action `New` (nouveau)

Cette action crée des rt-objets à partir d'objets modèles. Elle sert à obtenir une vue particulière de l'objet modèle pour une exécution et éventuellement pour une perception par l'utilisateur. Le processus de création de rt-objet, qui sort du cadre de la présente Recommandation, est fourni par le moteur MHEG.

Cette action a le paramètre suivant:

- ensemble de paramètres `Rt-Target Param` (paramètre de rt-cible).

Cette action peut être appliquée pendant la phase R1.

##### 42.3.1.1 Effet de l'action `New`

L'effet MHEG est exécuté immédiatement. Aucun effet utilisateur n'est associé. L'effet MHEG de cette action est le suivant:

- 1) Construire le rt-objet cible.  
Il appartient au moteur MHEG de décider s'il est nécessaire de copier les données de l'objet modèle.
- 2) Affecter l'identificateur du rt-objet. Il s'agit de la cible de l'action `New`.
- 3) Exécuter des effets particuliers de l'action `New` sur les objets suivants:
  - a) `rt-script`:
    - i) initialiser le comportement de passage de paramètres;
    - ii) mettre le comportement de terminaison du `rt-script` à «pas terminé»;
  - b) `rt-composant`:
    - i) initialiser les comportements de présentation;
    - ii) initialiser le comportement d'interaction;



- c) *rt-composite*:
- i) construire chaque composant dans le *rt-composite*.  
Chaque objet *composite* représente une génération. La construction est récursive lorsqu'un objet *composite* est associé à un élément de composition. Grâce à cette fonctionnalité, le moteur MHEG combine toutes les générations pour construire un *rt-composite* complet;
  - ii) appliquer l'action `Plug` (enfichage) à chaque réceptacle du *rt-composite* – Les données à enficher sont les données se trouvant dans le modèle associé de l'élément correspondant. Lorsque le modèle associé est vide, il faut enficher un *rt-objet* vide dans le réceptacle correspondant.
- 4) NOTE 1 – L'action `Plug` (enfichage) est appliquée automatiquement par le moteur MHEG. L'auteur n'a pas besoin de spécifier cette action. Il s'agit d'un avantage de la fonctionnalité de modèle associé dans l'objet *composite*.
  - 5) NOTE 2 – A vrai dire, on ne peut pas enficher une étiquette ou un objet *composant* directement dans un réceptacle. Lorsqu'une étiquette est associée, le moteur MHEG doit créer un objet *contenu* avec l'étiquette pour données puis créer un *rt-contenu* à partir de ce contenu. Lorsqu'un composant est associé, le moteur MHEG exécute une action `New` implicite pour créer un *rt-composant* à partir de ce composant puis il enfiche ce *rt-composant* dans le réceptacle.
  - 6) NOTE 3 – L'action `New` (nouveau) est propagée à la génération suivante dans le *rt-composite* via l'effet de l'action `Plug` (enfichage):
    - initialiser le comportement de dynamisme de présentation et de structure des réceptacles;
    - initialiser les comportements de présentation.
  - 7) Initialiser le comportement d'exécution à «pas en cours d'exécution».
  - 8) Fixer l'état de *rt-disponibilité* du *rt-objet* cible à «disponible». Le *rt-objet* créé entre en phase R2.
  - 9) Déclencher le lien de démarrage de *rt-disponibilité* dans un objet *composite*.

Lorsque l'objet modèle d'origine est en phase O1, une action `Prepare` (préparation) implicite est appliquée à l'objet modèle puis l'action `New` est appliquée comme ci-dessus.

#### 42.3.1.2 Conditions d'erreur supplémentaires de l'action `New`

- Si, dans l'une des cibles, l'identificateur d'objet d'origine désigne un objet qui n'est pas un objet modèle, l'action est ignorée pour cette cible.
- Si l'une des cibles ne désigne pas un *rt-objet*, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un *rt-objet* en phase R1, l'action est ignorée pour cette cible.

#### 42.3.2 Action `Delete` (suppression)

Cette action supprime des *rt-objets* du moteur MHEG. Elle sert à libérer des ressources dans le moteur MHEG. Cette action n'affecte pas les objets modèles à partir desquels les *rt-objets* sont créés.

Cette action a le paramètre suivant:

- ensemble de paramètres `Rt-Target Param` (paramètre de *rt-cible*).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 42.3.2.1 Effet de l'action `Delete`

Aucun effet utilisateur direct n'est associé; toutefois, il peut y avoir un effet utilisateur indirect dans certains cas. L'effet MHEG de cette action est le suivant:

- 1) exécuter des effets particuliers de cette action sur les cibles suivantes:
  - a) *rt-script*:
    - i) détruire le comportement de passage de paramètres;
    - ii) détruire le comportement de terminaison;
  - b) *rt-composant*:
    - i) détruire les comportements de présentation;
    - ii) détruire le comportement d'interaction;

c) *rt-composite*:

i) détruire le comportement de dynamisme de présentation et de structure des réceptacles.

L'action `Plug` (enfichage) définie dans ce comportement n'est plus appliquée aux réceptacles. Ce comportement n'est plus possible;

ii) détruire le comportement de présentation;

iii) propager l'action `Delete` (suppression) à chaque *rt-composant* enfiché dans chacun des réceptacles de ce *rt-composite*. Cette action détruit le *rt-composite* complet lorsque la propagation est terminée;

iv) si une étiquette est enfichée dans un réceptacle, détruire l'objet *contenu* créé dynamiquement qui a servi pour le processus d'action `Plug` (enfichage);

2) détruire le comportement d'exécution;

3) détruire le *rt-objet* cible;

4) fixer l'état de *rt-disponibilité* du *rt-objet* cible à «pas disponible». Le *rt-objet* entre en phase R1;

5) déclencher le lien d'arrêt de *rt-disponibilité* de l'objet *composite* correspondant.

#### 42.3.2.2 Conditions d'erreur supplémentaires de l'action `Delete`

- Si, dans l'une des cibles, l'identificateur d'objet d'origine désigne un objet qui n'est pas un objet modèle, l'action est ignorée pour cette cible.
- Si, dans l'une des cibles, l'identificateur d'objet d'origine désigne un objet modèle qui n'est pas en phase O2, l'action est ignorée pour cette cible car l'objet modèle n'est pas encore disponible.
- Si l'une des cibles n'est pas un *rt-objet* en phase R2, R3, ou R3.TD, l'action est ignorée pour cette cible.

## 42.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Rt-Availability Status` (extraction de l'état de *rt-disponibilité*) (voir 42.4.1).

### 42.4.1 Action `Get Rt-Availability Status` (extraction de l'état de *rt-disponibilité*)

Cette action récupère l'état de disponibilité de chaque *rt-objet* pour le moteur MHEG.

Cette action a le paramètre suivant:

- `Rt-Target Param` (paramètre de *rt-cible*).

Cette action peut être appliquée pendant les phases R1, R2, R3 et R3.TD.

#### 42.4.1.1 Action `Get Rt-Availability Status` appliquée à un *rt-objet*

L'effet MHEG de cette action est le suivant:

- 1) si le *rt-objet* cible est en phase R1, l'état de *rt-disponibilité* est «pas disponible»;
- 2) sinon, il est «disponible».

#### 42.4.1.2 Conditions d'erreur supplémentaires de l'action `Get Rt-Availability Status`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de *rt-composant* donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un *rt-objet*, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si, dans l'une des cibles, l'identificateur d'objet d'origine désigne un objet modèle qui n'est pas en phase O2, l'action est ignorée pour cette cible car l'objet modèle n'est pas encore disponible.

## 43 Comportement d'exécution des rt-objets

Ce comportement sert à exécuter et à arrêter des rt-objets.

### 43.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Running Status` (état d'exécution) (voir 43.2).

### 43.2 `Running Status` (état d'exécution)

Un rt-objet peut être exécuté par le moteur MHEG à différents instants. Lorsqu'un rt-composant est en cours d'exécution, il peut être présenté à l'utilisateur. Lorsqu'un rt-script est en cours d'exécution, le script lui-même subit un traitement.

Il est possible de déterminer l'état d'exécution de chaque rt-objet par le moteur MHEG.

Initialement, l'état d'exécution des rt-objets vaut «pas en cours d'exécution» et les rt-objets sont en phase R2. L'état d'exécution d'un rt-objet vaut «en cours d'exécution» si ce rt-objet est en cours d'exécution et sa présentation ou son traitement est commandé par le moteur MHEG.

### 43.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Run` (exécution) (voir 43.3.1),
- action `Stop` (arrêt) (voir 43.3.2).

#### 43.3.1 Action `Run` (exécution)

Cette action permet la présentation et le traitement du rt-objet par le moteur MHEG.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Target Param` (paramètre de rt-cible),
- `Number Of Performances Param` (paramètre nombre d'exécutions).

Cette action peut être appliquée pendant les phases R1 et R2.

##### 43.3.1.1 Effet MHEG de l'action `Run` appliquée à un seul rt-objet

Si l'état de rt-disponibilité de la cible est à «pas disponible», une action `New` implicite est appliquée à ce rt-objet. Si la cible est en phase R2, l'effet MHEG est exécuté immédiatement comme suit:

- 1) Placer la cible sous la commande du système de présentation ou de traitement.
- 2) Mettre le nombre d'exécutions terminées à 0.
- 3) Exécuter les effets particuliers suivants:
  - rt-composant: placer sa position temporelle courante à la position temporelle courante de son espace RPS.
- 4) NOTE 1 – Par défaut, lorsqu'un rt-composant est créé, sa position temporelle courante est placée au point de départ de sa durée OVD, c'est-à-dire à sa position temporelle initiale.
- 5) NOTE 2 – L'action `Run` commence la présentation du rt-composant à la position temporelle courante à laquelle la cible se trouvait à la fin de l'action `Run` précédente, c'est-à-dire que la position temporelle courante n'est pas réinitialisée par l'application de l'action `Run`.
- 6) Fixer l'état d'exécution à «en cours d'exécution». Le rt-objet entre en phase R3.
- 7) Si la position temporelle courante atteint la position temporelle terminale, incrémenter de 1 le nombre d'exécutions terminées.
- 8) Si le nombre d'exécutions spécifié est différent du nombre d'exécutions terminées, placer la position temporelle courante à la position temporelle initiale et reprendre la présentation au début de la durée de visualisation d'origine (OVD).
- 9) Faire intervenir la terminaison temporelle de la cible si le nombre d'exécutions spécifié est égal au nombre d'exécutions terminées.

### 43.3.1.2 Effet utilisateur de l'action `Run` appliquée à un seul rt-objet

L'effet utilisateur de cette action est le suivant:

- 1) `rt-script`: l'effet utilisateur est décrit dans les données de script. A la fin du traitement d'un `rt-script`, une action `Stop` (arrêt) implicite est appliquée au `rt-script` par le moteur MHEG.  
Dans ce cas, l'état de terminaison est mis à «terminé». Le traitement du `rt-script` a été réalisé jusqu'à la fin du script.
- 2) `rt-composant`:
  - a) la position temporelle courante de la cible progresse dans la durée OVD de la cible en tenant compte de la valeur courante du facteur GTF. Tous les attributs du comportement de présentation peuvent influencer sur l'effet utilisateur (position, taille, volume sonore par exemple);
  - b) lorsque la position temporelle terminale (c'est-à-dire la fin de la durée OVD) de la cible est franchie après le nombre d'exécutions spécifié, ce qui suit s'applique:
    - si la terminaison temporelle de la cible est égale à «gel», la valeur courante du facteur GTF est mémorisée et la valeur du facteur GTF est temporairement mise à 0 par le moteur MHEG. Le facteur GTF de la cible devient 0. La position temporelle courante reste constante égale à la valeur de la position temporelle terminale. Le `rt-composant` est toujours à «en cours d'exécution».
- 3) NOTE 1 – Si les informations présentées sont des informations vidéo, la dernière trame peut être affichée. Si ce sont des informations audio, il est possible qu'aucun son ne soit perçu. S'il s'agit d'une image fixe, celle-ci peut rester présentée.
- 4) NOTE 2 – Si la position temporelle courante passe à une autre position, la valeur mémorisée du facteur GTF est reprise.
  - Sinon (la terminaison temporelle de la cible est égale à «arrêt»), une action `Stop` implicite est appliquée au `rt-composant`.
- 5) NOTE 3 – Par défaut, la position temporelle terminale est placée à la fin de la durée d'origine du `rt-composant`. Si cette durée est infinie, une action `Stop` explicite doit être appliquée à ce `rt-composant` pour arrêter l'effet utilisateur de l'action `Run`.
- 6) NOTE 4 – Si un marqueur temporel est placé à la position temporelle terminale, la position temporelle courante du `rt-composant` franchit ce marqueur un nombre de fois égal au nombre d'exécutions spécifié. Si un `rt-composant` est mis à «gel» et si un tel marqueur temporel est positionné, ce marqueur est franchi chaque fois que la position temporelle courante arrive sur la position temporelle terminale.
- 7) `rt-composite`:
  - a) tous les effets utilisateur pour les `rt-composants` s'appliquent;
  - b) chaque fois que la position temporelle courante d'un `rt-composite` franchit une position temporelle à laquelle est rattaché un réceptacle enfant, une action `Run` implicite doit être appliquée à ce réceptacle.  
Un réceptacle peut être placé à une position temporelle de l'axe T de son parent (espace PRPS) au moyen de l'action `Set PVD Position` (fixation de la position de la durée PVD);
  - c) si la position temporelle terminale d'un `rt-composite` est franchie, une action `Set CTP` (fixation de la position temporelle courante) est appliquée à chaque réceptacle qui est encore en cours d'exécution pour le placer à sa position temporelle terminale.
- 8) NOTE 5 – La durée de visualisation (VD) des réceptacles qui sont en cours d'exécution et qui ne sont pas à leur position temporelle terminale est effectivement écourtée lorsque la position temporelle terminale du parent est franchie. Quelle que soit la position des réceptacles, si leur parent est terminé, ils sont terminés aussi.
- 9) NOTE 6 – Ils sont placés à leur position temporelle terminale de sorte qu'ils redémarrent à leur position temporelle initiale lorsqu'ils repassent à l'état «en cours d'exécution».

### 43.3.1.3 Conditions d'erreur supplémentaires de l'action `Run`

- Si la cible n'est pas un réceptacle dont le parent est dans une phase autre que R3 ou R3.TD, l'action est ignorée pour cette cible. Il est nécessaire que le parent soit en cours d'exécution pour pouvoir exécuter les enfants ou les descendants.
- Si l'une des cibles n'est pas un `rt-objet` en phase R1 ou R2, l'action est ignorée pour cette cible.
- Si le nombre d'exécutions est inférieur à 1, l'action est ignorée pour l'ensemble des cibles.

### 43.3.2 Action Stop (arrêt)

Cette action fait passer l'état d'exécution à «pas en cours d'exécution» et met fin aux éventuels effets utilisateur.

Cette action a le paramètre suivant:

- ensemble de paramètres `Rt-Target Param` (paramètre de rt-cible).

Cette action peut être appliquée pendant les phases R3 et R3.TD.

#### 43.3.2.1 Action Stop appliquée à un seul rt-objet

L'effet de cette action est le suivant:

- 1) Exécuter des effets particuliers sur les objets suivants:
  - a) `rt-composant`: la progression de la position temporelle courante est arrêtée et cette position reste à sa valeur courante;
  - b) `rt-composite`: l'action `Stop` est propagée à tous les réceptacles enfants, c'est-à-dire à tous les réceptacles descendants de manière récursive.
- 2) NOTE 1 – Un réceptacle ne peut pas être en cours d'exécution si son parent ne l'est pas.
- 3) NOTE 2 – L'exécution de tous les réceptacles qui se rapportent à cet espace de composition et qui sont directement associés à un canal est suspendue.
- 4) Mettre fin aux éventuels effets utilisateur de la cible.
- 5) Mettre l'état d'exécution à «pas en cours d'exécution». Le `rt-objet` entre en phase R2 et tous les effets utilisateur sont suspendus.

#### 43.3.2.2 Conditions d'erreur supplémentaires de l'action Stop

- Si l'une des cibles n'est pas un `rt-objet` en phase R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas un réceptacle dont le parent est dans une phase autre que R3 ou R3.TD, l'action est ignorée pour cette cible. Les enfants ou les descendants ne peuvent pas être en cours d'exécution lorsque le parent ne l'est pas. L'action `Stop` est inutile dans ce cas.

## 43.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Running Status` (extraction de l'état d'exécution) (voir 43.4.1).

### 43.4.1 Action Get Running Status (extraction de l'état d'exécution)

Cette action extrait l'état d'exécution de chaque `rt-objet`.

Cette action a le paramètre suivant:

- `Rt-Target Param` (paramètre de rt-cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 43.4.1.1 Effet MHEG

Cette action détermine l'état d'exécution comme suit:

- 1) si la cible est en phase R3 ou R3.TD, l'état d'exécution vaut «en cours d'exécution»;
- 2) sinon, il vaut «pas en cours d'exécution».

#### 43.4.1.2 Conditions d'erreur supplémentaires de l'action Get Running Status

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de `rt-composant` donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un `rt-objet` en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si, dans l'une des cibles, l'identificateur d'objet d'origine désigne un objet modèle qui n'est pas en phase O2, l'action est ignorée pour cette cible car l'objet modèle n'est pas encore disponible.

## 44 Comportement de passage de paramètres aux rt-scripts

Ce comportement sert à passer des paramètres aux rt-scripts.

### 44.1 Attributs et états du comportement

Aucun.

### 44.2 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Set Parameters` (fixation des paramètres) (voir 44.2.1).

#### 44.2.1 Action `Set Parameters` (fixation des paramètres)

Cette action permet de passer des paramètres aux rt-scripts et de renvoyer les informations provenant du traitement des rt-scripts. On peut définir un nombre quelconque de paramètres pour un rt-script donné.

Chaque paramètre de rt-script est spécifié sous forme de valeur générique ou d'objet *contenu*. Etant donné qu'une valeur générique est le résultat d'une action `Get` (extraction) ou une constante, utiliser une valeur générique revient à «appeler par valeur». Et utiliser un objet *contenu* revient à «appeler par référence» puisqu'un objet *contenu* peut contenir ou faire référence à une valeur générique, une image, des informations vidéo, etc. Cette action peut être appliquée à un instant quelconque au cours de la vie d'un rt-script même lorsque celui-ci est en cours de traitement par un moteur de script, et peut aussi être appliquée plusieurs fois au cours de la vie du rt-script.

La sémantique, l'ordre des paramètres, le type (entrée, sortie ou entrée/sortie) et les utilisations de chaque paramètre seront définis par le concepteur d'objet et seront pris en charge par une application utilisatrice et par un moteur de script.

Il est possible de ne spécifier aucun paramètre de passage. Ces paramètres peuvent servir à donner des indications au moteur de script ou au moteur MHEG. La sémantique de cette utilisation dépend de l'application utilisatrice.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Script Target Param` (paramètre de rt-script cible),
- ensemble de paramètres `Passing Param` (paramètre de passage).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 44.2.1.1 Effet de l'action `Set Parameters`

Aucun effet utilisateur direct n'est associé à cette action. Toutefois, il peut y avoir un effet utilisateur indirect dans certains cas.

L'effet MHEG de cette action consiste à transmettre à la cible les paramètres de passage spécifiés.

Les objets *contenu* situés dans les paramètres de passage peuvent servir à renvoyer des informations du moteur de script au moteur MHEG. Par conséquent, si un objet *contenu* est utilisé en tant que paramètre et si des rt-contents sont créés à partir de cet objet et mis en exécution, un rt-script peut entraîner des effets utilisateur indirects en raison de la modification des données dans l'objet *contenu*. Il est possible que la présentation d'un tel rt-contenu soit modifiée.

Supposons qu'un rt-script soit chargé d'accéder à une base de données pour récupérer des informations et que les résultats soient placés dans un objet *contenu*. Les résultats peuvent servir à remplir un formulaire, par exemple, comme suit:

- un rt-script RS1 est créé à partir d'un objet *script* modèle MS1: `New (RS1)`;
- un objet *contenu* CO1 et deux entiers génériques sont passés au RS1 sous forme de paramètres: `Set Parameters (RS1, CO1, 13, 15)`;
- RS1 est exécuté: `Run (RS1)`;
- pendant le processus d'exécution de RS1, les paramètres fixés par l'action `Set Parameters` sont utilisés. A la fin du script, le résultat de l'exécution est enregistré dans CO1;
- la valeur se trouvant dans CO1 peut être récupérée au moyen d'une condition de lien pour un traitement ultérieur dans le moteur MHEG: «Lorsque l'état d'exécution de RS1 passe à «pas en cours d'exécution»», `Get Data [extraction de données] (CO1)`.

#### 44.2.1.2 Conditions d'erreur supplémentaires de l'action `Set Parameters`

- Si l'une des cibles n'est pas un rt-script en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la référence à l'objet *contenu* servant à passer des paramètres n'est pas accessible ou ne désigne pas un objet *contenu*, ce paramètre est remplacé par la valeur «pas défini».

## 45 Comportement de terminaison des rt-scripts

Ce comportement détermine l'état de terminaison d'un processus de rt-script à l'intérieur d'un moteur de script pour le moteur MHEG.

### 45.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Termination Status` (état de terminaison) (voir 45.2).

### 45.2 `Termination Status` (état de terminaison)

Cet attribut spécifie l'état de terminaison d'un rt-script. Il vaut «terminé» si l'exécution du rt-script est terminée et «pas terminé» sinon.

Après la création d'un rt-script, l'état de terminaison est initialisé à «pas terminé». Après avoir reçu une action `Run`, le rt-script est exécuté par un moteur de script. Lorsque l'exécution est terminée (c'est-à-dire lorsque l'état de terminaison passe de «pas terminé» à «terminé»), une action `Stop` implicite est appliquée au rt-script. Si une action `Stop` est appliquée pendant l'exécution du rt-script, celui-ci passe aux états «pas en cours d'exécution» (pas en cours d'utilisation) et «pas terminé». Seuls les rt-objets qui sont à «pas terminé» peuvent être de nouveau exécutés. S'il faut exécuter deux fois le même script, il faut constituer deux rt-scripts différents et les exécuter. Il n'existe pas de relation directe entre l'état d'exécution et l'état de terminaison.

### 45.3 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Termination Status` (extraction de l'état de terminaison) (voir 45.3.1).

#### 45.3.1 Action `Get Termination Status` (extraction de l'état de terminaison)

Cette action récupère l'état de terminaison de la cible.

Cette action a le paramètre suivant:

- `Rt-Script Target Param` (paramètre de rt-script cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 45.3.1.1 Effet de l'action `Get Termination Status`

L'effet de cette action est le suivant:

- 1) si le traitement du rt-script cible n'est pas terminé, l'état de terminaison vaut «pas terminé»;
- 2) sinon, il vaut «terminé».

##### 45.3.1.2 Conditions d'erreur supplémentaires de l'action `Get Termination Status`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-script en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

## 46 Comportement de dynamisme de présentation et de structure des réceptacles

Ce comportement rattache ou détache un rt-composant d'un réceptacle. Cela peut se faire à tout moment au cours de la vie du rt-composite une fois qu'il est créé.

Ce comportement a les particularités suivantes:

- 1) rattachement d'un rt-contenu à un réceptacle. Ce réceptacle devient un réceptacle de rt-contenu et le rt-contenu est perçu via le réceptacle;
- 2) rattachement d'un rt-composite à un réceptacle. Ce réceptacle devient un réceptacle de rt-composite. Un nouvel ensemble de générations est rattaché au réceptacle. Si chaque génération contient un ensemble de réceptacles de rt-contenu, une nouvelle profondeur de présentation peut être perçue;
- 3) remplacement d'un rt-contenu dans un réceptacle de rt-contenu. Un nouveau rt-contenu peut être perçu;
- 4) remplacement d'un rt-composite dans un réceptacle de rt-composite. Une génération et tous ses descendants sont remplacés par un nouvel ensemble de générations. Une nouvelle génération peut être perçue;
- 5) détachement d'un rt-composant d'un réceptacle. Le réceptacle devient un réceptacle vide. Aucune présentation n'est perçue.

Initialement, dans chaque réceptacle est enfiché un rt-composant. Il s'agit d'un rt-objet vide ou d'un rt-composant créé à partir des informations du modèle associé transférées dans le composite.

### 46.1 Attributs et états du comportement

Aucun.

### 46.2 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Plug` (enfichage) (voir 46.2.1).

#### 46.2.1 Action `Plug` (enfichage)

Cette action permet de présenter et de structurer dynamiquement les réceptacles. Elle spécifie les informations à enficher dans un réceptacle. Elle sert à obtenir une présentation ou une structure différente à partir du même objet *composite* modèle.

Le parent de la cible doit être en phase R2, R3 ou R3.TD.

Cette action a les paramètres suivants:

- ensemble de paramètres `Socket Target Param` (paramètre de réceptacle cible),
- `Plug In Param` (paramètre d'informations à enficher).

Cette action peut être appliquée pendant une phase quelconque.

##### 46.2.1.1 Effet de l'action `Plug`

L'effet MHEG de cette action est le suivant:

- 1) Si le réceptacle spécifié existe, on applique ce qui suit:
  - a) si un rt-objet vide doit être enfiché:
    - i) une action `Delete` (suppression) implicite est appliquée au rt-composant actuellement enfiché dans le réceptacle cible;
    - ii) un rt-objet vide est rattaché au réceptacle cible. Le réceptacle devient un réceptacle vide;
    - iii) si le précédent état d'exécution du réceptacle était «en cours d'exécution», une action `Run` implicite est appliquée au réceptacle;
    - iv) il est possible que les effets utilisateur précédents cessent;
  - b) si un rt-composant doit être enfiché:
    - i) on fait une copie du rt-composant spécifié à enficher. Ce nouveau rt-composant n'a pas d'identificateur explicite et il n'y est fait référence que par l'intermédiaire de l'adresse du réceptacle dans lequel il doit être enfiché.



- 2) NOTE 1 – Il appartient au moteur MHEG d'affecter cet identificateur et de garantir son unicité. Un auteur ne doit pas pouvoir utiliser la même valeur.
- 3) NOTE 2 – Pour un rt-contenu, une action `New` (nouveau) peut être appliquée à l'objet modèle puis le comportement du rt-contenu cible peut être transféré dans ce rt-contenu nouvellement créé.
- 4) NOTE 3 – Pour un rt-composite, il n'est pas possible d'appliquer une action `New` à l'objet *composite* modèle pour en faire une copie. Le rt-composite créé n'est pas toujours identique au rt-composite cible en raison du comportement de dynamisme de présentation et de structure des réceptacles qui peut être appliqué au rt-composite.
  - ii) si l'état d'exécution du rt-composant nouvellement constitué est à «en cours d'exécution», il est mis à «pas en cours d'exécution»;
  - iii) une action `Delete` (suppression) implicite est appliquée au rt-composant actuellement enfiché dans le réceptacle cible;
  - iv) le rt-contenu nouvellement créé est rattaché au réceptacle cible avec ses comportements actuellement associés;
  - v) si le précédent état d'exécution du réceptacle était «en cours d'exécution», une action `Run` implicite est appliquée au réceptacle;
  - vi) l'action `Plug` (enfichage) peut entraîner la modification des effets utilisateur précédents;
- c) si un objet *composant* doit être enfiché:
  - i) un rt-composant est créé à partir de cet objet *composant* modèle. Ce rt-composant n'a pas d'identificateur explicite et il n'y est fait référence que par l'intermédiaire de l'adresse du réceptacle dans lequel il doit être enfiché;
  - ii) ce rt-composant nouvellement créé est rattaché au réceptacle cible avec ses comportements actuellement associés;
  - iii) si le précédent état d'exécution du réceptacle était «en cours d'exécution», une action `Run` implicite est appliquée au réceptacle;
  - iv) l'action `Plug` peut entraîner la modification des effets utilisateur précédents;
- d) si une étiquette doit être enfichée:
  - i) un objet *contenu* est créé dynamiquement. Cet objet *contenu* n'a pas d'identificateur explicite et il n'y est fait référence que par l'intermédiaire de l'adresse du réceptacle dans lequel un rt-contenu créé à partir de cet objet *contenu* doit être enfiché;
  - ii) on applique toutes les étapes du cas où un objet *composant* doit être enfiché.
- 5) Sinon (le réceptacle spécifié n'existe pas), il faut créer le réceptacle spécifié et appliquer ce qui suit:
  - a) si le premier indice de l'adresse du réceptacle n'existe pas, les réceptacles enfants sont complétés par des réceptacles vides jusqu'à cet indice de réceptacle. Par exemple, si le nombre de réceptacles vaut  $n$  et si le premier indice de l'adresse du réceptacle cible vaut  $m$  ( $n < m$ ), il faut indexer de nouveaux réceptacles de  $n + 1$  à  $m$ ;
  - b) si le second indice de l'adresse du réceptacle est fourni:
    - i) si le premier indice désigne un rt-composite et si le second indice n'existe pas, les réceptacles enfants de cette seconde génération sont complétés par des réceptacles vides comme cela est décrit dans le cas du premier indice;
    - ii) si le premier indice ne désigne pas un rt-composite:
      - une action `Delete` (suppression) implicite est appliquée au rt-contenu actuellement enfiché dans ce réceptacle;
      - on crée un nouveau rt-composite avec un nombre de réceptacles égal au deuxième indice. Chaque réceptacle créé dans ce rt-composite est un réceptacle vide. Ce nouveau rt-composite n'a pas d'identificateur explicite et il n'y est fait référence que par l'intermédiaire de l'adresse du réceptacle dans lequel il doit être enfiché;
      - ce nouveau rt-composite est enfiché dans le réceptacle désigné par le premier indice;

- c) les étapes ci-dessus sont répétées jusqu'au dernier indice de l'adresse du réceptacle cible;
- d) une fois que les créations de réceptacles sont terminées, la cible désigne un réceptacle vide et on applique les étapes décrites ci-dessus concernant l'adresse de réceptacle existant.

#### 46.2.1.2 Conditions d'erreur supplémentaires de l'action `Plug`

- Si l'une des cibles ne désigne pas un réceptacle qui existe ou qui n'existe pas, l'action est ignorée pour cette cible.
- Si l'une des cibles est un réceptacle dont le parent est en phase R1, l'action est ignorée pour cette cible.

## 47 Comportement de navigation des `rt-composites`

Ce comportement permet de naviguer dans les arborescences formées par les `rt-composites`.

### 47.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- `Rt-Composite Address` (adresse de `rt-composite`) (voir 47.2),
- `Navigation Command` (commande de navigation) (voir 47.3),
- `Child` (enfant) (voir 47.4),
- `EmptyChild` (enfant vide) (voir 47.5),
- `Sibling` (frère) (voir 47.6),
- `Ancestor` (ancêtre) (voir 47.7).

#### 47.2 `Rt-Composite Address` (adresse de `rt-composite`)

Il s'agit d'une adresse désignant un `rt-composite`.

#### 47.3 `Navigation Command` (commande de navigation)

Elle donne un trajet de navigation parmi les trajets suivants: «`Child`», «`EmptyChild`», «`Sibling`» et «`Ancestor`».

#### 47.4 `Child` (enfant)

Il s'agit du trajet de navigation permettant d'atteindre un enfant. Il est exprimé sous forme d'un entier  $N > 0$  pour le  $N^{\text{ième}}$  enfant, de 'last' (dernier) pour spécifier le dernier enfant indexé ou de 'random' (aléatoire) pour spécifier un enfant dont l'indice est choisi de manière aléatoire.

#### 47.5 `EmptyChild` (enfant vide)

Il s'agit du trajet de navigation permettant d'atteindre un réceptacle vide. Il est exprimé sous forme d'un entier  $N > 0$  pour le  $N^{\text{ième}}$  réceptacle vide ou de 'last' qui spécifie le dernier réceptacle vide dans un composite.

#### 47.6 `Sibling` (frère)

Il s'agit du trajet de navigation permettant d'atteindre un frère proche. Il est exprimé sous forme d'un entier, par exemple (1 = précédent, +1 = suivant, 0 = courant).

#### 47.7 `Ancestor` (ancêtre)

Il s'agit du trajet de navigation permettant d'atteindre un ancêtre. Il est exprimé sous forme d'un entier  $N > = 0$  pour la  $N^{\text{ième}}$  génération précédente ou de 'root' pour spécifier la racine. Tout ancêtre de la racine est la racine.

## 47.8 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Rt-Composite Address` (extraction de l'adresse de rt-composite) (voir 47.8.1).

### 47.8.1 Action `Get Rt-Composite Address` (extraction de l'adresse de rt-composite)

Cette action permet de remonter à la racine, de descendre jusqu'aux feuilles et d'explorer les niveaux intermédiaires (nœuds) d'un arbre de composition.

Cette action a les paramètres suivants:

- `Rt-Composite Target Param` (paramètre de rt-composite cible),
- `Navigation Command Param` (paramètre de commande de navigation).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 47.8.1.1 Effet de l'action `Get Rt-Composite Address`

Cette action a pour résultat une référence générique désignant un rt-composite spécifié par le paramètre de commande de navigation.

##### NOTES

1 – Il est clair que cette référence peut être utilisée comme cible d'autres actions.

2 – Cette référence peut être enregistrée dans un objet *contenu* avec l'action `Set Data [fixation des données]` (voir 72.8.1) pour être utilisée ultérieurement.

Cette action peut être utilisée récursivement.

#### 47.8.1.2 Conditions d'erreur supplémentaires de l'action `Get Rt-Composite Address`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si, dans l'une des cibles, l'identificateur d'objet d'origine désigne un objet qui n'est pas un objet modèle, l'action est ignorée pour cette cible.
- Si on navigue vers un réceptacle non existant, «pas défini» est renvoyé.
- Si le paramètre de commande de navigation spécifié n'est pas valide, «pas défini» est renvoyé.

## 48 Comportement d'affectation de rt-composants à un espace RPS

Ce comportement affecte des rt-composants à un espace de présentation relatif (RPS), c'est-à-dire à un espace de composition ou à un canal.

### 48.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `RPS Assignment` (affectation d'espace RPS) (voir 48.2).

### 48.2 `RPS Assignment` (affectation d'espace RPS)

Un espace RPS est un espace de composition ou un canal.

### 48.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Set RPS Assignment` (fixation de l'affectation de RPS) (voir 48.3.1).

### 48.3.1 Action Set RPS Assignment (fixation de l'affectation de RPS)

Cette action affecte un rt-composant à un espace RPS.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible),
- RPS Assignment Param (paramètre d'affectation d'espace RPS).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 48.3.1.1 Effet de l'action Set RPS Assignment

L'effet MHEG de cette action est le suivant:

- 1) si le paramètre RPS est spécifié comme une référence à un canal, l'affectation d'espace RPS courant de cette cible prend comme valeur le canal spécifié;
- 2) sinon, l'affectation d'espace RPS courant de cette cible prend comme valeur l'espace PRPS si la cible est un rt-composant et le canal par défaut si la cible est un rt-composant racine.

Après cette action, l'ancienne relation entre l'espace de présentation d'origine et l'espace de présentation relatif (OPS-RPS) est détruite et une nouvelle est établie. Par conséquent, la précédente projection de la position de la durée PVD et la précédente projection de la taille PVS de la cible sont entièrement détruites et réinitialisées.

#### 48.3.1.2 Conditions d'erreur supplémentaires de l'action Set RPS Assignment

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si l'affectation d'espace RPS précédent et l'affectation d'espace RPS spécifié sont identiques, l'action est ignorée pour cette cible.
- Si la référence spécifiée à un canal ne désigne pas un canal disponible (c'est-à-dire si le canal n'est pas en phase C2), cette référence est remplacée par la référence au canal par défaut et l'action doit être appliquée pour le canal par défaut.

## 48.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action Get RPS Assignment (extraction de l'affectation d'espace RPS) (voir 48.4.1 Action Get RPS Assignment).

### 48.4.1 Action Get RPS Assignment (extraction de l'affectation d'espace RPS)

Cette action récupère l'affectation d'espace RPS à un rt-composant et à un rt-composant racine.

Cette action a le paramètre suivant:

- Rt-Component Target Param (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 48.4.1.1 Effet de l'action Get RPS Assignment

L'effet MHEG de cette action consiste à récupérer l'affectation d'espace RPS à la cible. Ce qui suit s'applique:

- 1) si la cible est affectée à un canal, l'identificateur de ce canal est récupéré;
- 2) si la cible est affectée à l'espace PRPS, la valeur 'prps' est récupérée.

#### 48.4.1.2 Conditions d'erreur supplémentaires de l'action Get RPS Assignment

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

## 49 Comportement de perceptibilité des rt-composants

Ce comportement sert à représenter la perceptibilité des rt-composants et des rt-composants racine pour le moteur MHEG.

Les rt-composants et les rt-composants racine concernés par ce comportement peuvent être «visuels» ou «sonores».

### 49.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- `Perceptability` (perceptibilité) (voir 49.2),
- `Presentation Priority` (priorité de présentation) (voir 49.3).

### 49.2 `Perceptability` (perceptibilité)

La valeur de perceptibilité, définie dans l'intervalle 0% à 100%, est représentée par une fraction générique.

Chaque rt-contenu ou rt-contenu racine a une perceptibilité. La façon de percevoir un rt-contenu ou un rt-contenu racine est influencée par cette valeur. Si la valeur de perceptibilité est mise à zéro, on ne perçoit pas le rt-contenu ou le rt-contenu racine. Si elle est mise à 100%, on perçoit la version originale du rt-contenu ou du rt-contenu racine. Pour les autres valeurs de perceptibilité, la façon de percevoir un rt-contenu ou un rt-contenu racine dépend du moteur MHEG.

NOTE – On peut faire varier l'intensité d'une image entre 0% et 100%.

Chaque rt-composite ou rt-composite racine a aussi une perceptibilité. La façon de percevoir ses descendants est influencée par cette valeur. Si la valeur de perceptibilité est mise à zéro, aucun descendant n'est perçu. Si elle est mise à 100 %, on perçoit la version originale de tous les descendants compte tenu de leurs propres valeurs de perceptibilité. Pour les autres valeurs de perceptibilité, la façon de percevoir tous les descendants dépend du moteur MHEG.

Initialement, la perceptibilité de tous les rt-objets vaut 100%.

### 49.3 `Presentation Priority` (priorité de présentation)

Tous les rt-composants affectés à un même canal ont des priorités les uns par rapport aux autres. L'ordre de présentation est déterminé dans un seul espace RPS. Si cet espace RPS est l'espace d'un rt-composite, celui-ci peut aussi avoir une priorité de présentation par rapport aux autres rt-composants présentés dans le même espace RPS.

Une priorité de présentation est définie par un entier compris entre 0 et 255, où 0 représente la priorité la plus élevée dans l'ordre d'empilement de la présentation et 255 la priorité la plus faible.

Si deux rt-composants ont les mêmes valeurs de priorité de présentation dans un même espace RPS, l'ordre d'empilement est défini par les actions `Run` (exécution) appliquées à ces rt-composants. En d'autres termes, le rt-composant auquel l'action `Run` est appliquée en dernier a la priorité de présentation la plus élevée. Si les actions `Run` sont appliquées en parallèle pour ces rt-composants, l'ordre d'empilement dépend de l'implémentation. Toutefois, il n'y a aucune incidence sur leurs valeurs de priorité de présentation.

Initialement, la priorité de présentation de tous les rt-composants vaut 0.

### 49.4 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Set Perceptability` (fixation de la perceptibilité) (voir 49.4.1),
- action `Set Presentation Priority` (fixation de la priorité de présentation) (voir 49.4.2).

#### 49.4.1 Action `Set Perceptability` (fixation de la perceptibilité)

Cette action permet de modifier la perceptibilité d'un rt-composant ou d'un rt-composant racine.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `Perceptability Param` (paramètre de perceptibilité): spécifie la perceptibilité à affecter à un rt-composant;

- `Transition Duration Param` (paramètre de durée de transition): spécifie la perceptibilité à affecter à un rt-composant.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 49.4.1.1 Effet de l'action `Set Perceptability`

L'effet de cette action est le suivant:

- 1) la perceptibilité du rt-composant cible est fixée comme spécifié;
- 2) si la cible est en cours d'exécution, il est possible qu'un effet utilisateur soit associé à cette action;
- 3) si le paramètre de durée de transition est spécifié, la perceptibilité est modifiée progressivement.

#### 49.4.1.2 Conditions d'erreur supplémentaires de l'action `Set Perceptability`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la valeur du paramètre de priorité de présentation n'est pas dans l'intervalle (0%, 100%), elle est mise à 0.
- Si la durée de transition est négative, elle est mise à zéro.

#### 49.4.2 Action `Set Presentation Priority` (fixation de la priorité de présentation)

Cette action permet de modifier la priorité de présentation d'un rt-composant ou d'un rt-composant racine.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Presentation Priority Param` (paramètre de priorité de présentation),
- `Transition Duration Param` (paramètre de durée de transition).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 49.4.2.1 Effet de l'action `Set Presentation Priority`

L'effet de cette action est le suivant:

- 1) fixer la priorité de présentation du rt-composant cible comme spécifié;
- 2) si la priorité de présentation spécifiée est un entier, mettre la priorité de présentation de la cible à cette valeur;
- 3) si la priorité de présentation spécifiée est 'up-priority', décrémenter de 1 la priorité de présentation de la cible. Si la valeur est 0, elle reste à 0;
- 4) sinon (la priorité de présentation spécifiée est 'down-priority'), incrémenter de 1 la priorité de présentation de la cible. Si la valeur est 255, elle reste à 255.

Si la cible est en cours d'exécution, il est possible qu'un effet utilisateur soit associé à cette action.

Si la priorité de présentation vaut 'up-priority' ou 'down-priority', le paramètre de durée de transition est ignoré.

L'utilisation d'un paramètre de durée de transition permet à un rt-composant d'apparaître avec une priorité de présentation progressive. Typiquement, cette progressivité s'accompagne d'un effet spécial, par exemple la modification progressive de l'ordre d'empilement de la présentation.

#### 49.4.2.2 Conditions d'erreur supplémentaires de l'action `Set Presentation Priority`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la valeur du paramètre de priorité de présentation n'est pas dans l'intervalle (0, 255), elle est mise à 0.
- Si la durée de transition est négative, elle est mise à zéro.

## 49.5 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par les actions suivantes:

- action `Get Perceptability` (extraction de la perceptibilité) (voir 49.5.1),
- action `Get Presentation Priority` (extraction de la priorité de présentation) (voir 49.5.2).

### 49.5.1 Action `Get Perceptability` (extraction de la perceptibilité)

Cette action récupère la valeur de perceptibilité d'un rt-composant.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 49.5.1.1 Effet de l'action `Get Perceptability`

Cette action extrait la valeur courante de la perceptibilité qui est une fraction générique se trouvant dans l'intervalle (0, 1).

#### 49.5.1.2 Conditions d'erreur supplémentaires de l'action `Get Perceptability`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

### 49.5.2 Action `Get Presentation Priority` (extraction de la priorité de présentation)

Cette action récupère la valeur de priorité de présentation d'un rt-composant.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 49.5.2.1 Effet de l'action `Get Presentation Priority`

Cette action extrait la valeur courante de la priorité de présentation qui est un entier générique se trouvant dans l'intervalle (0, 255).

#### 49.5.2.2 Conditions d'erreur supplémentaires de l'action `Get Presentation Priority`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

## 50 Comportement temporel des rt-composants

Ce comportement sert à exprimer la présentabilité temporelle des rt-composants pour le moteur MHEG.

### 50.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- OD (durée d'origine) (voir 50.2),
- POD (durée d'origine projetée) (voir 50.3),
- OVD (durée de visualisation d'origine) (voir 50.4),

- PVD (durée de visualisation projetée) (voir 50.5),
- Temporal Termination (terminaison temporelle) (voir 50.6),
- PVD Position (position de la durée PVD) (voir 50.7),
- CTP (position temporelle courante) (voir 50.8),
- GTF (facteur temporel générique) (voir 50.9),
- Timestone Status (état de marqueur temporel) (voir 50.10),
- Timestone ID (identificateur de marqueur temporel) (voir 50.11),
- Expected OVD Result (résultat de durée OVD attendue) (voir 50.12),
- Expected PVD Result (résultat de durée PVD attendue) (voir 50.13).

## 50.2 OD (durée d'origine)

Chaque rt-composant et chaque canal ont une durée d'origine (OD) initialisée par le processus décrit dans l'espace de présentation d'origine. Pour les rt-composants, la durée d'origine est fixée une fois pour toutes au moment de la création et ne sera donc jamais modifiée.

## 50.3 POD (durée d'origine projetée)

Chaque durée d'origine (OD) est projetée dans son espace de présentation relatif (RPS). La projection donne la durée POD. La durée POD d'un rt-composant est calculée en fonction du facteur temporel générique (GTF) affecté au rt-composant:  $POD = OD \times GTF$ .

## 50.4 OVD (durée de visualisation d'origine)

Chaque rt-composant a une durée OVD qui est définie comme étant une partie de la durée d'origine (OD). Seule la partie spécifiée comme étant la durée OVD est perçue par l'utilisateur. La portion de la durée d'origine qui est en dehors de la durée OVD est coupée. La durée OVD est spécifiée par la position temporelle initiale et par la position temporelle terminale qui sont des positions se trouvant dans la durée d'origine. Les deux points sont spécifiés en unité temporelle générique d'origine (OGTU) dans l'espace de présentation d'origine (OPS).

Par défaut, la position temporelle initiale de la durée OVD est initialisée à 0 (c'est-à-dire à l'origine de la durée d'origine) et la position temporelle terminale de la durée OVD est initialisée à la longueur de la durée d'origine (c'est-à-dire au point final de la durée d'origine).

Il est possible de spécifier une position temporelle terminale inférieure à la position temporelle initiale (voir Figure 56). La longueur de la durée OVD est toujours définie de la position temporelle initiale à la position temporelle terminale, compte tenu du sens de l'axe T.

Si la position temporelle terminale est inférieure à la position temporelle initiale, la longueur de la durée de visualisation (VD) est la suivante:

- [point final de la durée d'origine (position temporelle initiale de la durée OVD) + (position temporelle terminale de la durée OVD) point de départ de la durée d'origine (0)]

Sinon:

- position temporelle terminale de la durée OVD (position temporelle initiale de la durée OVD).

## 50.5 PVD (durée de visualisation projetée)

La projection de chaque durée OVD dans son espace de présentation relatif (RPS) donne la durée PVD. La durée PVD d'un rt-composant est calculée en fonction du facteur temporel générique (GTF) affecté au rt-composant:  $PVD = OVD \times GTF$ .

## 50.6 Temporal Termination (terminaison temporelle)

La terminaison temporelle permet à un rt-composite d'indiquer le choix du processus de terminaison lorsque l'exécution de la présentation du rt-composite est terminée (voir 72.11.1).

Cet attribut vaut «gel» ou «arrêt». Initialement, il est mis à «gel».



## 50.7 PVD Position (position de la durée PVD)

La durée de visualisation d'origine (OVD) d'un rt-composite est projetée dans son espace de présentation relatif (RPS) pour donner la durée PVD. La longueur de la durée PVD est commandée par la valeur du facteur temporel générique (GTF) du rt-composite, et la position de la durée PVD est commandée implicitement ou explicitement par une action élémentaire comme suit:

- 1) la position de la durée PVD d'un rt-composant racine est implicitement déterminée par l'application de l'action `Run` (exécution) au rt-composant racine;
- 2) après l'action `Run`, la position temporelle courante (CTP) de la cible est placée à la position temporelle de son espace RPS courant (l'espace CPS) dans la vie temporelle de l'espace RPS. La durée PVD de la cible est placée par rapport à l'espace RPS en tenant compte de la position CTP;
- 3) la position CTP de la cible progresse en tenant compte de la valeur de son facteur GTF;
- 4) la position de la durée PVD d'un rt-composant est déterminée comme suit:
- 5) initialement, la durée PVD d'un rt-composant n'est pas placée par rapport à la durée d'origine de son espace RPS. Si une action `Run` est appliquée à ce rt-composant, la position CTP de la cible est implicitement placée à la position temporelle de son espace RPS courant dans la vie temporelle de l'espace RPS;
- 6) une fois un rt-composant placé par rapport à la durée d'origine de son espace RPS au moyen de l'action `Set PVD Position` (fixation de la position de la durée PVD), on applique ce qui suit (voir Figure 54):
- 7) la durée PVD du rt-composant est étroitement rattachée à la durée d'origine de l'espace RPS. Par conséquent, la position CTP du rt-composant est calculée automatiquement à partir de la position CTP de son espace RPS et ne sera pas modifiée par l'action `Set CTP` (fixation de la position CTP);
- 8) si la position CTP du parent se trouve en dehors de la durée PVD de l'enfant, la position CTP du rt-composant est «pas défini» (pas définie);
- 9) si le parent du rt-composite est en cours d'exécution et si la position CTP du parent entre dans la durée PVD d'un rt-composant, l'état d'exécution du rt-composant est mis à «en cours d'exécution» et la présentation commence en tenant compte du sens de lecture du parent (en avant ou en arrière). La progression des positions CTP des enfants est synchronisée avec celle de la position CTP du parent;
- 10) si le parent du rt-composite est en cours d'exécution et si la position CTP du parent sort de la durée PVD d'un rt-composant, l'état d'exécution du rt-composant est mis à «pas en cours d'exécution» et il est mis fin à sa présentation;
- 11) si une partie de la durée PVD sort de la durée OVD de son espace RPS, cette partie est coupée et ne sera pas affectée à l'espace RPS. La présentation d'une telle durée PVD peut commencer ou se terminer au milieu de la durée PVD correspondant à la limite de la durée OVD du parent;
- 12) le rattachement de la durée PVD du rt-composant à son espace RPS est effectif jusqu'à ce qu'il soit explicitement supprimé.

NOTE – La position de la durée PVD sert à réaliser la composition temporelle. Etant donné que la position est placée au niveau de l'objet rt-composite de l'espace RPS, le rt-composant est automatiquement démarré chaque fois que le rt-composite passe à l'état «en cours d'exécution».



Figure 54/T.171 – Présentation vers l'avant

## 50.8 CTP (position temporelle courante)

Chaque rt-composant a une position CTP dans l'intervalle de sa durée de visualisation d'origine (OVD). Initialement, la position CTP est mise à la position temporelle initiale de la durée OVD.

La position CTP est modifiée comme suit par les actions Run et Stop:

- 1) après l'action Run, la position CTP progresse à un pas correspondant au facteur GTF courant (voir 50.9) dans la durée PVD des chaînes OPS-RPS finales, et ce jusqu'à la position temporelle terminale de la durée PVD ou jusqu'à exécution de l'action Stop;

### NOTES

- 1 – Si le facteur GTF vaut 0, la position CTP reste à sa position courante.
- 2 – «Progresse» signifie «avance» si le facteur GTF est positif et «recule» si le facteur GTF est négatif.
- 2) après l'action Run, si la position CTP franchit le point final de la durée d'origine, elle est mise à zéro, c'est-à-dire au point de départ de la durée d'origine (voir Figure 55);
- 3) après l'action Run, si la position CTP franchit le point zéro (c'est-à-dire le point de départ de la durée d'origine), elle est mise au point final de la durée d'origine (voir Figure 56);
- 4) après l'action Stop, la position CTP reste à sa valeur courante.

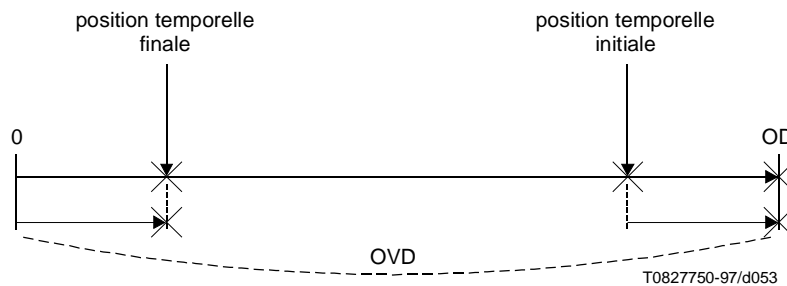


Figure 55/T.171 – Présentation vers l'avant avec rebouclage

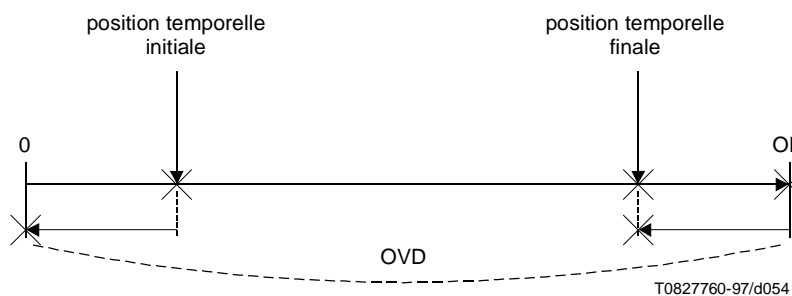


Figure 56/T.171 – Présentation vers l'arrière avec rebouclage

## 50.9 GTF (facteur temporel générique)

La vitesse de présentation d'un rt-composant est commandée par son facteur GTF courant. Le facteur GTF définit le nombre d'unités temporelles génériques relatives (RGTU) correspondant à une unité temporelle générique d'origine (OGTU). Le facteur GTF peut être négatif pour exprimer que le sens de lecture est «vers l'arrière». Dans ce cas, la position CTP d'un rt-composant recule s'il est en cours d'exécution.

NOTE – Les canaux n'ont pas de facteur GTF. On ne dispose d'aucun facteur temporel pour le mappage vers l'espace CPS.

## 50.10 Timestone Status (état de marqueur temporel)

Un marqueur temporel est un marqueur se trouvant à une position temporelle donnée. Un rt-composant peut avoir un nombre quelconque de marqueurs temporels définis le long de sa durée d'origine. Un marqueur temporel est composé:

- d'un identificateur de marqueur temporel: entier générique dont la valeur est fournie par l'auteur. La valeur 0 est réservée pour exprimer qu'aucun marqueur temporel n'a été rencontré pour la cible;
- d'une position temporelle: exprimée en unités OGTU dans l'intervalle de la durée d'origine.

Etant donné qu'un marqueur temporel contient une position temporelle et qu'il est défini dans l'espace OPS, la perception de ce point peut être différente sur l'axe T de l'espace RPS.

NOTE – Par exemple, si la valeur du facteur GTF est doublée, la durée PVD dans l'espace RPS est diminuée de moitié et la vitesse de progression de la position CTP est doublée. Dans ce cas, la position temporelle de l'espace OPS paraît plus proche dans l'espace RPS.

Les marqueurs temporels sont franchis par la position CTP lorsqu'un rt-composant est en cours d'exécution. L'état de marqueur temporel est défini comme étant l'identificateur du dernier marqueur temporel franchi par la position CTP.

Initialement, l'état de marqueur temporel vaut 0 et les rt-composants n'ont pas de marqueur temporel. Les marqueurs temporels sont spécifiés par l'auteur au moyen de l'action `Set Timestones` (fixation de marqueurs temporels).

## 50.11 Timestone ID (identificateur de marqueur temporel)

Entier générique. La valeur 0 est réservée.

## 50.12 Expected OVD Result (résultat de durée OVD attendue)

Choix entre 'initial-temporal-position', 'terminal-temporal-position' et 'duration'.

## 50.13 Expected PVD Result (résultat de durée PVD attendue)

Choix entre 'initial-temporal-position', 'terminal-temporal-position' et 'duration'.

## 50.14 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Set OVD` (fixation de la durée OVD) (voir 50.14.1),
- action `Set CTP` (fixation de la position CTP) (voir 50.14.2),
- action `Set Temporal Termination` (fixation de la terminaison temporelle) (voir 50.14.3),
- action `Set PVD Position` (fixation de la position de la durée PVD) (voir 50.14.4),
- action `Set GTF` (fixation du facteur GTF) (voir 50.14.5),
- action `Set Timestones` (fixation de marqueurs temporels) (voir 50.14.6).

### 50.14.1 Action `Set OVD` (fixation de la durée OVD)

Cette action spécifie une durée OVD dans une durée d'origine (OD).

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `Initial Point Spec Param` (paramètre de spécification de point initial);
- `Terminal Point Spec Param` (paramètre de spécification de point terminal).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.14.1.1 Effet de l'action Set OVD

Les positions temporelles initiale et terminale peuvent être spécifiées comme suit:

- 1) par une valeur absolue en unités OGTU de la cible.
- 2) par une valeur relative par rapport à la durée d'origine (OD). La valeur est déterminée par  $OD \times$  (valeur spécifiée). Si la valeur calculée est en dehors de la durée d'origine, elle est mise à 0 (si elle est négative) ou à la valeur de la durée d'origine (si elle est supérieure à la durée d'origine). Si la valeur spécifiée n'indique pas un point adressable, cette valeur sera arrondie.

Si la cible est en cours d'exécution, il est possible qu'un effet utilisateur soit associé, c'est-à-dire qu'il est possible que la durée de perception soit raccourcie ou allongée.

#### 50.14.1.2 Conditions d'erreur supplémentaires de l'action Set OVD

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si l'un des paramètres de position temporelle spécifié comme un point absolu n'est pas dans l'intervalle (0, OD), l'action est ignorée pour l'ensemble des cibles en entier.

#### 50.14.2 Action Set CTP (fixation de la position CTP)

Cette action spécifie la position CTP dans la durée OVD.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible),
- Current Point Spec Param (paramètre de spécification de point courant).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.14.2.1 Effet de l'action Set CTP

La position CTP peut être spécifiée comme suit:

- 1) si la position est spécifiée sous forme d'une valeur absolue, la position CTP est mise à cette valeur;
- 2) si la position est spécifiée sous forme d'une valeur relative, cette valeur est interprétée comme étant une valeur relative par rapport à la durée OVD de la cible. La position CTP est mise à  $OVD \times$  (valeur spécifiée);
- 3) si la position est spécifiée sous forme d'un facteur du point d'origine, la position CTP est mise à (position temporelle initiale)  $\times$  (valeur spécifiée);
- 4) sinon (la position est spécifiée sous forme d'un facteur du point courant), la position CTP est mise à  $CTP \times$  (valeur spécifiée);
- 5) dans les cas ci-dessus, si la valeur calculée est en dehors de la durée OVD, elle prend la valeur du point le plus proche à l'intérieur de la durée OVD;
- 6) si le point spécifié n'indique pas un point adressable, cette valeur sera arrondie.

Si la cible est en cours d'exécution, il est possible qu'un effet utilisateur soit associé. Si le sens de lecture est «vers l'avant» et si la position CTP est mise à une position inférieure, cela se traduit par une relecture d'une partie de la cible. Si le sens de lecture est «vers l'avant» et si la position CTP est mise à une position supérieure, cela se traduit par l'omission d'une partie de la cible.

L'action Set CTP (fixation de la position CTP) n'a aucune influence sur l'état de marqueur temporel. Si la cible est relue, un marqueur temporel déjà franchi peut être franchi à nouveau. Si une partie de la cible est omise, les marqueurs temporels figurant dans cette partie ne sont pas franchis.

#### 50.14.2.2 Conditions d'erreur supplémentaires de l'action Set CTP

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si le paramètre de position temporelle spécifié sous forme d'un point absolu n'est pas dans l'intervalle de la durée OVD, l'action est ignorée pour l'ensemble des cibles en entier.
- Si l'une des cibles est affectée à son espace RPS par l'action Set PVD Position (fixation de la position de la durée PVD), l'action est ignorée pour cette cible.

### 50.14.3 Action Set Temporal Termination (fixation de la terminaison temporelle)

Cette action spécifie le type de terminaison temporelle si la position CTP parvient à la fin de la présentation.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Target Param (paramètre de rt-cible),
- Temporal Termination Param (paramètre de terminaison temporelle).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.14.3.1 Effet de l'action Set Temporal Termination

Cette action fixe la terminaison temporelle de la cible. Si la cible se trouve à la fin de la présentation et si l'état d'exécution est à «en cours d'exécution», il est possible qu'un effet utilisateur fasse passer la précédente présentation de la terminaison temporelle à la nouvelle présentation spécifiée.

#### 50.14.3.2 Conditions d'erreur supplémentaires de l'action Set Temporal Termination

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 50.14.4 Action Set PVD Position (fixation de la position de la durée PVD)

Cette action spécifie la position dans la durée d'origine (OD) du parent à laquelle il faut rattacher la durée PVD d'un réceptacle.

Cette action a les paramètres suivants:

- ensemble de paramètres Socket Target Param (paramètre de réceptacle cible);
- Temporal Position Param (paramètre de position temporelle): spécifie la position temporelle soit par une valeur absolue le long d'un axe temporel soit par une valeur relative par rapport à une durée temporelle.

Cette action peut être appliquée pendant la phase R2.

#### 50.14.4.1 Effet de l'action Set PVD Position

La position de la durée PVD peut être spécifiée comme suit:

- 1) par une valeur absolue en unités RGTU de la cible;
- 2) par une valeur relative par rapport à la durée d'origine (OD) du parent. La valeur est déterminée par  $OD \times$  (valeur spécifiée). Si la valeur calculée est en dehors de la durée d'origine du parent, elle est mise à 0 (si elle est négative) ou à la valeur de la durée d'origine (si elle est supérieure à la durée d'origine). Si le point spécifié n'indique pas un point adressable, cette valeur sera arrondie.

Si le parent de la cible est en cours d'exécution, il est possible qu'un effet utilisateur soit associé. Après l'exécution de cette action, si la position CTP du parent entre dans la durée PVD de la cible, il faut procéder à la lecture de cette cible.

#### 50.14.4.2 Conditions d'erreur supplémentaires de l'action Set PVD Position

- Si l'une des cibles n'est pas un rt-composant en phase R2, l'action est ignorée pour cette cible.
- Si le paramètre de position temporelle spécifié sous forme d'un point absolu n'est pas dans l'intervalle de la durée d'origine du parent, l'action est ignorée pour l'ensemble des cibles en entier.
- Si le paramètre de position temporelle spécifié sous forme d'un point relatif n'est pas dans l'intervalle de la durée d'origine du parent, l'action est ignorée pour l'ensemble des cibles en entier.
- Si la cible n'a pas d'espace PRPS, l'action est ignorée.

### 50.14.5 Action Set GTF (fixation du facteur GTF)

Cette action définit la vitesse de présentation d'un rt-composant.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- GTF Param (paramètre de facteur GTF): fraction générique ou default-GF (facteur générique par défaut).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.14.5.1 Effet de l'action `Set GTF`

La valeur du facteur GTF de la cible est mise à la valeur spécifiée.

Si la cible est en cours d'exécution, il est possible qu'un effet utilisateur soit associé à cette action, à savoir la modification de la vitesse de présentation de la cible.

#### 50.14.5.2 Conditions d'erreur supplémentaires de l'action `Set GTF`

- Si l'une des cibles n'est pas un `rt-composant` en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 50.14.6 Action `Set Timestones` (fixation de marqueurs temporels)

Cette action spécifie un ensemble de marqueurs temporels à placer dans la durée d'origine (OD) de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de `rt-composant` cible),
- ensemble de paramètres `Timestone Spec Param` (paramètre de spécification de marqueur temporel): fraction générique ou `default-GF` (facteur générique par défaut).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 50.14.6.1 Effet de l'action `Set Timestones`

L'effet de cette action est le suivant:

- 1) conformément à la commande de mise à jour se trouvant dans le paramètre de spécification de marqueur temporel, des marqueurs temporels spécifiés sont ajoutés, supprimés ou remplacés;
- 2) si une valeur absolue est spécifiée pour la position d'un marqueur temporel, la valeur est interprétée comme étant la position dans la durée d'origine;
- 3) si une valeur relative est spécifiée pour la position d'un marqueur temporel, la valeur,  $OD \times$  (valeur spécifiée), est calculée et utilisée;
- 4) un même marqueur temporel peut être ajouté plusieurs fois ou indéfiniment, selon la spécification du «nombre de répétitions». Ce nombre n'est valide que pour la commande «`add`» (ajouter); il est ignoré pour les autres commandes. Si le nombre de répétitions spécifié est supérieur à 1, cela signifie qu'il faut ajouter un nombre de marqueurs temporels correspondants égal au nombre de positions temporelles spécifiées. Par exemple, l'action `Set Timestones` (cible, ((1, 5, 10), ajouter)) est répétée dix fois pour ajouter le marqueur temporel 1 toutes les 5 unités GTU à partir du début de la durée d'origine.

##### 50.14.6.2 Conditions d'erreur supplémentaires de l'action `Set Timestones`

- Si l'une des cibles n'est pas un `rt-composant` en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si le paramètre de position de marqueur temporel n'est pas dans l'intervalle de la durée d'origine, ce marqueur temporel est ignoré. Tous les autres marqueurs temporels sont appliqués.
- Si le nombre de répétitions est inférieur à 1 avec la commande «`add`» (ajouter), ce marqueur temporel est ignoré. Tous les autres marqueurs temporels sont appliqués.

### 50.15 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par les actions suivantes:

- action `Get OD` (extraction de la durée d'origine) (voir 50.15.1),
- action `Get POD` (extraction de la durée POD) (voir 50.15.2),
- action `Get OVD` (extraction de la durée OVD) (voir 50.15.3),
- action `Get PVD` (extraction de la durée PVD) (voir 50.15.4),
- action `Get CTP` (extraction de la position CTP) (voir 50.15.5),

- action `Get Temporal Termination` (extraction de la terminaison temporelle) (voir 50.15.6),
- action `Get PVD Position` (extraction de la position de la durée PVD) (voir 50.15.7),
- action `Get GTF` (extraction du facteur GTF) (voir 50.15.8),
- action `Get Timestone Status` (extraction de l'état de marqueur temporel) (voir 50.15.9).

### 50.15.1 Action `Get OD` (extraction de la durée d'origine)

Cette action récupère la durée d'origine (OD) de la cible exprimée en unités OGTU.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.15.1.1 Effet de l'action `Get OD`

L'effet MHEG de cette action consiste à récupérer la valeur de la durée d'origine de la cible exprimée en unités OGTU.

#### 50.15.1.2 Conditions d'erreur supplémentaires de l'action `Get OD`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 50.15.2 Action `Get POD` (extraction de la durée POD)

Cette action récupère la durée POD de la cible exprimée en unités RGTU.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.15.2.1 Effet de l'action `Get POD`

L'effet MHEG de cette action consiste à récupérer la valeur de la durée POD de la cible exprimée en unités RGTU.

#### 50.15.2.2 Conditions d'erreur supplémentaires de l'action `Get POD`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 50.15.3 Action `Get OVD` (extraction de la durée OVD)

Cette action récupère la durée OVD de la cible exprimée en unités OGTU.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Expected OVD Result Param` (paramètre de résultat de durée OVD attendue): fraction générique ou `default-GF` (facteur générique par défaut).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

### 50.15.3.1 Effet de l'action `Get OVD`

L'effet MHEG de cette action consiste à récupérer la valeur de la durée OVD de la cible exprimée en unités OGTU de la façon suivante:

- 1) si le paramètre de résultat de la durée OVD attendue est `'initial-temporal-position'`, cette action récupère la position temporelle initiale de la durée OVD;
- 2) si le paramètre de résultat de la durée OVD attendue est `'terminal-temporal-position'`, cette action récupère la position temporelle terminale de la durée OVD;
- 3) sinon (le paramètre de résultat de la durée OVD attendue est `'duration'`), cette action récupère la durée OVD.

### 50.15.3.2 Conditions d'erreur supplémentaires de l'action `Get OVD`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique `'*`' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 50.15.4 Action `Get PVD` (extraction de la durée PVD)

Cette action récupère la durée PVD de la cible exprimée en unités RGTU.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Expected PVD Result Param` (paramètre de résultat de durée PVD attendue): fraction générique ou `default-GF` (facteur générique par défaut).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 50.15.4.1 Effet de l'action `Get PVD`

L'effet MHEG de cette action consiste à récupérer la valeur de la durée PVD de la cible exprimée en unités RGTU de la façon suivante:

- 1) si le paramètre de résultat de la durée PVD attendue est `'initial-temporal-position'`, cette action récupère la position temporelle initiale de la durée PVD;
- 2) si le paramètre de résultat de la durée PVD attendue est `'terminal-temporal-position'`, cette action récupère la position temporelle terminale de la durée PVD;
- 3) sinon (le paramètre de résultat de la durée PVD attendue est `'duration'`), cette action récupère la durée PVD.

#### 50.15.4.2 Conditions d'erreur supplémentaires de l'action `Get PVD`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique `'*`' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

### 50.15.5 Action `Get CTP` (extraction de la position CTP)

Cette action récupère la valeur de la position CTP de la cible.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.



#### 50.15.5.1 Effet de l'action `Get CTP`

L'effet MHEG de cette action consiste à récupérer la valeur de la position CTP de la cible. Cette action détermine une position temporelle exprimée en unités OGTU.

#### 50.15.5.2 Conditions d'erreur supplémentaires de l'action `Get CTP`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si une cible est positionnée dans son espace PRPS par l'action `Set PVD Position` (fixation de la position de la durée PVD), cette action renvoie «pas défini» pour cette cible.

#### 50.15.6 Action `Get Temporal Termination` (extraction de la terminaison temporelle)

Cette action récupère la valeur de terminaison temporelle de la cible.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 50.15.6.1 Effet de l'action `Get Temporal Termination`

L'effet MHEG de cette action consiste à récupérer la valeur de la terminaison temporelle de la cible, à savoir «gel» ou «arrêt».

##### 50.15.6.2 Conditions d'erreur supplémentaires de l'action `Get Temporal Termination`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 50.15.7 Action `Get PVD Position` (extraction de la position de la durée PVD)

Cette action récupère la valeur de la position de la durée PVD de la cible dans son espace PRPS.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 50.15.7.1 Effet de l'action `Get PVD Position`

L'effet MHEG de cette action consiste à récupérer la valeur de la position de la durée PVD de la cible dans son espace PRPS. Cette action détermine une position temporelle exprimée en unités RGTU.

##### 50.15.7.2 Conditions d'erreur supplémentaires de l'action `Get PVD Position`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.

- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas positionnée dans son espace PRPS, cette action renvoie «pas défini».

#### **50.15.8 Action `Get GTF` (extraction du facteur GTF)**

Cette action récupère la valeur du facteur GTF de la cible.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### **50.15.8.1 Effet de l'action `Get GTF`**

L'effet MHEG de cette action consiste à récupérer la valeur du facteur GTF de la cible. Cette action détermine une fraction générique.

##### **50.15.8.2 Conditions d'erreur supplémentaires de l'action `Get GTF`**

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### **50.15.9 Action `Get Timestone Status` (extraction de l'état de marqueur temporel)**

Cette action récupère la valeur de l'état de marqueur temporel de la cible.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### **50.15.9.1 Effet de l'action `Get Timestone Status`**

L'effet MHEG de cette action consiste à récupérer la valeur de l'état de marqueur temporel de la cible. Cette action détermine un entier correspondant à l'identificateur du dernier marqueur temporel franchi par la position CTP. Si aucun marqueur temporel n'a été franchi, 0 est renvoyé.

##### **50.15.9.2 Conditions d'erreur supplémentaires de l'action `Get Timestone Status`**

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

## 51 Comportement spatial des rt-composants

Ce comportement décrit la présentabilité spatiale des rt-composants pour le moteur MHEG.

### 51.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- OS (taille d'origine) (voir 51.2);
- POS (taille d'origine projetée) (voir 51.3);
- Aspect Ratio (facteur de forme) (voir 51.4);
- Resizing Strategy (stratégie de redimensionnement) (voir 51.5);
- OVS (taille de visualisation d'origine) (voir 51.6);
- OAP (point de rattachement de taille de visualisation d'origine) (voir 51.7);
- OVS Position (position de la taille OVS) (voir 51.8);
- PVS (taille de visualisation projetée) (voir 51.9);
- OVS Proj Strategy (stratégie de projection de taille OVS) (voir 51.10);
- PAP (point de rattachement de taille de visualisation projetée) (voir 51.11);
- PVS Position (position de la taille PVS) (voir 51.12);
- GSF (facteur spatial générique) (voir 51.13);
- Spatial Control (commande spatiale) (voir 51.14);
- User Spatial Control (commande spatiale utilisateur) (voir 51.15);
- Expected Axis Result Param (paramètre de résultat d'axe attendu) (voir 51.16);
- Point Type Param (paramètre de type de point) (voir 51.17).

### 51.2 OS (taille d'origine)

Chaque rt-composant et chaque canal ont une taille d'origine affectée qui est définie dans son espace de présentation d'origine (OPS). Par conséquent, la taille d'origine est mesurée en unités spatiales génériques d'origine (OGSU) (voir «Espace de présentation d'origine» pour l'initialisation de chaque taille d'origine).

La taille d'origine d'un rt-contenu est fixée une fois pour toutes et ne sera pas modifiée. Mais la taille d'origine d'un rt-composite peut être modifiée au moyen de l'attribut de stratégie de redimensionnement (voir 51.5).

### 51.3 POS (taille d'origine projetée)

Chaque taille d'origine (OS) d'un rt-composant est projetée dans son espace de présentation relatif (RPS). La taille projetée est appelée taille POS et elle est donc mesurée en unités spatiales génériques relatives (RGSU). Elle est calculée par  $POS = OS \times GSF$  sur chaque axe.

### 51.4 Aspect Ratio (facteur de forme)

Le facteur de forme est le rapport entre la largeur, la hauteur et la profondeur le long des axes spatiaux.

La projection de la taille d'origine (OS) est commandée par le facteur spatial générique (GSF) et la relation  $POS = OS \times GSF$  est toujours vérifiée. Toutefois, afin de conserver le facteur de forme d'un rt-composant, la façon de remplir la taille POS est spécifiée indépendamment du facteur GSF. L'attribut de facteur de forme vaut 'preserved' (conservé) ou 'not-preserved' (pas conservé). Cet attribut commande la conservation du facteur de forme pour une seule projection OS-POS. Si on veut présenter le rt-composant dans un canal en conservant le facteur de forme, tous les attributs de facteur de forme dans les chaînes OPS-RPS depuis le rt-composant jusqu'au canal doivent être mis à 'preserved'.

Pour conserver le facteur de forme, on applique ce qui suit:

- 'preserved' (conservé): le rt-composant est mis à l'échelle en conservant le facteur de forme. Cette mise à l'échelle est indépendante du facteur GSF affecté, et la mise à l'échelle du rt-composant est telle que l'une au moins des valeurs de largeur, hauteur ou profondeur correspond à la taille POS. Ensuite, la taille POS est remplie avec ce rt-composant mis à l'échelle. Celui-ci peut ne pas correspondre entièrement à la taille POS, c'est-à-dire que certaines parties de la taille POS peuvent ne pas être remplies par le rt-composant mis à l'échelle. Si, sur un axe donné, le rt-composant mis à l'échelle a une taille plus petite que la taille POS, il est centré sur l'intervalle de cet axe. La façon dont sont remplies les portions de la taille POS qui ne sont pas couvertes par le rt-composant mis à l'échelle est soit définie par un style de présentation (enregistré) soit dépendante de la réalisation;
- 'not-preserved' (pas conservé): il faut remplir entièrement la taille POS avec le rt-composant en tenant compte du facteur GSF sur chaque axe.

Initialement, la valeur de l'attribut de facteur de forme est mise à 'preserved'.

#### NOTES

1 – La valeur de l'attribut de facteur de forme n'a aucune incidence sur les tailles POS, c'est-à-dire que le facteur de forme de la taille d'origine (OS) et le facteur de forme de la taille POS peuvent être différents sur chaque axe même si la valeur de l'attribut de facteur de forme est à 'preserved'.

2 – Si la taille d'origine sur un axe vaut zéro, cet axe n'est pas pris en considération pour la mise à l'échelle du rt-composant.

### 51.5 Resizing Strategy (stratégie de redimensionnement)

Il est possible, au moyen de l'attribut de stratégie de redimensionnement, de modifier la taille d'origine (OS) de chaque rt-composite afin de tenir compte de la modification de la taille et de la position des tailles de visualisation projetées (PVS) des rt-composants enfants. Cet attribut de rt-composite peut prendre l'une des trois valeurs suivantes:

- 'fixed' (fixe): la taille d'origine du rt-composite doit rester à la taille d'origine initiale. Elle ne peut pas être modifiée tant qu'une autre valeur d'attribut n'est pas affectée. Si cet attribut passe d'une autre valeur à la valeur 'fixed' et si la taille d'origine précédente est différente de la taille initiale en raison du redimensionnement à cet instant, la taille d'origine doit être remise à la taille initiale;
- 'minimum': il faut que le rt-composite ait une taille d'origine minimale qui soit suffisante pour contenir toutes les tailles PVS des rt-composants enfants et qui soit supérieure ou égale à la taille d'origine initiale de la cible sur chaque axe;
- 'grows-only' (croissance seulement): la taille d'origine du rt-composite ne doit pas être réduite. Si une partie de la taille PVS d'un rt-composant est placée en dehors de la taille d'origine, la taille d'origine prend la taille minimale sur chaque axe qui est suffisante pour inclure toutes les tailles PVS. Mais la taille d'origine n'est jamais réduite même si elle est supérieure à la taille minimale effective qui est suffisante pour toutes les tailles PVS.

La valeur initiale est 'fixed'.

Les portions des tailles PVS qui sortent de la taille d'origine d'un rt-composite sont coupées. Toutefois, les valeurs de ces tailles PVS ne changent pas. Cette situation ne se produit que si la valeur est 'fixed'.

Si des portions d'une taille PVS sortent de la taille d'origine d'un rt-composite et si son attribut de stratégie de redimensionnement vaut 'minimum' ou 'grows-only', le point PAP et la position de la taille PVS peuvent changer par suite du redimensionnement. Le moteur MHEG ne doit pas tenir compte de cette modification.

#### NOTES

1 – Les valeurs de ces tailles PVS restent identiques car ces rt-composants peuvent être affectés ultérieurement à un autre espace RPS ou peuvent devenir entièrement perceptibles en raison de la modification de la taille d'origine du parent.

2 – Les opérations sur les tailles décrites ci-dessus doivent être effectuées indépendamment les unes des autres sur chaque axe. Par exemple, dans le cas 'minimum', si la valeur initiale de la taille d'origine du rt-composite est  $x = 10$  et  $y = 10$ , et si le rt-composite a un rt-composant dont la taille PVS vaut  $x = 5$  et  $y = 20$ , la taille d'origine du rt-composite devient  $x = 10$  et  $y = 20$ .

Si l'attribut de facteur de forme d'un rt-composite est à 'preserved', la taille des données peut représenter une partie de la taille d'origine afin de conserver le facteur de forme.

## 51.6 OVS (taille de visualisation d'origine)

La taille OVS d'un rt-composant représente une partie de sa taille d'origine (OS), et elle est positionnée par rapport à la taille d'origine. La portion de la taille d'origine qui correspond à la taille OVS doit être présentée à l'utilisateur. La portion de la taille d'origine qui sort de la taille OVS est coupée. Si la taille OVS dépasse les limites de la taille d'origine, la portion qui dépasse est remplie par un arrière-plan qui dépend de la réalisation ou qui est spécifié par une fonctionnalité d'extension.

La taille OVS peut être spécifiée par l'une des méthodes suivantes (voir aussi `OVS Proj Strategy`):

- 1) elle peut être définie directement dans l'espace de présentation d'origine (OPS): la taille OVS est définie dans l'espace OPS par l'action `Set OVS` (fixation de la taille OVS) avec l'attribut `OVS Proj Strategy` valant 'calculated' (calculé). Dans ce cas, toute modification du facteur spatial générique (GSF) du rt-composant influe sur la taille PVS car la taille OVS est alors une constante [ $PVS = OVS (\text{constante}) \times GSF$ ]. Mais cette modification n'a d'influence ni sur le point PAP ni sur la position de la taille PVS;
- 2) elle peut être définie indirectement dans l'espace de présentation relatif (RPS): plutôt que de définir la taille OVS, la taille PVS est définie dans l'espace RPS par l'action `Set OVS` avec l'attribut `OVS Proj Strategy` valant 'fixed' (fixe). La taille OVS est calculée automatiquement par  $OVS = PVS / GSF$  sur chaque axe. Dans ce cas, la taille OVS peut varier dynamiquement en fonction de la valeur du facteur GSF. Et une modification du facteur GTF du rt-composant n'affecte pas la taille PVS, car celle-ci paraît constante [ $PVS (\text{constante}) = OVS \times GSF$ ]. Cette modification du facteur GTF n'affecte ni le point OAP (voir 51.7) ni la position de la taille OVS, mais la taille OVS elle-même.

NOTE 1 – Mais par cette modification, une certaine partie de la taille d'origine devient visible.

La taille OVS est initialisée à la taille d'origine correspondante dans l'espace OPS, exprimée en unités spatiales génériques d'origine (OGSU).

NOTE 2 – Typiquement, la taille OVS définie directement dans l'espace OPS sert à mettre à l'échelle un rt-composant. Par conséquent, les tailles PVS et POS varient en fonction du facteur GSF du rt-composant.

NOTE 3 – Typiquement, la taille OVS définie indirectement dans l'espace RPS sert à ne mettre à l'échelle que la taille d'origine d'un rt-composant qui est projetée et perçue à travers la taille PVS constante.

## 51.7 OAP (point de rattachement de taille de visualisation d'origine)

Un point OAP définit une position de rattachement dans la taille de visualisation d'origine (OVS). Cette position de rattachement ainsi que la position de la taille OVS servent à positionner la taille OVS dans sa taille d'origine. Initialement, le point OAP est placé à l'origine de la taille OVS, c'est-à-dire (0, 0, 0).

Le point OAP peut être spécifié par une position absolue dans la taille OVS, ou par une position relative par rapport à l'origine de la taille OVS exprimée par une fraction générique.

Si une taille OVS est spécifiée indirectement par l'action `Set OVS` (fixation de la taille OVS) avec l'attribut `OVS Proj Strategy` valant 'fixed', il est possible de modifier dynamiquement la taille OVS en fonction de la valeur du facteur GSF correspondant. Dans ce cas, le point OAP ne doit pas être affecté par la modification de la taille OVS. Il doit rester à la valeur courante. Si la valeur spécifiée est une valeur absolue, elle doit être conservée. Si la valeur spécifiée est une valeur relative, il faut conserver la valeur relative.

## 51.8 OVS Position (position de la taille OVS)

La position de la taille OVS définit une position par rapport à l'origine de la taille d'origine (OS) où le point OAP correspondant doit être positionné. La position de la taille OVS et le point OAP servent à positionner la taille OVS dans sa taille d'origine. Initialement, la position de la taille OVS est fixée à l'origine de la taille d'origine, c'est-à-dire (0, 0, 0).

La taille OVS peut être spécifiée par une position absolue dans la taille d'origine, ou par une position relative par rapport à la taille d'origine exprimée par une fraction générique.

Si une taille OVS est spécifiée indirectement par l'action `Set OVS` avec l'attribut `OVS Proj Strategy` valant 'fixed', il est possible de modifier dynamiquement la taille OVS en fonction de la valeur du facteur GSF correspondant. Dans ce cas, la position de la taille OVS ne doit pas être affectée par la modification de la taille OVS. Elle doit rester à la valeur courante. Si la valeur spécifiée est une valeur absolue, elle doit être conservée. Si la valeur spécifiée est une valeur relative, il faut conserver la valeur relative.

## 51.9 PVS (taille de visualisation projetée)

La taille PVS est la projetée de la taille OVS d'un rt-composant dans son espace RPS, et est positionnée par rapport à la taille d'origine de l'espace RPS.

La taille PVS peut être spécifiée par l'une des méthodes suivantes:

- elle peut être définie directement dans l'espace RPS: la taille PVS est définie dans l'espace RPS par l'action `Set OVS` (fixation de la taille OVS) avec l'attribut `OVS Proj Strategy` valant 'fixed' (fixe). Dans ce cas, la taille PVS est constante et la taille OVS correspondante est calculée automatiquement;
- elle peut être définie indirectement dans l'espace OPS: la taille OVS est définie dans l'espace OPS par l'action `Set OVS` avec l'attribut `OVS Proj Strategy` valant 'calculated' (calculé), par l'action `Set OAP` (fixation du point OAP) et par l'action `Set OVS Position` (fixation de la position de la taille OVS). Dans ce cas, la taille PVS est calculée en fonction de la valeur du facteur GSF et la taille OVS correspondante est constante.

La taille PVS est initialisée au moyen des valeurs initiales de la taille OVS et du facteur GSF.

## 51.10 OVS Proj Strategy (stratégie de projection de taille OVS)

Chaque rt-composant a un attribut `OVS Proj Strategy`. Cet attribut définit comment projeter une taille OVS en taille PVS correspondante. La relation entre les tailles OVS et PVS,  $PVS = OVS \times GSF$  sur chaque axe, est toujours respectée.

Cet attribut vaut 'fixed' (fixe) ou 'calculated' (calculé). On applique ce qui suit:

- si l'attribut `OVS Proj Strategy` vaut 'fixed', la taille PVS est toujours fixe;
- si la taille OVS varie, le facteur GSF est recalculé;
- si le facteur GSF varie, la taille OVS est recalculée;
- sinon ('calculated'), la taille PVS est toujours calculée;
- si la taille OVS varie, la taille PVS est recalculée;
- si le facteur GSF varie, la taille PVS est recalculée.

## 51.11 PAP (point de rattachement de taille de visualisation projetée)

Un point PAP définit une position de rattachement dans la taille PVS. Cette position de rattachement ainsi que la position de la taille PVS servent à positionner la taille PVS dans la taille d'origine de l'espace RPS. Initialement, le point PAP est placé à l'origine de la taille PVS, c'est-à-dire (0, 0, 0).

Le point PAP peut être spécifié par une position absolue dans la taille PVS, ou par une position relative par rapport à l'origine de la taille PVS exprimée par une fraction générique.

## 51.12 PVS Position (position de la taille PVS)

La position de la taille PVS définit une position par rapport à l'origine de la taille d'origine (OS) de l'espace RPS ou par rapport à une autre taille PVS dans le même espace RPS, où le point PAP correspondant doit être positionné. La position de la taille PVS ainsi que le point PAP servent à positionner la taille PVS dans la taille d'origine de l'espace RPS. Initialement, la position de la taille PVS est placée à l'origine de la taille d'origine de l'espace RPS, c'est-à-dire (0, 0, 0).

La position de la taille PVS peut être spécifiée par une position absolue dans la taille d'origine, ou par une position relative par rapport à la taille d'origine exprimée par une fraction générique.

## 51.13 GSF (facteur spatial générique)

La projection de l'espace OPS vers l'espace RPS est commandée par la valeur de son facteur GSF. Le facteur GSF définit le nombre d'unités spatiales génériques relatives (RGSU) correspondant à une unité spatiale générique d'origine (OGSU) pour chaque axe.

## 51.14 Spatial Control (commande spatiale)

On peut autoriser ou non à l'utilisateur de déplacer, redimensionner, mettre à l'échelle et faire défiler un rt-composant indépendamment des commandes d'actions élémentaires. Cet attribut indique si ces fonctions sont autorisées avec la combinaison de la valeur de l'attribut de commande spatiale utilisateur. L'attribut de commande spatiale prend l'une des valeurs suivantes:

- 'moving': indique un déplacement de la taille PVS;
- 'resizing': indique un redimensionnement de la taille PVS;
- 'scaling': indique une mise à l'échelle de la taille d'origine (OS);
- 'scrolling': indique un défilement de la taille d'origine.

NOTE 1 – Le défilement peut être autorisé pour chaque rt-composant et pour le parent des rt-composants indépendamment. Le défilement se rapporte toujours à l'espace de présentation (PS) du rt-composant.

Initialement, ces commandes utilisateur sont 'not-allowed' (pas autorisées).

Il appartient au moteur MHEG avec l'interface utilisateur d'informer l'utilisateur de son autorisation à exécuter une commande donnée et de fournir les moyens nécessaires (barres de défilement, manettes de redimensionnement par exemple).

Par exemple, si le défilement du rt-composant est autorisé, le moteur MHEG doit offrir un mécanisme permettant de faire défiler la taille d'origine telle qu'elle est perçue à travers la taille PVS. Il appartient au moteur MHEG de décider quels rendu et convivialité il faut adopter pour le mécanisme de défilement et s'il faut rattacher ce mécanisme au rt-composant.

Si la commande utilisateur n'est pas autorisée, il appartient au moteur MHEG d'en informer l'utilisateur (les barres de défilement ne sont pas affichées ou la frappe de touches de défilement produit un bip par exemple).

NOTE 2 – Typiquement, un mécanisme de défilement rattaché à un rt-composant peut être réalisé au moyen de deux barres de défilement.

NOTE 3 – Typiquement, un mécanisme de défilement non rattaché à un rt-composant peut être réalisé au moyen de touches ou de flèches.

## 51.15 User Spatial Control (commande spatiale utilisateur)

Cet attribut est utilisé avec l'attribut de commande spatiale pour indiquer si une certaine commande spatiale utilisateur est autorisée.

## 51.16 Expected Axis Result Param (paramètre de résultat d'axe attendu)

Choix entre 'x', 'y', 'z' et 'xyz'.

## 51.17 Point Type Param (paramètre de type de point)

Choix entre 'absolute' et 'relative'.

## 51.18 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action Set Aspect Ratio (fixation du facteur de forme) (voir 51.18.1);
- action Set Resizing Strategy (fixation de la stratégie de redimensionnement) (voir 51.18.2);
- action Set OVS Proj Strategy (fixation de la stratégie de projection de taille OVS) (voir 51.18.3);
- action Set OVS (fixation de la taille OVS) (voir 51.18.4);
- action Set OAP (fixation du point OAP) (voir 51.18.5);
- action Set OVS Position (fixation de la position de la taille OVS) (voir 51.18.6);
- action Set PAP (fixation du point PAP) (voir 51.18.7);
- action Set PVS Position (fixation de la position de la taille PVS) (voir 51.18.8);
- action Set GSF (fixation du facteur GSF) (voir 51.18.9);
- action Set User Spatial Control (fixation de la commande spatiale utilisateur) (voir 51.18.10).

### 51.18.1 Action Set Aspect Ratio (fixation du facteur de forme)

Cette action spécifie s'il faut conserver sur chaque axe le rapport entre la taille d'origine (OS) en unités OGSU et la taille (en unités RGSU) du rt-composant mis à l'échelle se trouvant dans la taille POS lors de la projection de la taille d'origine d'un rt-composant dans la taille POS.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- Aspect Ratio Param (paramètre de facteur de forme).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.18.1.1 Effet de l'action Set Aspect Ratio

L'effet de cette action est le suivant:

- 1) si le paramètre de facteur de forme vaut 'preserved', le facteur de forme du rt-composant mis à l'échelle doit être conservé;
- 2) sinon, le rt-composant doit remplir entièrement la taille POS.

Cet effet se poursuit jusqu'à destruction de la cible ou jusqu'à ce que cette action soit appliquée à nouveau avec une valeur différente. La poursuite de l'effet signifie que le rt-composant mis à l'échelle se trouvant dans la taille POS doit être recalculé à chaque modification de la taille POS afin de conserver le facteur de forme ou de remplir la taille POS.

Si la cible est en cours d'exécution, un effet utilisateur est associé, c'est-à-dire que la zone où les données sont présentées peut être soit élargie pour remplir entièrement la taille POS soit comprimée pour ne la remplir que partiellement.

#### 51.18.1.2 Conditions d'erreur supplémentaires de l'action Set Aspect Ratio

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 51.18.2 Action Set Resizing Strategy (fixation de la stratégie de redimensionnement)

Cette action spécifie la stratégie de redimensionnement de la taille d'origine (OS) d'un rt-composite compte tenu de la modification des tailles PVS des rt-composants enfants positionnés dans la taille d'origine du rt-composite.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Composite Target Param (paramètre de rt-composite cible);
- Resizing Strategy Param (paramètre de stratégie de redimensionnement).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.18.2.1 Effet de l'action Set Resizing Strategy

L'effet de l'action est le suivant:

- 1) selon la valeur du paramètre de stratégie de redimensionnement ['fixed', 'minimum' ou 'grows-only' (croissance seulement)], la taille d'origine du rt-composite cible peut être modifiée;
- 2) si 'minimum' ou 'grows-only' est affecté, la taille d'origine du rt-composite est recalculée chaque fois que les tailles PVS des enfants sont modifiées et chaque fois que la taille d'origine ne correspond pas à la stratégie spécifiée.

Cet effet se poursuit jusqu'à destruction de la cible ou jusqu'à ce que cette action soit appliquée à nouveau avec une valeur différente.

#### 51.18.2.2 Conditions d'erreur supplémentaires de l'action Set Resizing Strategy

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 51.18.3 Action Set OVS Proj Strategy (fixation de la stratégie de projection de taille OVS)

Cette action fixe la stratégie de projection de taille OVS pour un rt-composant.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- OVS Proj Strategy Param (paramètre de stratégie de projection de taille OVS).



Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.18.3.1 Effet de l'action Set OVS Proj Strategy

L'effet de cette action est le suivant:

- 1) l'attribut OVS Proj Strategy est mis à 'fixed' ou à 'calculated' selon les spécifications;
- 2) si l'attribut OVS Proj Strategy est mis à 'fixed', le paramètre de spécification de taille de l'action Set OVS (fixation de la taille OVS) est interprété comme étant exprimé en unités RGSU et l'action Set OVS fixe la taille PVS;
- 3) sinon, le paramètre de spécification de taille de l'action Set OVS est interprété comme étant exprimé en unités OGSU et l'action Set OVS fixe la taille OVS directement.

#### 51.18.3.2 Conditions d'erreur supplémentaires de l'action Set OVS Proj Strategy

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 51.18.4 Action Set OVS (fixation de la taille OVS)

Cette action fixe la taille OVS directement ou indirectement selon la valeur de l'attribut OVS Proj Strategy de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- Size Spec Param (paramètre de spécification de taille);
- Transition Duration Param (paramètre de durée de transition).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.18.4.1 Effet de l'action Set OVS

L'effet de cette action est le suivant:

- 1) le paramètre de spécification de taille peut être spécifié par une valeur absolue exprimée en unités GSU ou par une valeur relative exprimée sous forme d'une fraction générique;
- 2) si l'attribut OVS Proj Strategy vaut 'fixed':
  - a) les valeurs précisées dans le paramètre de spécification de taille sont interprétées comme étant relatives à l'espace RPS de la cible. On considère que les valeurs absolues sont exprimées en unités RGSU et que les valeurs relatives sont exprimées par rapport à la taille POS de la cible;
  - b) la taille PVS de la cible est définie directement par les valeurs interprétées;
  - c) la taille OVS est calculée automatiquement au moyen du facteur GSF courant de la cible;
  - d) si la taille OVS ou l'unité GSU est modifiée après l'application de cette action, la nouvelle unité GSU ou la nouvelle taille OVS est recalculée afin que l'équation  $PVS = OVS \times GSU$  reste vérifiée sur chaque axe;
- 3) sinon:
  - a) les valeurs précisées dans le paramètre de spécification de taille sont interprétées comme étant relatives à l'espace OPS de la cible. On considère que les valeurs absolues sont exprimées en unités OGSU et que les valeurs relatives sont exprimées par rapport à la taille d'origine de la cible;
  - b) la taille OVS de la cible est définie directement par les valeurs interprétées;
  - c) la taille PVS est calculée automatiquement au moyen du facteur GSF courant de la cible;
  - d) si la taille OVS ou l'unité GSU est modifiée après l'application de cette action, la nouvelle taille PVS est recalculée afin que l'équation  $PVS = OVS \times GSU$  reste vérifiée sur chaque axe.

Si une spécification de taille est omise sur un axe, on applique 100%.

Si une durée de transition non nulle est spécifiée, la taille OVS ou PVS varie progressivement selon la valeur de l'attribut OVS Proj Strategy. Un effet utilisateur est associé à cette action, c'est-à-dire que le contenu présenté dans la taille PVS varie progressivement ou que la taille PVS varie progressivement. La modification de la taille OVS ou PVS n'a aucune influence sur le point OAP, sur la position de la taille OVS, sur le point PAP et sur la position de la taille PVS.

#### 51.18.4.2 Conditions d'erreur supplémentaires de l'action Set OVS

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si une spécification de taille est omise sur un axe, on applique 100%.
- Si la valeur d'un paramètre de spécification de taille est négative, l'action est ignorée pour l'ensemble des cibles en entier.
- Si la durée de transition est négative, elle est mise à zéro.
- Si un facteur du point d'origine ou un facteur du point courant est utilisé, l'action est ignorée pour toutes les cibles.

#### 51.18.5 Action Set OAP (fixation du point OAP)

Cette action fixe le point OAP d'un rt-composant. Ce point sera spécifié à l'intérieur de la taille OVS.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- OAP Param (paramètre de point OAP): valeur absolue ou relative.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 51.18.5.1 Effet de l'action Set OAP

Le point OAP peut prendre des valeurs absolues ou des valeurs relatives à la taille OVS selon le paramètre de point OAP.

Il peut y avoir un effet utilisateur modifiant la portion de présentation de la taille d'origine.

##### 51.18.5.2 Conditions d'erreur supplémentaires de l'action Set OAP

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si l'un des paramètres de point OAP est spécifié par une valeur absolue et s'il n'est pas dans l'intervalle (0, OVS) sur un axe, ce paramètre est mis à 0.
- Si l'un des paramètres de point OAP est spécifié par une valeur relative et s'il n'est pas dans l'intervalle (0, 100 %) sur un axe, ce paramètre est mis à 0%.

#### 51.18.6 Action Set OVS Position (fixation de la position de la taille OVS)

Cette action spécifie la position de la taille OVS par rapport à la taille d'origine (OS).

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- OVS Position Param (paramètre de position de taille OVS): permet de spécifier un macroparamètre au lieu d'une valeur de position pour la taille OVS par rapport à la taille d'origine;
- Transition Duration Param (paramètre de durée de transition): permet de spécifier un macroparamètre au lieu d'une valeur de position pour la taille OVS par rapport à la taille d'origine.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 51.18.6.1 Effet de l'action Set OVS Position

La position de la taille OVS peut prendre des valeurs absolues dans l'espace OPS ou des valeurs relatives à la taille d'origine (OS) selon le paramètre de point OAP. L'effet de cette action est le suivant:

- fixer la position de la taille OVS de la cible.

Il peut y avoir un effet utilisateur modifiant la portion de présentation de la taille d'origine.

Si une durée de transition non nulle est spécifiée, la position de la taille OVS varie progressivement. Un effet utilisateur est associé à cette action, c'est-à-dire que la partie de présentation de la taille d'origine varie progressivement. La modification de la position de la taille OVS n'a aucune influence sur le point OAP.

## NOTES

1 – N'importe quelle valeur est autorisée pour le paramètre de position de taille OVS (valeurs absolues et valeurs relatives).

2 – Les portions de la taille d'origine qui sortent de la taille OVS ne sont pas perçues par l'utilisateur. A noter que l'action `Set OVS Position` peut positionner la taille OVS entièrement en dehors de la taille d'origine. Dans ce cas, la taille d'origine entière n'est pas perçue.

### 51.18.6.2 Conditions d'erreur supplémentaires de l'action `Set OVS Position`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.18.7 Action `Set PAP` (fixation du point PAP)

Cette action fixe le point PAP d'un rt-composant. Ce point sera spécifié à l'intérieur de la taille PVS.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `PAP Param` (paramètre de point PAP): valeur absolue ou relative.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.18.7.1 Effet de l'action `Set PAP`

Le point PAP peut prendre des valeurs absolues ou des valeurs relatives à la taille PVS selon le paramètre de point PAP. L'effet de cette action est le suivant:

- fixer le point PAP de la cible.

Il peut y avoir un effet utilisateur modifiant la position de présentation dans l'espace RPS.

#### 51.18.7.2 Conditions d'erreur supplémentaires de l'action `Set PAP`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si l'un des paramètres de point PAP est spécifié par une valeur absolue et s'il n'est pas dans l'intervalle (0, PVS) sur un axe, ce paramètre est mis à 0.
- Si l'un des paramètres de point PAP est spécifié par une valeur relative et s'il n'est pas dans l'intervalle (0, 100 %) sur un axe, ce paramètre est mis à 0%.

### 51.18.8 Action `Set PVS Position` (fixation de la position de la taille PVS)

Cette action spécifie la position de la taille PVS par rapport à la taille d'origine (OS) de l'espace RPS.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `PVS Position Param` (paramètre de position de taille PVS): valeur absolue ou relative;
- `Transition Duration Param` (paramètre de durée de transition): valeur absolue ou relative.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.18.8.1 Effet de l'action `Set PVS Position`

La position de la taille PVS peut prendre des valeurs absolues dans l'espace RPS ou des valeurs relatives à la taille d'origine (OS) de l'espace RPS selon le paramètre de point PAP. L'effet de cette action est le suivant:

- fixer la position de la taille PVS de la cible.

Il peut y avoir un effet utilisateur modifiant la position de présentation dans l'espace RPS.

Si une durée de transition non nulle est spécifiée, la position de la taille PVS varie progressivement. Un effet utilisateur est associé à cette action, c'est-à-dire que la position de présentation de la taille PVS varie progressivement. La modification de la position de la taille PVS n'a aucune influence sur le point PAP.

N'importe quelle valeur est autorisée pour le paramètre de position de taille PVS (valeurs absolues et valeurs relatives).

#### 51.18.8.2 Conditions d'erreur supplémentaires de l'action Set PVS Position

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

#### 51.18.9 Action Set GSF (fixation du facteur GSF)

Cette action fixe le facteur GSF de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- GSF Param (paramètre de facteur GSF): fraction générique;
- Transition Duration Param (paramètre de durée de transition): fraction générique.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 51.18.9.1 Effet de l'action Set GSF

Si la cible est en cours d'exécution, il peut y avoir un effet utilisateur. La taille du rt-composant de présentation peut varier.

Si une durée de transition non nulle est spécifiée, la taille de la cible varie progressivement jusqu'au facteur GSF spécifié.

#### 51.18.9.2 Conditions d'erreur supplémentaires de l'action Set GSF

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la valeur du paramètre GSF est négative, elle est mise à 0.
- Si la durée de transition est négative, elle est mise à zéro.

#### 51.18.10 Action Set User Spatial Control (fixation de la commande spatiale utilisateur)

Cette action spécifie si l'utilisateur est en mesure de commander certains rt-composants conformément aux fonctions spécifiées (voir 11.3).

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Component Target Param (paramètre de rt-composant cible);
- ensemble de paramètres Spatial Control Param (paramètre de commande spatiale): fraction générique;
- User Spatial Control Param (paramètre de commande spatiale utilisateur).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 51.18.10.1 Effet de l'action Set User Spatial Control

L'effet de cette action est le suivant:

- les paramètres de commande spatiale spécifiés sont mis à 'allowed' (autorisé) ou 'not-allowed' (pas autorisé) en fonction du paramètre de commande spatiale utilisateur.

Si l'une des fonctions de commande spatiale est autorisée, l'utilisateur peut interagir avec le rt-composant pour exécuter la fonction.

#### 51.18.10.2 Conditions d'erreur supplémentaires de l'action Set User Spatial Control

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.

## 51.19 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par les actions suivantes:

- action `Get OS` (extraction de la taille d'origine) (voir 51.19.1),
- action `Get POS` (extraction de la taille POS) (voir 51.19.2),
- action `Get Aspect Ratio` (extraction du facteur de forme) (voir 51.19.3),
- action `Get Resizing Strategy` (extraction de la stratégie de redimensionnement) (voir 51.19.4),
- action `Get OVS Proj Strategy` (extraction de la stratégie de projection de la taille OVS) (voir 51.19.5),
- action `Get OVS` (extraction de la taille OVS) (voir 51.19.6),
- action `Get OAP` (extraction du point OAP) (voir 51.19.7),
- action `Get OVS Position` (extraction de la position de la taille OVS) (voir 51.19.8),
- action `Get PVS` (extraction de la taille PVS) (voir 51.19.9),
- action `Get PAP` (extraction du point PAP) (voir 51.19.10),
- action `Get PVS Position` (extraction de la position de la taille PVS) (voir 51.19.11),
- action `Get GSF` (extraction du facteur GSF) (voir 51.19.12),
- action `Get User Spatial Control` (extraction de la commande spatiale utilisateur) (voir 51.19.13).

### 51.19.1 Action `Get OS` (extraction de la taille d'origine)

Cette action récupère la taille d'origine (OS) de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.1.1 Effet de l'action `Get OS`

L'effet de cette action est le suivant:

- 1) si le paramètre de résultat d'axe attendu vaut 'x', 'y' ou 'z', la valeur correspondante de la taille d'origine sur cet axe est renvoyée sous forme d'entier générique;
- 2) sinon ('xyz'), les trois valeurs de la taille d'origine sont renvoyées sous forme d'une liste générique contenant trois entiers génériques.

Les valeurs renvoyées sont mesurées en unités OGSU de la cible.

#### 51.19.1.2 Conditions d'erreur supplémentaires de l'action `Get OS`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.2 Action `Get POS` (extraction de la taille POS)

Cette action récupère la taille POS de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible);
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

### 51.19.2.1 Effet de l'action `Get POS`

L'effet de cette action est le suivant:

- 1) si le paramètre de résultat d'axe attendu vaut 'x', 'y' ou 'z', la valeur correspondante de la taille POS sur cet axe est renvoyée sous forme d'entier générique;
- 2) sinon ('xyz'), les trois valeurs de la taille POS sont renvoyées sous forme d'une liste générique contenant trois entiers génériques.

Les valeurs renvoyées sont mesurées en unités RGSU de la cible.

### 51.19.2.2 Conditions d'erreur supplémentaires de l'action `Get POS`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.3 Action `Get Aspect Ratio` (extraction du facteur de forme)

Cette action récupère l'attribut de facteur de forme de la cible.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.3.1 Effet de l'action `Get Aspect Ratio`

Cette action détermine la valeur de l'attribut de facteur de forme de la cible et le renvoie.

#### 51.19.3.2 Conditions d'erreur supplémentaires de l'action `Get Aspect Ratio`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.4 Action `Get Resizing Strategy` (extraction de la stratégie de redimensionnement)

Cette action récupère l'attribut de stratégie de redimensionnement de la cible.

Cette action a le paramètre suivant:

- `Rt-Composite Target Param` (paramètre de rt-composite cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.4.1 Effet de l'action `Get Resizing Strategy`

Cette action détermine la valeur de l'attribut de stratégie de redimensionnement de la cible et le renvoie.

#### 51.19.4.2 Conditions d'erreur supplémentaires de l'action `Get Resizing Strategy`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.5 Action Get OVS Proj Strategy (extraction de la stratégie de projection de la taille OVS)

Cette action récupère l'attribut OVS Proj Strategy de la cible.

Cette action a le paramètre suivant:

- Rt-Component Target Param (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.5.1 Effet de l'action Get OVS Proj Strategy

Cette action détermine la valeur de l'attribut OVS Proj Strategy de la cible et le renvoie.

#### 51.19.5.2 Conditions d'erreur supplémentaires de l'action Get OVS Proj Strategy

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.6 Action Get OVS (extraction de la taille OVS)

Cette action récupère la taille OVS de la cible.

Cette action a les paramètres suivants:

- Rt-Component Target Param (paramètre de rt-composant cible),
- Point Type Param (paramètre de type de point),
- Expected Axis Result Param (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.6.1 Effet de l'action Get OVS

L'effet de cette action est le suivant:

- 1) si le paramètre de type de point vaut 'absolute', cette action détermine la taille OVS par des valeurs absolues en unités OGSU et renvoie des entiers génériques;
- 2) sinon ('relative'), cette action détermine la taille OVS par des valeurs relatives par rapport à la taille d'origine correspondante et renvoie des fractions génériques;
- 3) si le paramètre de résultat d'axe attendu vaut 'x', 'y' ou 'z', la valeur correspondante de la taille OVS sur cet axe est renvoyée;
- 4) sinon ('xyz'), les trois valeurs de la taille OVS sont renvoyées sous forme d'une liste générique contenant trois éléments.

#### 51.19.6.2 Conditions d'erreur supplémentaires de l'action Get OVS

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.7 Action Get OAP (extraction du point OAP)

Cette action récupère le point OAP de la cible.

Cette action a les paramètres suivants:

- Rt-Component Target Param (paramètre de rt-composant cible),
- Point Type Param (paramètre de type de point),
- Expected Axis Result Param (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.7.1 Effet de l'action `Get OAP`

L'effet de cette action est le suivant:

- 1) si le paramètre de type de point vaut 'absolute', cette action détermine le point OAP par des valeurs absolues en unités OGSU dans la taille OVS et renvoie des entiers génériques;
- 2) sinon ('relative'), cette action détermine le point OAP par des valeurs relatives par rapport à la taille OVS correspondante et renvoie des fractions génériques;
- 3) si le paramètre de résultat d'axe attendu vaut 'x', 'y' ou 'z', la valeur correspondante du point OAP sur cet axe est renvoyée;
- 4) sinon ('xyz'), les trois valeurs du point OAP sont renvoyées sous forme d'une liste contenant trois éléments.

#### 51.19.7.2 Conditions d'erreur supplémentaires de l'action `Get OAP`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

#### 51.19.8 Action `Get OVS Position` (extraction de la position de la taille OVS)

Cette action récupère la position de la taille OVS de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Point Type Param` (paramètre de type de point),
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.8.1 Effet de l'action `Get OVS Position`

L'effet de cette action est le suivant:

- 1) si le paramètre de type de point vaut 'absolute', cette action détermine la position de la taille OVS par des valeurs absolues en unités OGSU et renvoie des entiers génériques;
- 2) sinon ('relative'), cette action détermine la position de la taille OVS par des valeurs relatives par rapport à la taille d'origine correspondante et renvoie des fractions génériques;
- 3) si le paramètre de résultat d'axe attendu vaut 'x', 'y' ou 'z', la valeur correspondante de la position de la taille OVS sur cet axe est renvoyée;
- 4) sinon ('xyz'), les trois valeurs de la position de la taille OVS sont renvoyées sous forme d'une liste contenant trois éléments.

#### 51.19.8.2 Conditions d'erreur supplémentaires de l'action `Get OVS Position`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

#### 51.19.9 Action `Get PVS` (extraction de la taille PVS)

Cette action récupère la taille PVS de la cible.



Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Point Type Param` (paramètre de type de point),
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### **51.19.9.1 Effet de l'action `Get PVS`**

L'effet de cette action est le suivant:

- 1) si le paramètre de type de point vaut `'absolute'`, cette action détermine la taille PVS par des valeurs absolues en unités RGSU de la cible et renvoie des entiers génériques;
- 2) sinon (`'relative'`), cette action détermine la taille PVS par des valeurs relatives par rapport à la taille d'origine du parent et renvoie des fractions génériques;
- 3) si le paramètre de résultat d'axe attendu vaut `'x'`, `'y'` ou `'z'`, la valeur correspondante de la taille PVS sur cet axe est renvoyée;
- 4) sinon (`'xyz'`), les trois valeurs de la taille PVS sont renvoyées sous forme d'une liste contenant trois éléments.

#### **51.19.9.2 Conditions d'erreur supplémentaires de l'action `Get PVS`**

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique `'*'` au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

#### **51.19.10 Action `Get PAP` (extraction du point PAP)**

Cette action récupère le point PAP de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Point Type Param` (paramètre de type de point),
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### **51.19.10.1 Effet de l'action `Get PAP`**

L'effet de cette action est le suivant:

- 1) si le paramètre de type de point vaut `'absolute'`, cette action détermine le point PAP par des valeurs absolues en unités RGSU de la cible dans la taille PVS et renvoie des entiers génériques;
- 2) sinon (`'relative'`), cette action détermine le point PAP par des valeurs relatives par rapport à la taille PVS correspondante et renvoie des fractions génériques;
- 3) si le paramètre de résultat d'axe attendu vaut `'x'`, `'y'` ou `'z'`, la valeur correspondante du point PAP sur cet axe est renvoyée;
- 4) sinon (`'xyz'`), les trois valeurs du point PAP sont renvoyées sous forme d'une liste contenant trois éléments.

#### **51.19.10.2 Conditions d'erreur supplémentaires de l'action `Get PAP`**

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique `'*'` au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.11 Action `Get PVS Position` (extraction de la position de la taille PVS)

Cette action récupère la position de la taille PVS de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Point Type Param` (paramètre de type de point),
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.11.1 Effet de l'action `Get PVS Position`

L'effet de cette action est le suivant:

- 1) si le paramètre de type de point vaut `'absolute'`, cette action détermine la position de la taille PVS par des valeurs absolues en unités RGSU de la cible et renvoie des entiers génériques;
- 2) sinon (`'relative'`), cette action détermine la position de la taille PVS par des valeurs relatives par rapport à la taille d'origine du parent et renvoie des fractions génériques;
- 3) si le paramètre de résultat d'axe attendu vaut `'x'`, `'y'` ou `'z'`, la valeur correspondante de la position de la taille PVS sur cet axe est renvoyée;
- 4) sinon (`'xyz'`), les trois valeurs de la position de la taille PVS sont renvoyées sous forme d'une liste contenant trois éléments.

#### 51.19.11.2 Conditions d'erreur supplémentaires de l'action `Get PVS Position`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique `'*'` au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.12 Action `Get GSF` (extraction du facteur GSF)

Cette action récupère le facteur GSF de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Expected Axis Result Param` (paramètre de résultat d'axe attendu).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 51.19.12.1 Effet de l'action `Get GSF`

L'effet de cette action est le suivant:

- 1) si le paramètre de résultat d'axe attendu vaut `'x'`, `'y'` ou `'z'`, le facteur GSF correspondant de la cible sur cet axe est renvoyé sous forme d'entier générique;
- 2) sinon (`'xyz'`), les trois facteurs GSF de la cible sont renvoyés sous forme d'une liste contenant trois entiers génériques.

#### 51.19.12.2 Conditions d'erreur supplémentaires de l'action `Get GSF`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique `'*'` au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 51.19.13 Action `Get User Spatial Control` (extraction de la commande spatiale utilisateur)

Cette action récupère la valeur de l'attribut de commande spatiale utilisateur.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Spatial Control Param` (paramètre de commande spatiale).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### **51.19.13.1 Effet de l'action `Get User Spatial Control`**

L'effet de cette action est le suivant:

- selon la spécification du paramètre de commande spatiale ('moving', 'resizing', 'scaling' ou 'scrolling'), cette action renvoie 'allowed' (autorisé) ou 'not-allowed' (pas autorisé).

#### **51.19.13.2 Conditions d'erreur supplémentaires de l'action `Get User Spatial Control`**

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

## **52 Comportement sonore des rt-composants**

Ce comportement décrit la présentation sonore des rt-composants. La présente Recommandation définit une valeur initiale pour ce comportement et des moyens permettant de le modifier et de récupérer sa valeur par l'utilisation d'actions.

### **52.1 Attributs et états du comportement**

Ce comportement est exprimé au moyen des attributs ou états suivants:

- `OV` (volume sonore d'origine) (voir 52.2),
- `CV` (volume sonore courant) (voir 52.3),
- `PCV` (volume sonore courant projeté) (voir 52.4),
- `GVF` (facteur de volume sonore générique) (voir 52.5).

### **52.2 `OV` (volume sonore d'origine)**

Chaque rt-contenu a un volume sonore d'origine (`OV`) défini dans la plage du volume sonore (`AVR`). Toutefois, aucun volume sonore d'origine n'est associé aux rt-composites. Ce volume est toujours mesuré en unités de volume sonore génériques (`GVU`) et sera dans la plage `AVR` (voir «Espace de présentation d'origine» pour l'initialisation de chaque volume sonore d'origine).

Chaque volume sonore d'origine d'un rt-contenu est fixé une fois pour toutes et ne sera pas modifié.

### **52.3 `CV` (volume sonore courant)**

Chaque rt-contenu a un volume sonore courant (`CV`) dont la valeur initiale est le volume sonore d'origine (`OV`). Il est possible de modifier le volume sonore courant au moyen de l'action `Set CV` (fixation du volume sonore courant), toutefois, ce volume restera dans la plage `AVR`.

### **52.4 `PCV` (volume sonore courant projeté)**

Le projeté du volume sonore courant (`CV`) dans l'espace de présentation relatif (`RPS`) est appelé volume sonore courant projeté (`PCV`). Le volume `PCV` est calculé en fonction du facteur de volume sonore générique (`GVF`) de son espace `RPS` (espace `PRPS` ou `CPS`) par  $CV \times GVF$ .

NOTE – Chaque rt-composant (rt-contenu et rt-composite) a un facteur `GTF` et un facteur `GSF`. Toutefois, les rt-contents n'ont pas de facteur `GVF`. Les rt-composites et les canaux ont un facteur `GVF`. Les rt-contents n'ont pas de facteur `GVF`, mais au lieu de cela leur volume sonore courant peut varier.

## 52.5 GVF (facteur de volume sonore générique)

La projection du volume sonore courant (CV) dans l'espace de présentation d'origine (OPS) pour donner le volume PCV dans l'espace RPS est commandée par le facteur GVF de l'espace RPS. Le facteur GVF définit le nombre d'unités de volume sonore génériques relatives (RGVU) correspondant à une unité de volume sonore générique d'origine (OGVU) dans l'espace OPS.

## 52.6 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Set CV` (fixation du volume sonore courant) (voir 52.6.1),
- action `Set GVF` (fixation du facteur GVF) (voir 52.6.2).

### 52.6.1 Action `Set CV` (fixation du volume sonore courant)

Cette action fixe le volume sonore courant (CV) de la cible selon les spécifications.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Content Target Param` (paramètre de rt-contenu cible);
- `CV Param` (paramètre de volume sonore courant): valeur absolue, valeur relative, facteur du point d'origine ou facteur du point courant;
- `Transition Duration Param` (paramètre de durée de transition): valeur absolue, valeur relative, facteur du point d'origine ou facteur du point courant.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 52.6.1.1 Effet de l'action `Set CV`

L'effet de cette action est le suivant:

- 1) si le volume sonore courant (CV) est spécifié par une valeur absolue, cette valeur est affectée au volume sonore courant de la cible;
- 2) si le volume sonore courant est spécifié par une valeur relative, cette valeur est interprétée comme étant une valeur relative associée à la plage AVR (0, 255) et le volume sonore est calculé par  $255 \times$  (valeur spécifiée). La valeur calculée est affectée au volume sonore courant de la cible;
- 3) si le volume sonore courant est spécifié par un facteur du point d'origine, on calcule la valeur  $OV \times$  (valeur spécifiée) puis on l'affecte au volume sonore courant de la cible;
- 4) sinon (le volume sonore courant est spécifié par un facteur du point courant), on calcule la valeur  $CV \times$  (valeur spécifiée) et on l'affecte au volume sonore courant;
- 5) dans les cas ci-dessus, si la valeur calculée sort de la plage AVR, la valeur est mise à 0 (si elle est négative) ou à 255 (si elle est supérieure à 255).

Si une durée de transition non nulle est spécifiée, le volume sonore courant varie progressivement jusqu'à la valeur spécifiée.

#### 52.6.1.2 Conditions d'erreur supplémentaires de l'action `Set CV`

- Si l'une des cibles n'est pas un rt-contenu en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si le paramètre CV défini par une valeur absolue n'est pas dans la plage AVR, l'action est ignorée pour l'ensemble des cibles en entier.
- Si la durée de transition est négative, elle est mise à zéro.

### 52.6.2 Action `Set GVF` (fixation du facteur GVF)

Cette action spécifie le facteur GVF d'un rt-composite ou d'un canal.

Cette action a les paramètres suivants:

- `GVF Target` (cible pour le facteur GVF),
- `GVF Param` (paramètre de facteur GVF): valeur du facteur GVF à affecter,
- `Transition Duration Param` (paramètre de durée de transition): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3, R3.TD et C2.

#### 52.6.2.1 Effet de l'action `Set GVF`

L'effet de cette action est le suivant:

- 1) le facteur GVF de la cible est mis à la valeur spécifiée;
- 2) si la cible est un rt-composite, tous les volumes sonores courants projetés (PCV) des descendants sont recalculés en fonction de cette valeur;
- 3) sinon (la cible est un canal), tous les volumes PCV des rt-composants affectés à ce canal sont recalculés en fonction de cette valeur.

Si une durée de transition non nulle est spécifiée, les volumes PCV des cibles varient progressivement.

#### 52.6.2.2 Conditions d'erreur supplémentaires de l'action `Set GVF`

- Si l'une des cibles n'est ni un rt-composite ni un canal en phase R2, R3, R3.TD ou C2, l'action est ignorée pour cette cible.
- Si la durée de transition est négative, elle est mise à zéro.

### 52.7 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par les actions suivantes:

- action `Get OV` (extraction du volume sonore d'origine) (voir 52.7.1),
- action `Get CV` (extraction du volume sonore courant) (voir 52.7.2),
- action `Get PCV` (extraction du volume PCV) (voir 52.7.3),
- action `Get GVF` (extraction du facteur GVF) (voir 52.7.4).

#### 52.7.1 Action `Get OV` (extraction du volume sonore d'origine)

Cette action récupère le volume sonore d'origine (OV) de la cible.

Cette action a le paramètre suivant:

- `Rt-Content Target Param` (paramètre de rt-contenu cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 52.7.1.1 Effet de l'action `Get OV`

L'effet de cette action est le suivant:

- récupérer le volume sonore d'origine (OV) de la cible exprimé en unités OGVU.

##### 52.7.1.2 Conditions d'erreur supplémentaires de l'action `Get OV`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-contenu en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 52.7.2 Action `Get CV` (extraction du volume sonore courant)

Cette action récupère le volume sonore courant (CV) de la cible.

Cette action a le paramètre suivant:

- `Rt-Content Target Param` (paramètre de rt-contenu cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 52.7.2.1 Effet de l'action `Get CV`

L'effet de cette action est le suivant:

- récupérer le volume sonore courant (CV) de la cible exprimé en unités OGVU.

#### 52.7.2.2 Conditions d'erreur supplémentaires de l'action `Get CV`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-contenu en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 52.7.3 Action `Get PCV` (extraction du volume PCV)

Cette action récupère le volume PCV de la cible.

Cette action a le paramètre suivant:

- `Rt-Content Target Param` (paramètre de rt-contenu cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 52.7.3.1 Effet de l'action `Get PCV`

L'effet de cette action est le suivant:

- récupérer le volume PCV de la cible exprimé en unités RGVU.

##### 52.7.3.2 Conditions d'erreur supplémentaires de l'action `Get PCV`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-contenu en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 52.7.4 Action `Get GVF` (extraction du facteur GVF)

Cette action récupère le facteur GVF de la cible.

Cette action a les paramètres suivants:

- `Rt-Composite Target Param` (paramètre de rt-composite cible),
- `Channel Target Param` (paramètre de canal cible).

Cette action peut être appliquée pendant les phases R2, R3, R3.TD et C2.

##### 52.7.4.1 Effet de l'action `Get GVF`

L'effet de cette action est le suivant:

- récupérer le facteur GVF de la cible sous forme de fraction générique.

##### 52.7.4.2 Conditions d'erreur supplémentaires de l'action `Get GVF`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est ni un rt-composant ni un canal en phase R2, R3, R3.TD ou C2, l'action est ignorée pour cette cible.

## 53 Comportement de choix de flux d'un rt-mux

Ce comportement décrit le choix des flux dans un rt-mux. Une fois les flux choisis, le rt-mux est responsable de ces flux choisis et de leur présentation.

### 53.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- `Stream Choice` (choix de flux) (voir 53.2),
- `Stream Chosen State` (état de choix de flux) (voir 53.3).

### 53.2 Stream Choice (choix de flux)

Un rt-mux est responsable de la présentation de l'ensemble des données multiplexées ou de la présentation de certains flux spécifiés choisis individuellement parmi les données multiplexées.

L'attribut de choix de flux comprend une liste des identificateurs des flux choisis.

Initialement, tous les flux d'un rt-mux sont choisis, c'est-à-dire que le rt-mux est responsable de la présentation de l'ensemble des données multiplexées. Cet attribut comprend donc les identificateurs de tous les flux.

Une fois que plusieurs flux d'un rt-mux sont choisis, le rt-mux est responsable de la présentation de ces flux lorsqu'il passe à l'état «en cours d'exécution». Dans ce cas, le processus de démultiplexage précède immédiatement la présentation des flux, le démultiplexage ne génère donc aucune nouvelle donnée.

### 53.3 Stream Chosen State (état de choix de flux)

Chaque flux d'un rt-mux a un état de choix de flux. Si le flux est choisi, l'état vaut 'chosen' (choisi). Sinon, l'état vaut 'not-chosen' (pas choisi).

Chaque rt-mux a aussi un état de choix de flux. Si tous les flux du rt-mux sont choisis, l'état vaut 'chosen'. Si certains de ses flux ne sont pas choisis, l'état vaut 'partially-chosen' (partiellement choisi). Sinon (aucun flux n'est choisi), l'état vaut 'not-chosen' (pas choisi).

### 53.4 Stream Identification (identification de flux)

Choix entre un identificateur de flux et 'all-streams' (tous les flux).

### 53.5 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action Set Stream Choice (fixation du choix de flux) (voir 53.5.1).

#### 53.5.1 Action Set Stream Choice (fixation du choix de flux)

L'action choisit les flux d'un rt-mux cible à présenter.

Cette action a les paramètres suivants:

- ensemble de paramètres Rt-Mux Target Param (paramètre de rt-mux cible);
- ensemble de paramètres Stream Spec Param (paramètre de spécification de flux): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

##### 53.5.1.1 Effet de l'action Set Stream Choice

L'effet de cette action est le suivant:

- 1) selon la commande de mise à jour spécifiée, des flux sont ajoutés, supprimés ou remplacés par un nouvel ensemble;
- 2) si une commande «remove» (supprimer) est utilisée et si aucun flux n'est spécifié, tous les flux alors choisis passent à l'état 'not-chosen' (pas choisi).

Cette action a pour résultat la modification des états de choix de flux de certains flux et du rt-mux. Il peut y avoir un effet utilisateur si la cible est en cours d'exécution.

##### 53.5.1.2 Conditions d'erreur supplémentaires de l'action Set Stream Choice

- Si l'une des cibles n'est pas un rt-mux en <phase valide>, l'action est ignorée pour cette cible.

### 53.6 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par les actions suivantes:

- action Get Stream Choice (extraction du choix de flux) (voir 53.6.1),
- action Get Stream Chosen State (extraction de l'état de choix de flux) (voir 53.6.2).

### 53.6.1 Action `Get Stream Choice` (extraction du choix de flux)

Cette action récupère la liste des identificateurs des flux choisis de la cible.

Cette action a le paramètre suivant:

- `Rt-Mux Target Param` (paramètre de rt-mux cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 53.6.1.1 Effet de l'action `Get Stream Choice`

L'effet de cette action est le suivant:

- récupération d'une liste d'identificateurs de flux choisis sous forme de liste générique.

#### 53.6.1.2 Conditions d'erreur supplémentaires de l'action `Get Stream Choice`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-mux en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

### 53.6.2 Action `Get Stream Chosen State` (extraction de l'état de choix de flux)

Cette action récupère l'état de flux de la cible.

Cette action a les paramètres suivants:

- `Rt-Mux Target Param` (paramètre de rt-mux cible);
- `Stream Identification Param` (paramètre d'identification de flux): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 53.6.2.1 Effet de l'action `Get Stream Chosen State`

L'effet de cette action est le suivant:

- 1) si un identificateur de flux est spécifié comme valeur du paramètre d'identification de flux, l'état du flux est récupéré;
- 2) sinon [`'all-streams'` (tous les flux) est spécifié comme valeur du paramètre d'identification de flux], l'état de flux du rt-mux cible est récupéré.

#### 53.6.2.2 Conditions d'erreur supplémentaires de l'action `Get Stream Chosen State`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-mux en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.

## 54 Comportement d'interaction

Ce comportement décrit l'interaction entre les rt-composants et l'utilisateur. Ce comportement assure l'interactivité du moteur MHEG et des entités MHEG côté utilisateur. Tous les rt-composants peuvent être interactifs. Initialement, aucun rt-composant n'est interactif.

Le comportement d'événement assure aussi l'interactivité du moteur MHEG et des entités MHEG côté utilisateur. Toutefois, ce comportement d'interaction se rapporte plus particulièrement à la sélection et à la modification par l'utilisateur qui sont fréquemment utilisées.



## 54.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- `Interaction Type` (type d'interaction) (voir 54.2),
- `Interaction Status` (état d'interaction) (voir 54.3),
- `Selection Status` (état de sélection) (voir 54.4),
- `Modification Status` (état de modification) (voir 54.5),
- `Interaction Ability` (capacité d'interaction) (voir 54.6),
- `Selectability` (sélectionnabilité) (voir 54.7),
- `Modifiability` (modifiabilité) (voir 54.8),
- `Min Interact Required` (interaction minimale requise) (voir 54.9),
- `Max Interact Required` (interaction maximale requise) (voir 54.10),
- `Number Of Interacted Sockets` (nombre de réceptacles en interaction) (voir 54.11).

## 54.2 Interaction Type (type d'interaction)

Il existe deux types d'interaction pour les rt-composants. L'un, 'selection', concerne la sélection par l'utilisateur, et l'autre, 'modification', concerne la modification par l'utilisateur. Les deux types de sélection sont indépendants l'un de l'autre de sorte qu'un rt-composant peut avoir l'un des types d'interaction ou les deux simultanément.

Le moteur MHEG doit gérer la sélection et la modification des rt-composants avec l'aide de l'interface utilisateur. Grâce à la valeur des attributs d'interaction suivants, le moteur MHEG connaît les interactions de l'utilisateur avec les rt-composants.

L'utilisateur ne peut interagir avec des rt-composants que s'ils sont en cours d'exécution. Si un rt-composant n'est pas en cours d'exécution, l'utilisateur ne peut pas interagir avec lui et les états concernant le comportement d'interaction ne seront pas modifiés.

Les deux types d'interactions sont commandés et récupérés par le même ensemble d'actions élémentaires et d'actions d'extraction spécifiant le type d'interaction en paramètre.

## 54.3 Interaction Status (état d'interaction)

Chaque rt-composant peut avoir un état de sélection indiquant s'il est sélectionné par l'utilisateur et un état de modification indiquant s'il est modifié par l'utilisateur.

L'état d'interaction d'un rt-composant n'est fixé que si ce rt-composant est à l'état «en cours d'exécution». L'état d'interaction est fixé explicitement par l'action `Set Interaction Status` (fixation de l'état d'interaction) ou implicitement par interaction de l'utilisateur.

Ce sont le moteur MHEG et l'interface utilisateur qui dictent la façon d'interagir avec les rt-composants.

### NOTES

1 – Les rt-contenus peuvent être sélectionnés par un positionnement et un cliquage de souris, et peuvent être modifiés par un positionnement de souris et la frappe d'une touche.

2 – Non seulement les rt-contenus visuels mais aussi certains contenus multiplexés et rt-contenus sonores peuvent être sélectionnés ou modifiés. La façon de sélectionner ou de modifier dépend de la réalisation du moteur MHEG et de l'application utilisatrice.

L'état d'interaction peut être perceptible (de manière visuelle ou sonore) pour montrer l'état courant d'un rt-composant à l'utilisateur. Toutefois, il s'agit d'une caractéristique du moteur MHEG et la présente Recommandation ne spécifie pas de méthode pour cela.

## 54.4 Selection Status (état de sélection)

L'état de sélection vaut 'selected' (sélectionné) ou 'not-selected' (pas sélectionné). Il indique les résultats de la sélection par l'utilisateur.

Initialement, tous les rt-composants sont à 'not-selected'.

#### 54.5 Modification Status (**état de modification**)

L'état de modification vaut 'modified' (modifié), 'not-modified' (pas modifié) ou 'modifying' (en cours de modification). Il indique les résultats de la modification par l'utilisateur.

Initialement, tous les rt-composants sont à 'not-modified'.

#### 54.6 Interaction Ability (**capacité d'interaction**)

Les deux types d'interaction peuvent être affectés séparément à chaque rt-composant. La possibilité de sélectionner un élément est appelée sélectionnabilité et la possibilité de modifier un élément est appelée modifiabilité.

La sélectionnabilité indique au moteur MHEG si un rt-composant peut être sélectionné. Et la modifiabilité indique au moteur MHEG si ce rt-composant peut être modifié. Si un rt-composant peut être sélectionné et modifié, l'utilisateur peut le sélectionner ou le modifier.

La capacité d'interaction peut être perceptible (de manière visuelle ou sonore) pour montrer la capacité d'interaction d'un rt-composant à l'utilisateur. Toutefois, il s'agit d'une caractéristique du moteur MHEG et la présente Recommandation ne spécifie aucune méthode pour cela.

#### 54.7 Selectability (**sélectionnabilité**)

Chaque rt-composant a une sélectionnabilité valant 'selectable' (sélectionnable) ou 'not-selectable' (pas sélectionnable). Cet attribut n'est pas spécifié directement par une valeur; il est spécifié par une valeur de l'attribut d'interaction maximale requise pour la sélection.

La valeur de l'attribut de sélectionnabilité est déterminée comme suit:

- si l'attribut d'interaction maximale requise pour la sélection vaut 0, l'attribut de sélectionnabilité vaut 'not-selectable'.

Sinon, il vaut 'selectable'.

Si un rt-composant est 'selectable', l'utilisateur peut interagir avec lui et le sélectionner. Sinon, la sélection par l'utilisateur est interdite.

#### 54.8 Modifiability (**modifiabilité**)

Chaque rt-composant a une modifiabilité valant 'modifiable' ou 'not-modifiable'. Cet attribut n'est pas spécifié directement par une valeur; il est spécifié par une valeur de l'attribut d'interaction maximale requise pour la modification.

La valeur de l'attribut de modifiabilité est déterminée comme suit:

- si l'attribut d'interaction maximale requise pour la modification vaut 0, l'attribut de modifiabilité vaut 'not-modifiable'.

Sinon, il vaut 'modifiable'.

Si un rt-composant est 'modifiable', l'utilisateur peut interagir avec lui et le modifier. Sinon, la modification par l'utilisateur est interdite.

#### 54.9 Min Interact Required (**interaction minimale requise**)

Les états de sélection et de modification d'un rt-contenu sont déterminés directement par une interaction de l'utilisateur car l'utilisateur peut directement sélectionner et modifier le rt-contenu. Mais les états de sélection et de modification d'un rt-composite ne sont pas déterminés directement par une interaction de l'utilisateur; ils sont déterminés dynamiquement comme suit:

- si l'action `Set Interaction Status` (fixation de l'état d'interaction) est appliquée à un rt-composite, celui-ci passe à l'état spécifié indépendamment des attributs d'interaction minimale requise et d'interaction maximale requise, et d'éventuels rt-composants enfants qui sont dans l'état 'modifying' (en cours de modification).

NOTE 1 – Il appartient à l'auteur de décider s'il faut conserver la cohérence entre ces valeurs d'attributs si une action `Set Interaction Status` explicite est utilisée.

Si l'un des rt-composants d'un rt-composite est dans l'état 'modifying', l'état du rt-composite est alors 'modifying'.

Si le nombre de rt-composants sélectionnés ou modifiés dans ce rt-composite est supérieur ou égal à la valeur de l'attribut d'interaction minimale requise, s'il est inférieur ou égal à la valeur de l'attribut d'interaction maximale requise, et si le processus d'interaction de l'utilisateur est terminé, l'état du rt-composite est alors 'selected' (sélectionné) ou 'modified' (modifié).

NOTE 2 – Le moteur MHEG doit fournir un mécanisme qui contrôle la fin du processus d'interaction de l'utilisateur. Il ne s'agit pas d'un processus automatique déterminant l'état d'interaction. L'interaction de l'utilisateur peut se poursuivre jusqu'à ce que le nombre de rt-composants sélectionnés ou modifiés atteigne la valeur de l'attribut d'interaction maximale requise même si le nombre est supérieur à la valeur de l'attribut d'interaction minimale requise.

Sinon, le rt-composite est 'not-selected' (pas sélectionné) ou 'not-modified' (pas modifié).

Par conséquent, les attributs d'interaction minimale requise et d'interaction maximale requise sont fournis au moteur MHEG à titre d'indication pour déterminer l'état d'interaction du rt-composite.

Le nombre de rt-composants en interaction dans un rt-composite peut changer dynamiquement à la suite d'une action Plug (enfichage). Il appartient au moteur MHEG de maintenir la cohérence entre les états des enfants, l'état du parent, et les attributs d'interaction minimale requise et d'interaction maximale requise du parent.

NOTE 3 – Par exemple, si un rt-composant sélectionné est enfiché dans un rt-composite dont l'attribut d'interaction maximale requise vaut 5 et le nombre de réceptacles sélectionnés est égal à 6, le moteur MHEG démarre le processus d'interaction de l'utilisateur et demande à l'utilisateur de désélectionner l'un des 6 réceptacles sélectionnés afin de satisfaire à la condition d'interaction maximale requise.

NOTE 4 – Par exemple, si l'attribut d'interaction minimale requise vaut 2 et si l'un des réceptacles sélectionnés est désenfiché, le moteur MHEG démarre le processus d'interaction de l'utilisateur et demande à l'utilisateur de sélectionner au moins un réceptacle de plus.

NOTE 5 – Par exemple, si l'attribut d'interaction minimale requise vaut 5 et si 4 réceptacles sont enfichés dans un rt-composite, le moteur MHEG continue le processus d'interaction de l'utilisateur jusqu'à ce que d'autres rt-composants soient enfichés et qu'au moins 5 réceptacles soient sélectionnés.

NOTE 6 – Si un rt-composite est à «pas en cours d'exécution», une certaine incohérence peut exister parmi les attributs. Etant donné que l'utilisateur ne peut interagir que si le rt-composite est en cours d'exécution, le moteur MHEG peut redemander ultérieurement à l'utilisateur d'interagir pour ce rt-composite si une autre action Run est appliquée. Supposons qu'un rt-composite avec une durée de 1 minute et un attribut de terminaison temporelle valant «arrêt» est en cours d'exécution et que son attribut d'interaction minimale requise vaut 1. Dans ce cas, si l'utilisateur n'interagit pas avec les réceptacles du rt-composite lorsqu'il est en cours d'exécution, le nombre de réceptacles en interaction est inférieur à la valeur de l'attribut d'interaction minimale requise. S'il est remis à «en cours d'exécution», le processus d'interaction de l'utilisateur recommence. Cela peut être considéré comme un mécanisme de temporisation de l'interaction utilisateur.

L'attribut d'interaction minimale requise n'est valable que pour les rt-composites. Pour les rt-contenus, la valeur est toujours ignorée et n'a pas de signification.

Si la valeur de l'attribut d'interaction minimale requise d'un rt-composite est supérieure ou égale à 1, cela signifie qu'il faut au moins un enfant en interaction pour modifier l'état d'interaction du rt-composite parent.

Initialement, l'attribut est mis à 0.

#### **54.10 Max Interact Required (interaction maximale requise)**

L'attribut d'interaction maximale requise sert à déterminer la capacité d'interaction d'un rt-composant.

Pour un rt-composite, l'attribut donne aussi une indication au moteur MHEG concernant la commande du nombre d'enfants en interaction. Si la valeur de l'attribut d'interaction maximale requise est supérieure ou égale à 1, cela signifie qu'un enfant en interaction est le nombre maximal d'enfants en interaction autorisés pour le rt-composite.

La valeur de l'attribut d'interaction maximale requise sera supérieure ou égale à la valeur de l'attribut d'interaction minimale requise pour chaque type d'interaction.

Le moteur MHEG empêchera que le nombre de rt-composants en interaction soit supérieur à la valeur spécifiée de l'attribut d'interaction maximale requise.

Initialement, l'attribut est mis à 0.

#### **54.11 Number of Interacted Sockets (nombre de réceptacles en interaction)**

Le nombre d'enfants en interaction peut être déterminé. L'attribut du nombre de réceptacles en interaction stocke la valeur qui peut être récupérée par l'action Get Number Of Interacted Sockets (extraction du nombre de réceptacles en interaction).

Initialement, aucun réceptacle n'est en interaction. Par conséquent, l'attribut est mis à 0.

## 54.12 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action `Set Interaction Ability` (fixation de la capacité d'interaction) (voir 54.12.1),
- action `Set Interaction Status` (fixation de l'état d'interaction) (voir 54.12.2).

### 54.12.1 Action `Set Interaction Ability` (fixation de la capacité d'interaction)

Cette action fixe la capacité d'interaction de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `Interaction Type Param` (paramètre de type d'interaction): valeur du facteur GVF à affecter;
- `Min Interact Required Param` (paramètre d'interaction minimale requise): valeur du facteur GVF à affecter;
- `Max Interact Required Param` (paramètre d'interaction maximale requise): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 54.12.1.1 Effet de l'action `Set Interaction Ability`

Selon le type d'interaction spécifié, la sélection ou la modification de la cible est fixée. L'effet de cette action est le suivant:

- 1) l'attribut d'interaction minimale requise est fixé à la valeur spécifiée;
- 2) l'attribut d'interaction maximale requise est fixé à la valeur spécifiée.

La valeur de l'attribut d'interaction minimale requise n'a pas de signification pour les rt-contenus.

#### 54.12.1.2 Conditions d'erreur supplémentaires de l'action `Set Interaction Ability`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si la valeur du paramètre d'interaction minimale requise est négative, elle est mise à 0.
- Si la valeur du paramètre d'interaction maximale requise est négative, elle est mise à 0.
- Si la valeur d'interaction minimale requise est supérieure à la valeur d'interaction maximale requise, l'action est ignorée pour l'ensemble des cibles en entier.
- Si la valeur du paramètre d'interaction minimale requise n'est pas dans l'intervalle (0, nombre de réceptacles enfants de la cible), l'action est ignorée pour l'ensemble des cibles en entier.
- Si la valeur du paramètre d'interaction maximale requise n'est pas dans l'intervalle (0, nombre de réceptacles enfants de la cible), l'action est ignorée pour l'ensemble des cibles en entier.
- Si le nombre courant de réceptacles en interaction est supérieur à la valeur d'interaction maximale requise spécifiée, l'action est ignorée pour cette cible.

### 54.12.2 Action `Set Interaction Status` (fixation de l'état d'interaction)

Cette action fixe l'état d'interaction de la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `Interaction Type Param` (paramètre de type d'interaction): valeur du facteur GVF à affecter;
- `Interaction Status Param` (paramètre d'état d'interaction): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R3 et R3.TD.

#### 54.12.2.1 Effet de l'action `Set Interaction Status`

L'effet de cette action consiste à fixer l'état d'interaction en fonction du type d'interaction spécifié comme suit:

- 1) si le type d'interaction spécifié est 'selection', l'état d'interaction de la cible est mis à 'selected' (sélectionné) ou 'not-selected' (pas sélectionné);
- 2) sinon (le type d'interaction spécifié est 'modification'), l'état d'interaction de la cible est mis à 'modified' (modifié), 'not-modified' (pas modifié) ou 'modifying' (en cours de modification).

Cette action fixe l'état d'interaction indépendamment des valeurs des autres attributs concernés. Il appartient à l'auteur de décider s'il faut maintenir la cohérence entre les valeurs des attributs.

#### 54.12.2.2 Conditions d'erreur supplémentaires de l'action `Set Interaction Status`

- Si l'une des cibles n'est pas un rt-composant en phase R3 ou R3.TD, l'action est ignorée pour cette cible.

### 54.13 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par les actions suivantes:

- action `Get Interaction Ability` (extraction de la capacité d'interaction) (voir 54.13.1),
- action `Get Min Interact Required` (extraction de l'interaction minimale requise) (voir 54.13.2),
- action `Get Max Interact Required` (extraction de l'interaction maximale requise) (voir 54.13.3 `Action Get Max Interact Required`),
- action `Get Interaction Status` (extraction de l'état d'interaction) (voir 54.13.4),
- action `Get Number Of Interacted Sockets` (extraction du nombre de réceptacles en interaction) (voir 54.13.5).

#### 54.13.1 Action `Get Interaction Ability` (extraction de la capacité d'interaction)

Cette action récupère la capacité d'interaction de la cible.

Cette action a les paramètres suivants:

- `Rt-Component Target Param` (paramètre de rt-composant cible),
- `Interaction Type Param` (paramètre de type d'interaction): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 54.13.1.1 Effet de l'action `Get Interaction Ability`

Selon le type d'interaction spécifié, l'effet de cette action est le suivant:

- 1) si le type d'interaction est 'selection', cette action renvoie 'selectable' ou 'not-selectable';
- 2) sinon (le type d'interaction est 'modification'), cette action renvoie 'modifiable' ou 'not-modifiable'.

#### 54.13.1.2 Conditions d'erreur supplémentaires de l'action `Get Interaction Ability`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3, R3.TD, l'action est ignorée pour cette cible.

### 54.13.2 Action Get Min Interact Required (extraction de l'interaction minimale requise)

Cette action récupère l'attribut d'interaction minimale requise de la cible.

Cette action a les paramètres suivants:

- Rt-Component Target Param (paramètre de rt-composant cible);
- Interaction Type Param (paramètre de type d'interaction): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 54.13.2.1 Effet de l'action Get Min Interact Required

Selon le type d'interaction spécifié (sélection ou modification), la valeur de l'attribut d'interaction minimale requise est récupérée.

#### 54.13.2.2 Conditions d'erreur supplémentaires de l'action Get Min Interact Required

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 54.13.3 Action Get Max Interact Required (extraction de l'interaction maximale requise)

Cette action récupère l'attribut d'interaction maximale requise de la cible.

Cette action a les paramètres suivants:

- Rt-Component Target Param (paramètre de rt-composant cible),
- Interaction Type Param (paramètre de type d'interaction): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 54.13.3.1 Effet de l'action Get Max Interact Required

Selon le type d'interaction spécifié (sélection ou modification), la valeur de l'attribut d'interaction maximale requise est récupérée.

#### 54.13.3.2 Conditions d'erreur supplémentaires de l'action Get Max Interact Required

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

### 54.13.4 Action Get Interaction Status (extraction de l'état d'interaction)

Cette action récupère l'état d'interaction de la cible.

Cette action a les paramètres suivants:

- Rt-Component Target Param (paramètre de rt-composant cible),
- Interaction Type Param (paramètre de type d'interaction): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 54.13.4.1 Effet de l'action `Get Interaction Status`

Conformément au type d'interaction spécifié, l'effet de cette action est le suivant:

- 1) si le type d'interaction est 'selection', l'état de sélection de la cible est renvoyé;
- 2) sinon (le type d'interaction est 'modification'), l'état de modification de la cible est renvoyé.

#### 54.13.4.2 Conditions d'erreur supplémentaires de l'action `Get Interaction Status`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

#### 54.13.5 Action `Get Number of Interacted Sockets` (extraction du nombre de réceptacles en interaction)

Cette action récupère le nombre de réceptacles en interaction de la cible.

Cette action a les paramètres suivants:

- `Rt-Composite Target Param` (paramètre de rt-composite cible),
- `Interaction Type Param` (paramètre de type d'interaction): valeur du facteur GVF à affecter.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 54.13.5.1 Effet de l'action `Get Number of Interacted Sockets`

Selon le type d'interaction spécifié, le nombre de réceptacles sélectionnés ou modifiés est récupéré.

#### 54.13.5.2 Conditions d'erreur supplémentaires de l'action `Get Number of Interacted Sockets`

- Si la cible désigne plusieurs entités (utilisation d'un alias qui désigne une liste de cibles ou utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné ou utilisation de postambules d'enfants ou de descendants par exemple), l'action est ignorée pour cette cible.
- Si la cible n'est pas un rt-composant, l'action est ignorée pour cette cible.
- Si la cible n'est pas accessible, par exemple si la référence ne peut pas être résolue, l'action est ignorée pour cette cible.
- Si l'une des cibles n'est pas un rt-composite en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

## 55 Comportement de style des rt-composants

Ce comportement décrit le style d'un rt-composant. Chaque rt-composant peut avoir un style. Un style peut modifier la perception par l'utilisateur des rt-contenus.

Un style est un mode de présentation, et peut être comparé à un comportement prédéfini ou à un style de présentation. L'application d'un style à une cible évite à l'auteur de décrire ce comportement ou cette présentation. Le moteur MHEG utilise les informations décrites dans un style pour projeter directement le style dans une fonctionnalité fournie par l'interface utilisateur, si c'est possible. Si l'interface utilisateur ne fournit pas une telle fonctionnalité, le moteur MHEG est capable d'interpréter le style avec la description du comportement prédéfini fourni pour chaque style.

Exemples du comportement de style:

- un rt-non-mux contenant un entier générique peut être affiché sous forme de curseur graphique avec le style «curseur»;
- les informations audio d'un rt-non-mux peuvent être écoutées sous cette forme ou peuvent être affichées sous forme d'onde en mode graphique.

## 55.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Style` (voir 55.2).

## 55.2 `Style`

Un style est une rubrique cataloguée, enregistrée ou hors norme. Cet attribut donne un style logique au rt-composant. La façon dont le style est perçu par l'utilisateur dépend du moteur MHEG et de l'interface graphique d'utilisateur (GUI).

## 55.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Set Style` (fixation du style) (voir 55.3.1).

### 55.3.1 Action `Set Style` (fixation du style)

Cette action affecte un style à la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Component Target Param` (paramètre de rt-composant cible);
- `Catalogued Style Param` (paramètre de style catalogué);
- `Additional Information Param` (paramètre d'informations supplémentaires): peut être utilisé par un style particulier pour transférer des informations supplémentaires à associer au style. Par exemple, une plage de valeurs pour un style «curseur». Le nombre de valeurs génériques qui sont nécessaires et la sémantique de ces valeurs génériques dépendent du style.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 55.3.1.1 Effet de l'action `Set Style`

L'effet de cette action est le suivant:

- selon les valeurs spécifiées pour le paramètre de style catalogué et le paramètre d'informations supplémentaires, un style est affecté à la cible;
- le style précédemment appliqué à la cible est supprimé.

Si une action `Run` est appliquée à la cible après application de cette action, la cible est présentée avec le style actuellement spécifié. Si la cible est en cours d'exécution au moment de l'application de cette action, un effet utilisateur peut être associé à cette action.

NOTE – Le moteur MHEG peut mapper directement la cible sur une fonctionnalité de style fournie par son interface utilisateur. Toutefois, certains attributs et états peuvent être utilisés dans des objets *liens*, par exemple l'état de sélection de la cible, la position et la taille de la cible. Le moteur MHEG doit les mettre à jour en fonction du style et en cohérence avec l'interaction utilisateur courante.

#### 55.3.1.2 Conditions d'erreur supplémentaires de l'action `Set Style`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

## 55.4 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Get Style` (extraction du style) (voir 55.4.1).

### 55.4.1 Action `Get Style` (extraction du style)

Cette action récupère le style courant associé à la cible.

Cette action a le paramètre suivant:

- `Rt-Component Target Param` (paramètre de rt-composant cible).

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.



#### 55.4.1.1 Effet de l'action `Get Style`

L'effet de cette action est le suivant:

- cette action détermine une liste correspondant aux styles catalogués qui sont appliqués à la cible.

#### 55.4.1.2 Conditions d'erreur supplémentaires de l'action `Get Style`

- Si l'une des cibles n'est pas un rt-composant en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.

## 56 Comportement d'ancrage des rt-contenus

Ce comportement décrit la fonction d'ancrage d'un rt-contenu. Une ancre est un rt-contenu créé à partir d'un objet *contenu*, qui contient des informations d'ancrage.

Une ancre est caractérisée de la façon suivante:

- une ancre se distingue d'un rt-contenu normal par l'attribut facultatif de classification des données de l'objet *contenu*. Si cet attribut de classification vaut 'Anchor' (ancre), cet objet *contenu* comporte alors des informations d'ancrage, et les rt-contenus créés à partir de cet objet *contenu* peuvent être utilisés comme des ancres;
- une ancre peut servir à créer une zone sensible sur d'autres rt-composants. Les données de contenu figurant dans une ancre servent à rattacher l'ancre à d'autres rt-contenus (point de départ et point final, zone rectangulaire, zone polygonale par exemple). Ces données de contenu sont appelées informations d'ancrage. Si un rt-contenu auquel est rattachée une ancre est modifié dans sa présentation (redimensionnement ou déplacement par exemple), l'ancre doit suivre sa modification et modifier automatiquement sa position ou sa forme. La sémantique de ces informations d'ancrage n'est pas définie par la présente Recommandation. Elle dépend des rt-contenus auxquels sont rattachées des ancres, du moteur MHEG ou de l'application utilisatrice;
- le crochet dans une ancre sert à indiquer la sémantique des informations d'ancrage. Il fournit un format de codage servant à décrire la reproduction de l'ancre, les points de rattachement de l'ancre dans le rt-contenu et la sémantique de l'ancre. Le format de codage étant enregistré, l'utilisation du crochet permet d'avoir une réalisation générique du mécanisme d'ancrage;
- les informations d'ancrage sont utilisées par l'interface graphique d'utilisateur (GUI) ou un autre système prenant en charge la reproduction des rt-contenus, et cette interface ou cet autre système doivent reproduire l'ancre de façon appropriée;
- une ancre étant un rt-contenu, son état de sélection indique au moteur MHEG si la zone sensible spécifiée par cette ancre est sélectionnée par une action utilisateur. Une action `Run` ou `Stop` peut faire passer une ancre à l'état actif ou inactif. L'action `Set Interaction Ability` (fixation de la capacité d'interaction) peut aussi faire passer une ancre à l'état actif ou inactif;
- une ancre a les comportements suivants et seulement ceux-là: préparation, rt-disponibilité, exécution, interaction et ancrage des rt-contenus. Si des comportements ne devant pas être appliqués à une ancre sont appliqués, ils sont ignorés par l'ancre.

La Figure 57 montre un exemple d'utilisation d'ancres. Sur cette figure, trois ancres (rt-contenu 1, rt-contenu 2, rt-contenu 3) sont rattachées au document pivot (rt-contenu A). Selon les informations d'ancrage, chaque ancre spécifie la forme et la position de la zone sensible dans le document pivot. Un système de reproduction affiche les ancres de façon appropriée. Si l'utilisateur sélectionne l'une des ancres, les objets *liens* associés à ces ancres sont déclenchés et conduisent l'utilisateur vers une autre présentation.

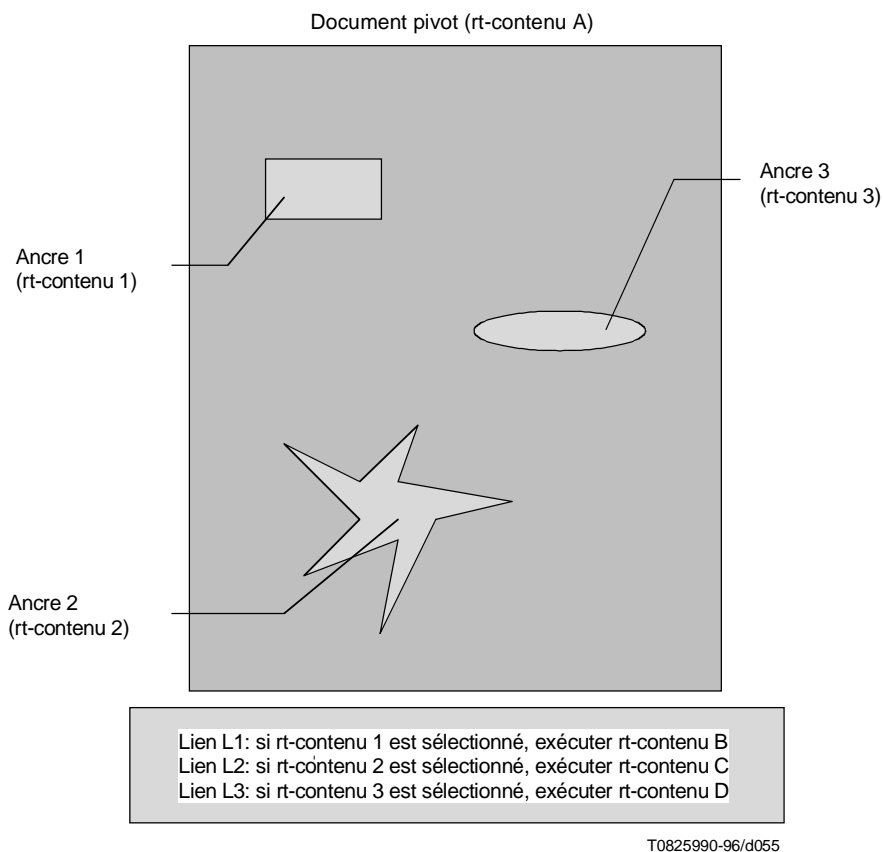
### 56.1 Attributs et états du comportement

Aucun.

### 56.2 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Attach Anchor` (rattachement d'ancre) (voir 56.2.1).



**Figure 57/T.171 – Exemple d'utilisation d'ancres**

### 56.2.1 Action Attach Anchor (rattachement d'ancre)

Cette action rattache des ancres à la cible.

Cette action a les paramètres suivants:

- ensemble de paramètres `Rt-Content Target Param` (paramètre de rt-contenu cible),
- ensemble de paramètres `Anchor Spec Param` (paramètre de spécification d'ancre): peut être utilisé par un style particulier pour transférer des informations supplémentaires à associer au style. Par exemple, une plage de valeurs pour un style «curseur». Le nombre de valeurs génériques qui sont nécessaires et la sémantique de ces valeurs génériques dépendent du style.

Cette action peut être appliquée pendant les phases R2, R3 et R3.TD.

#### 56.2.1.1 Effet de l'action Attach Anchor

L'effet de cette action est le suivant:

- selon la commande de mise à jour spécifiée, des ancres spécifiées sont ajoutées, supprimées ou remplacées.

#### 56.2.1.2 Conditions d'erreur supplémentaires de l'action Attach Anchor

- Si l'une des cibles n'est pas un rt-contenu en phase R2, R3 ou R3.TD, l'action est ignorée pour cette cible.
- Si l'une des ancres n'est pas une ancre [c'est-à-dire si elle ne contient pas d'attribut de classification cataloguée valant 'Anchor' (ancre)], l'action est ignorée pour cette ancre.

## SECTION 7 – COMPORTEMENT DES CANAUX

### 57 Comportement de disponibilité d'un canal

Ce comportement décrit la disponibilité des canaux. Chaque canal possède ce comportement.

#### 57.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut et de l'état suivant:

- Channel Availability Status (état de disponibilité d'un canal) (voir 57.2).

#### 57.2 Channel Availability Status (état de disponibilité d'un canal)

La disponibilité de chaque canal pour le moteur MHEG peut être déterminée. L'état de disponibilité d'un canal indique si un canal est disponible. Cet état prend une des deux valeurs suivantes: «disponible» ou 'not available' (non disponible).

L'état de disponibilité d'un canal doit être fixé à la valeur 'not available' (non disponible) si le canal ne peut être activé par le moteur MHEG, c'est-à-dire s'il est en phase C1 (voir 29.3, «disponibilité de canal»).

NOTE 1 – 'cannot' (ne peut pas) signifie que le canal n'a pas été formellement créé ou qu'il a été supprimé. Cela ne signifie pas que l'équipement requis pour exécuter le traitement n'est pas opérationnel.

Par exemple, le canal n'a pas encore été créé par le moteur MHEG, c'est-à-dire une action New (nouveau) n'a pas encore été appliquée.

L'état de disponibilité d'un canal doit être fixé à la valeur «disponible» si le canal est activé par le moteur MHEG, c'est-à-dire le canal a été créé par le moteur MHEG et est en phase C2.

NOTE 2 – Le moteur MHEG peut offrir à l'application utilisatrice des messages d'information et des diagnostics plus détaillés, par exemple, pour expliquer qu'un canal ne peut être créé, ou pour indiquer les conditions d'erreurs. Ces messages ne sont pas définis par la présente Recommandation, mais peuvent être définis par d'autres normes.

Le moteur MHEG suppose que tous les canaux sont initialisés à la valeur 'not available' (non disponible).

#### 57.3 Actions de modification du comportement

La présente Recommandation permet de modifier ce comportement par les actions suivantes:

- action New Channel (canal nouveau) (voir 57.3.1);
- action Delete Channel (suppression du canal) (voir 57.3.2).

##### 57.3.1 Action New Channel (canal nouveau)

Cette action crée des canaux. Elle sert à obtenir un espace de perception logique à travers lequel on perçoit les rt-composants affectés.

Le processus de création de canal, qui sort du cadre de la présente Recommandation, est fourni par le moteur MHEG.

Cette action possède le paramètre suivant:

- ensemble de paramètres Channel Target Param (paramètre de canal cible).

Cette action peut être appliquée pendant la phase C1.

##### 57.3.1.1 Effet de l'action New Channel

L'effet MHEG de cette action est le suivant:

- 1) créer le canal cible (voir 9.3);
- 2) affecter un identificateur de canal (cible de cette action) au canal créé;
- 3) initialiser le comportement de perceptibilité du canal à 'off' (arrêt);
- 4) positionner l'état de disponibilité du canal sur «disponible». Le canal créé passe en phase C2.

### 57.3.1.2 Conditions d'erreur supplémentaires de l'action `New Channel`

- Si l'une des cibles n'est pas un canal en phase C1, l'action est ignorée pour cette cible.
- Si un identificateur de canal est utilisé dans l'une des cibles, l'action est ignorée pour cette cible.
- Si l'une des cibles est le canal par défaut, l'action est ignorée pour cette cible.

### 57.3.2 Action `Delete` (suppression)

Cette action supprime les canaux du moteur MHEG. Elle sert à libérer des ressources dans le moteur MHEG.

Elle possède le paramètre suivant:

- ensemble de paramètres `Channel Target Param` (paramètre de canal cible).

Cette action s'applique en phase C2.

#### 57.3.2.1 Effet de l'action `Delete`

L'effet de cette action est le suivant:

- 1) détruire le comportement de perceptibilité du canal;  
le concepteur de l'objet doit savoir que si le canal était à l'état 'on' (en marche), le canal et tous ses rt-composants associés disparaîtront. Comment traiter cette disparition n'est pas défini par la présente Recommandation, ceci est laissé au moteur MHEG;
- 2) appliquer les actions `Implicit Stop` (arrêt implicite) à tous les rt-composants affectés aux cibles;
- 3) annuler l'affectation des rt-composants affectés à la cible et réaffecter implicitement ces rt-composants par une action `Set Channel Assignment` (affectation d'un canal) adressés au canal par défaut;
- 4) NOTE 1 – Le comportement de ces rt-composants reste identique après leur désaffectation.
- 5) NOTE 2 – Destruction de la cible et de son espace de présentation CPS.  
Positionner l'état de disponibilité du canal sur 'not available' (non disponible). Le canal passe en phase C1.

#### 57.3.2.2 Conditions d'erreur supplémentaires de l'action `Delete`

- Si l'une des cibles n'est pas un canal en phase C2, l'action est ignorée pour cette cible.
- Si l'une des cibles est le canal par défaut, l'action est ignorée pour cette cible.

## 57.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Channel Availability Status` (extraction de l'état de disponibilité du canal) (voir 57.4.1).

### 57.4.1 Action `Get Channel Availability Status` (extraction de l'état de disponibilité du canal)

Cette action récupère l'état de disponibilité du canal cible.

Elle possède le paramètre suivant:

- `Channel Target Param` (paramètre de canal cible).

Cette action s'applique en phases C1 et C2.

#### 57.4.1.1 Effet de l'action `Get Channel Availability Status`

L'effet de cette action est le suivant:

- 1) si la cible est en phase C1, cette action donne la valeur 'not available' (non disponible);
- 2) sinon (la cible est en phase C2), cette action donne la valeur «disponible».

#### 57.4.1.2 Conditions d'erreur supplémentaires de l'action `Get Channel Availability Status`

- L'action est appliquée à de multiples entités – utilisation d'un alias qui désigne par son adresse une liste de cibles, utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné, utilisation du postambule enfants ou descendants,...-: l'action est ignorée pour cette cible.
- La cible n'est pas un canal: l'action est ignorée pour cette cible.
- La cible n'est pas accessible – la référence n'est pas résolue,...-: l'action est ignorée pour cette cible.

## 58 Comportement de perceptibilité du canal

Ce comportement décrit la perceptibilité des canaux. Tous les canaux possèdent ce comportement.

### 58.1 Attributs et états du comportement

Ce comportement est exprimé au moyen de l'attribut ou état suivant:

- `Channel Perceptability` (perceptibilité du canal) (voir 58.2).

### 58.2 `Channel Perceptability` (perceptibilité du canal)

Chaque canal peut être en marche ou en arrêt. Si un canal est en marche, tous les rt-composants affectés à ce canal sont perçus par l'utilisateur. Si un canal est en arrêt, tous les rt-composants affectés à ce canal ne sont plus perceptibles.

NOTE – Les rt-composants peuvent être affectés à un canal non perceptible et y être positionnés. Si le canal est en marche, tous les rt-composants qui lui sont affectés apparaissent selon les spécifications antérieures.

La perceptibilité du canal peut prendre une des deux valeurs suivantes: 'on' (en marche), 'off' (arrêt).

Le canal par défaut est en marche et sa perceptibilité est initialisée à la valeur 'on' (en marche). Les autres canaux sont donc à l'arrêt et leur perceptibilité du canal à la valeur 'off' (arrêt).

### 58.3 Action de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Set Channel Perceptability` (positionner perceptibilité de canal) (voir 58.3.1).

#### 58.3.1 Action `Set Channel Perceptability` (positionner perceptibilité de canal)

Cette action change la perceptibilité du canal cible.

Elle possède les paramètres suivants:

- ensemble de paramètres `Channel Target Param` (paramètres de canal cible);
- `Channel Perceptability Param` (paramètre de perceptibilité de canal).

Cette action s'applique en phase C2.

##### 58.3.1.1 Effet de l'action `Set Channel Perceptability`

L'effet de cette action est le suivant:

- positionnement de l'attribut de perceptibilité du canal cible selon les indications suivantes:
  - 1) si le paramètre cible a la valeur 'off' (arrêt), le canal cible est mis à l'arrêt;
  - 2) sinon [le paramètre cible a la valeur 'on' (en marche)], le canal cible est mis en marche.

Il n'y a pas d'effet si la valeur spécifiée est identique à la valeur courante.

##### 58.3.1.2 Conditions d'erreur supplémentaires de l'action `Set Channel Perceptability`

- Si l'une des cibles n'est pas un canal en phase C2, l'action est ignorée pour cette cible.

## 58.4 Actions de récupération du comportement

La présente Recommandation permet de consulter ce comportement par l'action suivante:

- action `Get Channel Perceptability` (extraction de la perceptibilité du canal) (voir 58.4.1).

### 58.4.1 Action `Get Channel Perceptability` (extraction de la perceptibilité du canal)

Cette action récupère la perceptibilité du canal cible.

Elle possède le paramètre suivant:

- `Channel Target Param` (paramètre de canal cible).

Cette action s'applique en phase C2.

#### 58.4.1.1 Effet de l'action `Get Channel Perceptability`

Cette action évalue la valeur courante de l'attribut de perceptibilité du canal et renvoie l'état 'on' (en marche) ou 'off' (arrêt), selon les cas.

#### 58.4.1.2 Conditions d'erreur supplémentaires de l'action `Get Channel Perceptability`

- L'action est appliquée à de multiples entités – utilisation d'un alias qui désigne par son adresse une liste de cibles, utilisation du caractère générique '\*' au lieu d'un numéro de rt-composant donné, utilisation du postambule enfants ou descendants,...– : l'action est ignorée pour cette cible.
- L'une des cibles n'est pas un canal en phase C2: l'action est ignorée pour cette cible.
- L'une des cible n'est pas accessible – la référence n'est pas résolue,...–: l'action est ignorée pour cette cible.

## 59 Comportement de l'espace de présentation d'un canal

Ce comportement décrit l'espace de présentation d'un canal. Lors de la création d'un canal, son espace de présentation est initialisé aux valeurs par défaut. Ces valeurs peuvent être modifiées par la suite.

### 59.1 Attributs et états du comportement

Aucun.

### 59.2 Action de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- action `Set CPS` (positionnement de l'attribut CPS) (voir 59.2.1).

#### 59.2.1 Action `Set CPS` (positionnement de l'attribut CPS)

Cette action spécifie l'espace CPS d'un canal cible.

Elle possède les paramètres suivants:

- ensemble des paramètres `Channel Target Param` (paramètre de canal cible);
- `CPS Initialisation Param` (paramètre d'initialisation de l'attribut CPS).

Cette action s'applique en phase C2.

#### 59.2.1.1 Effet de l'action `Set CPS`

L'effet de cette action est le suivant:

- 1) fixer l'extrémité de l'axe temporel de l'espace CPS cible à la durée spécifiée. Par défaut, cette extrémité est fixée à l'infini;
- 2) fixer l'extrémité de chaque axe spatial de l'espace CPS cible à la taille spécifiée. Par défaut, l'extrémité d'un axe est fixée à 0.

Il peut y avoir un effet utilisateur. Si les axes spatiaux de l'espace CPS sont tronqués, certains des rt-composants projetés sur cet espace deviendront invisibles.

### 59.2.1.2 Conditions d'erreur supplémentaires de l'action `Set CPS`

- Si l'une des cibles n'est pas un canal en phase C2, l'action est ignorée pour cette cible.

## SECTION 8 – COMPORTEMENT DES CANAUX ET RT-COMPOSANTS

### 60 Comportement des événements des canaux et rt-composants

Ce comportement décrit les événements qui peuvent survenir sur les canaux et les rt-composants.

Le comportement permet au moteur MHEG d'extraire les événements qui peuvent survenir pas seulement du moteur MHEG mais aussi de certains systèmes ou de composants de systèmes qui lui sont extérieurs.

#### 60.1 Attributs et états du comportement

Ce comportement est exprimé au moyen des attributs ou états suivants:

- `Event` (événement) (voir 60.2);
- `Event Data` (donnée d'événement) (voir 60.3).

#### 60.2 `Event` (événement)

Un événement est identifié par une entrée cataloguée qui est une liste. L'identificateur d'événement 0 est réservé pour représenter l'état où aucun événement n'est survenu.

L'attribut d'événement contient l'identificateur du dernier événement survenu sur un canal ou un rt-composant. Chaque attribut d'événement est initialisé à 0. Ce comportement peut être modifié par la survenue d'un événement.

#### 60.3 `Event Data` (donnée d'événement)

Un attribut de donnée d'événement stocke l'information, associée au dernier événement reçu par un canal ou un rt-composant, si l'événement demande de stocker des informations supplémentaires. La valeur stockée dans un attribut de donnée d'événement est une valeur générique. Sa structure validée et ses valeurs permises sont cataloguées en association avec un événement.

La valeur d'un attribut de donnée d'événement est initialisée à une valeur particulière dépendant du codage de l'objet MHEG. Dans la présente Recommandation, il s'agit de la valeur `NULL` (néant) en ASN.1.

#### 60.4 Action de modification du comportement

La présente Recommandation permet de modifier ce comportement par l'action suivante:

- `action Set Event` (positionnement d'événement) (voir 60.4.1).

##### 60.4.1 `Action Set Event` (positionnement d'événement)

A l'initiative de l'auteur, cette action envoie un événement à la cible. Elle peut être implicitement appliquée si le moteur MHEG ou certains systèmes génèrent un événement et si ce dernier est catalogué.

Elle possède les paramètres suivants:

- ensemble de paramètres `Rt-Component Channel Tg Param` (paramètre de canal cible de rt-composant);
- `Event Param` (paramètre d'événement);
- `Event Data Param` (paramètre de donnée d'événement): spécifie les données associées à un événement.

Cette action s'applique en phases R2, R3, R3.TD et C2.

#### 60.4.1.1 Effet de l'action `Set Event` (positionnement d'événement)

L'effet de cette action est le suivant:

- 1) positionner les attributs d'événement de la cible selon les spécifications;
- 2) si des données d'événement sont associées à l'événement, les inscrire dans l'attribut donnée d'événement;
- 3) sinon (pas de donnée d'événement) affecter la valeur NULL (néant) à l'attribut donnée d'événement.

Il appartient à l'auteur de veiller à la cohérence de l'attribut d'événement et de l'attribut donnée d'événement, si cette action est explicitement utilisée.

Il appartient au moteur MHEG ou à l'application utilisatrice d'assurer cette cohérence si l'action est implicitement utilisée.

#### 60.4.1.2 Conditions d'erreur supplémentaires de l'action `Set Event`

- Si l'une des cibles n'est pas un rt-composant ou un canal en phases R2, R3, R3.TD et C2, l'action est ignorée pour cette cible.

### 60.5 Action de récupération du comportement

La présente Recommandation permet de consulter le comportement par les actions suivantes:

- action `Get Event` (extraction d'événement) (voir 60.5.1);
- action `Get Event Data` (extraction de donnée d'événement) (voir 60.5.2).

#### 60.5.1 Action `Get Event` (extraction d'événement)

Cette action récupère le dernier événement survenu à la cible.

Elle possède le paramètre suivant:

- `Rt-Component Channel Tg Param` (paramètre de canal cible de rt-composant): spécifie des données associées avec un événement.

Cette action s'applique en phases R2, R3, R3.TD et C2.

#### 60.5.1.1 Effet de l'action `Get Event`

L'effet de cette action est le suivant:

- évaluation de l'attribut d'événement et renvoi de la valeur stockée.

#### 60.5.1.2 Conditions d'erreur supplémentaires de l'action `Get Event`

- Si l'une des cibles n'est pas un rt-composant ou un canal en phases R2, R3, R3.TD et C2, l'action est ignorée par cette cible.

#### 60.5.2 Action `Get Event Data` (extraction de donnée d'événement)

Cette action récupère les données d'événement de la cible qui sont associées au dernier événement survenu.

Elle possède le paramètre suivant:

- `Rt-Component Channel Tg Param` (paramètre de canal cible de rt-composant): spécifie les données associées à un événement.

Cette action s'applique en phases R2, R3, R3.TD et C2.

#### 60.5.2.1 Effet de l'action `Get Event Data`

L'effet de cette action est le suivant:

- évaluation de l'attribut donnée d'événement et renvoi de la valeur stockée.

#### 60.5.2.2 Conditions d'erreur supplémentaires de l'action `Get Event Data`

- Si l'une des cibles n'est pas un rt-composant ou un canal en phases R2, R3, R3.TD et C2, l'action est ignorée par la cible.



**61 Attributs de représentation de la classe *objet MH*****Tableau 4/T.171 – Aperçu général des attributs de représentation de la classe *objet MH***

<p><b>MH-object Class</b> ::= Class Identification, Mheg ID?, Description?</p> <p><b>Class Identification</b> ::= #<i>joint-iso-itu-t</i>, #<i>mheg</i>, #<i>version</i>, Class ID</p> <p><b>Class ID</b> ::=  #<i>action-class-ID</i> / #<i>link-class-ID</i> / #<i>script-class-ID</i> / #<i>content-class-ID</i> / #<i>multiplexed-content-class-ID</i> / #<i>composite-class-ID</i> /  #<i>container-class-ID</i> / #<i>descriptor-class-ID</i> / Extensibility Provision</p> <p><b>Description</b> ::=  Name?, Owner?, Version?, Date?, Keywords?, Copyright?, Copyright ID?, Copyright Number?, Licence?, Cache Priority?,  Comments?, Extensibility Provision</p> <p><b>Name</b> ::= String</p> <p><b>Owner</b> ::= String</p> <p><b>Version</b> ::= String</p> <p><b>Date</b> ::= <i>\$UTCTime</i></p> <p><b>Keywords</b> ::= String*</p> <p><b>Copyright</b> ::= String</p> <p><b>Copyright ID</b> ::= <i>\$OCTET STRING</i></p> <p><b>Copyright Number</b> ::= <i>\$OCTET STRING</i></p> <p><b>Licence</b> ::= String</p> <p><b>Cache Priority</b> ::= Integer</p> <p><b>Comments</b> ::= String</p>
--

**61.1 MH-object Class (Classe *objet MH*)**

- La classe *objet MH* fournit la structure pour l'identification et le transfert de tous les objets MHEG. Elle fournit également le mécanisme d'adressage générique. La classe *objet MH* est une classe abstraite qui permet de procéder d'une manière cohérente et précise à l'identification et au transfert de tous les objets MHEG.

Hérite de: NÉANT.

Pointe vers: NÉANT.

Héritiers: classes *action*, *lien*, *modèle*, *conteneur* et *descripteur*.

Pointé par: NÉANT.

- Séquence de (Class Identification, Mheg ID?, Description?).

**61.2 Class Identification (Identification de classe)**

- Identification de la classe par:  
Norme MHEG: *joint-iso-itu* (2) *mheg* (19).  
Version de la norme MHEG: (1).  
Identificateur de classe correspondant à l'objet transféré.
- Séquence de (*joint-iso-itu*, *mheg*, *version*, Class ID).

### 61.3 Class ID (**Identificateur de classe**)

- Identificateur de la classe *objet MHEG* de l'objet transféré.
- Choix entre (action-class-ID, link-class-ID, script-class-ID, content-class-ID, multiplexed-content-class-ID, composite-class-ID, container-class-ID, descriptor-class-ID, Extensibility Provision).

### 61.4 Description (**Description**)

- Description plus précise d'un objet MHEG transféré.
- Séquence de (Name?, Owner?, Version?, Date?, Keywords?, Copyright?, Copyright ID?, Copyright Number?, Licence?, Cache Priority?, Comments?, Extensibility Provision).

### 61.5 Name (**Nom**)

- Nom de l'objet MHEG.
- String.

### 61.6 Owner (**Propriétaire**)

- Identification du propriétaire de l'objet MHEG.
- String.

### 61.7 Version (**Version**)

- Identification de la version de l'objet MHEG.
- String.

### 61.8 Date (**Date**)

- Dernière date de modification de l'objet MHEG. Dans la notation de base ASN.1, le temps universel (UTC), format ISO, est utilisé.
- UTC Time.

### 61.9 Keywords (**Mots clés**)

- Liste des mots clés qualifiant l'objet MHEG.
- Liste de (String).

### 61.10 Copyright (**Droit d'auteur**)

- Contenu du droit d'auteur lié à l'objet MHEG.
- String.

### 61.11 Copyright ID (**Identificateur de droit d'auteur**)

- Identificateur du code du type de travail.
- OCTET STRING.

### 61.12 Copyright Number (**Numéro du droit d'auteur**)

- Code d'identification unique de chaque travail.
- OCTET STRING.

### 61.13 Licence (**Licence**)

- Informations concernant la licence liée à l'objet MHEG.
- String.

### 61.14 Cache Priority (**Priorité d'antémémoire**)

- Fournit un moyen de gérer la mémoire et est utilisée par le moteur MHEG pour optimiser les actions Prepare (Préparer) et Destroy (Détruire). La valeur devrait être comprise dans l'intervalle (0, 255).
- Integer.

### 61.15 Comments (**Observations**)

- Observations en format libre liées à l'objet MHEG.
- String.

## 62 Attributs de représentation de la classe *action*

Tableau 5/T.171 – Aperçu général des attributs de représentation de la classe *action*

<p><b>Action Class</b> ::= MH-object Class, Synchro Indicator Param, Synchronised Action+, Extensibility Provision</p> <p><b>Synchro Indicator Param</b> ::= Synchro Indicator   Synchro Indicator Macro</p> <p><b>Synchro Indicator Macro</b> ::= Macro Def ID, Synchro Indicator?</p> <p><b>Synchro Indicator</b> ::= #serial   #parallel</p> <p><b>Synchronised Action</b> ::= Elementary Action   Action Object</p> <p><b>Action Object</b> ::= Action Object Reference   Action Class</p>
--

### 62.1 Action Class (**Classe *action***)

- La classe *action* fournit une structure cohérente pour le transfert d'ensembles d'actions. Elle fournit également une structure cohérente pour le transfert d'objets *action* de base et d'objets *action* emboîtés. Des objets *action*, des instances de la classe *action*, sont utilisés pour transférer un ensemble d'actions élémentaires organisé.

Hérite de: classe *objet MH*.

Pointe vers: classe *action*.

Héritiers: NÉANT.

Pointé par: classes *lien*, *composite* et *conteneur*.

- Séquence de (MH-object Class, Synchro Indicator Param, Synchronised Action+, Extensibility Provision).

### 62.2 Synchro Indicator Param (**Paramètre d'indicateur de synchronisation**)

- Paramètre d'indicateur de synchronisation.
- Choix entre (Synchro Indicator, Synchro Indicator Macro).

### 62.3 Synchro Indicator Macro (**Macro-paramètre d'indicateur de synchronisation**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'indicateur de synchronisation.
- Séquence de (Macro Def ID, Synchro Indicator?).

## 62.4 Synchro Indicator (**Indicateur de synchronisation**)

- Ce paramètre spécifie si le groupe d'actions synchronisées suivant doit être exécuté en parallèle ou en série.
- Choix entre (`serial`, `parallel`).

## 62.5 Synchronised Action (**Action synchronisée**)

- Définit une structure récurrente pour exprimer une combinaison d'actions. Chaque action synchronisée dans ce groupe est soit une action élémentaire simple soit un autre objet *action* emboîté.
- Choix entre (`Elementary Action`, `Action Object`).

## 62.6 Action Object (**Objet action**)

- L'objet *action* peut être référencé ou inclus dans l'objet.
- Choix entre (`Action Object Reference`, `Action Class`).

## 63 Attributs de représentation de la classe *lien*

Tableau 6/T.171 – Aperçu général des attributs de représentation de la classe *lien*

<b>Link Class</b> ::= MH-object Class, Link Condition, Link Effect, Extensibility Provision
<b>Link Condition</b> ::= Trigger Condition   Logical Combination
<b>Trigger Condition</b> ::= Source Value, Previous Condition?, Current Condition
<b>Constraint Condition</b> ::= Source Value, Current Condition
<b>Source Value</b> ::= Evaluated Value
<b>Comparison Operation</b> ::= Comparison Operator, Comparison Value
<b>Comparison Value</b> ::= Generic Value   Comparison Value Constant   <i>#unspecified</i>
<b>Previous Condition</b> ::= Comparison Operation
<b>Current Condition</b> ::= Comparison Operation
<b>Comparison Operator</b> ::= <i>#equal</i>   <i>#not-equal</i>   <i>#greater</i>   <i>#greater-equal</i>   <i>#less</i>   <i>#less-equal</i>
<b>Logical Combination</b> ::= Logical Operator, Condition+
<b>Logical Operator</b> ::= <i>#and</i>   <i>#or</i>   <i>#xor</i>   <i>#not</i>
<b>Condition</b> ::= Trigger Condition   Constraint Condition   Logical Combination
<b>Link Effect</b> ::= Macro Parameter Resolution*, Action Object
<b>Macro Parameter Resolution</b> ::= Macro Def ID, Usage Value
<b>Usage Value</b> ::= Generic Value

### 63.1 Link Class (**Classe *lien***)

- La classe *lien* fournit une structure cohérente pour le transfert d'actions conditionnelles ayant pour cible des objets, des objets d'exécution ou des canaux MHEG. Des objets *lien* MHEG, des instances de la classe *lien*, peuvent être échangés. Un objet *lien* est directif et il relie une ou plusieurs sources à une ou plusieurs cibles. Les actions définies dans le lien sont traitées sur les cibles indiquées, uniquement si les conditions sont satisfaites.

Hérite de: classe *objet MH*.

Pointe vers: classe *action*.

Héritiers: NÉANT.

Pointé par: classes *composite* et *conteneur*.

- Séquence de (MH-object Class, Link Condition, Link Effect, Extensibility Provision).

### 63.2 Link Condition (**Condition de lien**)

- Définit la condition de lien. L'effet de lien doit être exécuté chaque fois que la condition de lien est évaluée comme étant vraie.
- Choix entre (Trigger Condition, Logical Combination).

### 63.3 Trigger Condition (**Condition de déclenchement**)

- Définit une condition de déclenchement décrite par le changement de la valeur d'un attribut ou d'un état.
- Séquence de (Source Value, Previous Condition?, Current Condition).

### 63.4 Constraint Condition (**Contrainte**)

- Définit une contrainte testant l'état courant d'une valeur.
- Séquence de (Source Value, Current Condition).

### 63.5 Source Value (**Valeur source**)

- Identifie l'état ou l'attribut d'un objet à comparer dans une condition précédente ou courante. La valeur évaluée est le résultat d'une action d'extraction ayant pour cible un objet unique. Autrement, la valeur source est «pas définie».
- Evaluated Value.

### 63.6 Comparison Operation (**Opération de comparaison**)

- Décrit une opération de comparaison à exécuter sur une valeur source afin de remplir une condition.
- Séquence de (Comparison Operator, Comparison Value).

### 63.7 Comparison Value (**Valeur de comparaison**)

- Spécification de la valeur à comparer à la valeur source en utilisant un opérateur de comparaison. Il appartient au concepteur de l'objet de faire en sorte que le type de la valeur de comparaison courante soit compatible avec la valeur source.
- Choix entre (Generic Value, Comparison Value Constant, unspecified).

### 63.8 Previous Condition (**Condition précédente**)

- Définit la condition précédente appliquée à la valeur source.  
Si la condition précédente est omise, cela signifie qu'elle est évaluée comme: NON condition courante.
- Comparison Operation.

### 63.9 Current Condition (**Condition courante**)

- Définit la condition courante de la valeur source.
- Comparison Operation.

### 63.10 Comparison Operator (**Opérateur de comparaison**)

- Spécifie l'opération à effectuer entre la valeur source et la valeur de comparaison.
- Choix entre (equal, not-equal, greater, greater-equal, less, less-equal).

### 63.11 Logical Combination (**Combinaison logique**)

- Définit une structure arborescente pour exprimer une combinaison de conditions. L'opérateur logique constitue un nœud de l'arbre de conditions logique. L'ensemble de conditions constitue les feuilles du nœud. Lorsque l'opérateur logique est NON, une seule condition est autorisée. Tout autre opérateur a besoin d'au moins deux conditions.
- Séquence de (Logical Operator, Condition+).

### 63.12 Logical Operator (**Opérateur logique**)

- Définit l'opération logique à appliquer entre les conditions du nœud.
- Choix entre (and, or, xor, not).

### 63.13 Condition (**Condition**)

- Définit les conditions sur lesquelles l'opération logique doit être appliquée.
- Choix entre (Trigger Condition, Constraint Condition, Logical Combination).

### 63.14 Link Effect (**Effet de lien**)

- Définit la macro-résolution fournissant une valeur d'utilisation à chaque macro-paramètre, ainsi que l'objet *action* à exécuter lorsque le résultat du lien est déclenché.
- Séquence de (Macro Parameter Resolution\*, Action Object).

### 63.15 Macro Parameter Resolution (**Macro-résolution de paramètre**)

- Associe une valeur d'utilisation à un identificateur de macro-définition.
- Séquence de (Macro Def ID, Usage Value).

### 63.16 Usage Value (**Valeur d'utilisation**)

- Affecte une valeur d'utilisation à un identificateur de macro-définition. Il appartient au concepteur de l'objet de faire en sorte que la valeur d'utilisation soit compatible avec le paramètre spécifié au moyen d'un identificateur de macro-définition.
- Generic Value.

## 64 Attributs de représentation de la classe modèle

Tableau 7/T.171 – Aperçu général des attributs de représentation de la classe modèle

Model Class ::= MH-object Class
---------------------------------

### 64.1 Model Class (**Classe modèle**)

- La classe modèle est une classe abstraite. La classe *script* et la classe *composant* héritent cette classe.  
Hérite de: classe *objet MH*.  
Pointe vers: NÉANT.  
Héritiers: classe *script*, classe *composant*.  
Pointé par: NÉANT.
- MH-object Class.

## 65 Attributs de représentation de la classe *script*

Tableau 8/T.171 – Aperçu général des attributs de représentation de la classe *script*

<p><b>Script Class</b> ::= Model Class, Script Classification?, Script Hook, Script Data, Extensibility Provision</p> <p><b>Script Classification</b> ::= Catalogued Script Classification</p> <p><b>Script Data</b> ::= Script Inclusion   Data Reference</p> <p><b>Script Inclusion</b> ::= <i>§BIT STRING</i> / <i>§OCTET STRING</i> / InterchangedScript</p> <p><b>InterchangedScript</b> ::= <i>#Imported from ITU-T Rec. T.173</i></p>
--

### 65.1 Script Class (**Classe script**)

- La classe *script* fournit une structure cohérente pour le transfert d'actions complexes appliquées à des entités MHEG. Des objets *script* MHEG, des instances de la classe *script*, peuvent être transférés. Un objet *script* contient une indication du langage de script utilisé ainsi que le script codé lui-même. On part du principe que le langage de script utilisé dans un objet *script* est capable de faire référence à des entités MHEG et d'accéder à leurs attributs.

Hérite de: classe modèle.

Pointe vers: NÉANT.

Héritiers: NÉANT.

Pointé par: classe *conteneur*.

- Séquence de (Model Class, Script Classification?, Script Hook, Script Data, Extensibility Provision).

### 65.2 Script Classification (**Classification de scripts**)

- Une des classifications de scripts cataloguées; elle peut être enregistrée ou privée.
- Catalogued Script Classification.

### 65.3 Script Data (**Données de script**)

- Script codé au moyen du langage de script identifié par le crochet de script. Ces données de script peuvent être incluses dans l'objet *script* lui-même ou on peut y faire référence en utilisant le mécanisme de l'identificateur externe ou alias. On peut également utiliser des données script vides.
- Choix entre (Script Inclusion, Data Reference).

### 65.4 Script Inclusion (**Inclusion de script**)

- Les données de script sont incluses dans l'objet *script* transféré.
- Choix entre (BIT STRING, OCTET STRING, InterchangedScript).

### 65.5 Interchanged Script (**Script transféré**)

- La syntaxe du script transféré figure dans la Recommandation T.173.
- Importé de la représentation d'échange de script de la Recommandation T.173.

## 66 Attributs de représentation de la classe *composant*

Tableau 9/T.171 – Aperçu général des attributs de représentation de la classe *composant*

<p><b>Component Class</b> ::= Model Class, OPS Initialisation?</p>
--

## 66.1 Component Class (**Classe composant**)

- La classe *composant* est une classe abstraite. La classe *contenu* et la classe *composite* héritent cette classe.  
Hérite de: classe modèle.  
Pointe vers: NÉANT.  
Héritiers: classes *contenu* et *composite*.  
Pointé par: NÉANT.
- Séquence de (Model Class, OPS Initialisation?).

## 67 Attributs de représentation de la classe *contenu*

Tableau 10/T.171 – Aperçu général des attributs de représentation de la classe *contenu*

<p><b>Content Class</b> ::= Component Class, Cat Content Classification?, Content Hook, OV?, Content Data, Extensibility Provision</p> <p><b>Content Data</b> ::= Data Inclusion   Data Reference</p> <p><b>Data Inclusion</b> ::= \$BIT STRING / \$OCTET STRING / Generic Value</p>
--

### 67.1 Content Class (**Classe contenu**)

- La classe *contenu* fournit une structure cohérente pour le transfert de données codées. Des objets *contenu* MHEG, des instances de la classe *contenu*, peuvent être transférés. Un objet *contenu* assure l'identification du type et le transfert des données.  
Hérite de: classe *composant*.  
Pointe vers: NÉANT.  
Héritiers: classe *contenu multiplexé*.  
Pointé par: classes *composite* et *conteneur*.
- Séquence de (Component Class, Cat Content Classification?, Content Hook, OV?, Content Data, Extensibility Provision).

### 67.2 Content Data (**Données de contenu**)

- Données codées fournies par d'autres Recommandations et normes ou dans un format privé. Ces données sont incluses dans l'objet *contenu* lui-même ou il y est fait référence en utilisant le mécanisme de l'alias ou de l'identificateur extérieur offert. Une donnée vide peut également être spécifiée.  
Les données de contenu ne sont pas limitées aux schémas de codage prescrits par les normes internationales. La référence aux données de contenu dans la présente Recommandation propose un mécanisme de référence aux données de contenu qui est conforme à toutes les normes, y compris aux normes privées.
- Choix entre (Data Inclusion, Data Reference).

### 67.3 Data Inclusion (**Inclusion de données**)

- Les données sont incluses dans l'objet *contenu* transféré.
- Choix entre (BIT STRING, OCTET STRING, Generic Value).

## 68 Attributs de représentation de la classe *contenu multiplexé*

Tableau 11/T.171 – Aperçu général des attributs de représentation de la classe *contenu multiplexé*

<p><b>Multiplexed Content Class</b> ::= Content Class, Multiplexed Stream+, Extensibility Provision</p> <p><b>Multiplexed Stream</b> ::= Stream ID, Cat Content Classification?, Content Hook?</p>
--



## 68.1 Multiplexed Content Class (**Classe contenu multiplexé**)

- La classe *contenu multiplexé* fournit une structure cohérente pour l'échange de données monomédias multiplexées. Il s'agit d'une classe modèle qui est une sous-classe de la classe *contenu*. Elle contient la représentation codée de données monomédias multiplexées, ainsi qu'une description de chaque flux multiplexé, ou y fait référence. Une classe *contenu multiplexé* représente l'association entre une classe *contenu* et une liste ordonnée de flux décrivant les flux contenus dans les données multiplexées.

Hérite de: classe *contenu*.

Pointe vers: NÉANT.

Héritiers: NÉANT.

Pointé par: classes *composite* et *conteneur*.

- Séquence de (Content Class, Multiplexed Stream+, Extensibility Provision).

## 68.2 Multiplexed Stream (**Flux multiplexé**)

- Définit un flux de données multiplexées et donne des informations précises.
- Séquence de (Stream ID, Cat Content Classification?, Content Hook?).

## 69 Attributs de représentation de la classe *composite*

Tableau 12/T.171 – Aperçu général des attributs de représentation de la classe *composite*

<p><b>Composite Class ::=</b> Component Class, Availability Start-up?, Availability Close-down?, Rt-Availability Start-up?, Rt-Availability Close-down?, Action Object*, Link Object*, Nb Of Elements, Composition Element*, Extensibility Provision</p> <p><b>Availability Start-up ::=</b> Link Effect   #automatic-start-up-1   #automatic-start-up-2   #automatic start-up-3   #automatic-start-up-4   #automatic-start-up-5</p> <p><b>Availability Close-down ::=</b> Link Effect   #automatic-close-down-1   #automatic-close-down-2   #automatic-close-down-3</p> <p><b>Rt-Availability Start-up ::=</b> Link Effect   #automatic-rt-start-up</p> <p><b>Rt-Availability Close-down ::=</b> Link Effect</p> <p><b>Link Object ::=</b> Link Class   Link Object Reference</p> <p><b>Nb Of Elements ::=</b> Integer</p> <p><b>Composition Element ::=</b> Index, Associated Model</p> <p><b>Element Index ::=</b> Index</p> <p><b>Associated Model ::=</b> Component Object Reference   Content Class   Multiplexed Content Class   Composite Class   Label</p> <p><b>Label ::=</b> String</p>
--

### 69.1 Composite Class (**Classe composite**)

- La classe *composite* permet de synchroniser dans le temps et dans l'espace, de lier et d'encapsuler un ensemble d'objets, et de transférer ces derniers, d'une manière cohérente. Des objets *composites* MHEG, des instances de la classe *composite*, peuvent être transférés. Un objet *composite* pointe vers des actions et des liens pour décrire les comportements initial et dynamique des objets composants et des rt-objets créés à partir de ces objets composites.

Hérite de: classe *composant*.

Pointe vers: classes *lien*, *action*, *contenu*, *contenu multiplexé* et *composite*.

Héritiers: NÉANT.

Pointé par: classe *composite*.

- Séquence de (Component Class, Availability Start-up?, Availability Close-down?, Rt-Availability Start-up?, Rt-Availability Close-down?, Action Object\*, Link Object\*, Nb of Elements, Composition Element\*, Extensibility Provision).

#### 69.2 Availability Start-up (**Mise en disponibilité**)

- La mise en disponibilité du composite peut être utilisée comme un effet de préparation supplémentaire du composite. Il s'agit soit d'un effet de lien personnalisé fourni par l'auteur soit d'un des effets de lien par défaut figurant dans la présente Recommandation.
- Choix entre (Link Effect, automatic-start-up-1, automatic-start-up-2, automatic start-up-3, automatic-start-up-4, automatic-start-up-5).

#### 69.3 Availability Close-down (**Retrait de disponibilité**)

- Le retrait de disponibilité du composite peut être utilisé comme un effet de destruction supplémentaire du composite. Il s'agit soit d'un effet de lien personnalisé fourni par l'auteur soit d'un des effets de lien par défaut figurant dans la présente Recommandation.
- Choix entre (Link Effect, automatic-close-down-1, automatic-close-down-2, automatic-close-down-3).

#### 69.4 Rt-Availability Start-up (**Mise en disponibilité d'exécution**)

- La mise en disponibilité d'exécution peut être utilisée comme un effet de création supplémentaire du rt-composite. Il s'agit soit d'un effet de lien personnalisé fourni par l'auteur soit de l'effet de lien par défaut figurant dans la présente Recommandation.
- Choix entre (Link Effect, automatic-rt-start-up).

#### 69.5 Rt-Availability Close-down (**Retrait de disponibilité d'exécution**)

- Le retrait de disponibilité d'exécution peut être utilisé comme un effet de destruction supplémentaire du rt-composite. Il s'agit toujours d'un effet de lien personnalisé fourni par l'auteur.
- Link Effect.

#### 69.6 Link Object (**Objet lien**)

- L'objet *lien* est utilisé pour décrire le comportement de chaque élément et les relations parent-enfants. Cet objet peut être référencé ou inclus dans l'objet *composite*.
- Choix entre (Link Class, Link Object Reference).

#### 69.7 Nb of Elements (**Nombre d'éléments**)

- Indique le nombre d'éléments dans la composition.
- Integer.

#### 69.8 Composition Element (**Élément de composition**)

- Définit la structure interne du composite. Cette structure fournit un support aux rt-composites créés.  
Si aucun élément n'est inclus, le moteur MHEG en déduit que le composite a «nombre d'éléments» éléments vides.  
Si le dernier élément a un indice inférieur à «nombre d'éléments», le moteur MHEG en déduit que les éléments vides sont répétés à partir du dernier élément jusqu'au «nombre d'éléments» élément.  
Chaque élément a un indice et fait référence à un objet *composant* modèle ou à une étiquette. L'objet *composant* peut être transféré à l'intérieur ou à l'extérieur de l'objet composite. L'étiquette est toujours incluse.  
Les éléments vides ne sont pas codés, mais ils sont implicitement produits par le moteur MHEG.
- Séquence de (Index, Associated Model).

### 69.9 Element Index (**Indice d'élément**)

- Indice du modèle associé, qui est unique dans un composite.
- Index.

### 69.10 Associated Model (**Modèle associé**)

- L'élément comprend un objet *composant* ou une étiquette, ou pointe vers eux.
- Choix entre (Component Object Reference, Content Class, Multiplexed Content Class, Composite Class, Label).

### 69.11 Label (**Etiquette**)

- Une étiquette est une chaîne; elle peut être fournie sous la forme d'un élément de composite.
- String.

## 70 Attributs de représentation de la classe *conteneur*

Tableau 13/T.171 – Aperçu général des attributs de représentation de la classe *conteneur*

#### Container Class ::=

MH-object Class, Container Start-up?, Container Close-down?, Container Element+, Extensibility Provision

#### Container Start-up ::=

Link Effect | #automatic-container-start-up-1 | #automatic-container-start-up-2 | #automatic-container-start-up-3 | #automatic-container-start-up-4 | #automatic-container-start-up-5 | #automatic-container-start-up-6 | #automatic-container-start-up-7 | #automatic-container-start-up-8

#### Container Close-down ::= Link Effect | #automatic-container-close-down

#### Container Element ::=

MH-Reference | Action Class | Link Class | Script Class | Content Class | Multiplexed Content Class | Composite Class | Container Class | Descriptor Class

### 70.1 Container Class (**Classe conteneur**)

- La classe *conteneur* fournit une structure cohérente pour le traitement des informations multimédias et hypermédias. Ce regroupement est destiné à faciliter le transfert en regroupant en une seule unité l'ensemble des éléments à transférer. Les indices des composants ne doivent pas obligatoirement être codés car les éléments sont numérotés de 1 à n par le moteur MHEG.

Hérite de: classe *objet MH*.

Pointe vers: classes *lien, action, script, contenu, contenu multiplexé, composite, descripteur* et *conteneur*.

Héritiers: NÉANT.

Pointé par: NÉANT.

- Séquence de (MH-object Class, Container Start-up?, Container Close-down?, Container Element+, Extensibility Provision).

### 70.2 Container Start-up (**Mise en disponibilité du conteneur**)

- La mise en disponibilité du conteneur peut être utilisée comme un effet de préparation supplémentaire du conteneur. Il s'agit soit d'un effet de lien personnalisé fourni par l'auteur soit d'un des effets de lien par défaut figurant dans la présente Recommandation.
- Choix entre (Link Effect, automatic-container-start-up-1, automatic-container-start-up-2, automatic-container-start-up-3, automatic-container-start-up-4, automatic-container-start-up-5, automatic-container-start-up-6, automatic-container-start-up-7, automatic-container-start-up-8).

### 70.3 Container Close-down (**Retrait de disponibilité du conteneur**)

- Le retrait de disponibilité du conteneur peut être utilisé comme un effet de destruction supplémentaire du conteneur. Il s'agit soit d'un effet de lien personnalisé fourni par l'auteur soit de l'effet de lien par défaut figurant dans la présente Recommandation.
- Choix entre (Link Effect, automatic-container-close-down).

### 70.4 Container Element (**Élément de conteneur**)

- Un élément peut être inclus ou on peut y faire référence.
- Choix entre (MH-Reference, Action Class, Link Class, Script Class, Content Class, Multiplexed Content Class, Composite Class, Container Class, Descriptor Class).

## 71 Attributs de représentation de la classe *descripteur*

Tableau 14/T.171 – Aperçu général des attributs de représentation de la classe *descripteur*

**Descriptor Class ::=**

MH-object Class, Related Object\*, Other Descriptor\*, Readme?, System Readable Material?, Channel Information\*, Catalogued Style Information\*, Cat Ext elementary action Info\*, Cat Ext Attribute Info\*, Extensibility Provision

**Related Object ::=** MH-Reference, Object Information?

**Object Information ::=** Object Size?, Class ID, Class Specific Information?, Offset?

**Object Size ::=** Integer

**Class Specific Information ::=**

Script Class Information | Content Class Information | Mux Content Class Info

**Script Class Information: ::=** Script Classification?, Script Hook?

**Content Class Information ::=** Cat Content Classification?, Content Hook?, Alternative Object\*

**Mux Content Class Info ::=** Content Class Information, Number Of Streams?, Stream Information\*

**Number Of Streams ::=** Integer

**Stream Information ::=** Stream ID, Content Class Information

**Alternative Object ::=**

Content Object Ref, Content Hook?, Alternative Descriptor Object?, Alternative Readme?

**Alternative Descriptor Object ::=** Descriptor Object Reference

**Alternative Readme ::=** String

**Offset::=Integer**

**Other Descriptor ::=** Descriptor Object Reference

**Readme::=String**

**System Readable Material ::=** *\$BIT STRING* | *\$OCTET STRING*

**Channel Information ::=**

Channel ID, X min?, X max?, Y min?, Y max?, Z min?, Z max?, X Resolution?, Y Resolution?, Z Resolution?, T Resolution?, F min?, F max?, Audio Dynamic?, Channel Media Type\*, Event Mapping\*

**X min: ::=** Integer

**X max ::=** Integer

**Y min ::=** Integer

**Y max ::=** Integer

Tableau 14/T.171 – Aperçu général des attributs de représentation de la classe *descripteur* (fin)

<b>Z min</b> ::= Integer
<b>Z max</b> ::= Integer
<b>X Resolution</b> ::= Integer
<b>Y Resolution</b> ::= Integer
<b>Z Resolution</b> ::= Integer
<b>T Resolution</b> ::=Integer
<b>F min</b> ::= Integer
<b>F max</b> ::= Integer
<b>Audio Dynamic</b> ::= Integer
<b>Channel Media Type</b> ::= Catalogued Media Type
<b>Event Mapping</b> ::= Event, Catalogued Event?
<b>Catalogued Style Information</b> ::= Catalogued Style
<b>Cat Ext elementary action Info</b> ::= Catalogued Extended EA
<b>Cat Ext Attribute Info</b> ::= Cat Ext Attribute

### 71.1 Descriptor Class (**Classe *descripteur***)

- La classe *descripteur* définit une description d'objets qui doivent être transférés. Le but est de faciliter la négociation, l'installation, l'exploitation et la gestion des applications assurant le transfert des objets MHEG. Des objets *descripteur* MHEG, des instances de la classe *descripteur*, peuvent être transférés. Un objet *descripteur* décrit le support utilisé pour décrire d'autres objets transférés.
- Séquence de (MH-object Class, Related Object\*, Other Descriptor\*, Readme?, System Readable Material?, Channel Information\*, Catalogued Style Information\*, Cat Ext elementary action Info\*, Cat Ext Attribute Info\*, Extensibility Provision).

### 71.2 Related Object (**Objet concerné**)

- Spécifie le champ d'action du descripteur, c'est-à-dire la liste des objets MHEG concernés par ce descripteur.
- Séquence de (MH-Reference, Object Information?).

### 71.3 Object Information (**Information d'objet**)

- Définit l'objet concerné d'une manière plus précise.
- Séquence de (Object Size?, Class ID, Class Specific Information?, Offset?).

### 71.4 Object Size (**Taille d'objet**)

- Taille, en octets, de l'objet concerné codé.
- Integer.

### 71.5 Class Specific Information (**Information spécifique de classe**)

- Information spécifique d'objet.
- Choix entre (Script Class Information, Content Class Information, Mux Content Class Info).

### 71.6 Script Class Information (**Information de classe *script***)

- Achemine des informations spécifiques à la classe de *script*.
- Séquence de (Script Classification?, Script Hook?).

- 71.7 Content Class Information (Information de classe contenu)**
- Achemine des informations spécifiques à la classe *contenu*.
  - Séquence de (Cat Content Classification?, Content Hook?, Alternative Object\*).
- 71.8 Mux Content Class Info (Information de classe contenu multiplexé)**
- Achemine des informations spécifiques à la classe *contenu multiplexé*.
  - Séquence de (Content Class Information, Number of Streams?, Stream Information\*).
- 71.9 Number of Streams (Nombre de flux)**
- Indique le nombre de sous-flux.
  - Integer.
- 71.10 Stream Information (Information de flux)**
- D'autres informations concernant le flux.
  - Séquence de (Stream ID, Content Class Information).
- 71.11 Alternative Object (Objet de remplacement)**
- Fournit des informations sur des objets de remplacement possibles qu'un moteur peut utiliser à la place de l'objet concerné.
  - Séquence de (Content Object Ref, Content Hook?, Alternative Descriptor Object?, Alternative Readme?).
- 71.12 Alternative Descriptor Object (Objet descripteur de remplacement)**
- Fait référence au descripteur de remplacement là où les objets de remplacement sont décrits.
  - Descriptor Object Reference.
- 71.13 Alternative Readme («Lisez-moi» de remplacement)**
- Identifie la nature de l'objet de remplacement.
  - String.
- 71.14 Offset (Décalage)**
- Indique la position relative, en octets, d'un objet codé qui est inclus dans un autre objet codé.
  - Integer.
- 71.15 Other Descriptor (Autre descripteur)**
- Spécifie une référence à un autre objet *descripteur*.
  - Descriptor Object Reference.
- 71.16 Readme («Lisez-moi»)**
- Fournit un texte pouvant être lu par l'homme.
  - String.
- 71.17 System Readable Material (Informations en langage machine)**
- Fournit des informations pour des applications utilisatrices.
  - Choix entre (BIT STRING, OCTET STRING).

**71.18 Channel Information (Information de canal)**

- Contient des informations sur le traitement d'un canal.
- Séquence de (Channel ID, X min?, X max?, Y min?, Y max?, Z min?, Z max?, X Resolution?, Y Resolution?, Z Resolution?, T Resolution?, F min?, F max?, Audio Dynamic?, Channel Media Type\*, Event Mapping\*).

**71.19 X min (X min)**

- Définit la valeur minimale de l'abscisse pour le dispositif sur lequel ce canal doit être projeté.
- Integer.

**71.20 X max (X max)**

- Définit la valeur maximale de l'abscisse pour le dispositif sur lequel ce canal doit être projeté.
- Integer.

**71.21 Y min (Y min)**

- Définit la valeur minimale de l'ordonnée pour le dispositif sur lequel ce canal doit être projeté.
- Integer.

**71.22 Y max (Y max)**

- Définit la valeur maximale de l'ordonnée pour le dispositif sur lequel ce canal doit être projeté.
- Integer.

**71.23 Z min (Z min)**

- Définit la valeur minimale de la cote pour le dispositif sur lequel ce canal doit être projeté.
- Integer.

**71.24 Z max (Z max)**

- Définit la valeur maximale de la cote pour le dispositif sur lequel ce canal doit être projeté.
- Integer.

**71.25 X Resolution (Résolution X)**

- Nombre d'unités physiques spatiales adressables sur l'abscisse.
- Integer.

**71.26 Y Resolution (Résolution Y)**

- Nombre d'unités physiques spatiales adressables sur l'ordonnée.
- Integer.

**71.27 Z Resolution (Résolution Z)**

- Nombre d'unités physiques spatiales adressables sur la cote.
- Integer.

**71.28 T Resolution (Résolution T)**

- Nombre d'unités physiques spatiales adressables par seconde sur l'axe temporel.
- Integer.

- 71.29** `F min` (**F min**)
- Définit la fréquence minimale présentable pour le dispositif sur lequel ce canal doit être projeté.
  - Integer.
- 71.30** `F max` (**F max**)
- Définit la fréquence maximale présentable pour le dispositif sur lequel ce canal doit être projeté.
  - Integer.
- 71.31** `Audio Dynamic` (**Dynamique audio**)
- Définit en dB le dynamique audiofréquence du dispositif sur lequel le canal doit être projeté.
  - Integer.
- 71.32** `Channel Media Type` (**Type de support de canal**)
- Spécifie le type de support du canal; c'est une des valeurs cataloguées, qui peut être soit enregistrée soit privée.
  - Catalogued Media Type.
- 71.33** `Event Mapping` (**Mappage d'événement**)
- Spécifie les événements prévus et leur mappage à un événement catalogué.
  - Séquence de (`Event`, `Catalogued Event?`).
- 71.34** `Catalogued Style Information` (**Information de style catalogué**)
- Spécifie le style utilisé par les objets concernés.
  - Catalogued Style.
- 71.35** `Cat Ext elementary action Info` (**Information d'action élémentaire étendue cataloguée**)
- Spécifie l'action élémentaire cataloguée utilisée par les objets concernés.
  - Catalogued Extended EA.
- 71.36** `Cat Ext Attribute Info` (**Information d'attribut étendu catalogué**)
- Spécifie l'attribut étendu utilisé par les objets concernés.
  - `Cat Ext Attribute`.

## 72 Comportement

### 72.1 Comportement de renvoi

**Tableau 15/T.171 – Aperçu général du comportement de renvoi**

<b>Delay</b> ::= Temporal Unit Ref Param, Duration Param
<b>Temporal Unit Ref Param</b> ::= Temporal Unit Ref   Temporal Unit Ref Macro
<b>Temporal Unit Ref Macro</b> ::= Macro Def ID, Temporal Unit Ref?
<b>Temporal Unit Ref</b> ::= Rt-Component Reference   default-GF
<b>Duration Param</b> ::= Generic Integer Param



- 72.1.1 Delay (Retard)**
- Cette action permet de retarder le traitement d'autres actions.
  - Séquence de (Temporal Unit Ref Param, Duration Param).
- 72.1.2 Temporal Unit Ref Param (Paramètre de référence d'unité temporelle)**
- Paramètre de référence d'une unité temporelle.
  - Choix entre (Temporal Unit Ref, Temporal Unit Ref Macro).
- 72.1.3 Temporal Unit Ref Macro (Macro-paramètre de référence d'unité temporelle)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de référence d'unité temporelle.
  - Séquence de (Macro Def ID, Temporal Unit Ref?).
- 72.1.4 Temporal Unit Ref (Référence d'unité temporelle)**
- Définit comment retirer le facteur GTF afin d'interpréter le retard.
  - Choix entre (Rt-Component Reference, default-GF).
- 72.1.5 Duration Param (Paramètre de durée)**
- La durée est exprimée en unités GTU.
  - Generic Integer Param.
- 72.2 Comportement de retour**

**Tableau 16/T.171 – Aperçu général du comportement de retour**

**Return ::=**  
Return Target Param+, Return Indicator Param, Returned Generic Value Param\*, Content Object Ref Param\*

**Return Indicator Param ::=** Generic Numeric Param

**Returned Generic Value Param ::=** Generic Value Param

**Content Object Ref Param ::=** Content Object Ref | Content Object Ref Macro

**Content Object Ref Macro ::=** Macro Def ID, Content Object Ref?

- 72.2.1 Return (Retour)**
- Cette action permet de renvoyer une information à une application utilisatrice ou à des entités externes au moteur MHEG.
  - Séquence de (Return Target Param+, Return Indicator Param, Returned Generic Value Param\*, Content Object Ref Param\*).
- 72.2.2 Return Indicator Param (Paramètre d'indicateur de retour)**
- Identification de l'information renvoyée.
  - Generic Numeric Param.
- 72.2.3 Returned Generic Value Param (Paramètre des valeurs génériques renvoyées)**
- Une liste des valeurs génériques à renvoyer.
  - Generic Value Param.
- 72.2.4 Content Object Ref Param (Paramètre de référence d'objet contenu)**
- Une liste des objets *contenu* à renvoyer.
  - Choix entre (Content Object Ref, Content Object Ref Macro).

### 72.2.5 Content Object Ref Macro (**Macro-paramètre de référence d'objet contenu**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de référence d'objet *contenu*.
- Séquence de (Macro Def ID, Content Object Ref?).

## 72.3 Comportement des alias

Tableau 17/T.171 – Aperçu général du comportement des alias

<p><b>Set Alias</b> ::= Target Param+, Alias Spec Param+</p> <p><b>Alias Spec Param</b> ::= Alias Spec   Alias Spec Macro</p> <p><b>Alias Spec Macro</b> ::= Macro Def ID, Alias Spec?</p> <p><b>Alias Spec</b> ::= Alias+, Update Command</p>
--

### 72.3.1 Set Alias (**Fixer un alias**)

- Cette action permet d'affecter des alias aux références génériques.
- Séquence de (Target Param+, Alias Spec Param+).

### 72.3.2 Alias Spec Param (**Paramètre de spécification d'alias**)

- Paramètre de spécification d'alias.
- Choix entre (Alias Spec, Alias Spec Macro).

### 72.3.3 Alias Spec Macro (**Macro-paramètre de spécification d'alias**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification d'alias.
- Séquence de (Macro Def ID, Alias Spec?).

### 72.3.4 Alias Spec (**Spécification d'alias**)

- Des alias et la commande de mise à jour à appliquer.
- Séquence de (Alias+, Update Command).

## 72.4 Comportement d'extensibilité

Tableau 18/T.171 – Aperçu général du comportement d'extensibilité

<p><b>Catalogued Elementary Action</b> ::= Target Param+, Catalogued Extended EA Param, Elementary Action Param*</p> <p><b>Catalogued Extended EA Param</b> ::= Catalogued Extended EA   Catalogued Extended EA Macro</p> <p><b>Catalogued Extended EA Macro</b> ::= Macro Def ID, Catalogued Extended EA?</p> <p><b>Elementary Action Param</b> ::= Generic Value Param</p> <p><b>Set Catalogued Attribute</b> ::= Target Param+, Cat Ext Attribute Param, Ext Attribute Value Param, Transition Duration Param?</p> <p><b>Ext Attribute Value Param</b> ::= Generic Value Param</p>
---

- 72.4.1 Catalogued Elementary Action (Action élémentaire cataloguée)**
- Cette action permet d'utiliser une action élémentaire étendue enregistrée ou privée.
  - Séquence de (Target Param+, Catalogued Extended EA Param, Elementary Action Param\*).
- 72.4.2 Catalogued Extended EA Param (Paramètre d'action élémentaire étendue cataloguée)**
- Paramètre d'action élémentaire étendue cataloguée.
  - Choix entre (Catalogued Extended EA, Catalogued Extended EA Macro).
- 72.4.3 Catalogued Extended EA Macro (Macro-paramètre d'action élémentaire étendue cataloguée)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'action élémentaire étendue cataloguée.
  - Séquence de (Macro Def ID, Catalogued Extended EA?).
- 72.4.4 Elementary Action Param (Paramètre d'action élémentaire)**
- Utilisé pour transférer les paramètres spécifiques de l'action élémentaire étendue cataloguée.
  - Generic Value Param.
- 72.4.5 Set Catalogued Attribute (Fixer un attribut catalogué)**
- Cette action permet de fixer la valeur d'un attribut catalogué pour la cible.
  - Séquence de (Target Param+, Cat Ext Attribute Param, Ext Attribute Value Param, Transition Duration Param?).
- 72.4.6 Ext Attribute Value Param (Paramètre de valeur d'attribut étendu)**
- Valeur générique à affecter à un attribut étendu catalogué.
  - Generic Value Param.
- 72.5 Comportement de disponibilité des objets Mheg**

**Tableau 19/T.171 – Aperçu général du comportement de disponibilité des objets Mheg**

**Prepare** ::= MH-Target Param+

**Destroy** ::= MH-Target Param+

- 72.5.1 Prepare (Préparer)**
- Cette action rend l'action MHEG disponible pour le moteur MHEG.
  - Liste de (MH-Target Param).
- 72.5.2 Destroy (Détruire)**
- Cette action retire un objet MHEG du moteur MHEG afin de libérer des ressources.
  - Liste de (MH-Target Param).

**72.6 Comportement d'activation des objets *lien***

**Tableau 20/T.171 – Aperçu général du comportement d'activation des objets *lien***

**Activate** ::= Link Target Param+

**Deactivate** ::= Link Target Param+

### 72.6.1 Activate (**Activer**)

- Activer l'objet *lien*.
- Liste de (Link Target Param).

### 72.6.2 Deactivate (**Désactiver**)

- Désactiver l'objet *lien*.
- Liste de (Link Target Param).

## 72.7 Comportement d'abandon des objets *lien*

**Tableau 21/T.171 – Aperçu général du comportement d'abandon des objets *lien***

<b>Link Abort</b> ::= Link Target Param+
--

### 72.7.1 Link Abort (**Abandon de lien**)

- Abandon du traitement des liens spécifiés dans l'ensemble de cibles.
- Liste de (Link Target Param).

## 72.8 Comportement de stockage de la valeur générique de la classe *contenu*

**Tableau 22/T.171 – Aperçu général du comportement de stockage de la valeur générique de la classe *contenu***

<b>Set Data</b> ::= Content Target Param+, Substitution Indicator Param, Data Element Param*
--

<b>Substitution Indicator Param</b> ::= Substitution Indicator   Substitution Indicator Macro
---

<b>Substitution Indicator Macro</b> ::= Macro Def ID, Substitution Indicator?
---

<b>Substitution Indicator</b> ::= # <i>substitution</i> / # <i>no-substitution</i>
--

<b>Data Element Param</b> ::= Data Element   Data Element Macro
---

<b>Data Element Macro</b> ::= Macro Def ID, Data Element?
---

<b>Data Element</b> ::= Process Indicator, Generic List Elt ID?, Generic Value
--

<b>Process Indicator</b> ::= # <i>process</i> / # <i>no-process</i>
---

<b>Add</b> ::= Content Target Param+, Generic List Elt ID Param?, Generic Value Param?
--

<b>Substract</b> ::= Content Target Param+, Generic List Elt ID Param?, Generic Value Param?
--

### 72.8.1 Set Data (**Fixer des données**)

- Cette action permet de stocker ou de modifier la valeur générique dans un objet *contenu*.
- Séquence de (Content Target Param+, Substitution Indicator Param, Data Element Param\*).

### 72.8.2 Substitution Indicator Param (**Paramètre d'indicateur de substitution**)

- Paramètre d'indicateur de substitution.
- Choix entre (Substitution Indicator, Substitution Indicator Macro).

- 72.8.3** Substitution Indicator Macro (**Macro-paramètre d'indicateur de substitution**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour l'indicateur de substitution.
  - Séquence de (Macro Def ID, Substitution Indicator?).
- 72.8.4** Substitution Indicator (**Indicateur de substitution**)
- Valeurs autorisées pour l'indicateur de substitution.
  - Choix entre (substitution, no-substitution).
- 72.8.5** Data Element Param (**Paramètre d'élément de donnée**)
- Paramètre d'élément de donnée.
  - Choix entre (Data Element, Data Element Macro).
- 72.8.6** Data Element Macro (**Macro-paramètre d'élément de donnée**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'élément de donnée.
  - Séquence de (Macro Def ID, Data Element?).
- 72.8.7** Data Element (**Élément de donnée**)
- Désigne une donnée à stocker.
  - Séquence de (Process Indicator, Generic List Elt ID?, Generic Value).
- 72.8.8** Process Indicator (**Indicateur de traitement**)
- Désigne une donnée à stocker, qu'elle ait été évaluée ou pas.
  - Choix entre (process, no-process).
- 72.8.9** Add (**Ajouter**)
- Ajoute un chiffre générique, un entier générique ou un rapport générique à une donnée de contenu.
  - Séquence de (Content Target Param+, Generic List Elt ID Param?, Generic Value Param?).
- 72.8.10** Subtract (**Soustraire**)
- Offre de soustraire un chiffre générique, un entier générique ou un rapport générique d'une donnée de contenu.
  - Séquence de (Content Target Param+, Generic List Elt ID Param?, Generic Value Param?).
- 72.9** **Comportement de duplication de la classe *contenu***

**Tableau 23/T.171 – Aperçu général du comportement de duplication de la classe *contenu***

**Copy ::=** Content Target Param, Destination Param+

**Destination Param ::=** Content Target Param

- 72.9.1** Copy (**Copier**)
- Cette action spécifie un objet *contenu* cible comme source de l'opération de duplication et un ensemble d'objets *contenu* comme destination de l'opération de duplication.
  - Séquence de (Content Target Param, Destination Param+).
- 72.9.2** Destination Param (**Paramètre de destination**)
- Objets *contenu* dans lesquels les données cibles doivent être copiées.
  - Content Target Param.

## 72.10 Comportement de disponibilité des rt-objets

Tableau 24/T.171 – Aperçu général du comportement de disponibilité des rt-objets

<b>New</b> ::= Rt-Target Param+
<b>Delete</b> ::= Rt-Target Param+

### 72.10.1 New (Nouveau)

- Créé des rt-objets à partir d'objets modèles pour le moteur MHEG.
- Liste de (Rt-Target Param).

### 72.10.2 Delete (Supprimer)

- Supprime des rt-objets du moteur MHEG. Cette action peut être utilisée pour libérer des ressources.
- Liste de (Rt-Target Param).

## 72.11 Comportement d'exécution des rt-objets

Tableau 25/T.171 – Aperçu général du comportement d'exécution des rt-objets

<b>Run</b> ::= Rt-Target Param+, Number Of Performances Param?
<b>Number Of Performances Param</b> ::= Number Of Performances   Number Of Performances Macro
<b>Number Of Performances Macro</b> ::= Macro Def ID, Number Of Performances?
<b>Number Of Performances</b> ::= Generic Integer   <i>#infinite</i>
<b>Stop</b> ::= Rt-Target Param+

### 72.11.1 Run (Exécution)

- Permet la présentation des rt-objets au moyen du processus de présentation.
- Séquence de (Rt-Target Param+, Number of Performances Param?).

### 72.11.2 Number of Performances Param (Paramètre du nombre d'exécutions)

- Paramètre du nombre d'exécutions.
- Choix entre (Number of Performances, Number of Performances Macro).

### 72.11.3 Number of Performances Macro (Macro-paramètre de nombre d'exécutions)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre du nombre d'exécutions.
- Séquence de (Macro Def ID, Number of Performances?).

### 72.11.4 Number of Performances (Nombre d'exécutions)

- Spécifie le nombre d'exécutions.
- Choix entre (Generic Integer, infinite).

### 72.11.5 Stop (Arrêt)

- Met l'état d'exécution à «pas en cours d'exécution» et arrête les effets d'utilisateur possibles.
- Liste de (Rt-Target Param).

## 72.12 Comportement de passage de paramètre à un rt-script

Tableau 26/T.171 – Aperçu général du comportement de passage de paramètre à un rt-script

**Set Parameters** ::= Rt-Script Target Param+, Passing Param\*

**Passing Param** ::= Passing | Passing Macro

**Passing Macro** ::= Macro Def ID, Passing?

**Passing** ::= Generic Value | Content Object Ref

### 72.12.1 Set Parameters (Fixer paramètres)

- Offre le moyen de passer des paramètres à un rt-script.
- Séquence de (Rt-Script Target Param+, Passing Param\*).

### 72.12.2 Passing Param (Paramètre de passage)

- Paramètre de passage.
- Choix entre (Passing, Passing Macro).

### 72.12.3 Passing Macro (Macro-paramètre de passage)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de passage.
- Séquence de (Macro Def ID, Passing?).

### 72.12.4 Passing (Passage)

- Paramètres à passer à un rt-script.
- Choix entre (Generic Value, Content Object Ref).

## 72.13 Comportement de dynamisme de structure et de présentation des réceptacles

Tableau 27/T.171 – Aperçu général du comportement de dynamisme de structure et de présentation des réceptacles

**Plug** ::= Socket Target Param+, Plug In Param

**Plug In Param** ::= Plug In | Plug In Macro

**Plug In Macro** ::= Macro Def ID, Plug In?

**Plug In** ::= Rt-Component Reference | Component Object Reference | Label | Evaluated Reference

### 72.13.1 Plug (Enficher)

- Attache un rt-composant ou une étiquette à un réceptacle.
- Séquence de (Socket Target Param+, Plug In Param).

### 72.13.2 Plug In Param (Paramètre d'informations à enficher)

- Paramètre d'informations à enficher.
- Choix entre (Plug In, Plug In Macro).

### 72.13.3 Plug In Macro (Macro-paramètre d'informations à enficher)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'informations à enficher.
- Séquence de (Macro Def ID, Plug In?).

#### 72.13.4 Plug in (**Informations à enficher**)

- Spécifie les informations à enficher dans un réceptacle.
- Choix entre (Rt-Component Reference, Component Object Reference, Label, Evaluated Reference).

### 72.14 Comportement d'affectation d'espace RPS aux rt-composants

Tableau 28/T.171 – Aperçu général du comportement d'affectation d'espace RPS aux rt-composants

**Set RPS Assignment** ::= Rt-Component Target Param+, RPS Assignment Param

**RPS Assignment Param** ::= RPS Assignment | RPS Assignment Macro

**RPS Assignment Macro** ::= Macro Def ID, RPS Assignment?

#### 72.14.1 Set RPS Assignment (**Fixer l'affectation d'espace RPS**)

- Fixer l'affectation d'espace RPS de la cible.
- Séquence de (Rt-Component Target Param+, RPS Assignment Param).

#### 72.14.2 RPS Assignment Param (**Paramètre d'affectation d'espace RPS**)

- Paramètre d'affectation d'espace RPS.
- Choix entre (RPS Assignment, RPS Assignment Macro).

#### 72.14.3 RPS Assignment Macro (**Macro-paramètre d'affectation d'espace RPS**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'affectation d'espace RPS.
- Séquence de (Macro Def ID, RPS Assignment?).

### 72.15 Comportement de perceptibilité des rt-composants

Tableau 29/T.171 – Aperçu général du comportement de perceptibilité des rt-composants

**Set Perceptability** ::= Rt-Component Target Param+, Perceptability Param, Transition Duration Param?

**Perceptability Param** ::= Generic Ratio Param

**Set Presentation Priority** ::=

Rt-Component Target Param+, Presentation Priority Param, Transition Duration Param?

**Presentation Priority Param** ::= Presentation Priority | Presentation Priority Macro

**Presentation Priority Macro** ::= Macro Def ID, Presentation Priority?

#### 72.15.1 Set Perceptability (**Fixer la perceptibilité**)

- Cette action permet de changer la perceptibilité d'un rt-composant ou d'un rt-composant racine.
- Séquence de (Rt-Component Target Param+, Perceptability Param, Transition Duration Param?).

#### 72.15.2 Perceptability Param (**Paramètre de perceptibilité**)

- Perceptibilité à affecter à un rt-composant.
- Generic Ratio Param.



### 72.15.3 Set Presentation Priority (**Fixer la priorité de présentation**)

- Cette action permet de modifier la priorité de présentation d'un rt-composant ou d'un rt-composant racine.
- Séquence de (Rt-Component Target Param+, Presentation Priority Param, Transition Duration Param?).

### 72.15.4 Presentation Priority Param (**Paramètre de priorité de présentation**)

- Paramètre de priorité de présentation.
- Choix entre (Presentation Priority, Presentation Priority Macro).

### 72.15.5 Presentation Priority Macro (**Macro-paramètre de priorité de présentation**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de priorité de présentation.
- Séquence de (Macro Def ID, Presentation Priority?).

## 72.16 Comportement temporel des rt-composants

**Tableau 30/T.171 – Aperçu général du comportement temporel des rt-composants**

**Set OVD** ::= Rt-Component Target Param+, Initial Point Spec Param, Terminal Point Spec Param

**Set CTP** ::= Rt-Component Target Param+, Current Point Spec Param

**Set Temporal Termination** ::= Rt-Target Param+, Temporal Termination Param

**Temporal Termination Param** ::= Temporal Termination | Temporal Termination Macro

**Temporal Termination Macro** ::= Macro Def ID, Temporal Termination?

**Set PVD Position** ::= Socket Target Param+, Temporal Position Param

**Temporal Position Param** ::= Point Spec Param

**Set GTF** ::= Rt-Component Target Param+, GTF Param

**GTF Param** ::= GF Param

**Set Timestones** ::= Rt-Component Target Param+, Timestone Spec Param+

**Timestone Spec Param** ::= Timestone Spec | Timestone Spec Macro

**Timestone Spec Macro** ::= Macro Def ID, Timestone Spec?

**Timestone Spec** ::= Timestone+, Update Command

**Timestone** ::= Timestone ID, Timestone Position, Number Of Repetitions

**Timestone Position** ::= Point Spec

**Number Of Repetitions** ::= Generic Integer | *#infinite*

### 72.16.1 Set OVD (**Fixer la durée OVD**)

- Le premier paramètre de spécification de point spécifie la position temporelle initiale de la durée OVD. Le second en spécifie la position temporelle terminale.
- Séquence de (Rt-Component Target Param+, Initial Point Spec Param, Terminal Point Spec Param).

### 72.16.2 Set CTP (**Fixer la position CTP**)

- Cette action spécifie la position CTP dans la durée OVD.
- Séquence de (Rt-Component Target Param+, Current Point Spec Param).

### 72.16.3 Set Temporal Termination (**Fixer achèvement temporel**)

- Spécifie l'action à exécuter une fois que la position temporelle finale du rt-composant est atteinte.
- Séquence de (Rt-Target Param+, Temporal Termination Param).

- 72.16.4** Temporal Termination Param (**Paramètre d'achèvement temporel**)
- Paramètre d'achèvement temporel.
  - Choix entre (Temporal Termination, Temporal Termination Macro).
- 72.16.5** Temporal Termination Macro (**Macro-paramètre d'achèvement temporel**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'achèvement temporel.
  - Séquence de (Macro Def ID, Temporal Termination?).
- 72.16.6** Set PVD Position (**Fixer la position PVD**)
- Lier les durées PVD des enfants à la durée OVD de leur parent.
  - Séquence de (Socket Target Param+, Temporal Position Param).
- 72.16.7** Temporal Position Param (**Paramètre de position temporelle**)
- Spécifie la position temporelle soit par une valeur absolue le long d'un axe temporel soit par une valeur relative par rapport à une durée temporelle.
  - Point Spec Param.
- 72.16.8** Set GTF (**Fixer le facteur GTF**)
- Cette action définit la vitesse de présentation d'un rt-composant.
  - Séquence de (Rt-Component Target Param+, GTF Param).
- 72.16.9** GTF Param (**Paramètre GTF**)
- Fraction générique ou facteur générique par défaut.
  - GF Param.
- 72.16.10** Set Timestones (**Marqueurs temporels**)
- Fixe des marqueurs temporels.
  - Séquence de (Rt-Component Target Param+, Timestone Spec Param+).
- 72.16.11** Timestone Spec Param (**Paramètre de spécification de marqueur temporel**)
- Paramètre de spécification de marqueur temporel.
  - Choix entre (Timestone Spec, Timestone Spec Macro).
- 72.16.12** Timestone Spec Macro (**Macro-paramètre de spécification de marqueur temporel**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de marqueur temporel.
  - Séquence de (Macro Def ID, Timestone Spec?).
- 72.16.13** Timestone Spec (**Spécification de marqueur temporel**)
- Ensemble de marqueurs temporels et les commandes de mise à jour à appliquer.
  - Séquence de (Timestone+, Update Command).
- 72.16.14** Timestone (**Marqueur temporel**)
- Une paire composée d'un identificateur de marqueur temporel et d'une position de marqueur temporel, et le nombre de répétitions.
  - Séquence de (Timestone ID, Timestone Position, Number of Repetitions).
- 72.16.15** Timestone Position (**Position de marqueur temporel**)
- Spécifiée par la spécification de point.
  - Point Spec.
- 72.16.16** Number of Repetitions (**Nombre de répétitions**)
- Spécifie le nombre de répétitions.
  - Choix entre (Generic Integer, infinite).

## 72.17 Comportement spatial des rt-composants

Tableau 31/T.171 – Aperçu général du comportement spatial des rt-composants

<p><b>Set Aspect Ratio</b> ::= Rt-Component Target Param+, Aspect Ratio Param</p> <p><b>Aspect Ratio Param</b> ::= Aspect Ratio   Aspect Ratio Macro</p> <p><b>Aspect Ratio Macro</b> ::= Macro Def ID, Aspect Ratio?</p> <p><b>Set Resizing Strategy</b> ::= Rt-Composite Target Param+, Resizing Strategy Param</p> <p><b>Resizing Strategy Param</b> ::= Resizing Strategy   Resizing Strategy Macro</p> <p><b>Resizing Strategy Macro</b> ::= Macro Def ID, Resizing Strategy?</p> <p><b>Set OVS Proj Strategy</b> ::= Rt-Component Target Param+, OVS Proj Strategy Param</p> <p><b>OVS Proj Strategy Param</b> ::= OVS Proj Strategy   OVS Proj Strategy Macro</p> <p><b>OVS Proj Strategy Macro</b> ::= Macro Def ID, OVS Proj Strategy?</p> <p><b>Set OVS</b> ::= Rt-Component Target Param+, Size Spec Param, Transition Duration Param?</p> <p><b>Set OAP</b> ::= Rt-Component Target Param+, OAP Param</p> <p><b>OAP Param</b> ::= Spatial Position Spec Param</p> <p><b>Set OVS Position</b> ::= Rt-Component Target Param+, OVS Position Param, Transition Duration Param?</p> <p><b>OVS Position Param</b> ::= Spatial Position Spec Param</p> <p><b>Set PAP</b> ::= Rt-Component Target Param+, PAP Param</p> <p><b>PAP Param</b> ::= Spatial Position Spec Param</p> <p><b>Set PVS Position</b> ::= Rt-Component Target Param+, PVS Position Param, Transition Duration Param?</p> <p><b>PVS Position Param</b> ::= Spatial Position Spec Param</p> <p><b>Set GSF</b> ::= Rt-Component Target Param+, GSF Param, Transition Duration Param?</p> <p><b>GSF Param</b> ::= GF Param</p> <p><b>Set User Spatial Control</b> ::= Rt-Component Target Param+, Spatial Control Param+, User Spatial Control Param</p> <p><b>User Spatial Control Param</b> ::= User Spatial Control   User Spatial Control Macro</p> <p><b>User Spatial Control Macro</b> ::= Macro Def ID, User Spatial Control?</p>
--

### 72.17.1 Set Aspect Ratio (Fixer le facteur de forme)

- Spécifie si la taille d'origine est projetée avec le facteur de forme préservé.
- Séquence de (Rt-Component Target Param+, Aspect Ratio Param).

### 72.17.2 Aspect Ratio Param (Paramètre de facteur de forme)

- Spécifie la préservation du facteur de forme.
- Choix entre (Aspect Ratio, Aspect Ratio Macro).

### 72.17.3 Aspect Ratio Macro (Macro-paramètre de facteur de forme)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de facteur de forme.
- Séquence de (Macro Def ID, Aspect Ratio?).

- 72.17.4** Set Resizing Strategy (**Fixer la stratégie de redimensionnement**)
- Spécifie la stratégie de redimensionnement de la taille d'origine d'un rt-composite.
  - Séquence de (Rt-Composite Target Param+, Resizing Strategy Param).
- 72.17.5** Resizing Strategy Param (**Paramètre de stratégie de redimensionnement**)
- Spécifie la stratégie de redimensionnement.
  - Choix entre (Resizing Strategy, Resizing Strategy Macro).
- 72.17.6** Resizing Strategy Macro (**Macro-paramètre de stratégie de redimensionnement**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de stratégie de redimensionnement.
  - Séquence de (Macro Def ID, Resizing Strategy?).
- 72.17.7** Set OVS Proj Strategy (**Fixer la stratégie de projection de la taille OVS**)
- Spécifie la stratégie de projection de la taille OVS. Choix entre «fixe» et «calculé».
  - Séquence de (Rt-Component Target Param+, OVS Proj Strategy Param).
- 72.17.8** OVS Proj Strategy Param (**Paramètre de stratégie de projection de la taille OVS**)
- Paramètre de la stratégie de projection de la taille OVS.
  - Choix entre (OVS Proj Strategy, OVS Proj Strategy Macro).
- 72.17.9** OVS Proj Strategy Macro (**Macro-paramètre de stratégie de projection de la taille OVS**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de stratégie de projection de la taille OVS.
  - Séquence de (Macro Def ID, OVS Proj Strategy?).
- 72.17.10** Set OVS (**Fixer la taille OVS**)
- Fixer la taille OVS directement ou indirectement selon l'attribut OVS Proj Strategy (stratégie de projection de la taille OVS).
  - Séquence de (Rt-Component Target Param+, Size Spec Param, Transition Duration Param?).
- 72.17.11** Set OAP (**Fixer le point OAP**)
- Fixer le point OAP.
  - Séquence de (Rt-Component Target Param+, OAP Param).
- 72.17.12** OAP Param (**Paramètre de point OAP**)
- Peut être spécifié comme valeur absolue ou comme valeur relative.
  - Spatial Position Spec Param.
- 72.17.13** Set OVS Position (**Fixer la position OVS**)
- Fixer la position OVS.
  - Séquence de (Rt-Component Target Param+, OVS Position Param, Transition Duration Param?).
- 72.17.14** OVS Position Param (**Paramètre de position OVS**)
- Peut être spécifié comme valeur absolue ou comme valeur relative.
  - Spatial Position Spec Param.
- 72.17.15** Set PAP (**Fixer le point PAP**)
- Fixe le point PAP.
  - Séquence de (Rt-Component Target Param+, PAP Param).

#### 72.17.16 PAP Param (**Paramètre de point PAP**)

- Peut être spécifié comme valeur absolue ou comme valeur relative.
- Spatial Position Spec Param.

#### 72.17.17 Set PVS Position (**Fixer la position PVS**)

- Fixe la position PVS.
- Séquence de (Rt-Component Target Param+, PVS Position Param, Transition Duration Param?).

#### 72.17.18 PVS Position Param (**Paramètre de position PVS**)

- Peut être spécifié comme valeur absolue ou comme valeur relative.
- Spatial Position Spec Param.

#### 72.17.19 Set GSF (**Fixer le facteur GSF**)

- Fixe le facteur GSF.
- Séquence de (Rt-Component Target Param+, GSF Param, Transition Duration Param?).

#### 72.17.20 GSF Param (**Paramètre de facteur GSF**)

- Une fraction générique.
- GF Param.

#### 72.17.21 Set User Spatial Control (**Fixer une commande spatiale utilisateur**)

- Fixe une des commandes spatiales utilisateur.
- Séquence de (Rt-Component Target Param+, Spatial Control Param+, User Spatial Control Param).

#### 72.17.22 User Spatial Control Param (**Paramètre de commande spatiale utilisateur**)

- Choix entre «autorisé» et «pas autorisé».
- Choix entre (User Spatial Control, User Spatial Control Macro).

#### 72.17.23 User Spatial Control Macro (**Macro-paramètre de commande spatiale utilisateur**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de commande spatiale utilisateur.
- Séquence de (Macro Def ID, User Spatial Control?).

### 72.18 Comportement audible des rt-composants

**Tableau 32/T.171 – Aperçu général du comportement audible des rt-composants**

**Set CV** ::= Rt-Content Target Param+, CV Param, Transition Duration Param?

**CV Param** ::= Current Point Spec Param

**Set GVF** ::= GVF Target, GVF Param, Transition Duration Param?

**GVF Target** ::= Rt-Composite Targets Param | Channel Targets Param

**Rt-Composite Targets Param** ::= Rt-Composite Target Param+

**Channel Targets Param** ::= Channel Target Param+

**GVF Param** ::= GF Param

### 72.18.1 Set CV (**Fixer volume CV**)

- Cette action spécifie le volume CV d'un rt-contenu.
- Séquence de (Rt-Content Target Param+, CV Param, Transition Duration Param?).

### 72.18.2 CV Param (**Paramètre de volume CV**)

- Spécifié sous forme de valeur absolue, de valeur relative, de facteur de point d'origine ou de facteur de point courant.
- Current Point Spec Param.

### 72.18.3 Set GVF (**Fixer facteur GVF**)

- Cette action spécifie le facteur GVF d'un rt-composite ou d'un canal.
- Séquence de (GVF Target, GVF Param, Transition Duration Param?).

### 72.18.4 GVF Target (**Cible GVF**)

- Cibles pour le facteur GVF.
- Choix entre (Rt-Composite Targets Param, Channel Targets Param).

### 72.18.5 Rt-Composite Targets Param (**Paramètre de cibles rt-composites**)

- Cibles pour le facteur GVF.
- Liste de (Rt-Composite Target Param).

### 72.18.6 Channel Targets Param (**Paramètre de cibles canaux**)

- Cibles pour le facteur GVF.
- Liste de (Channel Targets Param).

### 72.18.7 GVF Param (**Paramètre GVF**)

- Valeur GVF à affecter.
- GF Param.

## 72.19 Comportement de choix de flux rt-mux

Tableau 33/T.171 – Aperçu général du comportement de choix de flux rt-mux

**Set Stream Choice** ::= Rt-Mux Target Param+, Stream Spec Param\*

**Stream Spec Param** ::= Stream Spec | Stream Spec Macro

**Stream Spec Macro** ::= Macro Def ID, Stream Spec?

**Stream Spec** ::= Stream ID Reference\*, Update Command

### 72.19.1 Set Stream Choice (**Fixer choix de flux**)

- Cette action spécifie une liste de flux à choisir parmi les données multiplexées et à affecter à la cible.
- Séquence de (Rt-Mux Target Param+, Stream Spec Param\*).

### 72.19.2 Stream Spec Param (**Paramètre de spécification de flux**)

- Paramètre de spécification de flux.
- Choix entre (Stream Spec, Stream Spec Macro).

### 72.19.3 Stream Spec Macro (**Macro-paramètre de spécification de flux**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de flux.
- Séquence de (Macro Def ID, Stream Spec?).

#### 72.19.4 Stream Spec (**Spécification de flux**)

- Combinaison d'une liste d'identificateurs de flux et une commande de mise à jour.
- Séquence de (Stream ID Reference\*, Update Command).

#### 72.20 Comportement d'interaction

**Tableau 34/T.171 – Aperçu général du comportement d'interaction**

<b>Interaction Status Param</b> ::= Interaction Status   Interaction Status Macro
<b>Interaction Status Macro</b> ::= Macro Def ID, Interaction Status?
<b>Min Interact Required Param</b> ::= Min Interact Required   Min Interact Required Macro
<b>Min Interact Required Macro</b> ::= Macro Def ID, Min Interact Required?
<b>Max Interact Required Param</b> ::= Max Interact Required   Max Interact Required Macro
<b>Max Interact Required Macro</b> ::= Macro Def ID, Max Interact Required?
<b>Set Interaction Ability</b> ::= Rt-Component Target Param+, Interaction Type Param, Min Interact Required Param, Max Interact Required Param
<b>Set Interaction Status</b> ::= Rt-Component Target Param+, Interaction Type Param, Interaction Status Param

##### 72.20.1 Interaction Status Param (**Paramètre d'état d'interaction**)

- Paramètre d'état d'interaction.
- Choix entre (Interaction Status, Interaction Status Macro).

##### 72.20.2 Interaction Status Macro (**Macro-paramètre d'état d'interaction**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'état d'interaction.
- Séquence de (Macro Def ID, Interaction Status?).

##### 72.20.3 Min Interact Required Param (**Paramètre d'interaction minimale requise**)

- Paramètre d'interaction minimale requise.
- Choix entre (Min Interact Required, Min Interact Required Macro).

##### 72.20.4 Min Interact Required Macro (**Macro-paramètre d'interaction minimale requise**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'interaction minimale requise.
- Séquence de (Macro Def ID, Min Interact Required?).

##### 72.20.5 Max Interact Required Param (**Paramètre d'interaction maximale requise**)

- Un entier plus grand ou égal à zéro pour indiquer l'interaction maximale requise pour un rt-composant.
- Choix entre (Max Interact Required, Max Interact Required Macro).

##### 72.20.6 Max Interact Required Macro (**Macro-paramètre d'interaction maximale requise**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'interaction maximale requise.
- Séquence de (Macro Def ID, Max Interact Required?).

##### 72.20.7 Set Interaction Ability (**Fixer capacité d'interaction**)

- Cette action peut avoir pour cible des rt-composants pour déclencher ou modifier leur comportement d'interaction.
- Séquence de (Rt-Component Target Param+, Interaction Type Param, Min Interact Required Param, Max Interact Required Param).

### 72.20.8 Set Interaction Status (**Fixer état d'interaction**)

- Cette action affecte une valeur à l'état de sélection d'un rt-composant.
- Séquence de (Rt-Component Target Param+, Interaction Type Param, Interaction Status Param).

### 72.21 Comportement de style des rt-composants

**Tableau 35/T.171 – Aperçu général du comportement de style des rt-composants**

**Set Style ::=** Rt-Component Target Param+, Catalogued Style Param, Additional Information Param?

**Catalogued Style Param ::=** Catalogued Style | Catalogued Style Macro

**Catalogued Style Macro ::=** Macro Def ID, Catalogued Style?

**Additional Information Param ::=** Generic Value Param

#### 72.21.1 Set Style (**Fixer style**)

- Spécifie un style à appliquer à une cible.
- Séquence de (Rt-Component Target Param+, Catalogued Style Param, Additional Information Param?).

#### 72.21.2 Catalogued Style Param (**Paramètre de style catalogué**)

- Paramètre de style catalogué.
- Choix entre (Catalogued Style, Catalogued Style Macro).

#### 72.21.3 Catalogued Style Macro (**Macro-paramètre de style catalogué**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de style catalogué.
- Séquence de (Macro Def ID, Catalogued Style?).

#### 72.21.4 Additional Information Param (**Paramètre d'informations supplémentaires**)

- Spécifie quelques informations supplémentaires nécessaires pour un style.
- Generic Value Param.

### 72.22 Comportement d'ancrage des rt-contenus

**Tableau 36/T.171 – Aperçu général du comportement d'ancrage des rt-contenus**

**Attach Anchor ::=** Rt-Content Target Param+, Anchor Spec Param+

**Anchor Spec Param ::=** Anchor Spec | Anchor Spec Macro

**Anchor Spec Macro ::=** Macro Def ID, Anchor Spec?

**Anchor Spec ::=** Anchor+, Update Command

**Anchor ::=** Rt-Content Reference | Evaluated Reference

#### 72.22.1 Attach Anchor (**Attacher ancre**)

- Attache des ancrs aux rt-contenus.
- Séquence de (Rt-Content Target Param+, Anchor Spec Param+).



#### 72.22.2 Anchor Spec Param (**Paramètre de spécification d'ancre**)

- Paramètre de spécification d'ancre.
- Choix entre (Anchor Spec, Anchor Spec Macro).

#### 72.22.3 Anchor Spec Macro (**Macro-paramètre de spécification d'ancre**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification d'ancre.
- Séquence de (Macro Def ID, Anchor Spec?).

#### 72.22.4 Anchor Spec (**Spécification d'ancre**)

- Une paire d'ancres et la commande de mise à jour.
- Séquence de (Anchor+, Update Command).

#### 72.22.5 Anchor (**Ancre**)

- Spécifie une ancre.
- Choix entre (Rt-Content Reference, Evaluated Reference).

### 72.23 Comportement de disponibilité des canaux

**Tableau 37/T.171 – Aperçu général du comportement de disponibilité des canaux**

**New Channel** ::= Channel Target Param+

**Delete Channel** ::= Channel Target Param+

#### 72.23.1 New Channel (**Nouveau canal**)

- Cette action crée des canaux.
- Liste de (Channel Target Param).

#### 72.23.2 Delete Channel (**Supprimer canal**)

- Cette action supprime des canaux.
- Liste de (Channel Target Param).

### 72.24 Comportement de perceptibilité des canaux

**Tableau 38/T.171 – Aperçu général du comportement de perceptibilité des canaux**

**Set Channel Perceptability** ::= Channel Target Param+, Channel Perceptability Param

**Channel Perceptability Param** ::= Channel Perceptability | Channel Perceptability Macro

**Channel Perceptability Macro** ::= Macro Def ID, Channel Perceptability?

#### 72.24.1 Set Channel Perceptability (**Fixer perceptibilité de canal**)

- Cette action modifie la perceptibilité du canal d'une cible.
- Séquence de (Channel Target Param+, Channel Perceptability Param).

#### 72.24.2 Channel Perceptability Param (**Paramètre de perceptibilité de canal**)

- Paramètre de perceptibilité d'un canal.
- Choix entre (Channel Perceptability, Channel Perceptability Macro).

### 72.24.3 Channel Perceptability Macro (**Macro-paramètre de perceptibilité de canal**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de perceptibilité de canal.
- Séquence de (Macro Def ID, Channel Perceptability?).

## 72.25 Comportement d'espace de présentation des canaux

Tableau 39/T.171 – Aperçu général du comportement d'espace de présentation des canaux

<p><b>Set CPS</b> ::= Channel Target Param+, CPS Initialisation Param?</p> <p><b>CPS Initialisation Param</b> ::= CPS Initialisation   CPS Initialisation Macro</p> <p><b>CPS Initialisation Macro</b> ::= Macro Def ID, CPS Initialisation?</p> <p><b>CPS Initialisation</b> ::= CPS Duration?, CPS Size?</p> <p><b>CPS Duration</b> ::= Integer   <i>#infinite</i> / Evaluated Integer</p> <p><b>CPS Size</b> ::= Size   Evaluated List</p>
---

### 72.25.1 Set CPS (**Fixer espace CPS**)

- Cette action spécifie un espace CPS de la cible.
- Séquence de (Channel Target Param+, CPS Initialisation Param?).

### 72.25.2 CPS Initialisation Param (**Paramètre d'initialisation d'espace CPS**)

- Paramètre d'initialisation d'espace CPS.
- Choix entre (CPS Initialisation, CPS Initialisation Macro).

### 72.25.3 CPS Initialisation Macro (**Macro-paramètre d'initialisation d'espace CPS**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'initialisation d'espace CPS.
- Séquence de (Macro Def ID, CPS Initialisation?).

### 72.25.4 CPS Initialisation (**Initialisation d'espace CPS**)

- Spécifie la durée et la taille de l'espace CPS.
- Séquence de (CPS Duration?, CPS Size?).

### 72.25.5 CPS Duration (**Durée CPS**)

- Spécifie la durée de l'espace CPS.
- Choix entre (Integer, *infinite*, Evaluated Integer).

### 72.25.6 CPS Size (**Taille CPS**)

- Spécifie la taille spatiale de l'espace CPS.
- Choix entre (Size, Evaluated List).

## 72.26 Comportement d'événement des canaux et des rt-composants

Tableau 40/T.171 – Aperçu général du comportement d'événement des canaux et des rt-composants

<p><b>Set Event</b> ::= Rt-Component Channel Tg Param+, Event Param, Event Data Param?</p> <p><b>Event Param</b> ::= Generic List Param</p> <p><b>Event Data Param</b> ::= Generic Value Param</p>
--

### 72.26.1 Set Event (**Fixer événement**)

- L'attribut Event Identifier (identificateur d'événement) de la cible est mis à la valeur spécifiée.
- Séquence de (Rt-Component Channel Tg Param+, Event Param, Event Data Param?).

### 72.26.2 Event Param (**Paramètre d'événement**)

- Spécifie un événement.
- Generic List Param.

### 72.26.3 Event Data Param (**Paramètre de données d'événement**)

- Spécifie des données associées à un événement.
- Generic Value Param.

## 73 Actions élémentaires

### 73.1 Liste d'actions élémentaires

**Tableau 41/T.171 – Aperçu général de la liste d'actions élémentaires**

**Elementary Action ::=**

Set Event | Set CPS | Set Channel Perceptability | New Channel | Delete Channel | Attach Anchor | Get Style | Set Style | Set Interaction Ability | Set Interaction Status | Set Stream Choice | Set CV | Set GVF | Set Aspect Ratio | Set Resizing Strategy | Set OVS | Set OAP | Set OVS Position | Set PAP | Set PVS Position | Set GSF | Set User Spatial Control | Set OVD | Set CTP | Set Temporal Termination | Set PVD Position | Set GTF | Set Timestones | Set Perceptability | Set Presentation Priority | Set RPS Assignment | Plug | Set Parameters | Run | Stop | New | Delete | Copy | Set Data | Add | Subtract | Link Abort | Activate | Deactivate | Prepare | Destroy | Catalogued Elementary Action | Set Catalogued Attribute | Set Alias | Return | Delay | Extensibility Provision

#### 73.1.1 Elementary Action (**Action élémentaire**)

- La présente Recommandation définit une liste d'actions élémentaires qui peuvent être incluses dans un objet *action* pour modifier le comportement d'objets MHEG.
- Choix entre (Set Event, Set CPS, Set Channel Perceptability, New Channel, Delete Channel, Attach Anchor, Get Style, Set Style, Set Interaction Ability, Set Interaction Status, Set Stream Choice, Set CV, Set GVF, Set Aspect Ratio, Set Resizing Strategy, Set OVS, Set OAP, Set OVS Position, Set PAP, Set PVS Position, Set GSF, Set User Spatial Control, Set OVD, Set CTP, Set Temporal Termination, Set PVD Position, Set GTF, Set Timestones, Set Perceptability, Set Presentation Priority, Set RPS Assignment, Plug, Set Parameters, Run, Stop, New, Delete, Copy, Set Data, Add, Subtract, Link Abort, Activate, Deactivate, Prepare, Destroy, Catalogued Elementary Action, Set Catalogued Attribute, Set Alias, Return, Delay, Extensibility Provision).

## 73.2 Entités MHEG, données, flux et macro-paramètres; définitions utiles concernant l'identification

Tableau 42/T.171 – Aperçu général des entités MHEG, des données, des flux, des macro-paramètres et de définitions utiles concernant l'identification

<b>Using-Application</b> ::= <i>\$NULL</i>
<b>Alias</b> ::= String
<b>Identifiant</b> ::= Integer
<b>Index</b> ::= Integer
<b>Tail Param</b> ::= Tail   Tail Macro
<b>Tail Macro</b> ::= Macro Def ID, Tail?
<b>Tail</b> ::= Index*
<b>External ID</b> ::= Public ID?, System ID?
<b>Public ID</b> ::= <i>\$PrintableString</i>
<b>System ID</b> ::= <i>\$OCTET STRING</i>
<b>Null-Data</b> ::= <i>\$NULL</i>
<b>Mheg ID</b> ::= Application ID?, Mh Number
<b>Application ID</b> ::= <i>\$OCTET STRING</i>
<b>Mh Number</b> ::= Identifiant
<b>Null-Mh</b> ::= <i>\$NULL</i>
<b>Stream ID</b> ::= Tail
<b>Rt-Number</b> ::= Identifiant
<b>Null-Root-Rt</b> ::= <i>\$NULL</i>
<b>Socket Identification</b> ::= Socket ID   Alias
<b>Socket ID</b> ::= Root Rt-Composite Reference, Tail
<b>Channel ID</b> ::= Identifiant
<b>Default-Channel</b> ::= <i>\$NULL</i>
<b>Macro Def ID</b> ::= String   Integer

### 73.2.1 Using-Application (**Application utilisatrice**)

- L'application utilisatrice est identifiée par un identificateur réservé: «Using-Application» (application utilisatrice).
- NULL.

### 73.2.2 Alias (**Alias**)

- Identificateur d'une entité MHEG, d'une donnée ou d'un flux. Il s'agit d'une séquence de caractères alphanumériques utilisée en remplacement de l'autre mécanisme d'adressage. Il appartient à l'application utilisatrice d'établir une correspondance entre ce nom et une adresse physique.
- String.

### 73.2.3 Identifiant (**Identificateur**)

- Identificateur d'une entité MHEG.
- Integer.

#### **73.2.4 Index (Indice)**

- Identificateur d'un élément interne. Les indices sont numérotés de 1 à n par défaut.
- Integer.

#### **73.2.5 Tail Param (Paramètre de postambule)**

- Paramètre d'un postambule.
- Choix entre (Tail, Tail Macro).

#### **73.2.6 Tail Macro (Macro-paramètre de postambule)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de postambule.
- Séquence de (Macro Def ID, Tail?).

#### **73.2.7 Tail (Postambule)**

- Identificateur d'un élément interne dans une hiérarchie. La liste d'indices désigne un chemin partant d'une entité externe et menant à un élément interne.
- Liste de (Index).

#### **73.2.8 External ID (Identificateur externe)**

- Identificateur d'une information extérieure (par exemple un objet MHEG, une donnée de contenu) comme prévu par l'ISO 9070 et l'ISO 8879.

La technique de référence à une information extérieure est fondée sur:

Les techniques utilisées dans l'ISO 8879 pour les «identificateurs publics formels».

L'ISO 9070, intitulée «Procédures d'enregistrement pour identificateurs de propriétaire de texte public», qui fournit la notation ASN.1 et SGML pour la représentation des identificateurs de propriétaire enregistrés.

La présence de l'un des deux champs est facultative.

- Séquence de (Public ID?, System ID?).

#### **73.2.9 Public ID (Identificateur public)**

- Chaîne de caractères dont la syntaxe est décrite au 10.2 de l'ISO 8879:1986.  
La présente Recommandation utilise une forme simplifiée de l'Annexe D (informative) de l'ISO 9070 pour le codage de l'identificateur public.
- PrintableString.

#### **73.2.10 System ID (Identificateur de système)**

- L'identificateur de système est un moyen, propre au système, qui permet d'identifier un fichier particulier dans ledit système. Il peut également être utilisé pour désigner le système fournissant l'objet. Cette information peut servir à décrire une demande d'information adressée à une base de données ou à un système de télécommunication.
- OCTET STRING.

#### **73.2.11 Null-Data (Données vides)**

- Les données vides sont identifiées par un identificateur réservé: «Null-Data» (données vides).
- NULL.

#### **73.2.12 MHEG ID (Identificateur MHEG)**

- Identificateur d'un objet MHEG transféré.
- Séquence de (Application ID?, MH Number).

#### **73.2.13 Application ID (Identificateur d'application)**

- Identificateur de l'application fournissant l'objet. La sémantique de l'identification est définie par l'application utilisatrice.
- OCTET STRING.

#### **73.2.14 MH Number (Numéro d'objet MHEG)**

- Identificateur unique de l'objet dans l'application. Un numéro d'objet MHEG réservé est utilisé pour désigner un objet vide: «Null-MH» (objet MHEG vide).
- Identifier.

#### **73.2.15 Null-MH (Objet MHEG vide)**

- L'objet MHEG vide est identifié par un identificateur réservé: «Null-MH» (objet MHEG vide).
- NULL.

#### **73.2.16 Stream ID (Identificateur de flux)**

- Identificateur unique d'un flux dans un multiplex.
- Tail.

#### **73.2.17 Rt Number (Numéro de rt-objet)**

- Identificateur unique d'un rt-objet racine créé à partir d'un objet modèle donné.
- Identifier.

#### **73.2.18 Null-Root-Rt (Rt-objet racine vide)**

- Le rt-objet racine vide est identifié par un identificateur réservé: «Null-Root-Rt» (rt-objet racine vide).
- NULL.

#### **73.2.19 Socket Identification (Identification de réceptacle)**

- Identification d'un réceptacle.
- Choix entre (Socket ID, Alias).

#### **73.2.20 Socket ID (Identificateur de réceptacle)**

- Identificateur d'un réceptacle, c'est-à-dire un élément de rt-composite.
- Séquence de (Root Rt-Composite Reference, Tail).

#### **73.2.21 Channel ID (Identificateur de canal)**

- Identificateur d'un canal.
- Identifier.

#### **73.2.22 Default-Channel (Canal par défaut)**

- Le canal par défaut est identifié par un identificateur réservé: «Default-Channel» (canal par défaut).
- NULL.

#### **73.2.23 Macro Def ID (Identificateur de macro-définition)**

- Identificateur d'un macro-paramètre dans une action.
- Choix entre (String, Integer).

### 73.3 Définitions utiles concernant les références

Tableau 43/T.171 – Aperçu général de définitions utiles concernant les références

<p><b>Reference ::=</b> Null-Data   Mheg ID Reference   External ID Reference   Container Element Reference   Null-Mh   This   Stream ID Reference   Root Rt-ID Reference   Null-Root-Rt   Socket ID Reference   Channel ID Reference   Default-Channel   Alias Reference</p> <p><b>Alias Reference ::=</b> Alias</p> <p><b>Mheg ID Reference ::=</b> Mheg ID</p> <p><b>External ID Reference ::=</b> External ID</p> <p><b>Tail Reference ::=</b> Tail?, Tail Complement?</p> <p><b>Tail Complement ::=</b> #children / #descendants</p> <p><b>Data Reference ::=</b> External ID   Null-Data   Alias Reference</p> <p><b>Mh-Reference ::=</b> Mheg ID Reference   External ID Reference   Container Element Reference   Null-Mh   This   Alias Reference</p> <p><b>Action Object Reference ::=</b> Mh-Reference</p> <p><b>Link Object Reference ::=</b> Mh-Reference</p> <p><b>Model Object Reference ::=</b> Mh-Reference</p> <p><b>Script Object Reference ::=</b> Mh-Reference</p> <p><b>Component Object Reference ::=</b> Mh-Reference</p> <p><b>Content Object Ref ::=</b> Mh-Reference</p> <p><b>Non-Mux Content Object Ref ::=</b> Mh-Reference</p> <p><b>Multiplexed Content Object Ref ::=</b> Mh-Reference</p> <p><b>Stream ID Reference ::=</b> Stream ID</p> <p><b>Composite Object Reference ::=</b> Mh-Reference</p> <p><b>Container Object Reference ::=</b> Mh-Reference</p> <p><b>Container Element Reference ::=</b> Container Object Reference, Tail Reference</p> <p><b>This ::=</b> §NULL</p> <p><b>Descriptor Object Reference ::=</b> Mh-Reference</p> <p><b>Root Rt-Reference ::=</b> Root Rt-ID Reference   Null-Root-Rt   Alias Reference</p> <p><b>Root Rt-ID Reference ::=</b> Model Object Reference, Rt-Number Reference</p> <p><b>Rt-Number Reference ::=</b> Rt-Number   #question-mark / #star</p> <p><b>Rt-Reference ::=</b> Root Rt-ID Reference   Null-Root-Rt   Socket ID Reference   Alias Reference</p> <p><b>Rt-Script Reference ::=</b> Rt-Script ID Reference   Null-Root-Rt   Alias Reference</p> <p><b>Rt-Script ID Reference ::=</b> Script Object Reference, Rt-Number Reference</p> <p><b>Root Rt-Component Reference ::=</b> Root Rt-Component ID Ref   Null-Root-Rt   Alias Reference</p> <p><b>Root Rt-Component ID Ref ::=</b> Component Object Reference, Rt-Number Reference</p> <p><b>Rt-Component Reference ::=</b> Root Rt-Component ID Ref   Null-Root-Rt   Socket ID Reference   Alias Reference</p> <p><b>Root Rt-Content Reference ::=</b> Root Rt-Content ID Ref   Null-Root-Rt   Alias Reference</p> <p><b>Root Rt-Content ID Ref ::=</b> Content Object Ref, Rt-Number Reference</p>
--

**Tableau 43/T.171 – Aperçu général de définitions utiles concernant les références (fin)**

<b>Rt-Content Reference</b> ::= Root Rt-Content ID Ref   Null-Root-Rt   Socket ID Reference   Alias Reference
<b>Root Rt-Mux Reference</b> ::= Root Rt-Mux ID Reference   Null-Root-Rt   Alias Reference
<b>Root Rt-Mux ID Reference</b> ::= Multiplexed Content Object Ref, Rt-Number Reference
<b>Rt-Mux Reference</b> ::= Root Rt-Mux ID Reference   Null-Root-Rt   Socket ID Reference   Alias Reference
<b>Root Rt-Composite Reference</b> ::= Root Rt-Composite ID Ref   Null-Root-Rt   Alias Reference
<b>Root Rt-Composite ID Ref</b> ::= Composite Object Reference, Rt-Number Reference
<b>Rt-Composite Reference</b> ::= Root Rt-Composite ID Ref   Null-Root-Rt   Socket ID Reference   Alias Reference
<b>Socket Reference</b> ::= Socket ID Reference   Alias Reference
<b>Socket ID Reference</b> ::= Root Rt-Composite Reference, Socket Tail Reference
<b>Socket Tail Reference</b> ::= Tail?, Socket Tail Complement?
<b>Socket Tail Complement</b> ::= Tail Complement   #question-mark-child   #question-mark-descendant
<b>Channel Reference</b> ::= Channel ID Reference   Default-Channel   Alias Reference
<b>Channel ID Reference</b> ::= Channel ID
<b>Rt-Component Channel Ref</b> ::= Rt-Component Reference   Channel Reference

### 73.3.1 Reference (**Référence**)

- Une référence est un moyen de désigner une entité MHEG, une donnée ou un flux.
- Choix entre (Null-Data, Mheg ID Reference, External ID Reference, Container Element Reference, Null-MH, This, Stream ID Reference, Root Rt ID Reference, Null-Root-Rt, Socket ID Reference, Channel ID Reference, Default-Channel, Alias Reference).

### 73.3.2 Alias Reference (**Référence à un alias**)

- Référence à une entité MHEG, à une donnée ou à un flux simple ou multiple, utilisant les alias affectés.
- Alias.

### 73.3.3 Mheg ID Reference (**Référence à un identificateur d'objet MHEG**)

- Référence à un objet MHEG, utilisant l'identificateur externe de l'objet.
- Mheg ID.

### 73.3.4 External ID Reference (**Référence à un identificateur externe**)

- Référence à une entité MHEG ou à une donnée, utilisant leur identificateur externe.
- External ID.

### 73.3.5 Tail Reference (**Référence à un postambule**)

- Référence à un élément donné, utilisant l'identificateur de postambule de l'élément; un complément de postambule peut être fourni. Le postambule peut être omis, auquel cas le complément de postambule est appliqué à l'entité externe elle-même.
- Séquence de (Tail?, Tail Complement?).

### 73.3.6 Tail Complement (**Complément de postambule**)

- Le complément de postambule permet de faire référence aux enfants (une génération) ou aux descendants (plusieurs générations) de l'élément référencé en utilisant son postambule ou de l'entité externe si le postambule est omis.
- Choix entre (children, descendants).



### 73.3.7 Data Reference (**Référence aux données**)

- Référence à une donnée d'objet *contenu* ou *script*, utilisant un de leurs identificateurs possibles.
- Choix entre (External ID, Null-Data, Alias Reference).

### 73.3.8 MH-Reference (**Référence à un objet MHEG**)

- Référence à un objet MHEG, utilisant un des identificateurs possibles de l'objet.
- Choix entre (Mheg ID Reference, External ID Reference, Container Element Reference, Null-MH, This, Alias Reference).

### 73.3.9 Action Object Reference (**Référence à un objet action**)

- Référence à un objet *action*.
- MH-Reference.

### 73.3.10 Link Object Reference (**Référence à un objet lien**)

- Référence à un objet *lien*.
- MH-Reference.

### 73.3.11 Model Object Reference (**Référence à un objet modèle**)

- Référence à un objet modèle. Lorsque l'objet modèle à référencer est un objet composite; «this» (cela) peut être utilisé dans les objets *lien* et *action* comme moyen d'adressage local dans l'objet *composite* ou *conteneur*.
- MH-Reference.

### 73.3.12 Script Object Reference (**Référence à un objet script**)

- Référence à un objet *script*.
- MH-Reference.

### 73.3.13 Component Object Reference (**Référence à un objet composant**)

- Référence à un objet *composant*.
- MH-Reference.

### 73.3.14 Content Object Ref (**Référence à un objet contenu**)

- Référence à un objet *contenu* ou à un objet *contenu multiplexé*.
- MH-Reference.

### 73.3.15 Non-Mux Content Object Ref (**Référence à un objet contenu non multiplexé**)

- Référence à un objet *contenu non multiplexé*, c'est-à-dire à un objet *contenu* pur.
- MH-Reference.

### 73.3.16 Multiplexed Content Object Ref (**Référence à un objet contenu multiplexé**)

- Référence à un objet *contenu multiplexé*.
- MH-Reference.

### 73.3.17 Stream ID Reference (**Référence à un identificateur de flux**)

- Référence à un flux, utilisant l'identificateur du flux.
- Stream ID.

### 73.3.18 Composite Object Reference (**Référence à un objet composite**)

- Référence à un objet *composite*; «this» (cela) peut être utilisé dans les objets *lien* et *action* comme moyen d'adressage local dans l'objet *composite*.
- MH-Reference.

- 73.3.19** Container Object Reference (**Référence à un objet conteneur**)
- Référence à un objet *conteneur*; «this» (cela) peut être utilisé dans les objets *lien* et *action* comme moyen d'adressage local dans l'objet *composite* ou *conteneur*.
  - MH-Reference.
- 73.3.20** Container Element Reference (**Référence à un élément de conteneur**)
- Référence à un élément de conteneur, utilisant une référence à l'objet *conteneur* définissant cet élément, suivie d'une référence à l'élément lui-même.
  - Séquence de (Container Object Reference, Tail Reference).
- 73.3.21** This (**Cela**)
- Lorsque l'entité MHEG à référencer est un objet *composite* ou *conteneur*, «this» (cela) peut être utilisé dans les objets *lien* et *action* comme moyen d'adressage local dans l'objet *composite* ou *conteneur*.
  - NULL.
- 73.3.22** Descriptor Object Reference (**Référence à un objet descripteur**)
- Référence à un objet *descripteur*.
  - MH-Reference.
- 73.3.23** Root Rt-Reference (**Référence à un rt-objet racine**)
- Référence à un rt-objet racine, utilisant un des identificateurs possibles du rt-objet racine.
  - Choix entre (Root Rt ID Reference, Null-Root-Rt, Alias Reference).
- 73.3.24** Root Rt-ID Reference (**Référence à un identificateur de rt-objet racine**)
- Référence à un identificateur de rt-objet racine, utilisant une référence à l'objet modèle à partir duquel le rt-objet racine est créé, suivie du rt-numéro qui est affecté au rt-objet racine.
  - Séquence de (Model Object Reference, Rt Number Reference).
- 73.3.25** Rt-Number Reference (**Référence à un numéro de rt-objet**)
- Référence à un rt-objet racine spécifié (numéro de rt-objet) ou à un rt-objet racine déterminé d'une manière dynamique (?) ou à tous les rt-objets racines créés à partir d'un objet modèle donné (\*).
  - Choix entre (Rt Number, question-mark, star).
- 73.3.26** Rt-Reference (**Référence à un rt-objet**)
- Référence à un rt-objet, c'est-à-dire soit une racine (rt-script, rt-composant racine) soit un réceptacle.
  - Choix entre (Root Rt ID Reference, Null-Root-Rt, Socket ID Reference, Alias Reference).
- 73.3.27** Rt-Script Reference (**Référence à un rt-script**)
- Référence à un rt-script, utilisant un des identificateurs possibles du rt-script.
  - Choix entre (Rt-Script ID Reference, Null-Root-Rt, Alias Reference).
- 73.3.28** Rt-Script ID Reference (**Référence à un identificateur de rt-script**)
- Référence à un identificateur de rt-script, utilisant une référence à l'objet modèle à partir duquel le rt-script est créé, suivie du rt-numéro qui est affecté au rt-script.
  - Séquence de (Script Object Reference, Rt Number Reference).
- 73.3.29** Root Rt-Component Reference (**Référence à un rt-composant racine**)
- Référence à un rt-composant racine, utilisant un des identificateurs possibles du rt-composant racine.
  - Choix entre (Root Rt-Component ID Ref, Null-Root-Rt, Alias Reference).
- 73.3.30** Root Rt-Component ID Ref (**Référence à un identificateur de rt-composant racine**)
- Référence à un identificateur de rt-composant racine, utilisant une référence à l'objet modèle à partir duquel le rt-composant racine est créé, suivie du rt-numéro qui est affecté au rt-composant racine.
  - Séquence de (Component Object Reference, Rt Number Reference).

- 73.3.31** Rt-Component Reference (**Référence à un rt-composant**)
- Référence à un rt-composant, c'est-à-dire soit une racine soit un réceptacle.
  - Choix entre (Root Rt-Component ID Ref, Null-Root-Rt, Socket ID Reference, Alias Reference).
- 73.3.32** Root Rt-Content Reference (**Référence à un rt-contenu racine**)
- Référence à un rt-contenu racine, un des identificateurs possibles du rt-contenu racine.
  - Choix entre (Root Rt-Content ID Ref, Null-Root-Rt, Alias Reference).
- 73.3.33** Root Rt-Content ID Ref (**Référence à un identificateur de rt-contenu racine**)
- Référence à un identificateur de rt-contenu racine, utilisant une référence à l'objet modèle à partir duquel le rt-contenu racine est créé, suivie du rt-numéro qui est affecté au rt-contenu racine.
  - Séquence de (Content Object Ref, Rt Number Reference).
- 73.3.34** Rt-Content Reference (**Référence à un rt-contenu**)
- Référence à un rt-contenu, c'est-à-dire soit une racine soit un réceptacle.
  - Choix entre (Root Rt-Content ID Ref, Null-Root-Rt, Socket ID Reference, Alias Reference).
- 73.3.35** Root Rt-Mux Reference (**Référence à un rt-mux racine**)
- Référence à un rt-mux racine, utilisant un des identificateurs possibles du rt-mux racine.
  - Choix entre (Root Rt-Mux ID Reference, Null-Root-Rt, Alias Reference).
- 73.3.36** Root Rt-Mux ID Reference (**Référence à un identificateur de rt-mux racine**)
- Référence à un identificateur de rt-mux racine, utilisant une référence à l'objet modèle à partir duquel le rt-mux racine est créé, suivie du rt-numéro qui est affecté au rt-mux racine.
  - Séquence de (Multiplexed Content Object Ref, Rt Number Reference).
- 73.3.37** Rt-Mux Reference (**Référence à un rt-mux**)
- Référence à un rt-mux, c'est-à-dire soit une racine soit un réceptacle.
  - Choix entre (Root Rt-Mux ID Reference, Null-Root-Rt, Socket ID Reference, Alias Reference).
- 73.3.38** Root Rt-Composite Reference (**Référence à un rt-composite racine**)
- Référence à un rt-composite racine, utilisant un des identificateurs possibles du rt-composite racine.
  - Choix entre (Root Rt-Composite ID Ref, Null-Root-Rt, Alias Reference).
- 73.3.39** Root Rt-Composite ID Ref (**Référence à un identificateur de rt-composite racine**)
- Référence à un identificateur de rt-composite racine, utilisant une référence à l'objet modèle à partir duquel le rt-composite racine est créé, suivie du rt-numéro qui est affecté au rt-composite racine.
  - Séquence de (Composite Object Reference, Rt Number Reference).
- 73.3.40** Rt-Composite Reference (**Référence à un rt-composite**)
- Référence à un rt-composite, c'est-à-dire soit une racine soit un réceptacle.
  - Choix entre (Root Rt-Composite ID Ref, Null-Root-Rt, Socket ID Reference, Alias Reference).
- 73.3.41** Socket Reference (**Référence à un réceptacle**)
- Référence à un réceptacle, utilisant un des identificateurs possibles du réceptacle. Il n'existe pas de référence spécifique à un réceptacle vide, car la référence à un tel réceptacle est la même que pour tout autre réceptacle.
  - Choix entre (Socket ID Reference, Alias Reference).

#### 73.3.42 Socket ID Reference (**Référence à un identificateur de réceptacle**)

- Référence à un identificateur de réceptacle, utilisant une référence au rt-composite définissant ce réceptacle suivie d'une référence au réceptacle lui-même. Un réceptacle est un élément de rt-composite.
- Séquence de (Root Rt-Composite Reference, Socket Tail Reference).

#### 73.3.43 Socket Tail Reference (**Référence à un postambule de réceptacle**)

- Référence à un réceptacle donné, utilisant l'identificateur de postambule du réceptacle; un complément de postambule peut être fourni. Le postambule peut être omis, auquel cas le complément de postambule du réceptacle est appliqué au rt-composite racine lui-même.
- Séquence de (Tail?, Socket Tail Complement?).

#### 73.3.44 Socket Tail Complement (**Complément de postambule de réceptacle**)

- Le complément de postambule de réceptacle permet de faire référence aux enfants ou aux descendants ou à un seul réceptacle déterminé de manière dynamique parmi les enfants (?enfant), ou à un seul réceptacle déterminé de manière dynamique parmi les descendants (?descendant) du réceptacle auquel il est fait référence au moyen du postambule ou du rt-composite racine si le postambule est omis.
- Choix entre (Tail Complement, question-mark-child, question-mark-descendant).

#### 73.3.45 Channel Reference (**Référence à un canal**)

- Référence à un canal, utilisant un des identificateurs possibles du canal.
- Choix entre (Channel ID Reference, Default-Channel, Alias Reference).

#### 73.3.46 Channel ID Reference (**Référence à un identificateur de canal**)

- Référence à un canal, utilisant l'identificateur du canal.
- Channel ID.

#### 73.3.47 Rt-Component Channel Ref (**Référence à un canal de rt-composant**)

- Référence à un rt-composant ou à un canal utilisant un des identificateurs possibles du rt-composant.
- Choix entre (Rt-Component Reference, Channel Reference).

### 73.4 Définitions utiles concernant les cibles

Tableau 44/T.171 – Aperçu général de définitions utiles concernant les cibles

<b>Target Param</b> ::= Target   Target Macro
<b>Target Macro</b> ::= Macro Def ID, Target?
<b>Target</b> ::= Generic Reference
<b>Mh-Target Param</b> ::= Mh-Target   Mh-Target Macro
<b>Mh-Target Macro</b> ::= Macro Def ID, Mh-Target?
<b>Mh-Target</b> ::= Mh-Reference   Evaluated Reference
<b>Link Target Param</b> ::= Mh-Target Param
<b>Content Target Param</b> ::= Mh-Target Param
<b>Rt-Target Param</b> ::= Rt-Target   Rt-Target Macro
<b>Rt-Target Macro</b> ::= Macro Def ID, Rt-Target?
<b>Rt-Target</b> ::= Rt-Reference   Evaluated Reference
<b>Rt-Script Target Param</b> ::= Rt-Script Target   Rt-Script Target Macro
<b>Rt-Script Target Macro</b> ::= Macro Def ID, Rt-Script Target?
<b>Rt-Script Target</b> ::= Rt-Script Reference   Evaluated Reference

**Tableau 44/T.171 – Aperçu général de définitions utiles concernant les cibles (*fin*)**

<b>Rt-Component Target Param</b> ::= Rt-Component Target   Rt-Component Target Macro
<b>Rt-Component Target Macro</b> ::= Macro Def ID, Rt-Component Target?
<b>Rt-Component Target</b> ::= Rt-Component Reference   Evaluated Reference
<b>Rt-Content Target Param</b> ::= Rt-Content Target   Rt-Content Target Macro
<b>Rt-Content Target Macro</b> ::= Macro Def ID, Rt-Content Target?
<b>Rt-Content Target</b> ::= Rt-Content Reference   Evaluated Reference
<b>Rt-Mux Target Param</b> ::= Rt-Mux Target   Rt-Mux Target Macro
<b>Rt-Mux Target Macro</b> ::= Macro Def ID, Rt-Mux Target?
<b>Rt-Mux Target</b> ::= Rt-Mux Reference   Evaluated Reference
<b>Rt-Composite Target Param</b> ::= Rt-Composite Target   Rt-Composite Target Macro
<b>Rt-Composite Target Macro</b> ::= Macro Def ID, Rt-Composite Target?
<b>Rt-Composite Target</b> ::= Rt-Composite Reference   Evaluated Reference
<b>Socket Target Param</b> ::= Socket Target   Socket Target Macro
<b>Socket Target Macro</b> ::= Macro Def ID, Socket Target?
<b>Socket Target</b> ::= Socket Reference   Evaluated Reference
<b>Return Target Param</b> ::= Generic Integer Param   Using-Application
<b>Channel Target Param</b> ::= Channel Target   Channel Target Macro
<b>Channel Target Macro</b> ::= Macro Def ID, Channel Target?
<b>Channel Target</b> ::= Channel Reference   Evaluated Reference
<b>Rt-Component Channel Tg Param</b> ::= Rt-Component Channel Tg   Rt-Component Channel Tg Macro
<b>Rt-Component Channel Tg Macro</b> ::= Macro Def ID, Rt-Component Channel Tg?
<b>Rt-Component Channel Tg</b> ::= Rt-Component Channel Ref   Evaluated Reference

#### 73.4.1 Target Param (**Paramètre de cible**)

- Paramètre d'une cible.
- Choix entre (Target, Target Macro).

#### 73.4.2 Target Macro (**Macro-paramètre de cible**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de cible.
- Séquence de (Macro Def ID, Target?).

#### 73.4.3 Target (**Cible**)

- L'attribut Target (cible) d'une action pointe vers une entité MHEG en utilisant la référence de l'entité.
- Generic Reference.

#### 73.4.4 MH-Target Param (**Paramètre de MH-cible**)

- Paramètre de MH-cible.
- Choix entre (MH-Target, MH-Target Macro).

#### 73.4.5 MH-Target Macro (**Macro-paramètre de MH-cible**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de MH-cible.
- Séquence de (Macro Def ID, MH-Target?).

#### 73.4.6 MH-Target (**MH-cible**)

- Le MH-cible d'une action pointe vers un objet MHEG en utilisant la référence de l'objet.
- Choix entre (MH-Reference, Evaluated Reference).

#### 73.4.7 Link Target Param (**Paramètre de lien cible**)

- Le lien cible d'une action pointe vers un objet *lien* en utilisant la référence de l'objet.
- MH-Target Param.

#### 73.4.8 Content Target Param (**Paramètre de contenu cible**)

- Le contenu cible d'une action pointe vers un objet *contenu* en utilisant la référence de l'objet.
- MH-Target Param.

#### 73.4.9 Rt-Target Param (**Paramètre de rt-cible**)

- Paramètre d'une rt-cible.
- Choix entre (Rt-Target, Rt-Target Macro).

#### 73.4.10 Rt-Target Macro (**Macro-paramètre de rt-cible**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de rt-cible.
- Séquence de (Macro Def ID, Rt-Target?).

#### 73.4.11 Rt-Target (**Rt-cible**)

- La rt-cible d'une action pointe vers un rt-objet en utilisant la référence de l'objet.
- Choix entre (Rt-Reference, Evaluated Reference).

#### 73.4.12 Rt-Script Target Param (**Paramètre de rt-script cible**)

- Paramètre d'une cible de rt-script.
- Choix entre (Rt-Script Target, Rt-Script Target Macro).

#### 73.4.13 Rt-Script Target Macro (**Macro-paramètre de rt-script cible**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de rt-script cible.
- Séquence de (Macro Def ID, Rt-Script Target?).

#### 73.4.14 Rt-Script Target (**Rt-script cible**)

- Le rt-script cible d'une action pointe vers un objet *rt-script* en utilisant la référence de l'objet.
- Choix entre (Rt-Script Reference, Evaluated Reference).

#### 73.4.15 Rt-Component Target Param (**Paramètre de rt-composant cible**)

- Paramètre d'un rt-composant cible.
- Choix entre (Rt-Component Target, Rt-Component Target Macro).

#### 73.4.16 Rt-Component Target Macro (**Macro-paramètre de rt-composant cible**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de rt-composant cible.
- Séquence de (Macro Def ID, Rt-Component Target?).

#### 73.4.17 Rt-Component Target (**Rt-composant cible**)

- Le rt-composant cible d'une action pointe vers un objet *rt-composant* en utilisant la référence de l'objet.
- Choix entre (Rt-Component Reference, Evaluated Reference).

#### 73.4.18 Rt-Content Target Param (**Paramètre de rt-contenu cible**)

- Paramètre d'un rt-contenu cible.
- Choix entre (Rt-Content Target, Rt-Content Target Macro).

- 73.4.19** Rt-Content Target Macro (**Macro-paramètre de rt-contenu cible**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de rt-contenu cible.
  - Séquence de (Macro Def ID, Rt-Content Target?).
- 73.4.20** Rt-Content Target (**Rt-contenu cible**)
- Le rt-contenu cible d'une action pointe vers un objet *rt-contenu* en utilisant la référence de l'objet.
  - Choix entre (Rt-Content Reference, Evaluated Reference).
- 73.4.21** Rt-Mux Target Param (**Paramètre de rt-mux cible**)
- Paramètre d'un rt-mux cible.
  - Choix entre (Rt-Mux Target, Rt-Mux Target Macro).
- 73.4.22** Rt-Mux Target Macro (**Macro-paramètre de rt-mux cible**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de rt-mux cible.
  - Séquence de (Macro Def ID, Rt-Mux Target?).
- 73.4.23** Rt-Mux Target (**Rt-mux cible**)
- Le rt-mux cible d'une action pointe vers un objet *rt-mux* en utilisant la référence de l'objet.
  - Choix entre (Rt-Mux Reference, Evaluated Reference).
- 73.4.24** Rt-Composite Target Param (**Paramètre de rt-composite cible**)
- Paramètre d'un rt-composite cible.
  - Choix entre (Rt-Composite Target, Rt-Composite Target Macro).
- 73.4.25** Rt-Composite Target Macro (**Macro-paramètre de rt-composite cible**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de rt-composite cible.
  - Séquence de (Macro Def ID, Rt-Composite Target?).
- 73.4.26** Rt-Composite Target (**Rt-composite cible**)
- Le rt-composite cible d'une action pointe vers un objet *rt-composite* en utilisant la référence de l'objet.
  - Choix entre (Rt-Composite Reference, Evaluated Reference).
- 73.4.27** Socket Target Param (**Paramètre de réceptacle cible**)
- Paramètre d'un réceptacle cible.
  - Choix entre (Socket Target, Socket Target Macro).
- 73.4.28** Socket Target Macro (**Macro-paramètre de réceptacle cible**)
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de réceptacle cible.
  - Séquence de (Macro Def ID, Socket Target?).
- 73.4.29** Socket Target (**Réceptacle cible**)
- Le réceptacle cible d'une action pointe vers un réceptacle en utilisant la référence du réceptacle.
  - Choix entre (Socket Reference, Evaluated Reference).
- 73.4.30** Return Target Param (**Paramètre de cible de retour**)
- Le cas échéant, il appartient à l'application d'établir la correspondance entre le paramètre et le destinataire approprié.
  - Choix entre (Generic Integer Param, Using-Application).
- 73.4.31** Channel Target Param (**Paramètre de canal cible**)
- Paramètre d'un canal cible.
  - Choix entre (Channel Target, Channel Target Macro).

#### 73.4.32 Channel Target Macro (**Macro-paramètre de canal cible**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de canal cible.
- Séquence de (Macro Def ID, Channel Target?).

#### 73.4.33 Channel Target (**Canal cible**)

- Le canal cible d'une action pointe vers un canal en utilisant la référence du canal.
- Choix entre (Channel Reference, Evaluated Reference).

#### 73.4.34 Rt-Component Channel Tg Param (**Paramètre de canal cible de rt-composant**)

- Paramètre du canal cible d'un rt-composant.
- Choix entre (Rt-Component Channel Tg, Rt-Component Channel Tg Macro).

#### 73.4.35 Rt-Component Channel Tg Macro (**Macro-paramètre de canal cible de rt-composant**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de canal cible de rt-composant.
- Séquence de (Macro Def ID, Rt-Component Channel Tg?).

#### 73.4.36 Rt-Component Channel Tg (**Canal cible de rt-composant**)

- Le rt-composant ou canal cible d'une action pointe vers un rt-composant ou un canal en utilisant leur référence.
- Choix entre (Rt-Component Channel Ref, Evaluated Reference).

### 73.5 Définitions utiles concernant les valeurs génériques

Tableau 45/T.171 – Aperçu général de définitions utiles concernant les valeurs génériques

<p><b>Generic Value Param</b> ::= Generic Value   Generic Value Macro</p> <p><b>Generic Value Macro</b> ::= Macro Def ID, Generic Value?</p> <p><b>Generic Value</b> ::= Value   Evaluated Value</p> <p><b>Value</b> ::= Boolean   Numeric   Ratio   String   List   Reference</p> <p><b>Generic Boolean Param</b> ::= Generic Boolean   Generic Boolean Macro</p> <p><b>Generic Boolean Macro</b> ::= Macro Def ID, Generic Boolean?</p> <p><b>Generic Boolean</b> ::= Boolean   Evaluated Boolean</p> <p><b>Boolean</b> ::= <i>\$BOOLEAN</i></p> <p><b>Generic Numeric Param</b> ::= Generic Numeric   Generic Numeric Macro</p> <p><b>Generic Numeric Macro</b> ::= Macro Def ID, Generic Numeric?</p> <p><b>Generic Numeric</b> ::= Numeric   Evaluated Numeric</p> <p><b>Numeric</b> ::= <i>\$INTEGER</i></p> <p><b>Generic Integer Param</b> ::= Generic Integer   Generic Integer Macro</p> <p><b>Generic Integer Macro</b> ::= Macro Def ID, Generic Integer?</p> <p><b>Generic Integer</b> ::= Integer   Evaluated Integer</p> <p><b>Integer</b> ::= <i>\$INTEGER</i></p> <p><b>Generic Ratio Param</b> ::= Generic Ratio   Generic Ratio Macro</p> <p><b>Generic Ratio Macro</b> ::= Macro Def ID, Generic Ratio?</p> <p><b>Generic Ratio</b> ::= Ratio   Evaluated Ratio</p>
---



**Tableau 45/T.171 – Aperçu général de définitions utiles concernant les valeurs génériques (fin)**

**Ratio** ::= Numerator, Denominator?  
**Numerator** ::= Integer  
**Denominator** ::= Integer  
**Generic String Param** ::= Generic String | Generic String Macro  
**Generic String Macro** ::= Macro Def ID, Generic String?  
**Generic String** ::= String | Evaluated String  
**String** ::= *\$GraphicString*  
**Generic Reference Param** ::= Generic Reference | Generic Reference Macro  
**Generic Reference Macro** ::= Macro Def ID, Generic Reference?  
**Generic Reference** ::= Reference | Evaluated Reference  
**Generic List Param** ::= Generic List | Generic List Macro  
**Generic List Macro** ::= Macro Def ID, Generic List?  
**Generic List** ::= List | Evaluated List  
**List** ::= Value\*  
**Generic List Elt ID Param** ::= Generic List Elt ID | Generic List Elt ID Macro  
**Generic List Elt ID Macro** ::= Macro Def ID, Generic List Elt ID?  
**Generic List Elt ID** ::= Tail

**73.5.1 Generic Value Param (Paramètre de valeur générique)**

- Paramètre de valeur générique.
- Choix entre (Generic Value, Generic Value Macro).

**73.5.2 Generic Value Macro (Macro-paramètre de valeur générique)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de valeur générique.
- Séquence de (Macro Def ID, Generic Value?).

**73.5.3 Generic Value (Valeur générique)**

- Mécanisme de spécification de valeurs d'une manière générique. En principe, les valeurs génériques sont utilisées pour exprimer des paramètres d'actions élémentaires, des conditions ou des attributs d'objets MHEG.
- Choix entre (Value, Evaluated Value).

**73.5.4 Value (Valeur)**

- Spécifications de toutes les valeurs possibles prises en charge par la présente Recommandation.
- Choix entre (Boolean, Numeric, Ratio, String, List, Reference).

**73.5.5 Generic Boolean Param (Paramètre de booléen générique)**

- Paramètre de booléen générique.
- Choix entre (Generic Boolean, Generic Boolean Macro).

**73.5.6 Generic Boolean Macro (Macro-paramètre de booléen générique)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de booléen générique.
- Séquence de (Macro Def ID, Generic Boolean?).

- 73.5.7 Generic Boolean (Booléen générique)**
- Un booléen générique a une des valeurs suivantes: Vrai, Faux, le résultat d'une action Get (extraire).
  - Choix entre (Boolean, Evaluated Boolean).
- 73.5.8 Boolean (Booléen)**
- Dans la présente Recommandation, un booléen est codé sous forme de booléens ASN.1.
  - BOOLEAN.
- 73.5.9 Generic Numeric Param (Paramètre de numérique générique)**
- Paramètre de numérique générique.
  - Choix entre (Generic Numeric, Generic Numeric Macro).
- 73.5.10 Generic Numeric Macro (Macro-paramètre de numérique générique)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de numérique générique.
  - Séquence de (Macro Def ID, Generic Numeric?).
- 73.5.11 Generic Numeric (Numérique générique)**
- Un numérique générique a une des valeurs suivantes: un numérique, le résultat d'une action Get (extraire).
  - Choix entre (Numeric, Evaluated Numeric).
- 73.5.12 Numeric (Numérique)**
- Le modèle de la présente Recommandation ne limite aucunement les valeurs numériques. Celles-ci peuvent être des nombres entiers, réels ou complexes. Les représentations codées peuvent toutefois imposer des limites. La syntaxe ASN.1 définie dans la présente Recommandation code les numériques sous forme d'entiers ASN.1, et il est recommandé aux auteurs d'adopter cette formule.
  - INTEGER.
- 73.5.13 Generic Integer Param (Paramètre d'entier générique)**
- Paramètre d'entier générique.
  - Choix entre (Generic Integer, Generic Integer Macro).
- 73.5.14 Generic Integer Macro (Macro-paramètre d'entier générique)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'entier générique.
  - Séquence de (Macro Def ID, Generic Integer?).
- 73.5.15 Generic Integer (Entier générique)**
- Un entier générique a une des valeurs suivantes: un entier, le résultat d'une action Get (extraire).
  - Choix entre (Integer, Evaluated Integer).
- 73.5.16 Integer (Entier)**
- Dans la présente Recommandation, un entier est codé en utilisant le type INTEGER ASN.1.
  - INTEGER.
- 73.5.17 Generic Ratio Param (Paramètre de fraction générique)**
- Paramètre de fraction générique.
  - Choix entre (Generic Ratio, Generic Ratio Macro).
- 73.5.18 Generic Ratio Macro (Macro-paramètre de fraction générique)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de fraction générique.
  - Séquence de (Macro Def ID, Generic Ratio?).

#### 73.5.19 Generic Ratio (**Fraction générique**)

- Une fraction générique a une des valeurs suivantes: une fraction, le résultat d'une action `Get` (extraire).
- Choix entre (`Ratio`, `Evaluated Ratio`).

#### 73.5.20 Ratio (**Fraction**)

- Une fraction est un couple d'entiers (m, n), m étant le numérateur et n le dénominateur. Si ce dernier est omis, on suppose qu'il est égal à 100, auquel cas le numérateur indique le pourcentage.  
Le numérateur est toujours supérieur à 1, et le dénominateur peut être négatif.
- Séquence de (`Enumerator`, `Denominator?`).

#### 73.5.21 Numerator (**Numérateur**)

- Numérateur d'une fraction.
- `Integer`.

#### 73.5.22 Denominator (**Dénominateur**)

- Dénominateur d'une fraction.
- `Integer`.

#### 73.5.23 Generic String Param (**Paramètre de chaîne générique**)

- Paramètre de chaîne générique.
- Choix entre (`Generic String`, `Generic String Macro`).

#### 73.5.24 Generic String Macro (**Macro-paramètre de chaîne générique**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de chaîne générique.
- Séquence de (`Macro Def ID`, `Generic String?`).

#### 73.5.25 Generic String (**Chaîne générique**)

- Une chaîne générique a une des valeurs suivantes: une chaîne, le résultat d'une action `Get` (extraire).
- Choix entre (`String`, `Evaluated String`).

#### 73.5.26 String (**Chaîne**)

- Dans la présente Recommandation, une chaîne est codée sous forme de chaîne graphique ASN.1. Cela permet d'établir une spécification de chaîne internationale.
- `GraphicString`.

#### 73.5.27 Generic Reference Param (**Paramètre de référence générique**)

- Paramètre de référence générique.
- Choix entre (`Generic Reference`, `Generic Reference Macro`).

#### 73.5.28 Generic Reference Macro (**Macro-paramètre de référence générique**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de référence générique.
- Séquence de (`Macro Def ID`, `Generic Reference?`).

#### 73.5.29 Generic Reference (**Référence générique**)

- Une référence générique a une des valeurs suivantes: une référence, le résultat d'une action `Get` (extraire).
- Choix entre (`Reference`, `Evaluated Reference`).

#### 73.5.30 Generic List Param (**Paramètre de liste générique**)

- Paramètre de liste générique.
- Choix entre (`Generic List`, `Generic List Macro`).

### 73.5.31 Generic List Macro (**Macro-paramètre de liste générique**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de liste générique.
- Séquence de (Macro Def ID, Generic List?).

### 73.5.32 Generic List (**Liste générique**)

- Une liste générique a une des valeurs suivantes: une liste, le résultat d'une action Get (extraire).
- Choix entre (List, Evaluated List).

### 73.5.33 List (**Liste**)

- Une liste de valeurs.
- Liste de (Value).

### 73.5.34 Generic List Elt ID Param (**Paramètre d'identificateur d'élément de liste générique**)

- Paramètre d'identificateur d'élément de liste générique.
- Choix entre (Generic List Elt ID, Generic List Elt ID Macro).

### 73.5.35 Generic List Elt ID Macro (**Macro-paramètre d'identificateur d'élément de liste générique**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'identificateur d'élément de liste générique.
- Séquence de (Macro Def ID, Generic List Elt ID?).

### 73.5.36 Generic List Elt ID (**Identificateur d'élément de liste générique**)

- Identification d'une valeur générique, qui est un élément d'une liste générique.
- Tail.

## 73.6 Définitions utiles concernant les valeurs évaluées

Tableau 46/T.171 – Aperçu général de définitions utiles concernant les valeurs évaluées

<b>Get Event Data ::= Rt-Component Channel Tg Param</b>
<b>Get Event ::= Rt-Component Channel Tg Param</b>
<b>Get Channel Perceptability ::= Channel Target Param</b>
<b>Get Channel Availability Status ::= Channel Target Param</b>
<b>Get Style ::= Rt-Component Target Param</b>
<b>Get Number Of Interacted Sockets ::= Rt-Composite Target Param, Interaction Type Param</b>
<b>Get Interaction Status ::= Rt-Component Target Param, Interaction Type Param</b>
<b>Get Max Interact Required ::= Rt-Component Target Param, Interaction Type Param</b>
<b>Get Min Interact Required ::= Rt-Component Target Param, Interaction Type Param</b>
<b>Get Interaction Ability ::= Rt-Component Target Param, Interaction Type Param</b>
<b>Get Stream Chosen State ::= Rt-Mux Target Param, Stream Identification Param</b>
<b>Get Stream Choice ::= Rt-Mux Target Param</b>
<b>Get GVF ::= Rt-Composite Target Param   Channel Target Param</b>
<b>Get PCV ::= Rt-Content Target Param</b>
<b>Get CV ::= Rt-Content Target Param</b>
<b>Get OV ::= Rt-Content Target Param</b>

**Tableau 46/T.171 – Aperçu général de définitions utiles concernant les valeurs évaluées (suite)**

**Get User Spatial Control** ::= Rt-Component Target Param, Spatial Control Param  
**Get GSF** ::= Rt-Component Target Param, Expected Axis Result Param  
**Get PVS Position** ::= Rt-Component Target Param, Point Type Param, Expected Axis Result Param  
**Get PAP** ::= Rt-Component Target Param, Point Type Param, Expected Axis Result Param  
**Get PVS** ::= Rt-Component Target Param, Point Type Param, Expected Axis Result Param  
**Get OVS Position** ::= Rt-Component Target Param, Point Type Param, Expected Axis Result Param  
**Get OAP** ::= Rt-Component Target Param, Point Type Param, Expected Axis Result Param  
**Get OVS** ::= Rt-Component Target Param, Point Type Param, Expected Axis Result Param  
**Get OVS Proj Strategy** ::= Rt-Component Target Param  
**Get Resizing Strategy** ::= Rt-Composite Target Param  
**Get Aspect Ratio** ::= Rt-Component Target Param  
**Get POS** ::= Rt-Component Target Param, Expected Axis Result Param  
**Get OS** ::= Rt-Component Target Param, Expected Axis Result Param  
**Get Timestone Status** ::= Rt-Component Target Param  
**Get GTF** ::= Rt-Component Target Param  
**Get PVD Position** ::= Rt-Component Target Param  
**Get Temporal Termination** ::= Rt-Component Target Param  
**Get CTP** ::= Rt-Component Target Param  
**Get PVD** ::= Rt-Component Target Param, Expected PVD Result Param  
**Get OVD** ::= Rt-Component Target Param, Expected OVD Result Param  
**Get POD** ::= Rt-Component Target Param  
**Get OD** ::= Rt-Component Target Param  
**Get Presentation Priority** ::= Rt-Component Target Param  
**Get Perceptability** ::= Rt-Component Target Param  
**Get RPS Assignment** ::= Rt-Component Target Param  
**Get Rt-Composite Address** ::= Rt-Composite Target Param, Navigation Command Param  
**Get Termination Status** ::= Rt-Script Target Param  
**Get Running Status** ::= Rt-Target Param  
**Get Rt-Availability Status** ::= Rt-Target Param  
**Get Data** ::= Content Target Param, Generic List Elt ID Param?  
**Get Activation Status** ::= Link Target Param  
**Get Preparation Status** ::= Mh-Target Param  
**Get Catalogued Attribute** ::= Target Param, Cat Ext Attribute Param  
**Get Boolean** ::= \$NULL  
**Get Numeric** ::= Get Integer

**Tableau 46/T.171 – Aperçu général de définitions utiles concernant les valeurs évaluées (fin)**

<p><b>Get Integer ::=</b>  Get Preparation Status   Get Activation Status   Get Rt-Availability Status   Get Running Status   Get Termination Status   Get Presentation Priority   Get OD   Get POD   Get PVD   Get CTP   Get Temporal Termination   Get PVD Position   Get Timestone Status   Get Aspect Ratio   Get Resizing Strategy   Get OVS Proj Strategy   Get User Spatial Control   Get CV   Get PCV   Get Stream Chosen State   Get Min Interact Required   Get Max Interact Required   Get Number Of Interacted Sockets   Get Channel Availability Status   Get Channel Perceptability</p> <p><b>Get Ratio ::=</b> Get Perceptability</p> <p><b>Get String ::=</b> \$NULL</p> <p><b>Get Reference ::=</b> Get Rt-Composite Address   Get RPS Assignment</p> <p><b>Get List ::=</b> Get Catalogued Attribute   Get Stream Choice   Get Style   Get Event</p> <p><b>Get Any ::=</b>  Get Data   Get OVD   Get GTF   Get OS   Get POS   Get OVS   Get OAP   Get OVS Position   Get PVS   Get PAP   Get PVS Position   Get GSF   Get GVF   Get Interaction Ability   Get Interaction Status   Get Event Data</p> <p><b>Evaluated Value ::=</b>  Get Boolean   Get Numeric   Get Ratio   Get String   Get Reference   Get List   Get Any   Extensibility Provision</p> <p><b>Evaluated Boolean ::=</b> Get Boolean   Get Any</p> <p><b>Evaluated Numeric ::=</b> Get Numeric   Get Integer   Get Any</p> <p><b>Evaluated Integer ::=</b> Get Integer   Get Any</p> <p><b>Evaluated Ratio ::=</b> Get Ratio   Get Any</p> <p><b>Evaluated String ::=</b> Get String   Get Any</p> <p><b>Evaluated Reference ::=</b> Get Reference   Get Any</p> <p><b>Evaluated List ::=</b> Get List   Get Any</p>
--

**73.6.1** Get Event Data (**Extraire données d'événement**)

- Récupère les données d'événement de la cible.
- Rt-Component Channel Tg Param.

**73.6.2** Get Event (**Extraire événement**)

- Récupère l'événement de la cible.
- Rt-Component Channel Tg Param.

**73.6.3** Get Channel Perceptability (**Extraire perceptibilité de canal**)

- Récupère la perceptibilité du canal affecté à la cible.
- Channel Target Param.

**73.6.4** Get Channel Availability Status (**Extraire état de disponibilité du canal**)

- Récupère l'état de disponibilité du canal.
- Channel Target Param.

**73.6.5** Get Style (**Extraire style**)

- Récupère le style courant associé à la cible.
- Rt-Component Target Param.

**73.6.6** Get Number of Interacted Sockets (**Extraire nombre de réceptacles en interaction**)

- Récupère la valeur de l'attribut Get Number of Interacted Sockets (extraire nombre de réceptacles en interaction) de la cible.
- Séquence de (Rt-Component Target Param, Interaction Type Param).

- 73.6.7** Get Interaction Status (**Extraire état d'interaction**)
- Récupère la valeur de l'état d'interaction de la cible. Cette action prend la valeur «sélectionné» ou «pas sélectionné», et «modifié», «en cours de modification» ou «pas modifié».
  - Séquence de (Rt-Component Target Param, Interaction Type Param).
- 73.6.8** Get Max Interact Required (**Extraire interaction maximale requise**)
- Récupère la valeur de l'attribut Max Interact Required (Interaction maximale requise) de la cible.
  - Séquence de (Rt-Component Target Param, Interaction Type Param).
- 73.6.9** Get Min Interact Required (**Extraire interaction minimale requise**)
- Récupère la valeur de l'attribut Min Interact Required (Interaction minimale requise) de la cible.
  - Séquence de (Rt-Component Target Param, Interaction Type Param).
- 73.6.10** Get Interaction Ability (**Extraire capacité d'interaction**)
- Récupère la capacité d'interaction de la cible.
  - Séquence de (Rt-Component Target Param, Interaction Type Param).
- 73.6.11** Get Stream Chosen State (**Extraire état sélectionné du flux**)
- Récupère l'état du flux.
  - Séquence de (Rt-Mux Target Param, Stream Identification Param).
- 73.6.12** Get Stream Choice (**Extraire choix de flux**)
- Récupère une liste d'identificateurs de flux choisis pour la cible.
  - Rt-Mux Target Param.
- 73.6.13** Get GVF (**Extraire facteur GVF**)
- Récupère le facteur GVF.
  - Choix entre (Rt-Composite Target Param, Channel Target Param).
- 73.6.14** Get PCV (**Extraire volume PCV**)
- Récupère le volume PCV.
  - Rt-Content Target Param.
- 73.6.15** Get CV (**Extraire volume sonore courant**)
- Récupère le volume sonore courant.
  - Rt-Content Target Param.
- 73.6.16** Get OV (**Extraire volume sonore d'origine**)
- Récupère le volume sonore d'origine.
  - Rt-Content Target Param.
- 73.6.17** Get User Spatial Control (**Extraire commande spatiale de l'utilisateur**)
- Récupère la valeur de l'attribut User Spatial Control (Commande spatiale de l'utilisateur).
  - Séquence de (Rt-Component Target Param, Spatial Control Param).
- 73.6.18** Get GSF (**Extraire facteur GSF**)
- Récupère le facteur GSF.
  - Séquence de (Rt-Component Target Param, Expected Axis Result Param).
- 73.6.19** Get PVS Position (**Extraire position PVS**)
- Récupère la position PVS.
  - Séquence de (Rt-Component Target Param, Point Type Param, Expected Axis Result Param).

- 73.6.20** Get PAP (**Extraire point PAP**)
- Récupère le point PAP.
  - Séquence de (Rt-Component Target Param, Point Type Param, Expected Axis Result Param).
- 73.6.21** Get PVS (**Extraire taille PVS**)
- Récupère la taille PVS.
  - Séquence de (Rt-Component Target Param, Point Type Param, Expected Axis Result Param).
- 73.6.22** Get OVS Position (**Extraire position OVS**)
- Récupère la position OVS.
  - Séquence de (Rt-Component Target Param, Point Type Param, Expected Axis Result Param).
- 73.6.23** Get OAP (**Extraire point OAP**)
- Récupère le point OAP.
  - Séquence de (Rt-Component Target Param, Point Type Param, Expected Axis Result Param).
- 73.6.24** Get OVS (**Extraire taille OVS**)
- Récupère la taille OVS.
  - Séquence de (Rt-Component Target Param, Point Type Param, Expected Axis Result Param).
- 73.6.25** Get OVS Proj Strategy (**Extraire stratégie de projection de la taille OVS**)
- Récupère l'attribut OVS Proj Strategy (stratégie de projection de la taille OVS).
  - Rt-Component Target Param.
- 73.6.26** Get Resizing Strategy (**Extraire stratégie de redimensionnement**)
- Récupère l'attribut Resizing Strategy (stratégie de redimensionnement).
  - Rt-Composite Target Param.
- 73.6.27** Get Aspect Ratio (**Extraire facteur de forme**)
- Récupère l'attribut Aspect Ratio (facteur de forme).
  - Rt-Component Target Param.
- 73.6.28** Get POS (**Extraire taille POS**)
- Récupère la taille POS.
  - Séquence de (Rt-Component Target Param, Expected Axis Result Param).
- 73.6.29** Get OS (**Extraire taille d'origine**)
- Récupère la taille d'origine.
  - Séquence de (Rt-Component Target Param, Expected Axis Result Param).
- 73.6.30** Get Timestone Status (**Extraire état de marqueur temporel**)
- Récupère la valeur de l'état du marqueur temporel.
  - Rt-Component Target Param.
- 73.6.31** Get GTF (**Extraire facteur GTF**)
- Récupère le facteur GTF de la cible.
  - Rt-Component Target Param.



- 73.6.32** Get PVD Position (**Extraire position PVD**)
- Récupère la position PVD associée à son espace PRPS.
  - Rt-Component Target Param.
- 73.6.33** Get Temporal Termination (**Extraire achèvement temporel**)
- Permet de récupérer l'achèvement temporel de la cible.
  - Rt-Component Target Param.
- 73.6.34** Get CTP (**Extraire position CTP**)
- Permet de récupérer la position CTP de la cible.
  - Rt-Component Target Param.
- 73.6.35** Get PVD (**Extraire durée PVD**)
- Récupère la durée PVD, la position temporelle initiale ou la position temporelle finale de la cible.
  - Séquence de (Rt-Component Target Param, Expected PVD Result Param).
- 73.6.36** Get OVD (**Extraire durée OVD**)
- Récupère la durée OVD, la position temporelle initiale ou la position temporelle finale de la cible.
  - Séquence de (Rt-Component Target Param, Expected OVD Result Param).
- 73.6.37** Get POD (**Extraire durée POD**)
- Récupère la durée POD de la cible.
  - Rt-Component Target Param.
- 73.6.38** Get OD (**Extraire durée d'origine**)
- Récupère la durée d'origine de la cible.
  - Rt-Component Target Param.
- 73.6.39** Get Presentation Priority (**Extraire priorité de présentation**)
- Récupère la priorité de présentation.
  - Rt-Component Target Param.
- 73.6.40** Get Perceptability (**Extraire perceptibilité**)
- Récupère la perceptibilité.
  - Rt-Component Target Param.
- 73.6.41** Get RPS Assignment (**Extraire affectation d'espace RPS**)
- Cette action récupère l'affectation d'espace RPS à un rt-composant et à un rt-composant racine.
  - Rt-Component Target Param.
- 73.6.42** Get Rt-Composite Address (**Extraire adresse de rt-composite**)
- Permet de monter jusqu'à la racine d'une arborescence, de descendre dans ses feuilles et d'en explorer les niveaux intermédiaires (nœuds).
  - Séquence de (Rt-Composite Target Param, Navigation Command Param).
- 73.6.43** Get Termination Status (**Extraire état d'achèvement**)
- Récupère l'état d'achèvement d'un rt-script. Le résultat est «terminé» ou «pas terminé».
  - Rt-Script Target Param.
- 73.6.44** Get Running Status (**Extraire état d'exécution**)
- Récupère l'état d'exécution d'un rt-objet ou d'un réceptacle. Le résultat est «en cours d'exécution» ou «pas en cours d'exécution».
  - Rt-Target Param.

- 73.6.45** Get Rt-Availability Status (**Extraire état de rt-disponibilité**)
- Permet de récupérer l'état de disponibilité d'un rt-objet. Le résultat est «disponible» ou «pas disponible».
  - Rt-Target Param.
- 73.6.46** Get Data (**Extraire données**)
- Cette action récupère une valeur générique ou un élément d'une liste générique stockée dans un objet *contenu*.
  - Séquence de (Content Target Param, Generic List Elt ID Param?).
- 73.6.47** Get Activation Status (**Extraire état d'activation**)
- Cette action récupère la valeur de l'état d'activation d'un objet *lien*.
  - Link Target Param.
- 73.6.48** Get Preparation Status (**Extraire état de préparation**)
- Récupère l'état de préparation d'un objet MHEG. Le résultat est «prêt» ou «pas prêt».
  - MH-Target Param.
- 73.6.49** Get Catalogued Attribute (**Extraire attribut catalogué**)
- Récupère la valeur d'un attribut catalogué étendu. Le résultat est une valeur générique décrite dans le catalogue correspondant.
  - Séquence de (Target Param, Cat Ext Attribute Param).
- 73.6.50** Get Boolean (**Extraire booléen**)
- L'action Get Boolean (extraire booléen) est une forme de l'action Get (extraire), qui retourne un booléen.
  - NULL.
- 73.6.51** Get Numeric (**Extraire numérique**)
- L'action Get Numeric (extraire numérique) est une forme de l'action Get (extraire), qui retourne un numérique.
  - Get Integer.
- 73.6.52** Get Integer (**Extraire entier**)
- L'action Get Integer (extraire entier) est une forme de l'action Get (extraire), qui retourne un entier.
  - Choix entre (Get Preparation Status, Get Activation Status, Get Rt-Availability Status, Get Running Status, Get Termination Status, Get Presentation Priority, Get OD, Get POD, Get PVD, Get CTP, Get Temporal Termination, Get PVD Position, Get Timestone Status, Get Aspect Ratio, Get Resizing Strategy, Get OVS Proj Strategy, Get User Spatial Control, Get CV, Get PCV, Get Stream Chosen State, Get Min Interact Required, Get Max Interact Required, Get Number of Interacted Sockets, Get Channel Availability Status, Get Channel Perceptability).
- 73.6.53** Get Ratio (**Extraire fraction**)
- L'action Get Ratio (extraire fraction) est une forme de l'action Get (extraire), qui retourne une fraction.
  - Get Perceptability.
- 73.6.54** Get String (**Extraire chaîne**)
- L'action Get String (extraire chaîne) est une forme de l'action Get (extraire), qui retourne une chaîne.
  - NULL.
- 73.6.55** Get Reference (**Extraire référence**)
- L'action Get Reference (extraire référence) est une forme de l'action Get (extraire), qui retourne une référence.
  - Choix entre (Get Rt-Composite Address, Get RPS Assignment).

### **73.6.56** Get List (**Extraire liste**)

- L'action Get List (extraire liste) est une forme de l'action Get (extraire), qui retourne une liste.
- Choix entre (Get Catalogued Attribute, Get Stream Choice, Get Style, Get Event).

### **73.6.57** Get Any (**Extraire valeur de type «quelconque»**)

- L'action Get Any (extraire valeur de type «quelconque») est une forme de l'action Get (extraire), qui retourne une valeur de type «quelconque».
- Choix entre (Get Data, Get OVD, Get GTF, Get OS, Get POS, Get OVS, Get OAP, Get OVS Position, Get PVS, Get PAP, Get PVS Position, Get GSF, Get GVF, Get Interaction Ability, Get Interaction Status, Get Event Data).

### **73.6.58** Evaluated Value (**Valeur évaluée**)

- Une valeur évaluée est le résultat d'une action Get (extraire).
- Choix entre (Get Boolean, Get Numeric, Get Ratio, Get String, Get Reference, Get List, Get Any, Extensibility Provision).

### **73.6.59** Evaluated Boolean (**Booléen évalué**)

- Un booléen évalué est le résultat d'une action Get (extraire) renvoyant un booléen ou une valeur de type «quelconque».
- Choix entre (Get Boolean, Get Any).

### **73.6.60** Evaluated Numeric (**Numérique évalué**)

- Un numérique évalué est le résultat d'une action Get (extraire) renvoyant un numérique, un entier ou une valeur de type «quelconque».
- Choix entre (Get Numeric, Get Integer, Get Any).

### **73.6.61** Evaluated Integer (**Entier évalué**)

- Un entier évalué est le résultat d'une action Get (extraire) renvoyant un entier ou une valeur de type «quelconque».
- Choix entre (Get Integer, Get Any).

### **73.6.62** Evaluated Ratio (**Fraction évaluée**)

- Une fraction évaluée est le résultat d'une action Get (extraire) renvoyant une fraction ou une valeur de type «quelconque».
- Choix entre (Get Ratio, Get Any).

### **73.6.63** Evaluated String (**Chaîne évaluée**)

- Une chaîne évaluée est le résultat d'une action Get (extraire) renvoyant une chaîne ou une valeur de type «quelconque».
- Choix entre (Get String, Get Any).

### **73.6.64** Evaluated Reference (**Référence évaluée**)

- Une référence évaluée est le résultat d'une action Get (extraire) renvoyant une référence ou une valeur de type «quelconque».
- Choix entre (Get Reference, Get Any).

### **73.6.65** Evaluated List (**Liste évaluée**)

- Une liste évaluée est le résultat d'une action Get (extraire) renvoyant une liste ou une valeur de type «quelconque».
- Choix entre (Get List, Get Any).

## 73.7 Crochets

Tableau 47/T.171 – Aperçu des crochets

**Content Hook** ::= Catalogued Content Encoding?, Content Encoding Description?

**Content Encoding Description** ::= *\$OCTET STRING*

**Script Hook** ::= Catalogued Script Encoding?, Script Encoding Description?

**Script Encoding Description** ::= *\$OCTET STRING*

### 73.7.1 Content Hook (Crochet de contenu)

- Informations concernant le codage et le décodage, permettant d'utiliser les données monomédias ou d'autres données codées. Ces informations contiennent un champ d'identification pour la norme de codage monomédia et un champ pour les paramètres associés au codage.

La sémantique de ces champs n'est pas définie par MHEG, mais elle sera placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG.

- Séquence de (Catalogued Content Encoding?, Content Encoding Description?).

### 73.7.2 Content Encoding Description (Description de codage de contenu)

- Informations supplémentaires se rapportant à des techniques de codage et de décodage spécifiques. La sémantique de cette description peut être fournie dans le catalogue de codage du contenu.
- OCTET STRING.

### 73.7.3 Script Hook (Crochet de script)

- Informations concernant le codage et le décodage, permettant d'utiliser le langage de script codé. Ces informations contiennent un champ d'identification pour la norme de codage de script et un champ pour les paramètres associés au codage.
- Séquence de (Catalogued Script Encoding?, Script Encoding Description?).

### 73.7.4 Script Encoding Description (Description de codage de script)

- Informations supplémentaires se rapportant à des techniques de codage et de décodage spécifiques. La sémantique de cette description peut être fournie dans le catalogue de codage du script.
- OCTET STRING.

## 73.8 Extensibilité

Tableau 48/T.171 – Aperçu général de l'extensibilité

**Catalogued Content Encoding** ::= Registered Content Encoding | Proprietary Content Encoding

**Registered Content Encoding** ::= *#Imported from ITU-T Rec. T.171 cat*

**Proprietary Content Encoding** ::= Prop Cat Entry ID

**Catalogued Script Encoding** ::= Registered Script Encoding | Proprietary Script Encoding

**Registered Script Encoding** ::= *#Imported from ITU-T Rec. T.171 cat*

**Proprietary Script Encoding** ::= Prop Cat Entry ID

**Cat Content Classification** ::= Registered Content Classification | Prop Content Classification

**Registered Content Classification** ::= *#Imported from ITU-T Rec. T.171 cat*

**Tableau 48/T.171 – Aperçu général de l'extensibilité (fin)**

<b>Prop Content Classification</b> ::= Prop Cat Entry ID
<b>Catalogued Script Classification</b> ::= Registered Script Classification   Prop Script Classification
<b>Registered Script Classification</b> ::= #Imported from ITU-T Rec. T.171 cat
<b>Prop Script Classification</b> ::= Prop Cat Entry ID
<b>Catalogued Media Type</b> ::= Registered Media Type   Proprietary Media Type
<b>Registered Media Type</b> ::= #Imported from ITU-T Rec. T.171 cat
<b>Proprietary Media Type</b> ::= Prop Cat Entry ID
<b>Catalogued Style</b> ::= Registered Style   Proprietary Style
<b>Registered Style</b> ::= #Imported from ITU-T Rec. T.171 cat
<b>Proprietary Style</b> ::= Prop Cat Entry ID
<b>Catalogued Event</b> ::= Registered Event   Proprietary Event
<b>Registered Event</b> ::= #Imported from ITU-T Rec. T.171 cat
<b>Proprietary Event</b> ::= Prop Cat Entry ID
<b>Catalogued Extended EA</b> ::= Registered Extended Elementary Action   Proprietary Extended Elementary Action
<b>Registered Extended Elementary Action</b> ::= #Imported from ITU-T Rec. T.171 cat
<b>Proprietary Extended Elementary Action</b> ::= Prop Cat Entry ID
<b>Cat Ext Attribute</b> ::= Registered Extended Attribute   Proprietary Extended Attribute
<b>Registered Extended Attribute</b> ::= #Imported from ITU-T Rec. T.171 cat
<b>Proprietary Extended Attribute</b> ::= Prop Cat Entry ID
<b>Prop Cat Entry ID</b> ::= Tail   §OCTET STRING
<b>Extensibility Provision</b> ::= Pointpointpoint
<b>Pointpointpoint</b> ::= #INTEGER ::= 9999

**73.8.1 Catalogued Content Encoding (Codage de contenu catalogué)**

- Identification du codage des données de contenu. Deux types de catalogues sont offerts: un catalogue enregistré et un catalogue privé.
- Choix entre (Registered Content Encoding, Proprietary Content Encoding).

**73.8.2 Registered Content Encoding (Codage de contenu enregistré)**

- Identification des techniques de codage et de décodage dans un catalogue de contenu enregistré.  
Le catalogue de contenu enregistré est placé sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
- Importé du module cat de l'ISO/CEI 13522-4.

**73.8.3 Proprietary Content Encoding (Codage de contenu privé)**

- Identification des techniques de codage et de décodage dans un catalogue de contenu privé.  
Il appartient à l'application utilisatrice de résoudre les références au catalogue de contenu privé.
- Prop Cat Entry ID.

**73.8.4 Catalogued Script Encoding (Codage de script catalogué)**

- Identification du codage du langage de script. Deux types de catalogues sont offerts: un catalogue enregistré et un catalogue privé.
- Choix entre (Registered Script Encoding, Proprietary Script Encoding).

- 73.8.5 Registered Script Encoding (Codage de script enregistré)**
- Identification du langage de script dans un catalogue de script enregistré.
  - Le catalogue de script enregistré est placé sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
  - Importé du module cat de l'ISO/CEI 13522-4.
- 73.8.6 Proprietary Script Encoding (Codage de script privé)**
- Identification du langage de script dans un catalogue de script privé.
  - Prop Cat Entry ID.
- 73.8.7 Cat Content Classification (Classification de contenu cataloguée)**
- Identification du type de donnée. Pour les données monomédias, la classification du contenu est utilisée pour indiquer le support de perception (texte, graphique, signaux audiophoniques). Ces informations peuvent être utilisées par un processus de négociation, une base de données ou un moteur MHEG pour choisir un décodeur. Il s'agit d'un moyen d'assistance facultatif pour déterminer le type de donnée. En cas d'incompatibilité entre le crochet et les informations sur la classification, la préférence va au crochet.
  - Choix entre (Registered Content Classification, Prop Content Classification).
- 73.8.8 Registered Content Classification (Classification de contenu enregistrée)**
- Identification du type de classification dans le catalogue de contenu enregistré.
  - La classification de contenu MHEG est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
  - Importé du module cat de l'ISO/CEI 13522-4.
- 73.8.9 Prop Content Classification (Classification de contenu privée)**
- Identification du type de donnée dans le catalogue de contenu privé.
  - Prop Cat Entry ID.
- 73.8.10 Catalogued Script Classification (Classification de script cataloguée)**
- Identification du type de script.
  - Choix entre (Registered Script Classification, Prop Script Classification).
- 73.8.11 Registered Script Classification (Classification de script enregistrée)**
- Identification du type de script dans le catalogue de script enregistré.
  - La classification de script enregistrée est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
  - Importé du module cat de l'ISO/CEI 13522-4.
- 73.8.12 Prop Script Classification (Classification de script privée)**
- Identification du type de script dans le catalogue de type de script privé.
  - Prop Cat Entry ID.
- 73.8.13 Catalogued Media Type (Type de support catalogué)**
- Identification du type de support.
  - Choix entre (Registered Media Type, Proprietary Media Type).
- 73.8.14 Registered Media Type (Type de support enregistré)**
- Identification du type de support dans le catalogue de support enregistré.
  - La classification de support enregistrée est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
  - Importé du module cat de l'ISO/CEI 13522-4.

- 73.8.15 Proprietary Media Type (Type de support privé)**
- Identification du type de support dans le catalogue de type de support privé.
  - Prop Cat Entry ID.
- 73.8.16 Catalogued Style (Style catalogué)**
- Identification du style.
  - Choix entre (Registered Style, Proprietary Style).
- 73.8.17 Registered Style (Style enregistré)**
- Identification du type de style dans le catalogue de style enregistré.
- La classification de style enregistrée est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
- Importé du module cat de l'ISO/CEI 13522-4.
- 73.8.18 Proprietary Style (Style privé)**
- Identification du type de style dans le catalogue de type de style privé.
  - Prop Cat Entry ID.
- 73.8.19 Catalogued Event (Événement catalogué)**
- Identification de l'événement.
  - Choix entre (Registered Event, Proprietary Event).
- 73.8.20 Registered Event (Événement enregistré)**
- Identification du type d'événement dans le catalogue d'événement enregistré.
- La classification d'événement enregistrée est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
- Importé du module cat de l'ISO/CEI 13522-4.
- 73.8.21 Proprietary Event (Événement privé)**
- Identification du type d'événement dans le catalogue de type d'événement privé.
  - Prop Cat Entry ID.
- 73.8.22 Catalogued Extended EA (Action élémentaire étendue cataloguée)**
- Identification d'une action élémentaire étendue cataloguée. L'action utilisée peut être enregistrée ou privée.
  - Choix entre (Registered Extended Elementary Action, Proprietary Extended Elementary Action).
- 73.8.23 Registered Extended Elementary Action (Action élémentaire étendue enregistrée)**
- Identification du type d'action élémentaire étendue dans le catalogue d'action élémentaire étendue enregistré.
- La classification d'action élémentaire étendue enregistrée est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.
- Importé du module cat de l'ISO/CEI 13522-4.
- 73.8.24 Proprietary Extended Elementary Action (Action élémentaire étendue privée)**
- Identification du type d'action élémentaire étendue dans le catalogue de type d'action élémentaire étendue privé.
  - Prop Cat Entry ID.
- 73.8.25 Cat Ext Attribute (Attribut étendu catalogué)**
- Identification d'un attribut étendu.
  - Choix entre (Registered Extended Attribute, Proprietary Extended Attribute).

### 73.8.26 Registered Extended Attribute (**Attribut étendu enregistré**)

- Identification du type d'attribut étendu dans le catalogue d'attribut étendu enregistré.

La classification d'attribut étendu enregistrée est placée sous le contrôle de l'Autorité chargée de l'enregistrement des identificateurs de format MHEG, comme il est spécifié dans l'ISO/CEI 13522-4.

- Importé du module cat de l'ISO/CEI 13522-4.

### 73.8.27 Proprietary Extended Attribute (**Attribut étendu privé**)

- Identification du type d'attribut étendu dans le catalogue de type d'attribut étendu privé.
- Prop Cat Entry ID.

### 73.8.28 Prop Cat Entry ID (**Identification de rubrique de catalogue privé**)

- Identification d'une rubrique dans un catalogue.
- Choix entre (Tail, OCTET STRING).

### 73.8.29 Extensibility Provision (**Disposition d'extensibilité**)

- Moyen pour une application utilisatrice d'étendre la syntaxe.
- Pointpointpoint.

## 73.9 Définitions utiles concernant l'espace de présentation

**Tableau 49/T.171 – Aperçu général de définitions utiles concernant l'espace de présentation**

```
OPS Initialisation ::= OD?, OS?
min-AVR ::= #INTEGER ::= 0
max-AVR ::= #INTEGER ::= 255
GF Param ::= GF | GF Macro
GF Macro ::= Macro Def ID, GF?
GF ::= Generic Ratio | default-GF
default-GF ::= #INTEGER ::= 1
X GF ::= GF
Y GF ::= GF
Z GF ::= GF
Length ::= Integer
Point ::= Integer | #terminal
Temporal Position ::= Point
Size ::= X Spatial Length?, Y Spatial Length?, Z Spatial Length?
X Spatial Length ::= Spatial Length
Y Spatial Length ::= Spatial Length
Z Spatial Length ::= Spatial Length
Spatial Length ::= Length | default-spatial-length
default-spatial-length ::= #INTEGER ::= 65536
Size Spec Param ::= Size Spec | Size Spec Macro
Size Spec Macro ::= Macro Def ID, Size Spec?
Size Spec ::= Lengths Spec | Get List | Get Any
```



**Tableau 49/T.171 – Aperçu général de définitions utiles concernant l'espace de présentation (fin)**

**Lengths Spec** ::= X Length Spec?, Y Length Spec?, Z Length Spec?  
**X Length Spec** ::= Length Spec  
**Y Length Spec** ::= Length Spec  
**Z Length Spec** ::= Length Spec  
**Length Spec Param** ::= Length Spec | Length Spec Macro  
**Length Spec Macro** ::= Macro Def ID, Length Spec?  
**Length Spec** ::= Length | Relative Length | Get Integer | Get Ratio | Get Any  
**Relative Length** ::= Ratio  
**Spatial Position** ::= X Point?, Y Point?, Z Point?  
**X Point** ::= Point  
**Y Point** ::= Point  
**Z Point** ::= Point  
**Spatial Position Spec Param** ::= Spatial Position Spec | Spatial Position Spec Macro  
**Spatial Position Spec Macro** ::= Macro Def ID, Spatial Position Spec?  
**Spatial Position Spec** ::= Points Spec | Get List | Get Any  
**Points Spec** ::= X Point Spec?, Y Point Spec?, Z Point Spec?  
**X Point Spec** ::= Point Spec  
**Y Point Spec** ::= Point Spec  
**Z Point Spec** ::= Point Spec  
**Point Spec Param** ::= Point Spec | Point Spec Macro  
**Point Spec Macro** ::= Macro Def ID, Point Spec?  
**Point Spec** ::= Point | Relative Point | Get Integer | Get Ratio | Get Any  
**Current Point Spec Param** ::= Current Point Spec | Current Point Spec Macro  
**Current Point Spec Macro** ::= Macro Def ID, Current Point Spec?  
**Current Point Spec** ::=  
 Point | Relative Point | Original Point Factor | Current Point Factor | Get Integer | Get Ratio | Get Any  
**Relative Point** ::= Ratio  
**Original Point Factor** ::= Ratio  
**Current Point Factor** ::= Ratio  
**Duration Spec** ::= Length Spec  
**Initial Point Spec Param** ::= Initial Point Spec | Initial Point Spec Macro  
**Initial Point Spec Macro** ::= Macro Def ID, Initial Point Spec?  
**Terminal Point Spec Param** ::= Terminal Point Spec | Terminal Point Spec Macro  
**Terminal Point Spec Macro** ::= Macro Def ID, Terminal Point Spec?  
**Initial Point Spec** ::= Point Spec  
**Terminal Point Spec** ::= Point Spec  
**Transition Duration Param** ::= Generic Integer Param

### 73.9.1 OPS Initialisation (**Initialisation d'espace OPS**)

- Fournit les informations nécessaires à la création de l'espace OPS d'un rt-composant ou d'un canal. Fournit également une durée d'origine et une taille d'origine pour déterminer les longueurs des axes temporel et spatial de l'espace OPS. La plage AVR est fixée sur la base de la fourchette de valeurs figurant dans la présente Recommandation.
- Séquence de (OD?, OS?).

### 73.9.2 min-AVR (**Plage AVR minimale**)

- Valeur minimale de la plage AVR dans la présente Recommandation.
- INTEGER ::= 0.

### 73.9.3 max-AVR (**Plage AVR maximale**)

- Valeur maximale de la plage AVR dans la présente Recommandation.
- INTEGER ::= 255.

### 73.9.4 GF Param (**Paramètre de facteur générique**)

- Paramètre de facteur générique.
- Choix entre (GF, GF Macro).

### 73.9.5 GF Macro (**Macro-paramètre de facteur générique**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de facteur générique.
- Séquence de (Macro Def ID, GF?).

### 73.9.6 GF (**Facteur générique**)

- Spécifie le facteur générique d'un rt-composant ou d'un canal.
- Choix entre (Generic Ratio, default-GF).

### 73.9.7 default-GF (**Facteur générique par défaut**)

- Valeur par défaut du facteur générique dans la présente Recommandation.
- INTEGER ::= 1.

### 73.9.8 X GF (**Facteur générique sur X**)

- Exprime le facteur générique sur l'abscisse.
- GF.

### 73.9.9 Y GF (**Facteur générique sur Y**)

- Exprime le facteur générique sur l'ordonnée.
- GF.

### 73.9.10 Z GF (**Facteur générique sur Z**)

- Exprime le facteur générique sur la cote.
- GF.

### 73.9.11 Length (**Longueur**)

- Spécifie la longueur d'un intervalle [Start Point, EndList Point]. Cette longueur est égale à EndList-Point - Start-Point + 1.
- Integer.

### 73.9.12 Point (**Point**)

- Spécifie une position dans une durée ou une taille. Le point est exprimé en unités GTU ou GSU. La valeur 0 représente la position initiale de la durée ou de la taille correspondante. Une valeur réservée «finale» représente la position finale de la durée ou de la taille correspondante.
- Choix entre (Integer, terminal).

### **73.9.13 Temporal Position (Position temporelle)**

- Spécifie une position temporelle exprimée en unités GTU. La position est toujours spécifiée dans une durée.
- Point.

### **73.9.14 Size (Taille)**

- Spécifie une taille en indiquant la longueur sur chaque axe: l'abscisse, l'ordonnée, la cote.
- Séquence de (X Spatial Length?, Y Spatial Length?, Z Spatial Length?).

### **73.9.15 X Spatial Length (Longueur spatiale sur X)**

- Indique la longueur sur l'abscisse.
- Spatial Length.

### **73.9.16 Y Spatial Length (Longueur spatiale sur Y)**

- Indique la longueur sur l'ordonnée.
- Spatial Length.

### **73.9.17 Z Spatial Length (Longueur spatiale sur Z)**

- Indique la longueur sur la cote.
- Spatial Length.

### **73.9.18 Spatial Length (Longueur spatiale)**

- Spécifie la longueur d'un axe spatial, exprimée en unités GSU.
- Choix entre (Length, default-spatial-length).

### **73.9.19 default-spatial-length (Longueur spatiale par défaut)**

- Valeur par défaut de la longueur de l'axe spatial, comme spécifié dans la présente Recommandation.
- INTEGER ::= 65536.

### **73.9.20 Size Spec Param (Paramètre de spécification de taille)**

- Paramètre de spécification de taille.
- Choix entre (Size Spec, Size Spec Macro).

### **73.9.21 Size Spec Macro (Macro-paramètre de spécification de taille)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de taille.
- Séquence de (Macro Def ID, Size Spec?).

### **73.9.22 Size Spec (Spécification de taille)**

- Ce paramètre d'action élémentaire spécifie différents moyens de définir une taille. Il est exprimé sous forme de longueur sur chaque axe: x, y, z.
- Choix entre (Lengths Spec, Get List, Get Any).

### **73.9.23 Lengths Spec (Spécification de longueur)**

- Exprimé sous forme de longueur sur chaque axe: x, y, z.
- Séquence de (X Length Spec?, Y Length Spec?, Z Length Spec?).

### **73.9.24 X Length Spec (Spécification de longueur sur X)**

- Indique la longueur sur l'abscisse.
- Length Spec.

### **73.9.25 Y Length Spec (Spécification de longueur sur Y)**

- Indique la longueur sur l'ordonnée.
- Length Spec.

### 73.9.26 Z Length Spec (**Spécification de longueur sur Z**)

- Indique la longueur sur la cote.
- Length Spec.

### 73.9.27 Length Spec Param (**Paramètre de spécification de longueur**)

- Paramètre de spécification de longueur.
- Choix entre (Length Spec, Length Spec Macro).

### 73.9.28 Length Spec Macro (**Macro-paramètre de spécification de longueur**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de longueur.
- Séquence de (Macro Def ID, Length Spec?).

### 73.9.29 Length Spec (**Spécification de longueur**)

- Chaque longueur peut être spécifiée comme une longueur absolue, une longueur relative ou comme le résultat d'une action `Get integer` (extraire entier), `Get ratio` (extraire fraction) ou `Get any` (extraire valeur de type «quelconque»).
- Choix entre (Length, Relative Length, Get Integer, Get Ratio, Get Any).

### 73.9.30 Relative Length (**Longueur relative**)

- Définit une longueur en tant que fraction d'un intervalle.
- Ratio.

### 73.9.31 Spatial Position (**Position spatiale**)

- Spécifie une position spatiale en indiquant un point sur chaque axe: X, Y, Z.
- Séquence de (X Point?, Y Point?, Z Point?).

### 73.9.32 X Point (**Point sur X**)

- Indique la position sur l'abscisse.
- Point.

### 73.9.33 Y Point (**Point Y**)

- Indique la position sur l'ordonnée.
- Point.

### 73.9.34 Z Point (**Point Z**)

- Indique la position sur la cote.
- Point.

### 73.9.35 Spatial Position Spec Param (**Paramètre de spécification de position spatiale**)

- Paramètre de spécification de position spatiale.
- Choix entre (Spatial Position Spec, Spatial Position Spec Macro).

### 73.9.36 Spatial Position Spec Macro (**Macro-paramètre de spécification de position spatiale**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de position spatiale.
- Séquence de (Macro Def ID, Spatial Position Spec?).

### 73.9.37 Spatial Position Spec (**Spécification de position spatiale**)

- Ce paramètre d'action élémentaire spécifie différents moyens de définir une position spatiale. Il est exprimé sous forme de longueur sur chaque axe: x, y, z.
- Choix entre (Points Spec, Get List, Get Any).

### **73.9.38 Points Spec (Spécification de point)**

- Définit une position. Ce paramètre est exprimé sous forme de longueur sur chaque axe: x, y, z.
- Séquence de (X Point Spec?, Y Point Spec?, Z Point Spec?).

### **73.9.39 X Point Spec (Spécification de point sur X)**

- Longueur sur l'abscisse.
- Point Spec.

### **73.9.40 Y Point Spec (Spécification de point sur Y)**

- Longueur sur l'ordonnée.
- Point Spec.

### **73.9.41 Z Point Spec (Spécification de point sur Z)**

- Longueur sur la cote.
- Point Spec.

### **73.9.42 Point Spec Param (Paramètre de spécification de point)**

- Paramètre de spécification de point.
- Choix entre (Point Spec, Point Spec Macro).

### **73.9.43 Point Spec Macro (Macro-paramètre de spécification de point)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de point.
- Séquence de (Macro Def ID, Point Spec?).

### **73.9.44 Point Spec (Spécification de point)**

- Chaque point peut être spécifié comme un point absolu, un point relatif dans une longueur ou comme le résultat d'une action `Get integer` (extraire entier), `Get ratio` (extraire fraction) ou `Get any` (extraire valeur de type «quelconque»). Un entier est interprété comme une valeur absolue. Une fraction est interprétée comme une valeur relative.
- Choix entre (Point, Relative Point, Get Integer, Get Ratio, Get Any).

### **73.9.45 Current Point Spec Param (Paramètre de spécification de point courant)**

- Paramètre de spécification de point courant.
- Choix entre (Current Point Spec, Current Point Spec Macro).

### **73.9.46 Current Point Spec Macro (Macro-paramètre de spécification de point courant)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de point courant.
- Séquence de (Macro Def ID, Current Point Spec?).

### **73.9.47 Current Point Spec (Spécification de point courant)**

- Chaque point courant peut être spécifié comme un point absolu, un point relatif dans une longueur, un facteur de la valeur du point d'origine, un facteur de la valeur du point courant, ou comme le résultat d'une action `Get integer` (extraire entier), `Get ratio` (extraire fraction) ou `Get any` (extraire valeur de type «quelconque»). Un entier est interprété comme une valeur absolue. Une fraction est interprétée comme une valeur relative.
- Choix entre (Point, Relative Point, Original Point Factor, Current Point Factor, Get Integer, Get Ratio, Get Any).

#### **73.9.48** Relative Point (**Point relatif**)

- Spécifié sous forme de fraction à mapper sur un intervalle.
- Ratio.

#### **73.9.49** Original Point Factor (**Facteur de point d'origine**)

- Facteur de mise à l'échelle par rapport à la valeur d'origine de ce point. Il est défini comme étant une fraction positive utilisée comme multiplicateur de la valeur de ce point.
- Ratio.

#### **73.9.50** Current Point Factor (**Facteur de point courant**)

- Facteur de mise à l'échelle par rapport à la valeur courante de ce point. Il est défini comme étant une fraction positive utilisée comme multiplicateur de la valeur courante de ce point.
- Ratio.

#### **73.9.51** Duration Spec (**Spécification de durée**)

- Ce paramètre d'action élémentaire spécifie différents moyens de définir une durée.
- Length Spec.

#### **73.9.52** Initial Point Spec Param (**Paramètre de spécification de point initial**)

- Paramètre de spécification de point initial.
- Choix entre (Initial Point Spec, Initial Point Spec Macro).

#### **73.9.53** Initial Point Spec Macro (**Macro-paramètre de spécification de point initial**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de point initial.
- Séquence de (Macro Def ID, Initial Point Spec?).

#### **73.9.54** Terminal Point Spec Param (**Paramètre de spécification de point final**)

- Paramètre de spécification de point final.
- Choix entre (Terminal Point Spec, Terminal Point Spec Macro).

#### **73.9.55** Terminal Point Spec Macro (**Macro-paramètre de spécification de point final**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de spécification de point final.
- Séquence de (Macro Def ID, Terminal Point Spec?).

#### **73.9.56** Initial Point Spec (**Spécification de point initial**)

- Position initiale sur l'axe temporel.
- Point Spec.

#### **73.9.57** Terminal Point Spec (**Spécification de point final**)

- Position finale sur l'axe temporel.
- Point Spec.

#### **73.9.58** Transition Duration Param (**Paramètre de durée de transition**)

- Paramètre facultatif, exprimé en unités GTU, qui peut être fourni pour certaines actions élémentaires dans le comportement de présentation. Lorsqu'une durée de transition est spécifiée, l'action élémentaire correspondante doit être traitée pendant la durée spécifiée, par exemple pour l'action Set audible volume (fixer volume audible), le volume audible passe progressivement de sa valeur précédente à la valeur spécifiée dans la durée de transition spécifiée.
- Generic Integer Param.

## 73.10 Définitions utiles concernant les constantes

Tableau 50/T.171 – Aperçu général de définitions utiles concernant les constantes

**Event Data** ::= Value  
**Event** ::= List  
**Channel Perceptability** ::= *#on / #off*  
**Channel Availability Status** ::= *#available / #not-available*  
**Style** ::= List  
**Max Interact Required** ::= Integer  
**Min Interact Required** ::= Integer  
**Modifiability** ::= *#modifiable / #not-modifiable*  
**Selectability** ::= *#selectable / #not-selectable*  
**Modification Status** ::= *#modified / #not-modified / #modifying*  
**Selection Status** ::= *#selected / #not-selected*  
**Interaction Status** ::= Selection Status | Modification Status  
**Interaction Type** ::= *#selection / #modification*  
**Interaction Type Macro** ::= Macro Def ID, Interaction Type?  
**Interaction Type Param** ::= Interaction Type | Interaction Type Macro  
**Stream Identification** ::= Stream ID | *#all-streams*  
**Stream Identification Macro** ::= Macro Def ID, Stream Identification?  
**Stream Identification Param** ::= Stream Identification | Stream Identification Macro  
**Stream Chosen State** ::= *#chosen / #not-chosen / #partially-chosen*  
**Stream Choice** ::= Generic List  
**GVF** ::= GF  
**PCV** ::= Integer | min-AVR | max-AVR  
**CV** ::= Integer | min-AVR | max-AVR  
**OV** ::= Integer | min-AVR | max-AVR  
**Point Type Param** ::= *#absolute / #relative*  
**Spatial Control Macro** ::= Macro Def ID, Spatial Control?  
**Spatial Control Param** ::= Spatial Control | Spatial Control Macro  
**Expected Axis Result Param** ::= *#x / #y / #z / #xyz*  
**User Spatial Control** ::= *#allowed / #not-allowed*  
**Spatial Control** ::= *#moving / #resizing / #scaling / #scrolling*  
**GSF** ::= X GF?, Y GF?, Z GF?  
**PVS Position** ::= Spatial Position Spec  
**PAP** ::= Spatial Position Spec  
**OVS Proj Strategy** ::= *#fixed / #calculated*  
**OVS Position** ::= Spatial Position Spec  
**OAP** ::= Spatial Position Spec  
**OVS** ::= Size Spec

**Tableau 50/T.171 – Aperçu général de définitions utiles concernant les constantes (fin)**

**Resizing Strategy** ::= #fixed / #minimum / #grows-only

**Aspect Ratio** ::= #preserved / #not-preserved

**OS** ::= Size

**Expected PVD Result** ::= #initial-temporal-position / #terminal-temporal-position / #duration

**Expected PVD Result Macro** ::= Macro Def ID, Expected PVD Result?

**Expected PVD Result Param** ::= Expected PVD Result | Expected PVD Result Macro

**Expected OVD Result** ::= #initial-temporal-position / #terminal-temporal-position / #duration

**Expected OVD Result Macro** ::= Macro Def ID, Expected OVD Result?

**Expected OVD Result Param** ::= Expected OVD Result | Expected OVD Result Macro

**Timestone ID** ::= Generic Integer

**Temporal Termination** ::= #freeze / #stop

**OD** ::= Integer | #infinite

**Presentation Priority** ::= Generic Integer | #up-priority / #down-priority

**Perceptability** ::= Generic Ratio

**RPS Assignment** ::= Channel Reference | Evaluated Reference | #prps

**Ancestor** ::= Integer | #root

**Sibling** ::= Integer

**EmptyChild** ::= Integer | #last

**Child** ::= Generic Integer | #last / #random

**Navigation Command** ::= Child | EmptyChild | Sibling | Ancestor

**Navigation Command Macro** ::= Macro Def ID, Navigation Command?

**Navigation Command Param** ::= Navigation Command | Navigation Command Macro

**Termination Status** ::= #terminated / #not-terminated

**Running Status** ::= #running / #not-running

**Rt-Availability Status** ::= #available / #not-available

**Activation Status** ::= #active / #inactive

**Preparation Status** ::= #ready / #not-ready

**Cat Ext Attribute Macro** ::= Macro Def ID, Cat Ext Attribute?

**Cat Ext Attribute Param** ::= Cat Ext Attribute | Cat Ext Attribute Macro

**Update Command** ::= #add / #remove / #replace

### 73.10.1 Event Data (Données d'événement)

- Spécifie une donnée associée à un événement.
- Value.

### 73.10.2 Event (Événement)

- Spécifie un événement.
- List.



### **73.10.3 Channel Perceptability (Perceptibilité de canal)**

- Spécifie la perceptibilité d'un canal. Choix entre «on» (oui) et «off» (non).
- Choix entre (on, off).

### **73.10.4 Channel Availability Status (Etat de disponibilité de canal)**

- Indique si un canal est disponible.
- Choix entre (available, not-available).

### **73.10.5 Style (Style)**

- Une rubrique cataloguée, enregistrée ou privée, qui donne un style logique à un rt-composant.
- List.

### **73.10.6 Max Interact Required (Interaction maximale requise)**

- Un entier plus grand ou égal à zéro pour indiquer l'interaction maximale requise pour un rt-composant.
- Integer.

### **73.10.7 Min Interact Required (Interaction minimale requise)**

- Un entier plus grand ou égal à zéro pour indiquer l'interaction minimale requise pour un rt-composant.
- Integer.

### **73.10.8 Modifiability (Possibilité de modification)**

- Indique si un rt-composant est modifiable.
- Choix entre (modifiable, not-modifiable).

### **73.10.9 Selectability (Possibilité de sélection)**

- Spécifie une valeur de l'attribut Maximum interaction required (interaction maximale requise) pour la sélection d'un rt-composant.
- Choix entre (selectable, not-selectable).

### **73.10.10 Modification Status (Etat de modification)**

- Indique le résultat de la modification par l'utilisateur.
- Choix entre (modified, not-modified, modifying).

### **73.10.11 Selection Status (Etat de sélection)**

- Indique le résultat de la sélection par l'utilisateur.
- Choix entre (selected, not-selected).

### **73.10.12 Interaction Status (Etat d'interaction)**

- Indique si un rt-composant peut être modifié ou sélectionné.
- Choix entre (Selection Status, Modification Status).

### **73.10.13 Interaction Type (Type d'interaction)**

- Spécifie le type d'interaction.
- Choix entre (selection, modification).

### **73.10.14 Interaction Type Macro (Macro-paramètre de type d'interaction)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de type d'interaction.
- Séquence de (Macro Def ID, Interaction Type?).

### **73.10.15 Interaction Type Param (Paramètre de type d'interaction)**

- Paramètre de type d'interaction.
- Choix entre (Interaction Type, Interaction Type Macro).

#### 73.10.16 Stream Identification (**Identification de flux**)

- Choix entre un identificateur de flux et «tous les flux» («all streams»).
- Choix entre (Stream ID, all-streams).

#### 73.10.17 Stream Identification Macro (**Macro-paramètre d'identification de flux**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'identification de flux.
- Séquence de (Macro Def ID, Stream Identification?).

#### 73.10.18 Stream Identification Param (**Paramètre d'identification de flux**)

- Paramètre d'identification de flux.
- Choix entre (Stream Identification, Stream Identification Macro).

#### 73.10.19 Stream Chosen State (**Etat de choix de flux**)

- Indique l'état de choix de flux pour chaque flux dans un rt-mux, et pour chaque rt-mux.
- Choix entre (chosen, not-chosen, partially-chosen).

#### 73.10.20 Stream Choice (**Choix de flux**)

- Une liste d'identificateurs de flux choisis dans un rt-mux.
- Generic List.

#### 73.10.21 GVF (**Facteur GVF**)

- Spécifie le facteur GVF à appliquer pendant la projection de l'espace OPS dans son espace RPS.
- GF.

#### 73.10.22 PCV (**Volume PCV**)

- Spécifie le volume PCV d'un objet *contenu* qui est une projection du volume sonore courant dans son espace RPS parent.
- Choix entre (Integer, min-AVR, max-AVR).

#### 73.10.23 CV (**Volume sonore courant**)

- Spécifie le volume sonore courant d'un objet *contenu*. Ce volume est initialisé par le volume sonore d'origine et il peut être modifié en utilisant l'action Set CV (fixer volume sonore courant).
- Choix entre (Integer, min-AVR, max-AVR).

#### 73.10.24 OV (**Volume sonore d'origine**)

- Spécifie le volume sonore d'origine d'un objet *contenu*. Ce volume est défini dans la plage AVR. Il sert à initialiser le volume sonore courant de chaque rt-contenu créé à partir de ce contenu.
- Choix entre (Integer, min-AVR, max-AVR).

#### 73.10.25 Point Type Param (**Paramètre de type de point**)

- Choix entre «absolu» et «relatif».
- Choix entre (absolute, relative).

#### 73.10.26 Spatial Control Macro (**Macro-paramètre de commande spatiale**)

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de commande spatiale.
- Séquence de (Macro Def ID, Spatial Control?).

#### 73.10.27 Spatial Control Param (**Paramètre de commande spatiale**)

- Choix entre «déplacement», «redimensionnement», «mise à l'échelle» et «défilement».
- Choix entre (Spatial Control, Spatial Control Macro).

**73.10.28 Expected Axis Result Param (Paramètre de résultat d'axe attendu)**

- Choix entre «x», «y», «z» ou «xyz».
- Choix entre (x, y, z, xyz).

**73.10.29 User Spatial Control (Commande spatiale de l'utilisateur)**

- Indique les fonctions de la commande spatiale de l'utilisateur qui sont autorisées et celles qui ne le sont pas.
- Choix entre (allowed, not-allowed).

**73.10.30 Spatial Control (Commande spatiale)**

- Indique les fonctions de la commande spatiale de l'utilisateur.
- Choix entre (moving, resizing, scaling, scrolling).

**73.10.31 GSF (Facteur GSF)**

- Un ensemble de trois facteurs génériques utilisés pour chaque axe.
- Séquence de (X GF?, Y GF?, Z GF?).

**73.10.32 PVS Position (Position PVS)**

- Spécifie la position de la taille PVS par rapport à la taille d'origine ou à une autre taille PVS.
- Spatial Position Spec.

**73.10.33 PAP (Point PAP)**

- Spécifie le point PAP de la taille PVS. Il est utilisé pour placer la taille PVS dans l'espace RPS.
- Spatial Position Spec.

**73.10.34 OVS Proj Strategy (Stratégie de projection de la taille OVS)**

- Spécifie la stratégie de projection de la taille OVS dans la taille PVS.
- Choix entre (fixed, calculated).

**73.10.35 OVS Position (Position OVS)**

- Spécifie la position de la taille OVS par rapport à la taille d'origine.
- Spatial Position Spec.

**73.10.36 OAP (Point OAP)**

- Spécifie le point OAP de la taille OVS. Il est utilisé pour placer la taille OVS dans sa taille d'origine.
- Spatial Position Spec.

**73.10.37 OVS (Taille OVS)**

- Spécifie la taille OVS d'un rt-composant. Cette taille est spécifiée comme un sous-ensemble de la taille d'origine.
- Size Spec.

**73.10.38 Resizing Strategy (Stratégie de redimensionnement)**

- Définit la stratégie de redimensionnement d'un rt-composite.
- Choix entre (fixed, minimum, grows-only).

**73.10.39 Aspect Ratio (Facteur de forme)**

- Indique si le facteur de forme d'un rt-composant doit être préservé.
- Choix entre (preserved, not-preserved).

**73.10.40 OS (Taille d'origine)**

- Spécifie la taille d'origine d'un objet *composant* ou d'un canal. Sert à initialiser la taille d'origine d'un rt-composant. Sert également à établir les axes spatiaux de l'espace OPS, de chaque rt-composant créé à partir de ce composant, ou de l'espace CPS.
- Size.

- 73.10.41 Expected PVD Result (Résultat PVD attendu)**
- Choix entre (initial-temporal-position, terminal-temporal-position, duration).
- 73.10.42 Expected PVD Result Macro (Macro-paramètre de résultat PVD attendu)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de résultat PVD attendu.
  - Séquence de (Macro Def ID, Expected PVD Result?).
- 73.10.43 Expected PVD Result Param (Paramètre de résultat PVD attendu)**
- Paramètre de résultat PVD attendu.
  - Choix entre (Expected PVD Result, Expected PVD Result Macro).
- 73.10.44 Expected OVD Result (Résultat OVD attendu)**
- Choix entre (initial-temporal-position, terminal-temporal-position, duration).
- 73.10.45 Expected OVD Result Macro (Macro-paramètre de résultat OVD attendu)**
- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de résultat OVD attendu.
  - Séquence de (Macro Def ID, Expected OVD Result?).
- 73.10.46 Expected OVD Result Param (Paramètre de résultat OVD attendu)**
- Paramètre de résultat OVD attendu.
  - Choix entre (Expected OVD Result, Expected OVD Result Macro).
- 73.10.47 Timestone ID (Identificateur de marqueur temporel)**
- Un entier générique. La valeur 0 est réservée.
  - Generic Integer.
- 73.10.48 Temporal Termination (Achèvement temporel)**
- Détermine la présentation lorsqu'une cible atteint la position temporelle finale.
  - Choix entre (freeze, stop).
- 73.10.49 OD (Durée d'origine)**
- Spécifie la durée d'origine d'un objet *composant* ou d'un canal. Sert à initialiser la durée d'origine d'un rt-composant. Sert également à établir les axes spatiaux de l'espace OPS, de chaque rt-composant créé à partir de ce composant, ou de l'espace CPS.
  - Choix entre (Integer, infinite).
- 73.10.50 Presentation Priority (Priorité de présentation)**
- Détermine l'ordre d'empilage des présentations de rt-composants affectés au même espace RPS.
  - Choix entre (Generic Integer, up-priority, down-priority).
- 73.10.51 Perceptability (Perceptibilité)**
- Une fraction qui définit la perceptibilité d'un rt-composant. Si l'état de perceptibilité est mis à zéro, le rt-contenu n'est pas perçu. S'il est mis à 100, la perception d'origine du rt-contenu est perçue.
  - Generic Ratio.
- 73.10.52 RPS Assignment (Affectation d'espace RPS)**
- L'espace RPS est une référence de canal, une référence évaluée ou un espace PRPS.
  - Choix entre (Channel Reference, Evaluated Reference, prps).
- 73.10.53 Ancestor (Ancêtre)**
- Chemin de navigation pour atteindre un ancêtre.
  - Choix entre (Integer, root).

#### **73.10.54 Sibling (Frère)**

- Chemin de navigation pour atteindre un frère.
- Integer.

#### **73.10.55 EmptyChild (Enfant vide)**

- Chemin de navigation pour atteindre un réceptacle vide.
- Choix entre (Integer, last).

#### **73.10.56 Child (Enfant)**

- Chemin de navigation pour atteindre un enfant.
- Choix entre (Generic Integer, last, random).

#### **73.10.57 Navigation Command (Commande de navigation)**

- Indique un trajet de navigation.
- Choix entre (Child, EmptyChild, Sibling, Ancestor).

#### **73.10.58 Navigation Command Macro (Macro-paramètre de commande de navigation)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre de commande de navigation.
- Séquence de (Macro Def ID, Navigation Command?).

#### **73.10.59 Navigation Command Param (Paramètre de commande de navigation)**

- Paramètre de commande de navigation.
- Choix entre (Navigation Command, Navigation Command Macro).

#### **73.10.60 Termination Status (Etat d'achèvement)**

- Représente le traitement d'un rt-script.
- Choix entre (terminated, not-terminated).

#### **73.10.61 Running Status (Etat d'exécution)**

- Indique si un rt-objet est exécuté.
- Choix entre (running, not-running).

#### **73.10.62 Rt-Availability Status (Etat de rt-disponibilité)**

- Indique si le rt-objet a été créé à partir de l'objet modèle.
- Choix entre (available, not-available).

#### **73.10.63 Activation Status (Etat d'activation)**

- Indique si un objet *lien* a été activé.
- Choix entre (active, inactive).

#### **73.10.64 Preparation Status (Etat de préparation)**

- Indique la disponibilité des objets MHEG pour la machine MHEG.
- Choix entre (ready, not-ready).

#### **73.10.65 Cat Ext Attribute Macro (Macro-paramètre d'attribut étendu catalogué)**

- Permet de spécifier un macro-paramètre au lieu d'une valeur spécifique pour le paramètre d'attribut étendu catalogué.
- Séquence de (Macro Def ID, Cat Ext Attribute?).

### 73.10.66 Cat Ext Attribute Param (**Paramètre d'attribut étendu catalogué**)

- Paramètre d'attribut étendu catalogué.
- Choix entre (Cat Ext Attribute, Cat Ext Attribute Macro).

### 73.10.67 Update Command (**Commande de mise à jour**)

- Lorsque l'attribut d'une entité MHEG prend une liste de valeurs, ce paramètre est fourni pour la mise à jour de cette liste de valeurs. Ce paramètre est utilisé dans les actions Set Alias (Fixer alias), Set Timestone (Fixer marqueur temporel), Set Stream Choice (Fixer choix de flux) et Attach Anchor (Attacher ancre). Il est toujours utilisé comme spécification d'attribut = Liste de (List of attribute value, Update Command)\*.

Les dispositions suivantes s'appliquent:

si la commande de mise à jour est égale à «remplacer», la liste des valeurs d'attribut courantes de la cible est remplacée par la liste de valeurs d'attribut fournie comme paramètre de l'action élémentaire;

si la commande de mise à jour est égale à «ajouter», la liste de valeurs d'attribut fournie comme paramètre de l'action élémentaire est ajoutée à la liste des valeurs d'attribut courantes de la cible. Elle est ignorée pour ajouter un attribut qui existe déjà dans la liste des valeurs d'attribut courantes;

si la commande de mise à jour est égale à «supprimer», la liste de valeurs d'attribut fournie comme paramètre de l'action élémentaire est supprimée de la liste des valeurs d'attribut courantes de la cible. Elle est ignorée pour supprimer un attribut qui n'existe pas dans la liste des valeurs d'attribut courantes;

la combinaison des commandes de mise à jour «ajouter» et «supprimer» peut être utilisée dans une liste de spécification d'attribut donnée, auquel cas chaque commande doit être évaluée en suivant l'ordre de la liste;

la commande de mise à jour «remplacer» ne doit pas être utilisée avec «ajouter» ou «supprimer» dans une liste de spécification d'attribut donnée, autrement l'action élémentaire est ignorée.

- Choix entre (add, remove, replace).

## 73.11 Constantes de valeurs de comparaison

**Tableau 51/T.171 – Aperçu général de constantes de valeurs de comparaison**

#### **Comparison Value Constant ::=**

Preparation Status | Activation Status | Rt-Availability Status | Running Status | Termination Status | #prps / #infinite |  
Temporal Termination | Aspect Ratio | Resizing Strategy | OVS Proj Strategy | User Spatial Control | Stream Chosen State |  
Selection Status | Modification Status | Channel Availability Status | Channel Perceptability

### 73.11.1 Comparison Value Constant (**Constante de valeur de comparaison**)

- Spécification des constantes définies par la présente Recommandation et qui peuvent être utilisées comme valeurs de comparaison dans une condition de lien d'un objet *lien*.
- Choix entre (Preparation Status, Activation Status, Rt-Availability Status, Running Status, Termination Status, prps, infinite, Temporal Termination, Aspect Ratio, Resizing Strategy, OVS Proj Strategy, User Spatial Control, Stream Chosen State, Selection Status, Modification Status, Channel Availability Status, Channel Perceptability).

## Annexe A

### Notations ASN.1 (Niveau C) Représentation codée (Niveau D)

Recommandation T.171 (10/96)

#### Useful definitions

-- Droits d'auteur:

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definitions(9)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

#### EXPORTS

Aspect-Ratio,  
Cat-Content-Classification,  
Cat-Ext-Attribute,  
Cat-Ext-Attribute-Param,  
Catalogued-Event,  
Catalogued-Extended-Elementary-Action,  
Catalogued-Media-Type,  
Catalogued-Script-Classification,  
Catalogued-Style,  
Channel-Perceptability,  
Channel-Target-Param,  
Class-ID,  
Comparison-Value-Constant,  
Component-Class,  
Content-Hook,  
Current-Point-Spec-Param,  
Data-Reference,  
Evaluated-Integer,  
Evaluated-List,  
Evaluated-Reference,  
Evaluated-Value,  
Generic-Integer,  
Generic-Integer-Param,  
Generic-List-Elt-ID-Param,  
Generic-List-Param,  
Generic-Numeric-Param,  
Generic-Ratio-Param,  
Generic-Value,  
Generic-Value-Param,  
GF-Param,  
Initial-Point-Spec-Param,  
Interaction-Status,  
Interaction-Type-Param,  
Macro-Def-ID,  
Mh-object-Class,  
Mh-Reference,  
Mh-Target-Param,  
Model-Class,  
OVS-Proj-Strategy,  
Point-Spec,  
Point-Spec-Param,  
Presentation-Priority,  
Resizing-Strategy,  
Return-Target-Param,  
RPS-Assignment,  
Rt-Component-Channel-Tg-Param,

**Rt-Component-Reference,**  
**Rt-Component-Target-Param,**  
**Rt-Composite-Target-Param,**  
**Rt-Content-Reference,**  
**Rt-Content-Target-Param,**  
**Rt-Mux-Target-Param,**  
**Rt-Script-Target-Param,**  
**Rt-Target-Param,**  
**Script-Hook,**  
**Size,**  
**Size-Spec-Param,**  
**Socket-Target-Param,**  
**Spatial-Control-Param,**  
**Spatial-Position-Spec-Param,**  
**Target-Param,**  
**Temporal-Termination,**  
**Terminal-Point-Spec-Param,**  
**Update-Command,**  
**User-Spatial-Control,**  
**Value,**  
**OV;**

#### IMPORTS

**Registered-Content-Encoding,**  
**Registered-Script-Encoding,**  
**Registered-Content-Classification,**  
**Registered-Script-Classification,**  
**Registered-Media-Type,**  
**Registered-Style,**  
**Registered-Event,**  
**Registered-Extended-Elementary-Action,**  
**Registered-Extended-Attribute FROM ISOMHEG-cat {joint-iso-itu-t(2) mheg(19) version(1) catalogues(12)};**

```

Mh-object-Class ::= SEQUENCE {
mheg-id                Mheg-ID OPTIONAL,
description            [0] Description OPTIONAL
}
Class-ID ::= CHOICE {
action-class-id       [0] INTEGER { action-class-id (1)},
link-class-id        [1] INTEGER { link-class-id (2)},
script-class-id      [2] INTEGER { script-class-id (3)},
content-class-id     [3] INTEGER { content-class-id (4)},
multiplexed-content-class-id [4] INTEGER { multiplexed-content-class-id (5)},
composite-class-id   [5] INTEGER { composite-class-id (6)},
container-class-id   [6] INTEGER { container-class-id (7)},
descriptor-class-id  [7] INTEGER { descriptor-class-id (8)},
...
}
Description ::= SEQUENCE {
name                  GraphicString OPTIONAL,
owner                [0] GraphicString OPTIONAL,
version              [1] GraphicString OPTIONAL,
date                 UTCTime OPTIONAL,
keywords             SEQUENCE OF GraphicString OPTIONAL,
copyright            [2] GraphicString OPTIONAL,
copyright-id         OCTET STRING OPTIONAL,
copyright-number     [3] OCTET STRING OPTIONAL,
licence              [4] GraphicString OPTIONAL,
cache-priority       INTEGER OPTIONAL,
comments            [5] GraphicString OPTIONAL,
...
}
Model-Class ::= SEQUENCE {
COMPONENTS OF Mh-object-Class
}
Component-Class ::= SEQUENCE {
COMPONENTS OF Model-Class,
ops-initialisation   [1] OPS-Initialisation OPTIONAL
}

```



```

Tail-Param ::= CHOICE
tail
tail-macro
}
Tail-Macro ::= SEQUENCE
macro-def-id
tail
}
External-ID ::= SEQUENCE
public-id
system-id
}
Mheg-ID ::= SEQUENCE
application-id
mh-number
}
Socket-Identification ::= CHOICE
socket-id
alias
}
Socket-ID ::= SEQUENCE
root-rt-composite-reference
tail
}
Macro-Def-ID ::= CHOICE
string
integer
}
Reference ::= CHOICE
null-data
mheg-id-reference
external-id-reference
container-element-reference
null-mh
this
stream-id-reference
root-rt-id-reference
null-root-rt
socket-id-reference
channel-id-reference
default-channel
alias-reference
}
Tail-Reference ::= SEQUENCE
tail
tail-complement
}
Tail-Complement ::= ENUMERATED
children (1),
descendants (2)
}
Data-Reference ::= CHOICE
external-id
null-data
alias-reference
}
Mh-Reference ::= CHOICE
mheg-id-reference
external-id-reference
container-element-reference
null-mh
this
alias-reference
}
Container-Element-Reference ::= SEQUENCE
container-object-reference
tail-reference
}

```

```

Root-Rt-Reference ::= CHOICE {
root-rt-id-reference      Root-Rt-ID-Reference,
null-root-rt              NULL,
alias-reference           GraphicString
}
Root-Rt-ID-Reference ::= SEQUENCE {
model-object-reference    Mh-Reference,
rt-number-reference      Rt-Number-Reference
}
Rt-Number-Reference ::= CHOICE {
rt-number                INTEGER,
question-mark            [0] INTEGER { question-mark (1)},
star                     [1] INTEGER { star (2)}
}
Rt-Reference ::= CHOICE {
root-rt-id-reference      Root-Rt-ID-Reference,
null-root-rt              NULL,
socket-id-reference       [0] Socket-ID-Reference,
alias-reference           GraphicString
}
Rt-Script-Reference ::= CHOICE {
rt-script-id-reference    Rt-Script-ID-Reference,
null-root-rt              NULL,
alias-reference           GraphicString
}
Rt-Script-ID-Reference ::= SEQUENCE {
script-object-reference   Mh-Reference,
rt-number-reference      Rt-Number-Reference
}
Root-Rt-Component-Reference ::= CHOICE {
root-rt-component-id-ref  Root-Rt-Component-ID-Ref,
null-root-rt              NULL,
alias-reference           GraphicString
}
Root-Rt-Component-ID-Ref ::= SEQUENCE {
component-object-reference Mh-Reference,
rt-number-reference      Rt-Number-Reference
}
Rt-Component-Reference ::= CHOICE {
root-rt-component-id-ref  [10] Root-Rt-Component-ID-Ref,
null-root-rt              [11] NULL,
socket-id-reference       [12] Socket-ID-Reference,
alias-reference           [13] GraphicString
}
Root-Rt-Content-Reference ::= CHOICE {
root-rt-content-id-ref    Root-Rt-Content-ID-Ref,
null-root-rt              NULL,
alias-reference           GraphicString
}
Root-Rt-Content-ID-Ref ::= SEQUENCE {
content-object-ref        Mh-Reference,
rt-number-reference      Rt-Number-Reference
}
Rt-Content-Reference ::= CHOICE {
root-rt-content-id-ref    Root-Rt-Content-ID-Ref,
null-root-rt              NULL,
socket-id-reference       [0] Socket-ID-Reference,
alias-reference           GraphicString
}
Root-Rt-Mux-Reference ::= CHOICE {
root-rt-mux-id-reference  Root-Rt-Mux-ID-Reference,
null-root-rt              NULL,
alias-reference           GraphicString
}
Root-Rt-Mux-ID-Reference ::= SEQUENCE {
multiplexed-content-object-ref Mh-Reference,
rt-number-reference      Rt-Number-Reference
}

```

```

Rt-Mux-Reference ::= CHOICE {
root-rt-mux-id-reference      Root-Rt-Mux-ID-Reference,
null-root-rt                  NULL,
socket-id-reference           [0] Socket-ID-Reference,
alias-reference                GraphicString
}
Root-Rt-Composite-Reference ::= CHOICE {
root-rt-composite-id-ref      Root-Rt-Composite-ID-Ref,
null-root-rt                  NULL,
alias-reference                GraphicString
}
Root-Rt-Composite-ID-Ref ::= SEQUENCE {
composite-object-reference     Mh-Reference,
rt-number-reference            Rt-Number-Reference
}
Rt-Composite-Reference ::= CHOICE {
root-rt-composite-id-ref      [28] Root-Rt-Composite-ID-Ref,
null-root-rt                  [29] NULL,
socket-id-reference           [30] Socket-ID-Reference,
alias-reference                [31] GraphicString
}
Socket-Reference ::= CHOICE {
socket-id-reference            Socket-ID-Reference,
alias-reference                GraphicString
}
Socket-ID-Reference ::= SEQUENCE {
root-rt-composite-reference    Root-Rt-Composite-Reference,
socket-tail-reference          Socket-Tail-Reference
}
Socket-Tail-Reference ::= SEQUENCE {
tail                           SEQUENCE OF INTEGER OPTIONAL,
socket-tail-complement         Socket-Tail-Complement OPTIONAL
}
Socket-Tail-Complement ::= CHOICE {
tail-complement                Tail-Complement,
question-mark-child            INTEGER { question-mark-child (1)},
question-mark-descendant       [0] INTEGER { question-mark-descendant (2)}
}
Channel-Reference ::= CHOICE {
channel-id-reference           [34] INTEGER,
default-channel                [35] NULL,
alias-reference                [36] GraphicString
}
Rt-Component-Channel-Ref ::= CHOICE {
rt-component-reference         [41] Rt-Component-Reference,
channel-reference              [42] Channel-Reference
}
Target-Param ::= CHOICE {
target                          Generic-Reference,
target-macro                    [45] Target-Macro
}
Target-Macro ::= SEQUENCE {
macro-def-id                   Macro-Def-ID,
target                          Generic-Reference OPTIONAL
}
Mh-Target-Param ::= CHOICE {
mh-target                       [2] Mh-Target,
mh-target-macro                 [3] Mh-Target-Macro
}
Mh-Target-Macro ::= SEQUENCE {
macro-def-id                   Macro-Def-ID,
mh-target                       Mh-Target OPTIONAL
}
Mh-Target ::= CHOICE {
mh-reference                    Mh-Reference,
evaluated-reference             Evaluated-Reference
}

```

```

Rt-Target-Param ::= CHOICE    {
rt-target          [5] Rt-Target,
rt-target-macro    [6] Rt-Target-Macro
}
Rt-Target-Macro ::= SEQUENCE {
macro-def-id      Macro-Def-ID,
rt-target         Rt-Target OPTIONAL
}
Rt-Target ::= CHOICE        {
rt-reference      Rt-Reference,
evaluated-reference  Evaluated-Reference
}
Rt-Script-Target-Param ::= CHOICE    {
rt-script-target  [8] Rt-Script-Target,
rt-script-target-macro [9] Rt-Script-Target-Macro
}
Rt-Script-Target-Macro ::= SEQUENCE {
macro-def-id      Macro-Def-ID,
rt-script-target  Rt-Script-Target OPTIONAL
}
Rt-Script-Target ::= CHOICE    {
rt-script-reference  Rt-Script-Reference,
evaluated-reference  Evaluated-Reference
}
Rt-Component-Target-Param ::= CHOICE {
rt-component-target  Rt-Component-Target,
rt-component-target-macro [15] Rt-Component-Target-Macro
}
Rt-Component-Target-Macro ::= SEQUENCE {
macro-def-id      Macro-Def-ID,
rt-component-target  Rt-Component-Target OPTIONAL
}
Rt-Component-Target ::= CHOICE {
rt-component-reference  Rt-Component-Reference,
evaluated-reference     [14] Evaluated-Reference
}
Rt-Content-Target-Param ::= CHOICE {
rt-content-target      [26] Rt-Content-Target,
rt-content-target-macro [27] Rt-Content-Target-Macro
}
Rt-Content-Target-Macro ::= SEQUENCE {
macro-def-id      Macro-Def-ID,
rt-content-target  Rt-Content-Target OPTIONAL
}
Rt-Content-Target ::= CHOICE {
rt-content-reference  Rt-Content-Reference,
evaluated-reference   Evaluated-Reference
}
Rt-Mux-Target-Param ::= CHOICE {
rt-mux-target        [45] Rt-Mux-Target,
rt-mux-target-macro  [46] Rt-Mux-Target-Macro
}
Rt-Mux-Target-Macro ::= SEQUENCE {
macro-def-id      Macro-Def-ID,
rt-mux-target     Rt-Mux-Target OPTIONAL
}
Rt-Mux-Target ::= CHOICE {
rt-mux-reference  Rt-Mux-Reference,
evaluated-reference  Evaluated-Reference
}
Rt-Composite-Target-Param ::= CHOICE {
rt-composite-target  Rt-Composite-Target,
rt-composite-target-macro [33] Rt-Composite-Target-Macro
}
Rt-Composite-Target-Macro ::= SEQUENCE {
macro-def-id      Macro-Def-ID,
rt-composite-target  Rt-Composite-Target OPTIONAL
}

```

```

Rt-Composite-Target ::= CHOICE {
rt-composite-reference      Rt-Composite-Reference,
evaluated-reference        [32] Evaluated-Reference
}
Socket-Target-Param ::= CHOICE {
socket-target              Socket-Target,
socket-target-macro        [0] Socket-Target-Macro
}
Socket-Target-Macro ::= SEQUENCE {
macro-def-id               Macro-Def-ID,
socket-target              Socket-Target OPTIONAL
}
Socket-Target ::= CHOICE {
socket-reference           Socket-Reference,
evaluated-reference       Evaluated-Reference
}
Return-Target-Param ::= CHOICE {
generic-integer-param     Generic-Integer-Param,
using-application        NULL
}
Channel-Target-Param ::= CHOICE {
channel-target            Channel-Target,
channel-target-macro     [38] Channel-Target-Macro
}
Channel-Target-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
channel-target            Channel-Target OPTIONAL
}
Channel-Target ::= CHOICE {
channel-reference         Channel-Reference,
evaluated-reference     [37] Evaluated-Reference
}
Rt-Component-Channel-Tg-Param ::= CHOICE {
rt-component-channel-tg   Rt-Component-Channel-Tg,
rt-component-channel-tg-macro [44] Rt-Component-Channel-Tg-Macro
}
Rt-Component-Channel-Tg-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
rt-component-channel-tg   Rt-Component-Channel-Tg OPTIONAL
}
Rt-Component-Channel-Tg ::= CHOICE {
rt-component-channel-ref  Rt-Component-Channel-Ref,
evaluated-reference      [43] Evaluated-Reference
}
Generic-Value-Param ::= CHOICE {
generic-value             Generic-Value,
generic-value-macro      [50] Generic-Value-Macro
}
Generic-Value-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
generic-value            Generic-Value OPTIONAL
}
Generic-Value ::= CHOICE {
value                     Value,
evaluated-value          Evaluated-Value
}
Value ::= CHOICE {
boolean                   BOOLEAN,
numeric                   INTEGER,
ratio                     Ratio,
string                    GraphicString,
list                      [0] SEQUENCE OF Value,
reference                  [1] Reference
}
Generic-Boolean-Param ::= CHOICE {
generic-boolean           Generic-Boolean,
generic-boolean-macro    Generic-Boolean-Macro
}

```

```

Generic-Boolean-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-boolean        Generic-Boolean OPTIONAL
}
Generic-Boolean ::= CHOICE {
boolean                BOOLEAN,
evaluated-boolean     Evaluated-Boolean
}
Generic-Numeric-Param ::= CHOICE {
generic-numeric        Generic-Numeric,
generic-numeric-macro Generic-Numeric-Macro
}
Generic-Numeric-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-numeric        Generic-Numeric OPTIONAL
}
Generic-Numeric ::= CHOICE {
numeric                INTEGER,
evaluated-numeric     Evaluated-Numeric
}
Generic-Integer-Param ::= CHOICE {
generic-integer        Generic-Integer,
generic-integer-macro Generic-Integer-Macro
}
Generic-Integer-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-integer        Generic-Integer OPTIONAL
}
Generic-Integer ::= CHOICE {
integer                INTEGER,
evaluated-integer     Evaluated-Integer
}
Generic-Ratio-Param ::= CHOICE {
generic-ratio          Generic-Ratio,
generic-ratio-macro   [0] Generic-Ratio-Macro
}
Generic-Ratio-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-ratio          Generic-Ratio OPTIONAL
}
Generic-Ratio ::= CHOICE {
ratio                  Ratio,
evaluated-ratio       Evaluated-Ratio
}
Ratio ::= SEQUENCE {
numerator              INTEGER,
denominator            INTEGER OPTIONAL
}
Generic-String-Param ::= CHOICE {
generic-string         Generic-String,
generic-string-macro  Generic-String-Macro
}
Generic-String-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-string         Generic-String OPTIONAL
}
Generic-String ::= CHOICE {
string                 GraphicString,
evaluated-string      Evaluated-String
}
Generic-Reference-Param ::= CHOICE {
generic-reference      Generic-Reference,
generic-reference-macro [45] Generic-Reference-Macro
}

```

```

Generic-Reference-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-reference      Generic-Reference OPTIONAL
}
Generic-Reference ::= CHOICE {
reference              Reference,
evaluated-reference   Evaluated-Reference
}
Generic-List-Param ::= CHOICE {
generic-list          Generic-List,
generic-list-macro    [1] Generic-List-Macro
}
Generic-List-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-list           Generic-List OPTIONAL
}
Generic-List ::= CHOICE {
list                  SEQUENCE OF Value,
evaluated-list        Evaluated-List
}
Generic-List-Elt-ID-Param ::= CHOICE {
generic-list-elt-id    SEQUENCE OF INTEGER,
generic-list-elt-id-macro [0] Generic-List-Elt-ID-Macro
}
Generic-List-Elt-ID-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
generic-list-elt-id    SEQUENCE OF INTEGER OPTIONAL
}
Get-Number-Of-Interacted-Sockets ::= SEQUENCE {
rt-composite-target-param Rt-Composite-Target-Param,
interaction-type-param    Interaction-Type-Param
}
Get-Interaction-Status ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
interaction-type-param    Interaction-Type-Param
}
Get-Max-Interact-Required ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
interaction-type-param    Interaction-Type-Param
}
Get-Min-Interact-Required ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
interaction-type-param    Interaction-Type-Param
}
Get-Interaction-Ability ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
interaction-type-param    Interaction-Type-Param
}
Get-Stream-Chosen-State ::= SEQUENCE {
rt-mux-target-param      Rt-Mux-Target-Param,
stream-identification-param Stream-Identification-Param
}
Get-GVF ::= CHOICE {
rt-composite-target-param Rt-Composite-Target-Param,
channel-target-param      Channel-Target-Param
}
Get-User-Spatial-Control ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
spatial-control-param     Spatial-Control-Param
}
Get-GSF ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
expected-axis-result-param Expected-Axis-Result-Param
}
Get-PVS-Position ::= SEQUENCE {
rt-component-target-param Rt-Component-Target-Param,
point-type-param         Point-Type-Param,
expected-axis-result-param Expected-Axis-Result-Param
}

```

```

Get-PAP ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
point-type-param               Point-Type-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-PVS ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
point-type-param               Point-Type-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-OVS-Position ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
point-type-param               Point-Type-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-OAP ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
point-type-param               Point-Type-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-OVS ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
point-type-param               Point-Type-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-POS ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-OS ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
expected-axis-result-param     Expected-Axis-Result-Param
}
Get-PVD ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
expected-pvd-result-param      Expected-PVD-Result-Param
}
Get-OVD ::= SEQUENCE {
rt-component-target-param      Rt-Component-Target-Param,
expected-ovd-result-param      Expected-OVD-Result-Param
}
Get-Rt-Composite-Address ::= SEQUENCE
{
rt-composite-target-param      Rt-Composite-Target-Param,
navigation-command-param      Navigation-Command-Param
}
Get-Data ::= SEQUENCE {
content-target-param           Mh-Target-Param,
generic-list-elt-id-param      Generic-List-Elt-ID-Param OPTIONAL
}
Get-Catalogued-Attribute ::= SEQUENCE {
target-param                   Target-Param,
cat-ext-attribute-param        Cat-Ext-Attribute-Param
}
Get-Integer ::= CHOICE {
get-preparation-status         Mh-Target-Param,
get-activation-status          [4] Mh-Target-Param,
get-rt-availability-status     Rt-Target-Param,
get-running-status             [7] Rt-Target-Param,
get-termination-status         Rt-Script-Target-Param,
get-presentation-priority      Rt-Component-Target-Param,
get-od                         [16] Rt-Component-Target-Param,
get-pod                        [17] Rt-Component-Target-Param,
get-pvd                        [18] Get-PVD,
get-ctp                        [19] Rt-Component-Target-Param,
get-temporal-termination       [20] Rt-Component-Target-Param,
get-pvd-position              [21] Rt-Component-Target-Param,
}

```



get-timestone-status	[22] Rt-Component-Target-Param,
get-aspect-ratio	[23] Rt-Component-Target-Param,
get-resizing-strategy	Rt-Composite-Target-Param,
get-ovs-proj-strategy	[24] Rt-Component-Target-Param,
get-user-spatial-control	[25] Get-User-Spatial-Control,
get-cv	Rt-Content-Target-Param,
get-pcv	[34] Rt-Content-Target-Param,
get-stream-chosen-state	[35] Get-Stream-Chosen-State,
get-min-interact-required	[36] Get-Min-Interact-Required,
get-max-interact-required	[37] Get-Max-Interact-Required,
get-number-of-interacted-sockets	[38] Get-Number-Of-Interacted-Sockets,
get-channel-availability-status	[39] Channel-Target-Param,
get-channel-perceptability	[40] Channel-Target-Param
}	
Get-Reference ::= CHOICE	{
get-rt-composite-address	[9] Get-Rt-Composite-Address,
get-rps-assignment	Rt-Component-Target-Param
}	
Get-List ::= CHOICE	{
get-catalogued-attribute	[44] Get-Catalogued-Attribute,
get-stream-choice	Rt-Mux-Target-Param,
get-style	[47] Rt-Component-Target-Param,
get-event	[48] Rt-Component-Channel-Tg-Param
}	
Get-Any ::= CHOICE	{
get-data	[16] Get-Data,
get-ovd	[17] Get-OVD,
get-gtf	[18] Rt-Component-Target-Param,
get-os	[19] Get-OS,
get-pos	[20] Get-POS,
get-ovs	[21] Get-OVS,
get-oap	[22] Get-OAP,
get-ovs-position	[23] Get-OVS-Position,
get-pvs	[24] Get-PVS,
get-pap	[25] Get-PAP,
get-pvs-position	[26] Get-PVS-Position,
get-gsf	[27] Get-GSF,
get-gvf	Get-GVF,
get-interaction-ability	[39] Get-Interaction-Ability,
get-interaction-status	[40] Get-Interaction-Status,
get-event-data	Rt-Component-Channel-Tg-Param
}	
Evaluated-Value ::= CHOICE	{
get-boolean	NULL,
get-numeric	Get-Integer,
get-ratio	[41] Rt-Component-Target-Param,
get-string	[42] NULL,
get-reference	[43] Get-Reference,
get-list	Get-List,
get-any	[49] Get-Any,
...	
}	
Evaluated-Boolean ::= CHOICE	{
get-boolean	NULL,
get-any	Get-Any
}	
Evaluated-Numeric ::= CHOICE	{
get-numeric	Get-Integer,
get-integer	[0] Get-Integer,
get-any	[1] Get-Any
}	
Evaluated-Integer ::= CHOICE	{
get-integer	Get-Integer,
get-any	[41] Get-Any
}	
Evaluated-Ratio ::= CHOICE	{
get-ratio	Rt-Component-Target-Param,
get-any	Get-Any
}	

```

Evaluated-String ::= CHOICE      {
get-string                       NULL,
get-any                           Get-Any
}
Evaluated-Reference ::= CHOICE {
get-reference                     Get-Reference,
get-any                             Get-Any
}
Evaluated-List ::= CHOICE       {
get-list                           Get-List,
get-any                               [0] Get-Any
}
Content-Hook ::= SEQUENCE      {
catalogued-content-encoding       Catalogued-Content-Encoding OPTIONAL,
content-encoding-description     [0] OCTET STRING OPTIONAL
}
Script-Hook ::= SEQUENCE      {
catalogued-script-encoding        Catalogued-Script-Encoding OPTIONAL,
script-encoding-description     [0] OCTET STRING OPTIONAL
}
Catalogued-Content-Encoding ::= CHOICE {
registered-content-encoding       Registered-Content-Encoding,
proprietary-content-encoding     Prop-Cat-Entry-ID
}
Catalogued-Script-Encoding ::= CHOICE {
registered-script-encoding        Registered-Script-Encoding,
proprietary-script-encoding     Prop-Cat-Entry-ID
}
Cat-Content-Classification ::= CHOICE {
registered-content-classification Registered-Content-Classification,
prop-content-classification      Prop-Cat-Entry-ID
}
Catalogued-Script-Classification ::= CHOICE {
registered-script-classification  Registered-Script-Classification,
prop-script-classification       Prop-Cat-Entry-ID
}
Catalogued-Media-Type ::= CHOICE {
registered-media-type             Registered-Media-Type,
proprietary-media-type           Prop-Cat-Entry-ID
}
Catalogued-Style ::= CHOICE   {
registered-style                 Registered-Style,
proprietary-style               Prop-Cat-Entry-ID
}
Catalogued-Event ::= CHOICE   {
registered-event                 Registered-Event,
proprietary-event               Prop-Cat-Entry-ID
}
Catalogued-Extended-Elementary-Action ::= CHOICE {
registered-extended-elementary-action Registered-Extended-Elementary-Action,
proprietary-extended-elementary-action Prop-Cat-Entry-ID
}
Cat-Ext-Attribute ::= CHOICE  {
registered-extended-attribute     Registered-Extended-Attribute,
proprietary-extended-attribute   Prop-Cat-Entry-ID
}
Prop-Cat-Entry-ID ::= CHOICE  {
tail                             SEQUENCE OF INTEGER,
octet-string                       OCTET STRING
}
OPS-Initialisation ::= SEQUENCE {
od                                 OD OPTIONAL,
os                                 Size OPTIONAL
}

```

```

GF-Param ::= CHOICE      {
gf                        GF,
gf-macro                  [0] GF-Macro
}
GF-Macro ::= SEQUENCE   {
macro-def-id              Macro-Def-ID,
gf                        GF OPTIONAL
}
GF ::= CHOICE            {
generic-ratio              Generic-Ratio,
default-gf                 INTEGER { default-gf (1)}
}
Point ::= CHOICE        {
integer                    INTEGER,
terminal                   [0] INTEGER { terminal (1)}
}
Size ::= SEQUENCE       {
x-spatial-length          Spatial-Length OPTIONAL,
y-spatial-length          [1] Spatial-Length OPTIONAL,
z-spatial-length          [2] Spatial-Length OPTIONAL
}
Spatial-Length ::= CHOICE {
length                     INTEGER,
default-spatial-length     [0] INTEGER { default-spatial-length (65536)}
}
Size-Spec-Param ::= CHOICE {
size-spec                  Size-Spec,
size-spec-macro           [0] Size-Spec-Macro
}
Size-Spec-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
size-spec                 Size-Spec OPTIONAL
}
Size-Spec ::= CHOICE    {
lengths-spec              Lengths-Spec,
get-list                  Get-List,
get-any                   [50] Get-Any
}
Lengths-Spec ::= SEQUENCE {
x-length-spec             [2]Length-Spec OPTIONAL,
y-length-spec             [0] Length-Spec OPTIONAL,
z-length-spec             [1] Length-Spec OPTIONAL
}
Length-Spec-Param ::= CHOICE {
length-spec              Length-Spec,
length-spec-macro       [0] Length-Spec-Macro
}
Length-Spec-Macro ::= SEQUENCE {
macro-def-id            Macro-Def-ID,
length-spec             Length-Spec OPTIONAL
}
Length-Spec ::= CHOICE  {
length                  INTEGER,
relative-length         Ratio,
get-integer             Get-Integer,
get-ratio               [46] Rt-Component-Target-Param,
get-any                 [45] Get-Any
}
Spatial-Position ::= SEQUENCE {
x-point                 Point OPTIONAL,
y-point                 [1] Point OPTIONAL,
z-point                 [2] Point OPTIONAL
}
Spatial-Position-Spec-Param ::= CHOICE {
spatial-position-spec   Spatial-Position-Spec,
spatial-position-spec-macro [0] Spatial-Position-Spec-Macro
}

```

```

Spatial-Position-Spec-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
spatial-position-spec  Spatial-Position-Spec OPTIONAL
}
Spatial-Position-Spec ::= CHOICE {
points-spec           Points-Spec,
get-list             Get-List,
get-any             [51] Get-Any
}
Points-Spec ::= SEQUENCE {
x-point-spec        [3] Point-Spec OPTIONAL,
y-point-spec        [1] Point-Spec OPTIONAL,
z-point-spec        [2] Point-Spec OPTIONAL
}
Point-Spec-Param ::= CHOICE {
point-spec          Point-Spec,
point-spec-macro    [1] Point-Spec-Macro
}
Point-Spec-Macro ::= SEQUENCE {
macro-def-id        Macro-Def-ID,
point-spec          Point-Spec OPTIONAL
}
Point-Spec ::= CHOICE {
point              Point,
relative-point     Ratio,
get-integer        Get-Integer,
get-ratio          [46] Rt-Component-Target-Param,
get-any           [45] Get-Any
}
Current-Point-Spec-Param ::= CHOICE {
current-point-spec  Current-Point-Spec,
current-point-spec-macro [3] Current-Point-Spec-Macro
}
Current-Point-Spec-Macro ::= SEQUENCE {
macro-def-id        Macro-Def-ID,
current-point-spec  Current-Point-Spec OPTIONAL
}
Current-Point-Spec ::= CHOICE {
point              Point,
relative-point     Ratio,
original-point-factor [1] Ratio,
current-point-factor [2] Ratio,
get-integer        [47] Get-Integer,
get-ratio          [46] Rt-Component-Target-Param,
get-any           [45] Get-Any
}
Initial-Point-Spec-Param ::= CHOICE {
initial-point-spec  Point-Spec,
initial-point-spec-macro [1] Initial-Point-Spec-Macro
}
Initial-Point-Spec-Macro ::= SEQUENCE {
macro-def-id        Macro-Def-ID,
initial-point-spec  Point-Spec OPTIONAL
}
Terminal-Point-Spec-Param ::= CHOICE {
terminal-point-spec  Point-Spec,
terminal-point-spec-macro [1] Terminal-Point-Spec-Macro
}
Terminal-Point-Spec-Macro ::= SEQUENCE {
macro-def-id        Macro-Def-ID,
terminal-point-spec  Point-Spec OPTIONAL
}
Channel-Perceptability ::= ENUMERATED {
on (1),
off (2)
}

```

```

Channel-Availability-Status ::= ENUMERATED {
available (1),
not-available (2)
}
Modifiability ::= ENUMERATED {
modifiable (1),
not-modifiable (2)
}
Selectability ::= ENUMERATED {
selectable (1),
not-selectable (2)
}
Modification-Status ::= ENUMERATED {
modified (1),
not-modified (2),
modifying (3)
}
Selection-Status ::= ENUMERATED {
selected (1),
not-selected (2)
}
Interaction-Status ::= CHOICE {
selection-status      Selection-Status,
modification-status  [0] Modification-Status
}
Interaction-Type ::= ENUMERATED {
selection (1),
modification (2)
}
Interaction-Type-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
interaction-type      Interaction-Type OPTIONAL
}
Interaction-Type-Param ::= CHOICE {
interaction-type      Interaction-Type,
interaction-type-macro Interaction-Type-Macro
}
Stream-Identification ::= CHOICE {
stream-id             SEQUENCE OF INTEGER,
all-streams          INTEGER { all-streams (1) }
}
Stream-Identification-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
stream-identification Stream-Identification OPTIONAL
}
Stream-Identification-Param ::= CHOICE {
stream-identification Stream-Identification,
stream-identification-macro [0] Stream-Identification-Macro
}
Stream-Chosen-State ::= ENUMERATED {
chosen (1),
not-chosen (2),
partially-chosen (3)
}
PCV ::= CHOICE {
integer              INTEGER,
min-avr              [0] INTEGER { min-avr (0)},
max-avr              [1] INTEGER { max-avr (255)}
}
CV ::= CHOICE {
integer              INTEGER,
min-avr              [0] INTEGER { min-avr (0)},
max-avr              [1] INTEGER { max-avr (255)}
}
OV ::= CHOICE {
integer              INTEGER,
min-avr              [0] INTEGER { min-avr (0)},
max-avr              [1] INTEGER { max-avr (255)}
}

```

```

Point-Type-Param ::= ENUMERATED {
absolute (1),
relative (2)
}
Spatial-Control-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
spatial-control       Spatial-Control OPTIONAL
}
Spatial-Control-Param ::= CHOICE {
spatial-control       Spatial-Control,
spatial-control-macro Spatial-Control-Macro
}
Expected-Axis-Result-Param ::= ENUMERATED {
x (1),
y (2),
z (3),
xyz (4)
}
User-Spatial-Control ::= ENUMERATED {
allowed (1),
not-allowed (2)
}
Spatial-Control ::= ENUMERATED {
moving (1),
resizing (2),
scaling (3),
scrolling (4)
}
GSF ::= SEQUENCE {
x-gf          GF OPTIONAL,
y-gf          [0] GF OPTIONAL,
z-gf          [1] GF OPTIONAL
}
OVS-Proj-Strategy ::= ENUMERATED {
fixed (1),
calculated (2)
}
Resizing-Strategy ::= ENUMERATED {
fixed (1),
minimum (2),
grows-only (3)
}
Aspect-Ratio ::= ENUMERATED {
preserved (1),
not-preserved (2)
}
Expected-PVD-Result ::= ENUMERATED {
initial-temporal-position (1),
terminal-temporal-position (2),
duration (3)
}
Expected-PVD-Result-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
expected-pvd-result   Expected-PVD-Result OPTIONAL
}
Expected-PVD-Result-Param ::= CHOICE {
expected-pvd-result   Expected-PVD-Result,
expected-pvd-result-macro Expected-PVD-Result-Macro
}
Expected-OVD-Result ::= ENUMERATED {
initial-temporal-position (1),
terminal-temporal-position (2),
duration (3)
}
Expected-OVD-Result-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
expected-ovd-result   Expected-OVD-Result OPTIONAL
}

```

```

Expected-OVD-Result-Param ::= CHOICE {
  expected-ovd-result      Expected-OVD-Result,
  expected-ovd-result-macro Expected-OVD-Result-Macro
}
Temporal-Termination ::= ENUMERATED {
  freeze (1),
  stop (2)
}
OD ::= CHOICE {
  integer      INTEGER,
  infinite     [0] INTEGER { infinite (1)}
}
Presentation-Priority ::= CHOICE {
  generic-integer      Generic-Integer,
  up-priority          [1] INTEGER { up-priority (1)},
  down-priority        [0] INTEGER { down-priority (2)}
}
RPS-Assignment ::= CHOICE {
  channel-reference    Channel-Reference,
  evaluated-reference  [0] Evaluated-Reference,
  prps                 INTEGER { prps (1)}
}
Ancestor ::= CHOICE {
  integer      [4] INTEGER,
  root         [5] INTEGER { root (1)}
}
EmptyChild ::= CHOICE {
  integer      [1] INTEGER,
  last         [2] INTEGER { last (1)}
}
Child ::= CHOICE {
  generic-integer [6] Generic-Integer,
  last            INTEGER { last (1)},
  random          [0] INTEGER { random (2)}
}
Navigation-Command ::= CHOICE {
  child          Child,
  emptychild     EmptyChild,
  sibling         [3] INTEGER,
  ancestor       Ancestor
}
Navigation-Command-Macro ::= SEQUENCE {
  macro-def-id      Macro-Def-ID,
  navigation-command Navigation-Command OPTIONAL
}
Navigation-Command-Param ::= CHOICE {
  navigation-command      Navigation-Command,
  navigation-command-macro Navigation-Command-Macro
}
Termination-Status ::= ENUMERATED {
  terminated (1),
  not-terminated (2)
}
Running-Status ::= ENUMERATED {
  running (1),
  not-running (2)
}
Rt-Availability-Status ::= ENUMERATED {
  available (1),
  not-available (2)
}
Activation-Status ::= ENUMERATED {
  active (1),
  inactive (2)
}

```

```

Preparation-Status ::= ENUMERATED {
ready (1),
not-ready (2)
}
Cat-Ext-Attribute-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
cat-ext-attribute     Cat-Ext-Attribute OPTIONAL
}
Cat-Ext-Attribute-Param ::= CHOICE {
cat-ext-attribute     Cat-Ext-Attribute,
cat-ext-attribute-macro [0] Cat-Ext-Attribute-Macro
}
Update-Command ::= ENUMERATED {
add (1),
remove (2),
replace (3)
}
Comparison-Value-Constant ::= CHOICE {
preparation-status    Preparation-Status,
activation-status     [0] Activation-Status,
rt-availability-status [1] Rt-Availability-Status,
running-status        [2] Running-Status,
termination-status    [3] Termination-Status,
prps                  INTEGER { prps (1)},
infinite               [4] INTEGER { infinite (2)},
temporal-termination [5] Temporal-Termination,
aspect-ratio           [6] Aspect-Ratio,
resizing-strategy      [7] Resizing-Strategy,
ovs-proj-strategy      [8] OVS-Proj-Strategy,
user-spatial-control  [9] User-Spatial-Control,
stream-chosen-state    [10] Stream-Chosen-State,
selection-status       [11] Selection-Status,
modification-status    [12] Modification-Status,
channel-availability-status [13] Channel-Availability-Status,
channel-perceptability [14] Channel-Perceptability
}
END

```

#### Action Class

-- Droits d'auteur:

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-ac {joint-iso-itu-t(2) mheg(19) version(1) action-class(1)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

#### EXPORTS

Action-Object,  
Action-Class;

#### IMPORTS

Mh-object-Class,  
Macro-Def-ID,  
Mh-Reference FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) action-class(9)}  
Elementary-Action FROM ISOMHEG-ea {joint-iso-itu-t(2) mheg(19) version(1) action-class(10)};

```

Action-Class ::= SEQUENCE {
OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) action-class(1)},
COMPONENTS OF Mh-object-Class,
synchro-indicator-param [1] Synchro-Indicator-Param,
synchronised-action-list SEQUENCE OF Synchronised-Action,
...
}

```



```

Synchro-Indicator-Param ::= CHOICE {
synchro-indicator          Synchro-Indicator,
synchro-indicator-macro    Synchro-Indicator-Macro
}
Synchro-Indicator-Macro ::= SEQUENCE {
macro-def-id               Macro-Def-ID,
synchro-indicator          Synchro-Indicator OPTIONAL
}
Synchro-Indicator ::= ENUMERATED {
serial (1),
parallel (2)
}
Synchronised-Action ::= CHOICE {
elementary-action          Elementary-Action,
action-object              [60] Action-Object
}
Action-Object ::= CHOICE {
action-object-reference    [0] Mh-Reference,
action-class                [1] Action-Class
}
END

```

#### Link Class

-- Droits d'auteur:

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-1k {joint-iso-itu-t(2) mheg(19) version(1) link-class(2)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

#### EXPORTS

Link-Class,  
Link-Effect;

#### IMPORTS

Mh-object-Class,  
Evaluated-Value,  
Generic-Value,  
Comparison-Value-Constant,  
Macro-Def-ID FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definitions(9)}  
Action-Object FROM ISOMHEG-ac {joint-iso-itu-t(2) mheg(19) version(1) action-class(1)};

```

Link-Class ::= SEQUENCE {
OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) link-class(2)},
COMPONENTS OF Mh-object-Class,
link-condition              [1] Link-Condition,
link-effect                  Link-Effect,
...
}

```

```

Link-Condition ::= CHOICE {
trigger-condition           Trigger-Condition,
logical-combination         [0] Logical-Combination
}

```

```

Trigger-Condition ::= SEQUENCE {
source-value                Evaluated-Value,
previous-condition          Comparison-Operation OPTIONAL,
current-condition           [0] Comparison-Operation
}

```

```

Constraint-Condition ::= SEQUENCE {
source-value                Evaluated-Value,
current-condition           Comparison-Operation
}

```

```

Comparison-Operation ::= SEQUENCE {
comparison-operator          Comparison-Operator,
comparison-value            Comparison-Value
}
Comparison-Value ::= CHOICE {
generic-value                Generic-Value,
comparison-value-constant    [50] Comparison-Value-Constant,
unspecified                  [51] INTEGER { unspecified (1)}
}
Comparison-Operator ::= ENUMERATED {
equal (1),
not-equal (2),
greater (3),
greater-equal (4),
less (5),
less-equal (6)
}
Logical-Combination ::= SEQUENCE {
logical-operator            Logical-Operator,
condition-list              SEQUENCE OF Condition
}
Logical-Operator ::= ENUMERATED {
and (1),
or (2),
xor (3),
not (4)
}
Condition ::= CHOICE {
trigger-condition           Trigger-Condition,
constraint-condition       [0] Constraint-Condition,
logical-combination        [1] Logical-Combination
}
Link-Effect ::= SEQUENCE {
macro-parameter-resolution-list SEQUENCE OF Macro-Parameter-Resolution OPTIONAL,
action-object                Action-Object
}
Macro-Parameter-Resolution ::= SEQUENCE {
macro-def-id                 Macro-Def-ID,
usage-value                  Generic-Value
}
END

```

**Script Class**

-- *Droits d'auteur:*

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

**ISOMHEG-sc {joint-iso-itu-t(2) mheg(19) version(1) script-class(3)}**

**DEFINITIONS**

**IMPLICIT TAGS**

**::= BEGIN**

**EXPORTS**

**Script-Class;**

**IMPORTS**

**Model-Class,**

**Catalogued-Script-Classification,**

**Script-Hook,**

**Data-Reference FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definitions(9)}**

**InterchangedScript FROM ISOMHEG-sir {joint-iso-itu-t(2) mheg(19) version(1) script-interchange-representation(11)};**

```

Script-Class ::= SEQUENCE      {
OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) script-class(3)},
COMPONENTS OF Model-Class,
script-classification          [1] Catalogued-Script-Classification OPTIONAL,
script-hook                    [2] Script-Hook,
script-data                    Script-Data,
...
}
Script-Data ::= CHOICE        {
script-inclusion                 Script-Inclusion,
data-reference                 Data-Reference
}
Script-Inclusion ::= CHOICE    {
bit-string                    BIT STRING,
octet-string                  OCTET STRING,
interchangedscript           InterchangedScript
}
END

```

#### Content Class

-- Droits d'auteur:

-----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-ct {joint-iso-itu-t(2) mheg(19) version(1) content-class(4)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

##### EXPORTS

Content,  
Content-Class;

##### IMPORTS

OV,  
Component-Class,  
Cat-Content-Classification,  
Content-Hook,  
Data-Reference,  
Generic-Value FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) content-class(9)};

```

Content-Class ::= SEQUENCE      {
OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) content-class(4)},
COMPONENTS OF Content,
...
}

```

...

```

Content ::= SEQUENCE          {
COMPONENTS OF Component-Class,
cat-content-classification    [2] Cat-Content-Classification OPTIONAL,
content-hook                  [3] Content-Hook,
ov                             [51] OV OPTIONAL,
content-data                  Content-Data
}

```

```

Content-Data ::= CHOICE      {
data-inclusion                 Data-Inclusion,
data-reference                [50] Data-Reference
}

```

```

Data-Inclusion ::= CHOICE     {
bit-string                   BIT STRING,
octet-string                 OCTET STRING,
generic-value                Generic-Value
}
END

```

## Multiplexed Content Class

-- Droits d'auteur:

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-mu {joint-iso-itu-t(2) mheg(19) version(1) multiplexed-content-class(5)}

### DEFINITIONS

#### IMPLICIT TAGS

::= BEGIN

### EXPORTS

Multiplexed-Content-Class;

### IMPORTS

Content FROM ISOMHEG-ct {joint-iso-itu-t(2) mheg(19) version(1) content-class(4)}

Cat-Content-Classification,

Content-Hook FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definitions(9)};

Multiplexed-Content-Class ::= SEQUENCE {

OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) multiplexed-content-class(5)},

COMPONENTS OF Content,

multiplexed-stream-list SEQUENCE OF Multiplexed-Stream,

...

}

Multiplexed-Stream ::= SEQUENCE {

stream-id SEQUENCE OF INTEGER,

cat-content-classification Cat-Content-Classification OPTIONAL,

content-hook [0] Content-Hook OPTIONAL

}

END

## Composite Class

-- Droits d'auteur:

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-co {joint-iso-itu-t(2) mheg(19) version(1) composite-class(6)}

### DEFINITIONS

#### IMPLICIT TAGS

::= BEGIN

### EXPORTS

Composite-Class;

### IMPORTS

Component-Class,

Mh-Reference FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definition(9)}

Link-Effect,

Link-Class FROM ISOMHEG-lk {joint-iso-itu-t(2) mheg(19) version(1) link-class(2)}

Action-Object FROM ISOMHEG-ac {joint-iso-itu-t(2) mheg(19) version(1) action-class(1)}

Content-Class FROM ISOMHEG-ct {joint-iso-itu-t(2) mheg(19) version(1) content-class(4)}

Multiplexed-Content-Class FROM ISOMHEG-mu {joint-iso-itu-t(2) mheg(19) version(1) multiplexed-content-class(5)};

Composite-Class ::= SEQUENCE {

OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) composite-class(6)},

COMPONENTS OF Component-Class,

availability-start-up [14] Availability-Start-up OPTIONAL,

availability-close-down Availability-Close-down OPTIONAL,

rt-availability-start-up Rt-Availability-Start-up OPTIONAL,

```

rt-availability-close-down      [10] Link-Effect OPTIONAL,
action-object-list              [11] SEQUENCE OF Action-Object OPTIONAL,
link-object-list                [12] SEQUENCE OF Link-Object OPTIONAL,
nb-of-elements                  [13] INTEGER,
composition-element-list       SEQUENCE OF Composition-Element OPTIONAL,
...
}
Availability-Start-up ::= CHOICE {
link-effect                     Link-Effect,
automatic-start-up-1           INTEGER { automatic-start-up-1 (1)},
automatic-start-up-2           [0] INTEGER { automatic-start-up-2 (2)},
automatic-start-up-3           [1] INTEGER { automatic-start-up-3 (3)},
automatic-start-up-4           [2] INTEGER { automatic-start-up-4 (4)},
automatic-start-up-5           [3] INTEGER { automatic-start-up-5 (5)}
}
Availability-Close-down ::= CHOICE {
link-effect                     [4] Link-Effect,
automatic-close-down-1         [5] INTEGER { automatic-close-down-1 (1)},
automatic-close-down-2         [6] INTEGER { automatic-close-down-2 (2)},
automatic-close-down-3         [7] INTEGER { automatic-close-down-3 (3)}
}
Rt-Availability-Start-up ::= CHOICE {
link-effect                     [8] Link-Effect,
automatic-rt-start-up          [9] INTEGER { automatic-rt-start-up (1)}
}
Link-Object ::= CHOICE {
link-class                     Link-Class,
link-object-reference          [0] Mh-Reference
}
Composition-Element ::= SEQUENCE {
index                          INTEGER,
associated-model               Associated-Model
}
Associated-Model ::= CHOICE {
component-object-reference     Mh-Reference,
content-class                  [3] Content-Class,
multiplexed-content-class      [4] Multiplexed-Content-Class,
composite-class                [5] Composite-Class,
label                          [6] GraphicString
}
END

```

#### Container Class

-- Droits d'auteur:

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-cr {joint-iso-itu-t(2) mheg(19) version(1) container-class(7)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

#### EXPORTS;

#### IMPORTS

Mh-object-Class,

Mh-Reference FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definition(9)}

Link-Effect,

Link-Class FROM ISOMHEG-lk {joint-iso-itu-t(2) mheg(19) version(1) link-class(2)}

Action-Class FROM ISOMHEG-ac {joint-iso-itu-t(2) mheg(19) version(1) action-class(1)}

Script-Class FROM ISOMHEG-sc {joint-iso-itu-t(2) mheg(19) version(1) script-class(3)}

Content-Class FROM ISOMHEG-ct {joint-iso-itu-t(2) mheg(19) version(1) content-class(4)}

Multiplexed-Content-Class FROM ISOMHEG-mu {joint-iso-itu-t(2) mheg(19) version(1) multiplexed-content-class(5)}

Composite-Class FROM ISOMHEG-co {joint-iso-itu-t(2) mheg(19) version(1) composite-class(6)}

Descriptor-Class FROM ISOMHEG-de {joint-iso-itu-t(2) mheg(19) version(1) descriptor-class(8)};

```

Container-Class ::= SEQUENCE {
OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) container-class(7)},
COMPONENTS OF Mh-object-Class,
container-start-up          [10] Container-Start-up OPTIONAL,
container-close-down        Container-Close-down OPTIONAL,
container-element-list      [9] SEQUENCE OF Container-Element,
...
}

```

```

Container-Start-up ::= CHOICE {
link-effect                 Link-Effect,
automatic-container-start-up-1 INTEGER { automatic-container-start-up-1 (1)},
automatic-container-start-up-2 [0] INTEGER { automatic-container-start-up-2 (2)},
automatic-container-start-up-3 [1] INTEGER { automatic-container-start-up-3 (3)},
automatic-container-start-up-4 [2] INTEGER { automatic-container-start-up-4 (4)},
automatic-container-start-up-5 [3] INTEGER { automatic-container-start-up-5 (5)},
automatic-container-start-up-6 [4] INTEGER { automatic-container-start-up-6 (6)},
automatic-container-start-up-7 [5] INTEGER { automatic-container-start-up-7 (7)},
automatic-container-start-up-8 [6] INTEGER { automatic-container-start-up-8 (8)}
}

```

```

Container-Close-down ::= CHOICE{
link-effect                 [7] Link-Effect,
automatic-container-close-down [8] INTEGER { automatic-container-close-down (1)}
}

```

```

Container-Element ::= CHOICE {
mh-reference                Mh-Reference,
action-class                [3] Action-Class,
link-class                  [4] Link-Class,
script-class                [5] Script-Class,
content-class               [6] Content-Class,
multiplexed-content-class   [7] Multiplexed-Content-Class,
composite-class             [8] Composite-Class,
container-class             [9] Container-Class,
descriptor-class           [10] Descriptor-Class,
...
}
END

```

#### Descriptor Class

-- Droits d'auteur:

-----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-de {joint-iso-itu-t(2) mheg(19) version(1) descriptor-class(8)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

##### EXPORTS

Descriptor-Class;

##### IMPORTS

Mh-object-Class,

Mh-Reference,

Catalogued-Style,

Catalogued-Extended-Elementary-Action,

Cat-Ext-Attribute,

Class-ID,

Catalogued-Script-Classification,

Script-Hook,

Cat-Content-Classification,

Content-Hook,

Catalogued-Media-Type,

Value,

Catalogued-Event FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definitions(9)};

```

Descriptor-Class ::= SEQUENCE {
OBJECT IDENTIFIER DEFAULT {joint-iso-itu-t(2) mheg(19) version(1) descriptor-class(8)},
COMPONENTS OF Mh-object-Class,
related-object-list          [1] SEQUENCE OF Related-Object OPTIONAL,
other-descriptor-list        [2] SEQUENCE OF Mh-Reference OPTIONAL,
readme                       GraphicString OPTIONAL,
system-readable-material     System-Readable-Material OPTIONAL,
channel-information-list     [3] SEQUENCE OF Channel-Information OPTIONAL,
catalogued-style-information [4] SEQUENCE OF Catalogued-Style OPTIONAL,
cat-ext-elementary-action-info [5] SEQUENCE OF Catalogued-Extended-Elementary-Action OPTIONAL,
cat-ext-attribute-info-list  [6] SEQUENCE OF Cat-Ext-Attribute OPTIONAL,
...
}
Related-Object ::= SEQUENCE {
mh-reference                 Mh-Reference,
object-information           Object-Information OPTIONAL
}
Object-Information ::= SEQUENCE {
object-size                  INTEGER OPTIONAL,
class-id                     Class-ID,
class-specific-information   Class-Specific-Information OPTIONAL,
offset                       INTEGER OPTIONAL
}
Class-Specific-Information ::= CHOICE {
script-class-information     Script-Class-Information,
content-class-information    [0] Content-Class-Information,
mux-content-class-info      [1] Mux-Content-Class-Info
}
Script-Class-Information ::= SEQUENCE {
script-classification        Catalogued-Script-Classification OPTIONAL,
script-hook                  [0] Script-Hook OPTIONAL
}
Content-Class-Information ::= SEQUENCE {
cat-content-classification   Cat-Content-Classification OPTIONAL,
content-hook                 [0] Content-Hook OPTIONAL,
alternative-object-list      [1] SEQUENCE OF Alternative-Object OPTIONAL
}
Mux-Content-Class-Info ::= SEQUENCE {
content-class-information    Content-Class-Information,
number-of-streams           INTEGER OPTIONAL,
stream-information-list      SEQUENCE OF Stream-Information OPTIONAL
}
Stream-Information ::= SEQUENCE {
stream-id                    SEQUENCE OF INTEGER,
content-class-information    Content-Class-Information
}
Alternative-Object ::= SEQUENCE {
content-object-ref           Mh-Reference,
content-hook                 Content-Hook OPTIONAL,
alternative-descriptor-object [0] Mh-Reference OPTIONAL,
alternative-readme           GraphicString OPTIONAL
}
System-Readable-Material ::= CHOICE {
bit-string                   BIT STRING,
octet-string                 OCTET STRING
}
Channel-Information ::= SEQUENCE {
channel-id                   INTEGER,
x-min                        INTEGER OPTIONAL,
x-max                        [0] INTEGER OPTIONAL,
y-min                        [1] INTEGER OPTIONAL,
y-max                        [2] INTEGER OPTIONAL,
z-min                        [3] INTEGER OPTIONAL,
z-max                        [4] INTEGER OPTIONAL,
x-resolution                 [5] INTEGER OPTIONAL,
y-resolution                 [6] INTEGER OPTIONAL,

```

```

z-resolution          [7] INTEGER OPTIONAL,
t-resolution          [8] INTEGER OPTIONAL,
f-min                 [9] INTEGER OPTIONAL,
f-max                 [10] INTEGER OPTIONAL,
audio-dynamic         [11] INTEGER OPTIONAL,
channel-media-type-list SEQUENCE OF Catalogued-Media-Type OPTIONAL,
event-mapping-list   [12] SEQUENCE OF Event-Mapping OPTIONAL
}
Event-Mapping ::= SEQUENCE {
event                SEQUENCE OF Value,
catalogued-event     Catalogued-Event OPTIONAL
}
END

```

#### Elementary Action Class

-- *Droits d'auteur:*

-- -----

-- (c) UIT-T/Organisation internationale pour la normalisation 1996.

-- L'autorisation de reproduction est donnée pour utilisation avec les applications et machines MHEG

-- conformes selon ce qui a été défini dans la Recommandation UIT-T T.171, sous réserve de reproduire

-- la présente note dans toutes les copies.

--

ISOMHEG-ea {joint-iso-itu-t(2) mheg(19) version(1) elementary-actions(10)}

#### DEFINITIONS

##### IMPLICIT TAGS

::= BEGIN

#### EXPORTS

Elementary-Action;

#### IMPORTS

Generic-Integer-Param,  
Macro-Def-ID,  
Rt-Component-Reference,  
Return-Target-Param,  
Generic-Numeric-Param,  
Generic-Value-Param,  
Mh-Reference,  
Target-Param,  
Update-Command,  
Catalogued-Extended-Elementary-Action,  
Cat-Ext-Attribute-Param,  
Mh-Target-Param,  
Generic-Value,  
Generic-List-Elt-ID-Param,  
Rt-Target-Param,  
Generic-Integer,  
Rt-Script-Target-Param,  
Socket-Target-Param,  
Evaluated-Reference,  
Rt-Component-Target-Param,  
RPS-Assignment,  
Generic-Ratio-Param,  
Presentation-Priority,  
Initial-Point-Spec-Param,  
Terminal-Point-Spec-Param,  
Current-Point-Spec-Param,  
Temporal-Termination,  
Point-Spec-Param,  
GF-Param,  
Point-Spec,  
Aspect-Ratio,  
Rt-Composite-Target-Param,  
Resizing-Strategy,  
OVS-Proj-Strategy,  
Size-Spec-Param,  
Spatial-Position-Spec-Param,  
Spatial-Control-Param,  
User-Spatial-Control,



**Rt-Content-Target-Param,**  
**Channel-Target-Param,**  
**Rt-Mux-Target-Param,**  
**Interaction-Status,**  
**Interaction-Type-Param,**  
**Catalogued-Style,**  
**Rt-Content-Reference,**  
**Channel-Perceptability,**  
**Size,**  
**Evaluated-Integer,**  
**Evaluated-List,**  
**Rt-Component-Channel-Tg-Param,**  
**Generic-List-Param FROM ISOMHEG-ud {joint-iso-itu-t(2) mheg(19) version(1) useful-definitions(9)};**

```

Delay ::= SEQUENCE {
temporal-unit-ref-param      Temporal-Unit-Ref-Param,
duration-param              Generic-Integer-Param
}
Temporal-Unit-Ref-Param ::= CHOICE {
temporal-unit-ref          Temporal-Unit-Ref,
temporal-unit-ref-macro   Temporal-Unit-Ref-Macro
}
Temporal-Unit-Ref-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
temporal-unit-ref        Temporal-Unit-Ref OPTIONAL
}
Temporal-Unit-Ref ::= CHOICE {
rt-component-reference    Rt-Component-Reference,
default-gf                INTEGER { default-gf (1) }
}
Return ::= SEQUENCE {
return-target-param-list  SEQUENCE OF Return-Target-Param,
return-indicator-param    Generic-Numeric-Param,
returned-generic-value-param-list SEQUENCE OF Generic-Value-Param OPTIONAL,
content-object-ref-param-list [0] SEQUENCE OF Content-Object-Ref-Param OPTIONAL
}
Content-Object-Ref-Param ::= CHOICE {
content-object-ref        Mh-Reference,
content-object-ref-macro [3] Content-Object-Ref-Macro
}
Content-Object-Ref-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
content-object-ref        Mh-Reference OPTIONAL
}
Set-Alias ::= SEQUENCE {
target-param-list         SEQUENCE OF Target-Param,
alias-spec-param-list     SEQUENCE OF Alias-Spec-Param
}
Alias-Spec-Param ::= CHOICE {
alias-spec                Alias-Spec,
alias-spec-macro          [0] Alias-Spec-Macro
}
Alias-Spec-Macro ::= SEQUENCE {
macro-def-id              Macro-Def-ID,
alias-spec                Alias-Spec OPTIONAL
}
Alias-Spec ::= SEQUENCE {
alias-list                SEQUENCE OF GraphicString,
update-command            Update-Command
}
Catalogued-Elementary-Action ::= SEQUENCE {
target-param-list         SEQUENCE OF Target-Param,
catalogued-extended-elementary-action-param Cat-Ext-Elementary-Action-Param,
elementary-action-param-list SEQUENCE OF Generic-Value-Param OPTIONAL
}
Cat-Ext-Elementary-Action-Param ::= CHOICE {
cat-ext-elementary-action Catalogued-Extended-Elementary-Action,
cat-ext-elementary-action-macro [0] Cat-Ext-Elementary-Action-Macro
}

```

```

Cat-Ext-Elementary-Action-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,

catalogued-extended-elementary-action  Catalogued-Extended-Elementary-Action OPTIONAL
}
Set-Catalogued-Attribute ::= SEQUENCE {
target-param-list     SEQUENCE OF Target-Param,
cat-ext-attribute-param  Cat-Ext-Attribute-Param,
ext-attribute-value-param  Generic-Value-Param,
transition-duration-param  Generic-Integer-Param OPTIONAL
}
Set-Data ::= SEQUENCE {
content-target-param-list  SEQUENCE OF Mh-Target-Param,
substitution-indicator-param  Substitution-Indicator-Param,
data-element-param-list    SEQUENCE OF Data-Element-Param OPTIONAL
}
Substitution-Indicator-Param ::= CHOICE {
substitution-indicator      Substitution-Indicator,
substitution-indicator-macro  Substitution-Indicator-Macro
}
Substitution-Indicator-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
substitution-indicator  Substitution-Indicator OPTIONAL
}
Substitution-Indicator ::= ENUMERATED {
substitution (1),
no-substitution (2)
}
Data-Element-Param ::= CHOICE {
data-element          Data-Element,
data-element-macro    [0] Data-Element-Macro
}
Data-Element-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
data-element          Data-Element OPTIONAL
}
Data-Element ::= SEQUENCE {
process-indicator      Process-Indicator,
generic-list-elt-id    SEQUENCE OF INTEGER OPTIONAL,
generic-value          [0] Generic-Value
}
Process-Indicator ::= ENUMERATED {
process (1),
no-process (2)
}
Add ::= SEQUENCE {
content-target-param-list  SEQUENCE OF Mh-Target-Param,
generic-list-elt-id-param  Generic-List-Elt-ID-Param OPTIONAL,
generic-value-param        [1] Generic-Value-Param OPTIONAL
}
Substract ::= SEQUENCE {
content-target-param-list  SEQUENCE OF Mh-Target-Param,
generic-list-elt-id-param  Generic-List-Elt-ID-Param OPTIONAL,
generic-value-param        [1] Generic-Value-Param OPTIONAL
}
Copy ::= SEQUENCE {
content-target-param      Mh-Target-Param,
destination-param-list    SEQUENCE OF Mh-Target-Param
}
Run ::= SEQUENCE {
rt-target-param-list      SEQUENCE OF Rt-Target-Param,
number-of-performances-param  [0] Number-Of-Performances-Param OPTIONAL
}
Number-Of-Performances-Param ::= CHOICE {
number-of-performances    Number-Of-Performances,
number-of-performances-macro  [1] Number-Of-Performances-Macro
}

```

```

Number-Of-Performances-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
number-of-performances [0] Number-Of-Performances OPTIONAL
}
Number-Of-Performances ::= CHOICE {
generic-integer      Generic-Integer,
infinite             [0] INTEGER { infinite (1)}
}
Set-Parameters ::= SEQUENCE {
rt-script-target-param-list SEQUENCE OF Rt-Script-Target-Param,
passing-param-list        SEQUENCE OF Passing-Param OPTIONAL
}
Passing-Param ::= CHOICE {
passing              Passing,
passing-macro        Passing-Macro
}
Passing-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
passing               Passing OPTIONAL
}
Passing ::= CHOICE {
generic-value        [1] Generic-Value,
content-object-ref   [0] Mh-Reference
}
Plug ::= SEQUENCE {
socket-target-param-list SEQUENCE OF Socket-Target-Param,
plug-in-param          Plug-In-Param
}
Plug-In-Param ::= CHOICE {
plug-in              Plug-In,
plug-in-macro        [45] Plug-In-Macro
}
Plug-In-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
plug-in               Plug-In OPTIONAL
}
Plug-In ::= CHOICE {
rt-component-reference Rt-Component-Reference,
component-object-reference Mh-Reference,
label                 [3] GraphicString,
evaluated-reference   [4] Evaluated-Reference
}
Set-RPS-Assignment ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
rps-assignment-param          RPS-Assignment-Param
}
RPS-Assignment-Param ::= CHOICE {
rps-assignment          RPS-Assignment,
rps-assignment-macro    RPS-Assignment-Macro
}
RPS-Assignment-Macro ::= SEQUENCE {
macro-def-id          Macro-Def-ID,
rps-assignment        RPS-Assignment OPTIONAL
}
Set-Perceptability ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
perceptability-param          Generic-Ratio-Param,
transition-duration-param      Generic-Integer-Param OPTIONAL
}
Set-Presentation-Priority ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
presentation-priority-param      Presentation-Priority-Param,
transition-duration-param        Generic-Integer-Param OPTIONAL
}
Presentation-Priority-Param ::= CHOICE {
presentation-priority          Presentation-Priority,
presentation-priority-macro    Presentation-Priority-Macro
}

```

```

Presentation-Priority-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
presentation-priority  Presentation-Priority OPTIONAL
}
Set-OVD ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
initial-point-spec-param      Initial-Point-Spec-Param,
terminal-point-spec-param     Terminal-Point-Spec-Param
}
Set-CTP ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
current-point-spec-param      Current-Point-Spec-Param
}
Set-Temporal-Termination ::= SEQUENCE {
rt-target-param-list         SEQUENCE OF Rt-Target-Param,
temporal-termination-param   Temporal-Termination-Param
}
Temporal-Termination-Param ::= CHOICE {
temporal-termination        Temporal-Termination,
temporal-termination-macro  Temporal-Termination-Macro
}
Temporal-Termination-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
temporal-termination  Temporal-Termination OPTIONAL
}
Set-PVD-Position ::= SEQUENCE {
socket-target-param-list SEQUENCE OF Socket-Target-Param,
temporal-position-param  Point-Spec-Param
}
Set-GTF ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
gtf-param                      GF-Param
}
Set-Timestones ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
timestone-spec-param-list      SEQUENCE OF Timestone-Spec-Param
}
Timestone-Spec-Param ::= CHOICE {
timestone-spec           Timestone-Spec,
timestone-spec-macro    [0] Timestone-Spec-Macro
}
Timestone-Spec-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
timestone-spec         Timestone-Spec OPTIONAL
}
Timestone-Spec ::= SEQUENCE {
timestone-list         SEQUENCE OF Timestone,
update-command        Update-Command
}
Timestone ::= SEQUENCE {
timestone-id          Generic-Integer,
timestone-position    Point-Spec,
number-of-repetitions Number-Of-Repetitions
}
Number-Of-Repetitions ::= CHOICE {
generic-integer        Generic-Integer,
infinite              [0] INTEGER { infinite (1) }
}
Set-Aspect-Ratio ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
aspect-ratio-param           Aspect-Ratio-Param
}
Aspect-Ratio-Param ::= CHOICE {
aspect-ratio           Aspect-Ratio,
aspect-ratio-macro    Aspect-Ratio-Macro
}

```

```

Aspect-Ratio-Macro ::= SEQUENCE{
macro-def-id           Macro-Def-ID,
aspect-ratio          Aspect-Ratio OPTIONAL
}
Set-Resizing-Strategy ::= SEQUENCE {
rt-composite-target-param-list SEQUENCE OF Rt-Composite-Target-Param,
resizing-strategy-param       Resizing-Strategy-Param
}
Resizing-Strategy-Param ::= CHOICE {
resizing-strategy          Resizing-Strategy,
resizing-strategy-macro    Resizing-Strategy-Macro
}
Resizing-Strategy-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
resizing-strategy       Resizing-Strategy OPTIONAL
}
Set-OVS-Proj-Strategy ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
ovs-proj-strategy-param       OVS-Proj-Strategy-Param
}
OVS-Proj-Strategy-Param ::= CHOICE {
ovs-proj-strategy          OVS-Proj-Strategy,
ovs-proj-strategy-macro    OVS-Proj-Strategy-Macro
}
OVS-Proj-Strategy-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
ovs-proj-strategy       OVS-Proj-Strategy OPTIONAL
}
Set-OVS ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
size-spec-param              Size-Spec-Param,
transition-duration-param     Generic-Integer-Param OPTIONAL
}
Set-OAP ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
oap-param                    Spatial-Position-Spec-Param
}
Set-OVS-Position ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
ovs-position-param            Spatial-Position-Spec-Param,
transition-duration-param     Generic-Integer-Param OPTIONAL
}
Set-PAP ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
pap-param                    Spatial-Position-Spec-Param
}
Set-PVS-Position ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
pvs-position-param            Spatial-Position-Spec-Param,
transition-duration-param     Generic-Integer-Param OPTIONAL
}
Set-GSF ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
gsf-param                    GF-Param,
transition-duration-param     Generic-Integer-Param OPTIONAL
}
Set-User-Spatial-Control ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
spatial-control-param-list     SEQUENCE OF Spatial-Control-Param,
user-spatial-control-param     User-Spatial-Control-Param
}
User-Spatial-Control-Param ::= CHOICE {
user-spatial-control          User-Spatial-Control,
user-spatial-control-macro    User-Spatial-Control-Macro
}
User-Spatial-Control-Macro ::= SEQUENCE {
macro-def-id           Macro-Def-ID,
user-spatial-control   User-Spatial-Control OPTIONAL
}

```

```

Set-CV ::= SEQUENCE {
rt-content-target-param-list SEQUENCE OF Rt-Content-Target-Param,
cv-param Current-Point-Spec-Param,
transition-duration-param Generic-Integer-Param OPTIONAL
}
Set-GVF ::= SEQUENCE {
gvf-target GVF-Target,
gvf-param GF-Param,
transition-duration-param Generic-Integer-Param OPTIONAL
}
GVF-Target ::= CHOICE {
rt-composite-targets-param SEQUENCE OF Rt-Composite-Target-Param,
channel-targets-param [0] SEQUENCE OF Channel-Target-Param
}
Set-Stream-Choice ::= SEQUENCE {
rt-mux-target-param-list SEQUENCE OF Rt-Mux-Target-Param,
stream-spec-param-list SEQUENCE OF Stream-Spec-Param OPTIONAL
}
Stream-Spec-Param ::= CHOICE {
stream-spec Stream-Spec,
stream-spec-macro [0] Stream-Spec-Macro
}
Stream-Spec-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
stream-spec Stream-Spec OPTIONAL
}
Stream-Spec ::= SEQUENCE {
stream-id-reference-list SEQUENCE OF SEQUENCE OF INTEGER OPTIONAL,
update-command Update-Command
}
Interaction-Status-Param ::= CHOICE {
interaction-status Interaction-Status,
interaction-status-macro Interaction-Status-Macro
}
Interaction-Status-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
interaction-status Interaction-Status OPTIONAL
}
Min-Interact-Required-Param ::= CHOICE {
min-interact-required INTEGER,
min-interact-required-macro Min-Interact-Required-Macro
}
Min-Interact-Required-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
min-interact-required INTEGER OPTIONAL
}
Max-Interact-Required-Param ::= CHOICE {
max-interact-required INTEGER,
max-interact-required-macro Max-Interact-Required-Macro
}
Max-Interact-Required-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
max-interact-required INTEGER OPTIONAL
}
Set-Interaction-Ability ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
interaction-type-param Interaction-Type-Param,
min-interact-required-param Min-Interact-Required-Param,
max-interact-required-param Max-Interact-Required-Param
}
Set-Interaction-Status ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
interaction-type-param Interaction-Type-Param,
interaction-status-param Interaction-Status-Param
}

```

```

Set-Style ::= SEQUENCE {
rt-component-target-param-list SEQUENCE OF Rt-Component-Target-Param,
catalogued-style-param Catalogued-Style-Param,
additional-information-param Generic-Value-Param OPTIONAL
}
Catalogued-Style-Param ::= CHOICE {
catalogued-style Catalogued-Style,
catalogued-style-macro [0] Catalogued-Style-Macro
}
Catalogued-Style-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
catalogued-style Catalogued-Style OPTIONAL
}
Attach-Anchor ::= SEQUENCE {
rt-content-target-param-list SEQUENCE OF Rt-Content-Target-Param,
anchor-spec-param-list SEQUENCE OF Anchor-Spec-Param
}
Anchor-Spec-Param ::= CHOICE {
anchor-spec Anchor-Spec,
anchor-spec-macro [0] Anchor-Spec-Macro
}
Anchor-Spec-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
anchor-spec Anchor-Spec OPTIONAL
}
Anchor-Spec ::= SEQUENCE {
anchor-list SEQUENCE OF Anchor,
update-command Update-Command
}
Anchor ::= CHOICE {
rt-content-reference Rt-Content-Reference,
evaluated-reference Evaluated-Reference
}
Set-Channel-Perceptability ::= SEQUENCE {
channel-target-param-list SEQUENCE OF Channel-Target-Param,
channel-perceptability-param Channel-Perceptability-Param
}
Channel-Perceptability-Param ::= CHOICE {
channel-perceptability Channel-Perceptability,
channel-perceptability-macro Channel-Perceptability-Macro
}
Channel-Perceptability-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
channel-perceptability Channel-Perceptability OPTIONAL
}
Set-CPS ::= SEQUENCE {
channel-target-param-list SEQUENCE OF Channel-Target-Param,
cps-initialisation-param CPS-Initialisation-Param OPTIONAL
}
CPS-Initialisation-Param ::= CHOICE {
cps-initialisation CPS-Initialisation,
cps-initialisation-macro [0] CPS-Initialisation-Macro
}
CPS-Initialisation-Macro ::= SEQUENCE {
macro-def-id Macro-Def-ID,
cps-initialisation CPS-Initialisation OPTIONAL
}
CPS-Initialisation ::= SEQUENCE {
cps-duration CPS-Duration OPTIONAL,
cps-size [2] CPS-Size OPTIONAL
}
CPS-Duration ::= CHOICE {
integer INTEGER,
infinite [0] INTEGER { infinite (1)},
evaluated-integer [1] Evaluated-Integer
}

```

```

CPS-Size ::= CHOICE      {
size                    Size,
evaluated-list         Evaluated-List
}
Set-Event ::= SEQUENCE  {
rt-component-channel-tg-param-list SEQUENCE OF Rt-Component-Channel-Tg-Param,
event-param             Generic-List-Param,
event-data-param       Generic-Value-Param OPTIONAL
}
Elementary-Action ::= CHOICE {
set-event              Set-Event,
set-cps                [0] Set-CPS,
set-channel-perceptability [1] Set-Channel-Perceptability,
new-channel            [59] SEQUENCE OF Channel-Target-Param,
delete-channel        [2] SEQUENCE OF Channel-Target-Param,
attach-anchor         [3] Attach-Anchor,
get-style              Rt-Component-Target-Param,
set-style              [4] Set-Style,
set-interaction-ability [5] Set-Interaction-Ability,
set-interaction-status [6] Set-Interaction-Status,
set-stream-choice      [7] Set-Stream-Choice,
set-cv                 [8] Set-CV,
set-gvf                [9] Set-GVF,
set-aspect-ratio       [16] Set-Aspect-Ratio,
set-resizing-strategy [17] Set-Resizing-Strategy,
set-ovs                [18] Set-OVS,
set-oap                [19] Set-OAP,
set-ovs-position       [20] Set-OVS-Position,
set-pap                [21] Set-PAP,
set-pvs-position       [22] Set-PVS-Position,
set-gsf                [23] Set-GSF,
set-user-spatial-control [24] Set-User-Spatial-Control,
set-ovd                [25] Set-OVD,
set-ctp                [26] Set-CTP,
set-temporal-termination [27] Set-Temporal-Termination,
set-pvd-position       [28] Set-PVD-Position,
set-gtf                [29] Set-GTF,
set-timestones         [30] Set-Timestones,
set-perceptability     [31] Set-Perceptability,
set-presentation-priority [32] Set-Presentation-Priority,
set-rps-assignment     [33] Set-RPS-Assignment,
plug                   [39] Plug,
set-parameters         [40] Set-Parameters,
run                    [41] Run,
stop                   [42] SEQUENCE OF Rt-Target-Param,
new                    [43] SEQUENCE OF Rt-Target-Param,
delete                 [44] SEQUENCE OF Rt-Target-Param,
copy                   [45] Copy,
set-data               [46] Set-Data,
add                    [47] Add,
substract              [48] Substract,
link-abort             [49] SEQUENCE OF Mh-Target-Param,
activate               [50] SEQUENCE OF Mh-Target-Param,
deactivate             [51] SEQUENCE OF Mh-Target-Param,
prepare                [52] SEQUENCE OF Mh-Target-Param,
destroy                [53] SEQUENCE OF Mh-Target-Param,
catalogued-elementary-action [54] Catalogued-Elementary-Action,
set-catalogued-attribute [55] Set-Catalogued-Attribute,
set-alias              [56] Set-Alias,
return                 [57] Return,
delay                  [58] Delay,
...
}
END

```



## Annexe B

### Exemples de systèmes MHEG

La présente Recommandation ne définit pas la structure du moteur MHEG, pas plus qu'elle ne décrit comment celui-ci traite les objets transférés. On trouvera dans la présente annexe un exemple de moteur MHEG.

#### B.1 Exemple de moteur MHEG

La Figure B.1 illustre un exemple de réalisation de moteur MHEG et montre la relation qui existe entre le moteur, l'application, les services de présentation et les services d'accès. Le moteur MHEG est responsable de tous ses processus internes et de l'interfonctionnement avec les autres modules. Il peut utiliser les moyens offerts par un système d'exploitation ou par des services de gestion-systèmes, qui ne sont pas présentés dans la figure ci-après.

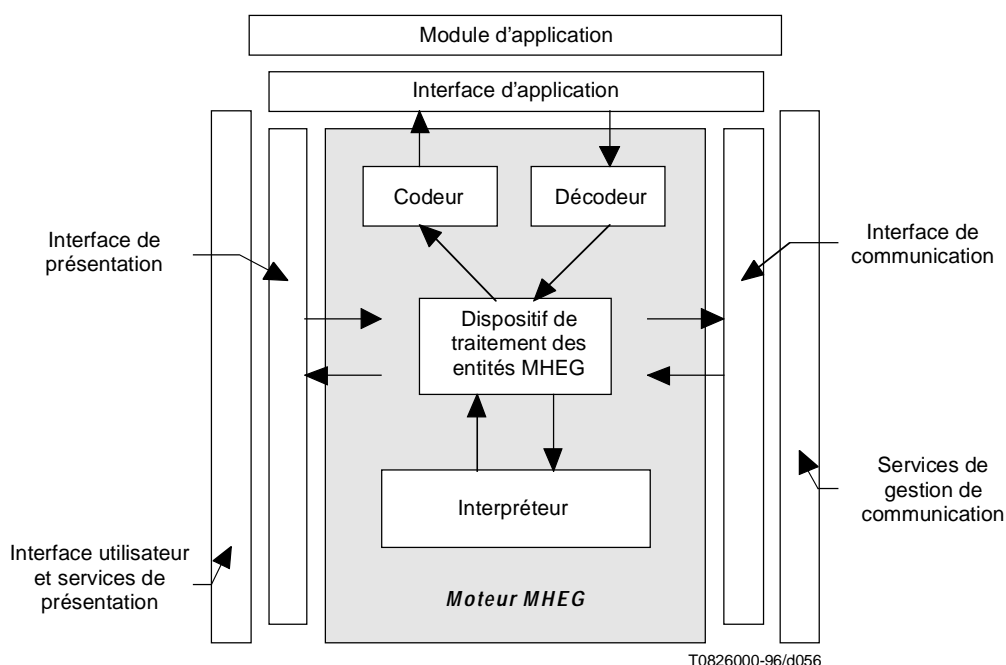


Figure B.1/T.171 – Exemple de moteur MHEG

Le moteur MHEG peut être composé de plusieurs modules comme on peut le voir dans la Figure B.1.

##### B.1.1 Décodeur MHEG

Ce module convertit les objets MHEG codés en langage ASN.1 en un format interne propre au moteur MHEG en vue de leur traitement.

##### B.1.2 Codeur MHEG

Ce module met les objets internes MHEG sous forme de données ASN.1 pour qu'ils puissent être transférés vers un autre système. Il est utilisé lorsque le moteur MHEG envoie un objet MHEG à un autre système. Ce module peut être requis, par exemple, dans un système d'assistance aux auteurs.

### B.1.3 Dispositif de traitement des entités MHEG

Ce module traite les objets MHEG, les rt-objets et les canaux dans leurs formats internes, affecte les entités et commande la gestion de la mémoire.

#### B.1.3.1 Résolution des références

Le dispositif de traitement des entités MHEG peut également résoudre les références de ces entités. Une référence peut être résolue de la manière indiquée dans la Figure B.2. Celle-ci illustre un exemple de processus de décision permettant de résoudre des références d'entités MHEG.

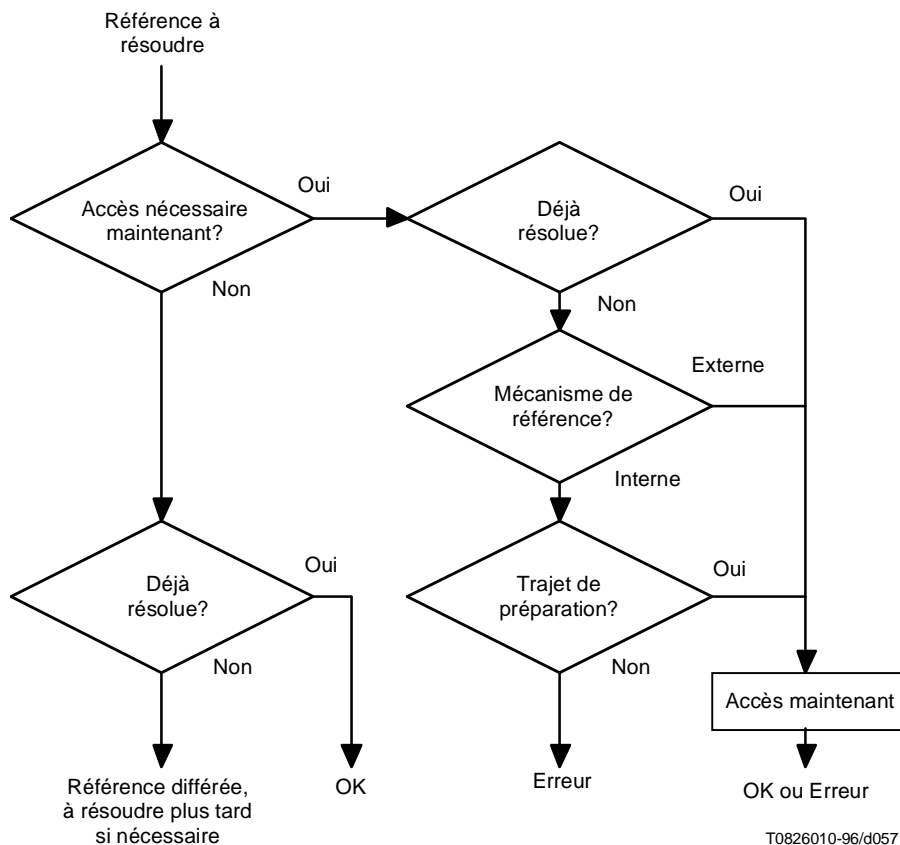


Figure B.2/T.171 – Résolution d'une référence

Lorsque le moteur MHEG doit résoudre une référence, deux cas peuvent se présenter, comme on peut le voir dans la Figure B.2:

- 1) il n'est pas nécessaire d'accéder immédiatement à l'entité. Par exemple, si une entité MHEG est la cible d'une action spécifiée dans l'effet de lien d'un objet *lien* et qu'une action *Prepare* (préparer) est appliquée à cet objet, il n'est pas nécessaire de préparer l'objet MHEG à ce moment. Si l'entité n'est pas disponible pour la machine MHEG, la résolution de la référence à cette entité peut être complétée plus tard. Une telle référence est dite «reportée». Si l'entité est accessible pour le moteur MHEG au moment de la demande, aucun traitement supplémentaire n'est nécessaire;
- 2) il est nécessaire d'accéder immédiatement à l'entité. Par exemple, si un objet *action* est pointé dans un objet *lien* et qu'une action *Prepare* (préparer) est appliquée à cet objet *lien*, il est alors nécessaire de

préparer l'objet *action* pointé. Dans ce cas, il est nécessaire d'accéder, de décoder et de préparer l'entité. Deux cas peuvent alors se présenter;

- 3) si l'entité référencée n'est pas encore résolue, le moteur MHEG essaie de résoudre la référence. Deux cas peuvent alors se présenter;
- 4) si l'entité est pointée au moyen d'un identificateur externe, on suppose que le moteur MHEG sait le reconnaître et le résoudre, et on considère que la référence est résolue;
- 5) si l'entité est pointée au moyen d'un identificateur interne, le moteur MHEG essaie de résoudre la référence. Deux cas peuvent alors se présenter;
- 6) si le moteur MHEG dispose d'un trajet de préparation, la référence est considérée comme résolue. Le trajet de préparation, qui n'est pas défini dans la présente Recommandation, dépend de l'application utilisatrice, du moteur MHEG et de la pratique locale. Par exemple, un trajet de préparation peut indiquer l'emplacement du conteneur dans lequel se trouvent les objets requis, ou être représenté sous la forme d'une demande SQL, d'une adresse de serveur, d'une spécification de fichier, etc.;
- 7) si le moteur MHEG ne peut pas résoudre la référence, il s'agit d'une erreur;
- 8) si l'entité référencée est considérée comme accessible pour le moteur MHEG, celui-ci essaie d'accéder à l'objet en utilisant les services de communication.

Si aucune entité n'est adressée pendant la résolution de la référence, le moteur MHEG ne tente alors d'accéder à aucun objet.

Selon le contexte, cela peut créer une condition d'erreur. Par exemple, si la cible de l'action `Prepare` (préparer) n'est pas accessible, elle reste non disponible pour le moteur MHEG. C'est ce dernier qui doit signaler cette condition à l'application utilisatrice. Par exemple, si une condition de lien est décrite sur une entité MHEG inexistante, cette condition ne peut être satisfaite sur le champ, sans pour autant qu'il s'agisse d'une erreur. Ce lien n'est pas déclenché sur le moment, mais il peut l'être par la suite.

#### **B.1.4 Interpréteur MHEG**

Ce module traite les entités MHEG conformément à la présente Recommandation. L'interpréteur MHEG peut toutefois contenir divers processus; on peut ainsi distinguer les processus suivants.

##### **B.1.4.1 Processus de préparation**

Ce processus prépare les entités MHEG avant leur traitement par le moteur MHEG. Par exemple, la récupération des données de contenu d'une séquence audiovisuelle figurant sur un disque peut nécessiter tellement de temps qu'il peut être utile de commencer à charger la séquence avant d'en avoir besoin.

Le processus de préparation peut interpréter les actions élémentaires suivantes: `Prepare` (préparer), `Destroy` (détruire) et `Get Preparation Status` (extraire état de préparation).

##### **B.1.4.2 Processus de création de rt-objets et de canaux**

Ce processus crée les rt-objets et les canaux.

Le processus de création peut interpréter les actions élémentaires suivantes: `New` (nouveau), `Delete` (supprimer), `Get Rt-Availability Status` (extraire état de disponibilité d'exécution), `New Channel` (nouveau canal), `Delete Channel` (supprimer canal) et `Get Channel-Availability Status` (extraire état de disponibilité de canal).

##### **B.1.4.3 Processus d'activation**

Ce processus active les rt-composants qui doivent être transmis à un processus de présentation ainsi que les rt-scripts qui doivent être transférés au moteur de script.

Ce processus peut interpréter les actions élémentaires suivantes: `Run` (exécuter), `Stop` (arrêter) et `Get Running Status` (extraire état d'exécution).

##### **B.1.4.4 Processus de script**

Ce processus interprète une donnée de script telle qu'elle est décrite dans un objet *script*. Des paramètres peuvent être échangés dans les deux sens entre le moteur MHEG et le processus de script.

Ce processus peut interpréter l'action élémentaire `Set parameters` (fixer paramètres).

#### **B.1.4.5 Processus de présentation**

Ce processus prend en charge la présentation des rt-composants. Il peut fonctionner avec l'interface utilisateur et les services de présentation.

Il peut interpréter toutes les actions élémentaires relatives à la présentation. Ces actions peuvent modifier dynamiquement les rendus des rt-composants et l'affectation des canaux: comportements temporels, comportements spatiaux et comportements sonores. Ce processus peut également interpréter les styles spécifiés par des attributs.

Une application utilisatrice peut étendre les fonctionnalités de présentation de manière que celles-ci incorporent des caractéristiques de rendu qui lui sont propres (couleur, polices de caractères). Ces caractéristiques de rendu particulières peuvent être transférées entre objets MHEG à l'aide d'extensions et de descriptions contenues dans des objets *descripteur*. C'est à l'application utilisatrice de s'assurer que le processus de présentation dont dispose le moteur MHEG est capable de présenter les objets MHEG avec ces caractéristiques.

#### **B.1.4.6 Processus d'interaction**

Ce processus interagit avec l'utilisateur et interfonctionne avec les outils de l'interface GUI qui existent dans le système. Il peut traiter le comportement de l'interaction, par exemple la sélection et la modification.

#### **B.1.4.7 Processus de traitement de lien**

Ce processus évalue en parallèle toutes les conditions relatives aux entités MHEG décrites dans les conditions de lien des objets *lien* activés, et traite les actions décrites dans un effet de lien si la condition de lien correspondante est satisfaite.

#### **B.1.5 Interface utilisateur et services de présentation**

Ce module prend en charge la présentation des données multimédias à l'utilisateur et l'acquisition des données d'utilisateur. Le module de présentation peut également être utilisé par le module d'application pour présenter directement des informations de l'application utilisatrice.

#### **B.1.6 Module d'application**

C'est l'application utilisatrice qui traite, échange et gère les entités MHEG. Elle traite ces entités via l'interface décrite dans la présente Recommandation (par des actions appliquées à chaque entité MHEG), en usant éventuellement d'extensions répertoriées dans un catalogue privé. L'application utilisatrice peut également avoir besoin d'accéder directement à l'interface utilisateur.

On trouvera ci-après des exemples de fonctionnalités de commande et de données pour l'interface du moteur MHEG, qui peuvent être utilisés par le module d'application.

- Lancer le moteur MHEG.
- Arrêter le moteur MHEG.
- Fournir des entités MHEG ou des références à ces entités au moteur MHEG.
- Accepter des entités MHEG ou des références à ces entités fournies par le moteur MHEG.
- Déclencher le traitement d'un objet spécifié.
- Arrêter le traitement d'un objet spécifié.
- Fonctionnalités équivalentes à celles qui sont assurées par les actions élémentaires.
- Fonctionnalités pour accéder à des données dans le moteur MHEG, par exemple des attributs, des valeurs d'état ou des états du moteur.
- Fonctionnalités pour accepter les données fournies par le moteur MHEG, par exemple à la suite d'une action Return (retourner).
- Traiter les informations de gestion-systèmes indiquant les résultats des fonctions ci-dessus, par exemple les erreurs.

#### **B.1.7 Module de gestion des communications**

Ce module traite la transmission des entités reçues ou transmises et coordonne l'accès aux données nécessaires au moteur MHEG. Les données peuvent être stockées sur des dispositifs locaux (par exemple un lecteur de CD) ou distants (par exemple une base de données accessible par le biais d'un réseau). Ce module prend également en charge la transmission de l'information du moteur MHEG à l'application utilisatrice.

Le protocole de transfert n'est pas défini dans la présente Recommandation. Pendant le transfert, une identification donnée peut être ajoutée aux entités MHEG.

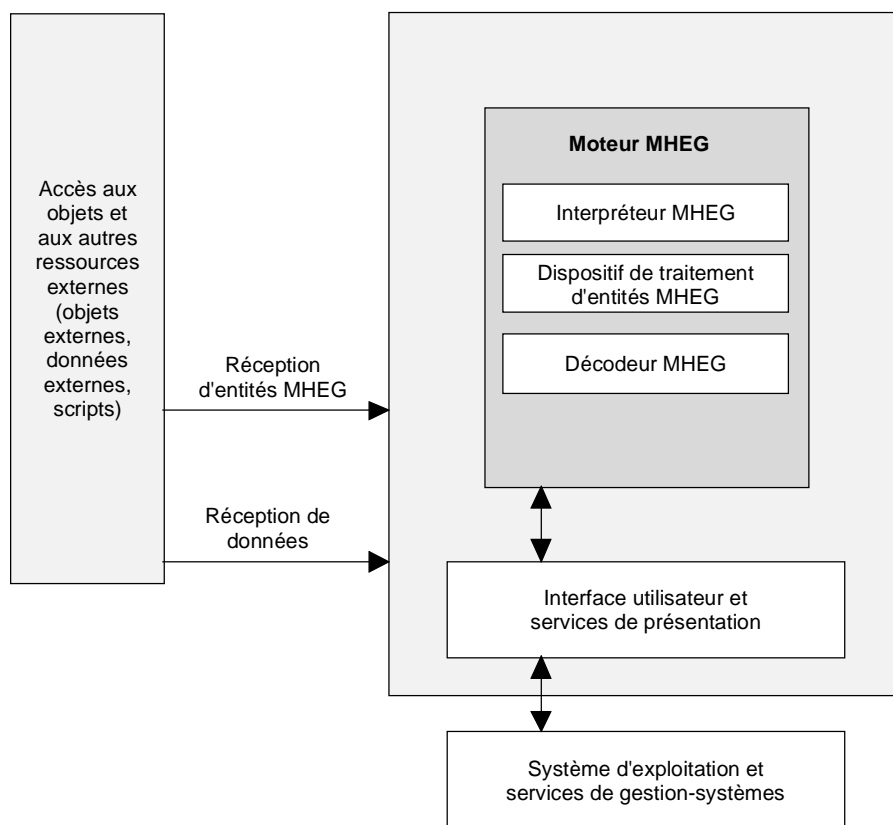
Le module de communication peut également être utilisé par le module d'application pour accéder directement à des parties de logiciel, de scripts, de données externes, etc.

## B.2 Exemples d'applications

On trouvera dans le présent sous-paragraphe la description de diverses utilisations et configurations de systèmes multimédias et hypermédias, dans des domaines tels que la formation, la présentation passive ou interactive d'informations multimédias, la rédaction conjointe et la collectique.

### B.2.1 Système de présentation passif

La Figure B.3 montre un système de présentation multimédia simple purement passif et qui sert uniquement à recevoir et à présenter des entités MHEG. Celles-ci sont envoyées par un processus externe, reçues par un module de communication et présentées sans aucune interaction avec l'utilisateur.



T0826020-96/d058

Figure B.3/T.171 – Exemple de configuration d'un système de présentation passif

## B.2.2 Système de présentation passif amélioré

La Figure B.4 montre un système de présentation multimédia passif, qui prend en charge jusqu'à un certain point les applications utilisatrices pour certaines opérations spécifiques. Dans cette configuration, il n'existe aucune interaction avec l'utilisateur, mais le moteur MHEG peut accéder aux entités MHEG sous le contrôle de l'application utilisatrice.

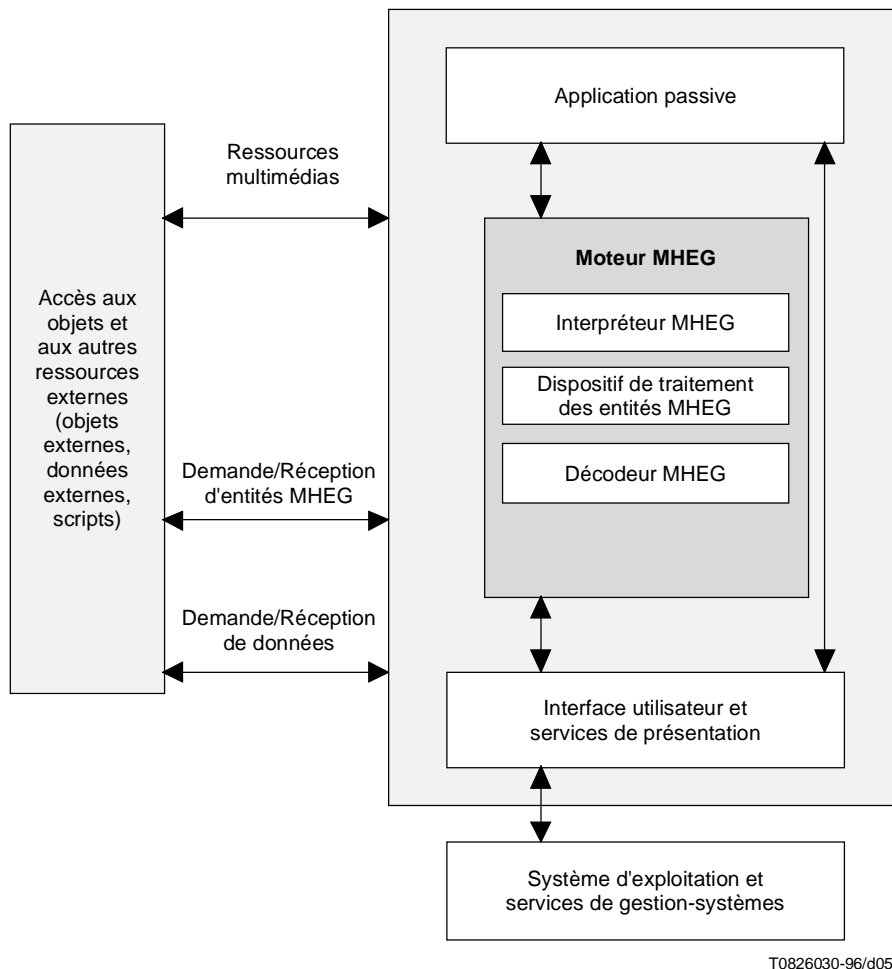


Figure B.4/T.171 – Exemple de configuration d'un système de présentation passif amélioré

### B.2.3 Système de présentation interactif

La Figure B.5 montre un système de présentation multimédia/hypermédia qui est pleinement interactif avec l'utilisateur.

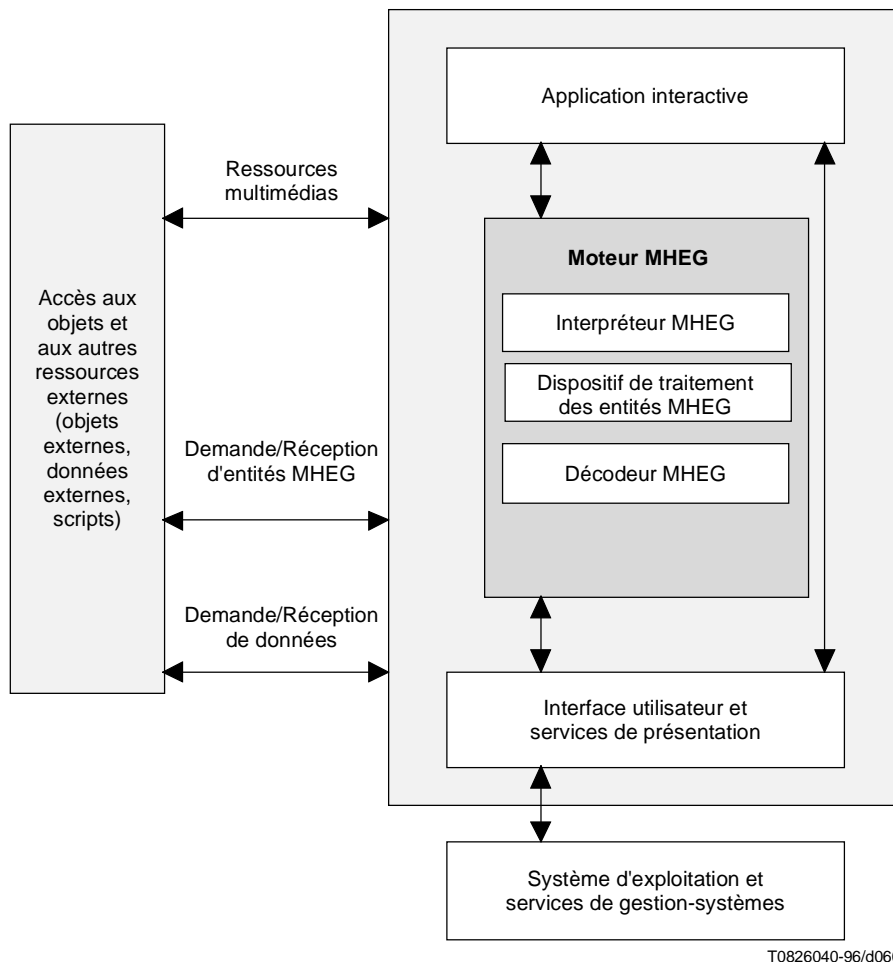
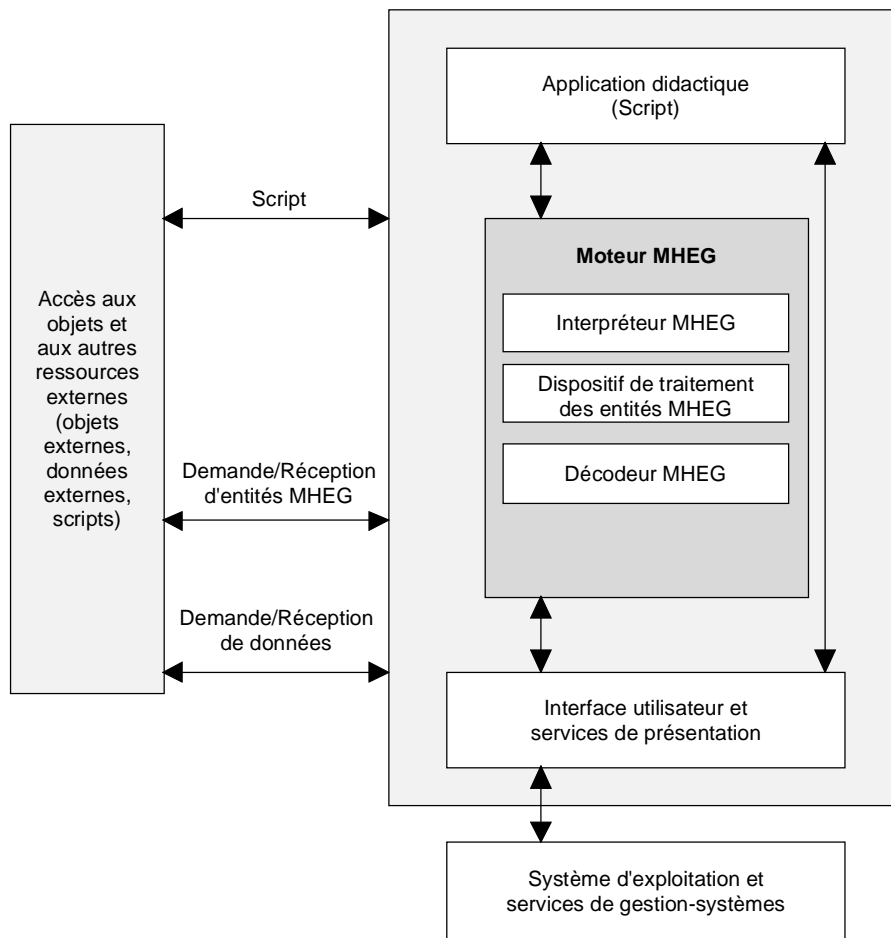


Figure B.5/T.171 – Exemple de configuration d'un système de présentation interactif

## B.2.4 Système télématique interactif

La Figure B.6 montre un système télématique interactif dans l'environnement didacticiel exposé dans la Recommandation F.740. Dans ce système, un étudiant peut réagir à la présentation multimédia et dialoguer avec le système hypermédia. Selon l'interaction, des entités MHEG sont échangées entre le terminal et le serveur. Un traitement de données ou une interaction complexe peut avoir lieu au moyen d'une application utilisatrice qui peut utiliser un script.

Le moteur MHEG analyse les réactions de l'étudiant et peut demander de nouvelles entités MHEG ou des données de contenu requises par les entités MHEG déjà reçues. Les réponses formelles de l'étudiant sont renvoyées au serveur sous la forme d'objets MHEG quelconques, et peuvent être reçues par le système le plus sophistiqué utilisé par l'enseignant.



T0826050-96/d061

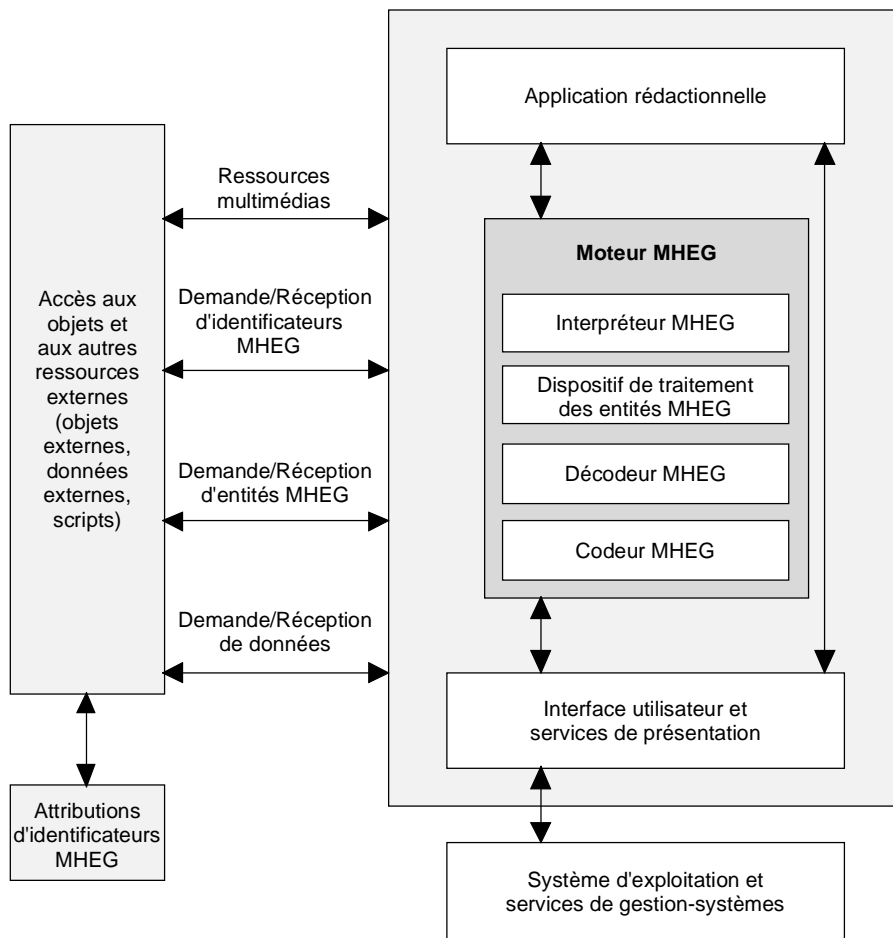
Figure B.6/T.171 – Exemple de configuration d'un service télématique interactif



### B.2.5 Système rédactionnel

La Figure B.7 montre un système rédactionnel qui crée des objets MHEG et utilise différents services du moteur MHEG pour dialoguer avec l'auteur, par exemple l'attribution d'identificateurs MHEG aux objets et l'échange d'objets MHEG.

Dans ce système, le moteur MHEG et l'application peuvent accéder au module chargé d'attribuer des identificateurs uniques aux différents objets MHEG au moment de la création.



T0826060-96/d062

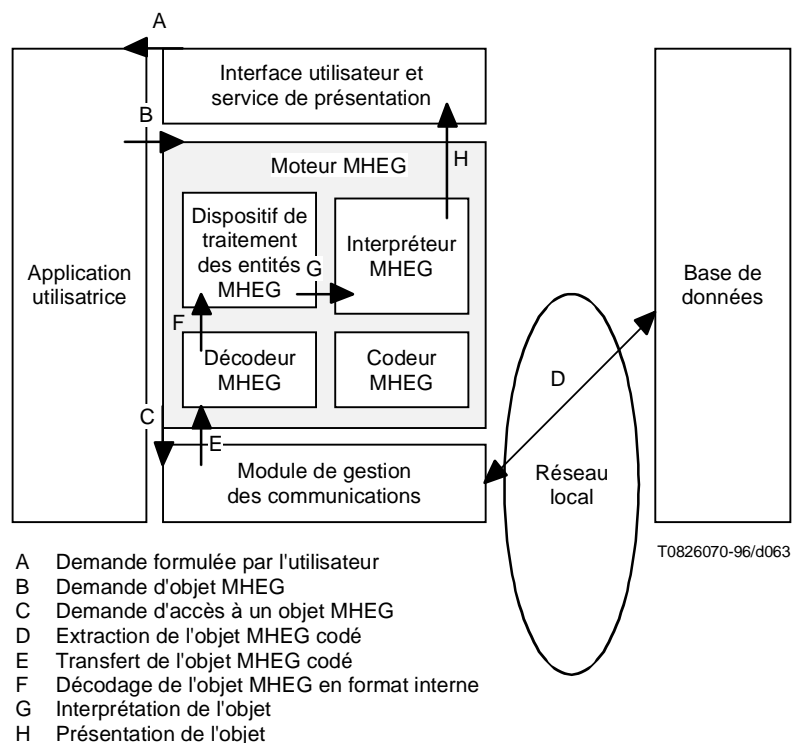
Figure B.7/T.171 – Exemple de configuration d'un système auteur

## B.2.6 Système médical

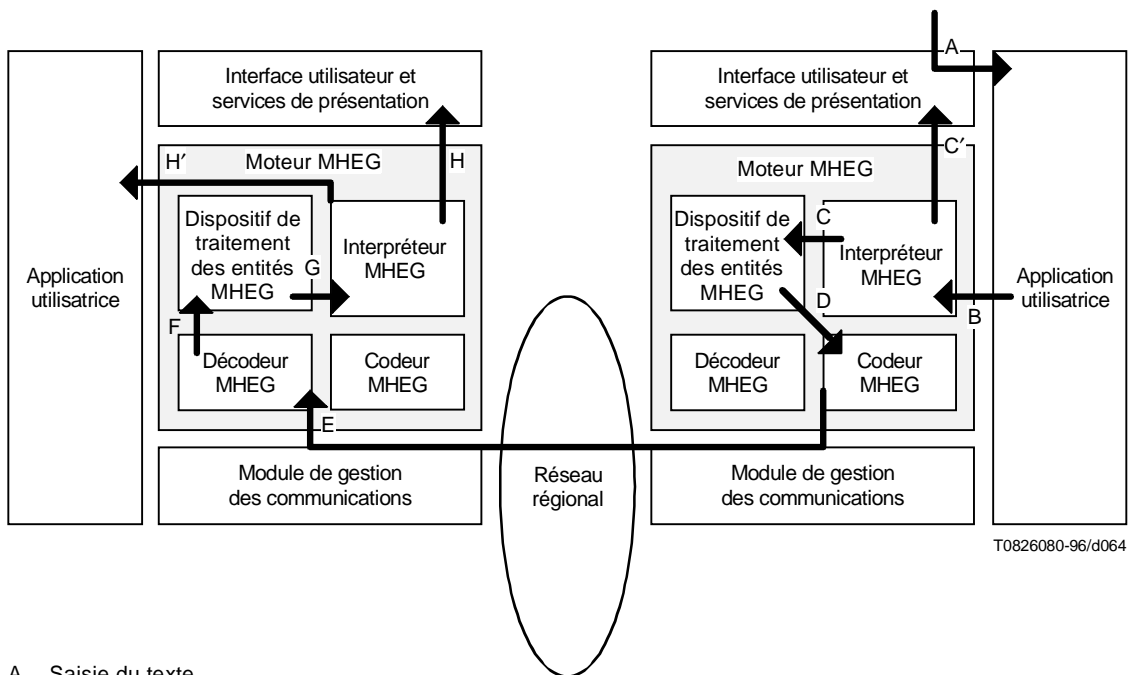
Un système de support d'information cherchant des données sur des patients, chargeant ces données dans la base de données et permettant à des médecins de dialoguer sur un réseau est un exemple type de système d'information multimédia/hypermédia et de système de travail collectif.

Les applications suivantes sont typiques:

- accès à la base de données et mise à jour de celle-ci (voir Figure B.8): par l'intermédiaire d'un réseau local, des médecins peuvent accéder à la base de données dans laquelle sont stockées des informations médicales (sous forme de texte, d'image, de son ou de signaux vidéo) et la mettre à jour. Ils peuvent consulter et mettre à jour les dossiers médicaux sur écran comme s'ils travaillaient sur des documents en papier. Lorsqu'un médecin demande à consulter un dossier, une application utilisatrice demande l'objet MHEG correspondant au moteur MHEG. Celui-ci récupère l'objet de la base de données et le présente en collaboration avec le gestionnaire des communications et le décodeur MHEG;
- travail collectif: les médecins et les spécialistes travaillant dans divers hôpitaux échangent des dossiers médicaux et se consultent sur les cas difficiles. A cette fin, ils ont également besoin d'un système de vidéocommunication interactif pour améliorer la convivialité des échanges;
- introduction de texte par un médecin: la Figure B.9 montre le flux informationnel correspondant au texte introduit par un médecin dans un hôpital local et apparaissant sur l'écran d'un autre médecin se trouvant dans un autre hôpital;
- introduction de signaux vidéo depuis une caméra: la Figure B.10 montre le flux informationnel correspondant aux signaux vidéo produits par une caméra installée dans un hôpital distant et affichés sur l'écran d'un médecin de l'hôpital local.



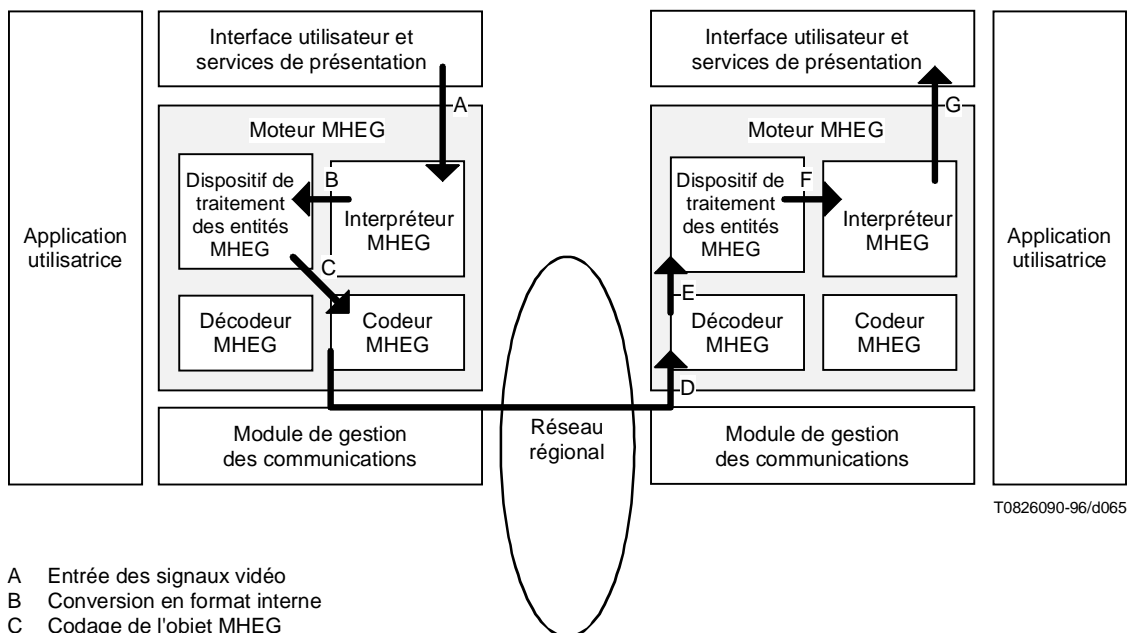
**Figure B.8/T.171 – Procédure d'accès à une base de données et de mise à jour de celle-ci**



T0826080-96/d064

- A Saisie du texte
- B Demande de modification de l'objet MHEG correspondant
- C Conversion en format interne
- C' Présentation de l'objet mis à jour
- D Codage de l'objet MHEG
- E Transfert de l'objet MHEG codé
- F Décodage de l'objet MHEG et conversion en format interne
- G Interprétation de l'objet
- H Présentation de l'objet
- H' Indication de la modification à l'application utilisatrice

**Figure B.9/T.171 – Cheminement de l'information contenue dans le texte saisi**



T0826090-96/d065

- A Entrée des signaux vidéo
- B Conversion en format interne
- C Codage de l'objet MHEG
- D Transfert de l'objet MHEG codé
- E Décodage de l'objet MHEG et conversion en format interne
- F Interprétation de l'objet
- G Présentation de l'objet

**Figure B.10/T.171 – Cheminement de l'information contenue dans les signaux vidéo introduits**

## Annexe C

### Interfaçage avec les Normes et Recommandations relatives aux informations monomédias

#### C.1 Exemple d'objet *contenu* de type image fixe

L'exemple suivant montre un objet *contenu* ayant pour données de contenu une image fixe codée. La composition de ces données est la suivante (voir Figure C.1):

- 256 (absc.) × 128 (ord.) pixels
- 150 points/pouce (absc.) × 150 points/pouce (ord.)
- 256 niveaux de gris (8 bits/pixel)
- Compression JPEG, de base et hiérarchique (4 niveaux).

L'objet *contenu* peut contenir les informations suivantes:

- 1) données de contenu: données codées correspondantes;
- 2) classification des données: image fixe;
- 3) perception d'origine du contenu: pas codé. Ces informations doivent être extraites du crochet et des données de contenu;
- 4) information de crochet de contenu:
  - 5) identification du codage du contenu: ISO 10918 (JPEG) dans le catalogue enregistré;
  - 6) description du codage du contenu:
    - 7) octet 1: image de luminance
    - 8) octets 2 et 3: densité de pixels en ordonnée (150 points par pouce)
    - 9) octets 4 et 5: densité de pixels en abscisse (150 points par pouce)
    - 10) octet 6: 8 bits/pixel
    - 11) octets 7 et 8: 256 (absc.) pixels
    - 12) octets 9 et 10: 128 (ord.) pixels
    - 13) octets 11 et 12: de base et hiérarchique
    - 14) octets 13 et 128: 128 octets de commentaires: «casa mía».



T0826100-96/d066

Figure C.1/T.171 – Image fixe

## C.2 Exemple d'objet *contenu* de type audio

Le présent exemple montre un objet *contenu* ayant pour données de contenu des informations audio. La composition de ces données est la suivante:

- 1) durée: 10 secondes
- 2) fréquence d'échantillonnage: 48 kHz
- 3) mode stéréo
- 4) signaux audiophoniques MPEG, couche 2
- 5) débit: 256 kbit/s
- 6) longueur: 352 000 octets
- 7) droit d'auteur détenu par Deutsch Gramophon
- 8) «morceau de musique baroque composée par Haydn».

L'objet *contenu* peut contenir les informations suivantes:

- 1) données de contenu: les données codées correspondantes figurent dans les données de contenu;
- 2) classification des données: audio;
- 3) perception d'origine du contenu: pas codé. Ces informations doivent être extraites du crochet et des données de contenu;
- 4) information de crochet de contenu:
  - 5) identification du codage du contenu: ISO 11172, signaux audiophoniques MPEG et Annexe E/T.101 dans le catalogue enregistré;
  - 6) description du codage du contenu:
    - 7) octet 1: débit en kbit/s: 256
    - 8) octet 2: signaux audiophoniques JPEG, couche 2
    - 9) octet 3: mode stéréo
    - 10) octet 4: fréquence d'échantillonnage en kHz: 48
    - 11) octet 5: durée en secondes: 10
    - 12) octets 6 à 10: longueur en octets: 352 000
    - 13) octet 11: droits d'auteur? Oui
    - 14) octets 12 et 128: 128 octets de commentaires: «morceau de musique baroque de Haydn; droits d'auteur détenus par Deutsch Gramophon».

## Annexe D

### Support hypertexte/hypermédia

#### D.1 Introduction

La présente Recommandation propose divers moyens de réaliser des systèmes hypertextes et hypermédias, autrement dit des hyperliens. Les modalités de la réalisation dépendent entièrement de l'application. La présente annexe a pour objet d'exposer plusieurs manières de procéder et de fournir quelques indices aux personnes impliquées de la réalisation.

Un des points essentiels qu'il faut considérer pour réaliser des systèmes hypertextes/hypermédias est la façon de créer un mécanisme dit d'ancrage et de navigation. L'ancre est une entité particulière qui indique que d'autres entités sont raccordées à un point ou à une zone donnée d'une certaine entité. Le point ou la zone à laquelle une ancre est associée est appelé zone sensible. Un système de présentation peut conférer un rendu particulier à une telle zone pour indiquer qu'une ancre y est rattachée, par exemple par l'affichage des mots dans une couleur différente ou par leur soulignement. Le mécanisme de navigation permet d'examiner une autre présentation d'entité via l'ancre reliant les deux entités.

## D.2 Mécanisme pour hypertexte et hypermédia

Les mécanismes pour hypertexte et hypermédia exposés dans la présente annexe sont de trois types:

- objet transparent;
- ancre attachée à une entité;
- traitement des événements.

### D.2.1 Objet transparent

A partir d'un objet *contenu*, on peut créer un objet transparent et l'utiliser comme ancre. Pour ce faire, on pourra procéder de la manière suivante:

- 1) créer un objet *contenu* contenant certaines données, par exemple une phototrame;
- 2) créer un rt-contenu à partir de ce modèle;
- 3) mettre son attribut de perceptibilité à 0;
- 4) mettre son attribut de sélectionnabilité à sélectionnable;
- 5) attacher ce rt-contenu à un objet MHEG en le positionnant à l'endroit où une ancre doit être placée;
- 6) créer un lien avec la condition «si ce rt-contenu est sélectionné, actionner un autre objet MHEG», par exemple afficher un autre texte.

L'objet transparent peut être considéré comme une ancre. Sa zone sensible peut être déterminée par certaines actions élémentaires de placement. La navigation est spécifiée par le lien associé au rt-objet transparent.

L'objet transparent peut contenir tous types de données, du moment que le crochet en indique le contenu. Celui-ci doit contenir au moins une indication de la taille de l'objet, car cette taille est également adoptée pour la zone sensible. Si la perceptibilité est mise à 0, l'apparence de l'objet, c'est-à-dire la façon dont un objet sélectionnable est affiché à l'écran, dépend entièrement de l'interface GUI. L'attribut de perceptibilité pourrait être mis non pas à 0 mais autour de 0. Dans ce cas, l'objet peut être perçu en semi-transparence, ce qui permet à l'utilisateur de distinguer la zone sensible. Le positionnement de ce rt-contenu par rapport à un certain objet MHEG est une opération complexe. Si l'objet MHEG change de présentation (déplacement, redimensionnement, changement de police de caractères), le rt-contenu qui y est attaché doit suivre. Etant donné que la présente annexe définit la forme de présentation finale, cela pourrait ne pas être un gros problème dans certains cas.

### D.2.2 Ancre rattachée à une entité

En utilisant l'ancre définie dans la présente Recommandation, un mécanisme d'ancrage peut être réalisé d'une manière simple. Pour ce faire, on pourra procéder de la manière suivante:

- 1) créer une ancre;
- 2) créer un rt-objet à partir de cette ancre;
- 3) mettre son attribut de sélectionnabilité à sélectionnable;
- 4) attacher ce rt-objet (rt-ancre) à un objet MHEG par l'action élémentaire «attach anchor» (attacher ancre);
- 5) créer un lien avec la condition «si ce rt-contenu est sélectionné, actionner l'autre objet MHEG», par exemple afficher un autre texte.

L'objet *ancre* contient des informations concernant l'ancre qui varient en fonction du contenu auquel l'ancre est attachée. C'est un type de schéma d'adressage qui indique l'emplacement de la zone sensible pour l'ancre. Le crochet peut être utilisé pour désigner le schéma d'adressage utilisé dans cette ancre. Sa zone sensible doit être positionnée par une interface GUI utilisatrice interprétant l'information d'ancrage. Sa présentation dépend également de cette interface.

Etant donné qu'une ancre et un objet *contenu* ne diffèrent que par leur présentation, les attributs d'objet contenu peuvent servir à contrôler certains comportements des ancres.

Si la présentation de l'objet MHEG auquel l'ancre est attachée change, celle-ci suit la position spécifiée par l'information d'adresse, car l'interface GUI est informée du changement de présentation et peut modifier la position de l'ancre en conséquence.

### D.2.3 Traitement des événements

Un moteur MHEG peut recevoir certains événements de l'extérieur. Ce mécanisme de traitement d'événement peut également être utilisé pour la réalisation de systèmes hypermédias et hypertextes. Pour ce faire, on pourra procéder de la manière suivante:

- 1) déclarer les événements nécessaires à l'intérieur d'un descripteur;
- 2) créer un objet *contenu* correspondant au descripteur;
- 3) créer un rt-contenu à partir de ce modèle;
- 4) créer un lien avec la condition «si un certain événement a lieu, actionner l'autre objet MHEG», par exemple afficher un autre texte.

L'attribut *événement* et l'attribut *donnée d'événement* associé peuvent contrôler le mécanisme hyperlien.

Par exemple, pour réaliser une phototrame qui dépende de la position sur laquelle est cliqué le dispositif de pointage (une souris par exemple), on peut:

- 1) déclarer un événement «cliquement de souris» à l'intérieur d'un descripteur;
- 2) créer un objet *contenu* contenant une phototrame;
- 3) créer un rt-contenu à partir de cet objet;
- 4) créer un lien avec la condition «si l'événement cliquement de souris survient sur ce rt-objet, actionner un objet MHEG dépendant de l'information de position stockée dans l'attribut *donnée d'événement*».

Il ne s'agit là que d'un exemple. Etant donné que le mécanisme d'événement fournit une méthode générale pour assurer la synchronisation et l'interaction tant entre le moteur MHEG et les systèmes extérieurs qu'entre les objets MHEG, de nombreuses autres méthodes peuvent être mises au point.

### D.2.4 Autre possibilité

Un objet composite peut être réalisé à partir d'un certain nombre d'objets *contenu* contenant chacun un morceau ou une unité d'information. L'information complète est obtenue en regroupant ces éléments. Par exemple, un objet *texte* peut être réalisé sous forme d'objet composite au moyen d'un certain nombre de contenus correspondant chacun à un mot. Dans ce cas, chaque objet *contenu* peut être une zone sensible si chacun de ces objets est disposé de manière appropriée et si leur attribut de sélectionnabilité est mis à sélectionnable.

Cette méthode peut s'avérer difficile et compliquée pour un objet complexe.

### D.2.5 Réalisation et coût

Dans une application réelle, il peut être utile non seulement d'utiliser les méthodes décrites plus haut mais aussi de les combiner. En outre, la présente annexe montre comment obtenir d'autres réalisations de la fonction hyperlien. Il est recommandé de choisir une méthode appropriée en fonction du coût de la réalisation et des spécifications du système.

## D.3 Exemple d'interactions entre un système MHEG et le logiciel de navigation du Web

Le réseau mondial Web est un système hypermédia largement utilisé sur Internet. Le Web emploie le HTML (Langage de balisage hypertexte) pour le balisage hypertexte et l'URL (Identificateur uniforme de ressources) pour l'identification des objets Web. Un logiciel de navigation du Web, associé à un système de présentation, est responsable du traitement de toutes les ancrs qui apparaissent dans l'hypertexte en langage HTML.

Le Web utilise une technique d'identification d'objet, appelée URL, qui est défférente de celle qui est employée dans la présente Recommandation, à savoir l'identificateur public défini dans l'ISO/CEI 9070. Il y a donc deux questions à examiner:

- les ancrs qui se trouvent dans un document HTML pointant sur des objets MHEG;
- les ancrs qui se trouvent dans un objet MHEG pointant sur des documents HTML.

Pour harmoniser les techniques, la solution pourrait consister à utiliser l'identificateur public décrit dans l'ISO/CEI 9070.

Dans ce cas, un identificateur public aura la forme suivante:

Identificateur public = +IETF//<Protocole>://xxx.yyy.edu/.../zzz.mhg

où zzz.mhg est le type MIME associé, et où <Protocole> prendra par exemple la valeur 'T.176', 'ftp' ou 'http'.

L'IETF devra demander à l'autorité compétente de l'ISO/CEI 9070 d'enregistrer son identificateur. En outre, un type MIME pour les objets MHEG conformes à la présente Recommandation devra être enregistré auprès de l'IETF. Il appartient à l'UIT-T d'enregistrer le protocole T.176.

Un décodeur d'identificateurs publics, utilisé conjointement par le moteur MHEG et les logiciels de navigation du Web, devra figurer dans le système. Ce décodeur doit être capable de résoudre les références. Si la référence est celle d'un objet Web, cette information est passée au logiciel de navigation du Web; si l'information est un objet MHEG, elle est passée au moteur MHEG. D'un autre côté, si la référence est celle d'un objet MHEG, cette information est passée au moteur MHEG et celui-ci passe l'information à une interface GUI.

## Annexe E

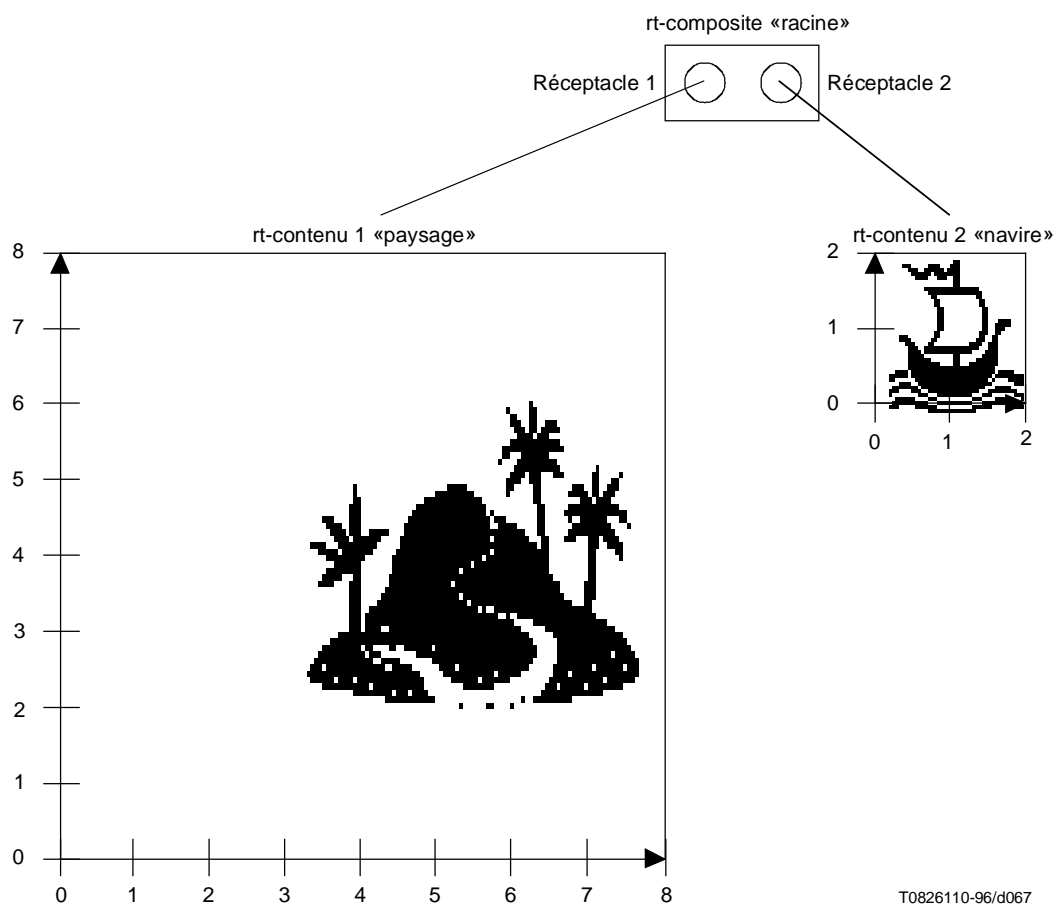
### Exemples de comportements spatiaux

#### E.1 Exemple 1

Le présent exemple montre quelques images de l'évolution de la présentation d'un rt-composite composé de deux réceptacles dans lesquels sont enfichés deux rt-contenus. Ces objets sont affichés dans la taille d'origine (OS) de leur parent, c'est-à-dire leur taille PRGS.

##### E.1.1 Structure du rt-composite et valeurs d'origine

La Figure E.1 présente la structure du rt-composite et la taille d'origine des rt-composants.



T0826110-96/d067

Figure E.1/T.171 – Taille d'origine du rt-composite et du rt-contenu



La taille d'origine du rt-composite est composée des réceptacles décrits ci-après. Les valeurs d'origine de ces trois rt-objets figurent dans le Tableau E.1.

**Tableau E.1/T.171 – Etablissement des valeurs d'origine de l'exemple 1**

	OS	Affectation d'espace RPS	Stratégie de redimensionnement	Stratégie de projection de la taille OVS	Facteur GSF	Priorité de présentation
Rt-composite «racine»	8 × 8	canal	fixe	fixe	1,0	0
Rt-contenu 1 «paysage»	8 × 8	parent	–	calculée	1,0	1
Rt-contenu 2 «navire»	2 × 2	parent	–	calculée	1,0	0

### E.1.2 Présentation initiale dans l'espace CGS

La Figure E.2 montre le mappage des tailles PVS de deux rt-contenus à la taille d'origine de leur parent.

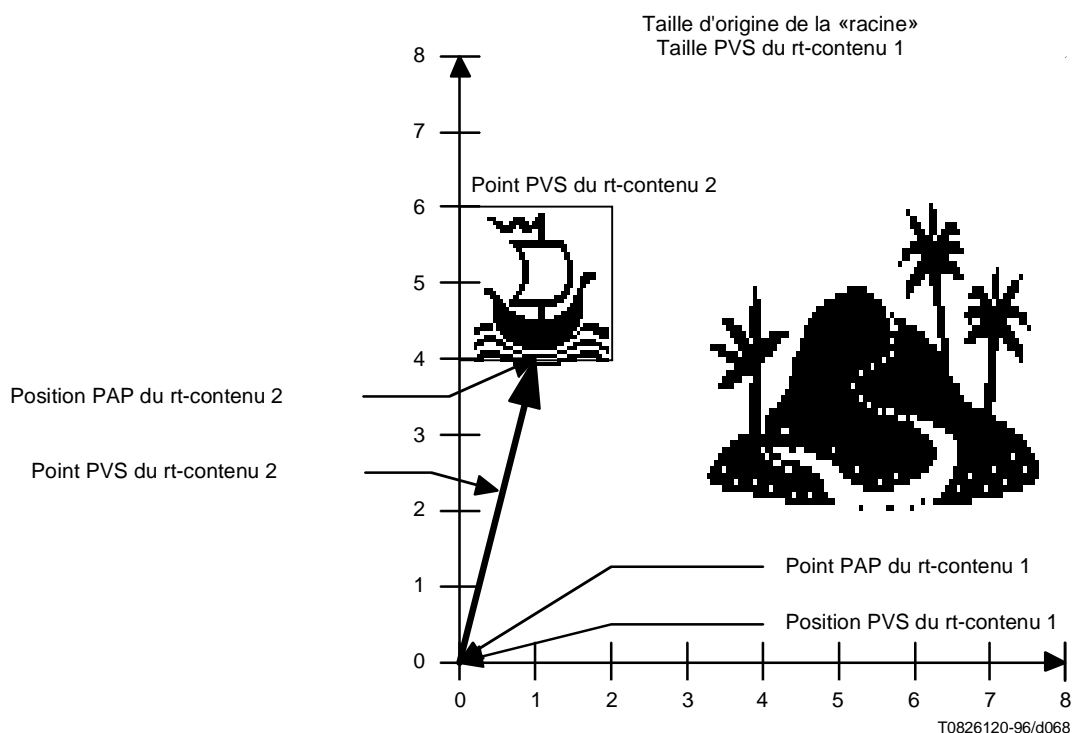
Aucune action n'est appliquée au rt-contenu 1. Les valeurs par défaut sont appliquées pour fixer la taille OVS, le point OAP, la position OVS, le point PAP et la position PVS.

Les actions suivantes sont appliquées au rt-contenu 2:

- fixer le point PAP (rt-contenu 2, x: 1, y: 0);
- fixer la position PVS (rt-contenu 2, x: 1, y: 4).

Aucune action n'étant appliquée à la taille OVS, au point OAP et à la position OVS, les valeurs par défaut correspondantes sont adoptées.

Le rt-contenu 1 masque le rt-contenu 2 car sa priorité de présentation est supérieure.



**Figure E.2/T.171 – Détermination de la taille PVS des rt-contenus par rapport à la taille d'origine de la racine**

La Figure E.3 illustre le mappage de la taille d'origine du rt-composite par rapport à son espace CPS.

Les actions suivantes sont appliquées:

- fixer la taille OVS (racine, x: 4, y: 4);
- fixer le point OAP (racine, x: 50%, y: 50%);
- fixer la position OVS (racine, x: 2, y: 4);
- fixer la position PVS (racine, x: 2, y: 4).

L'action Set PAP (fixer point PAP) n'est pas utilisée. La valeur par défaut du point PAP de la racine est retenue.

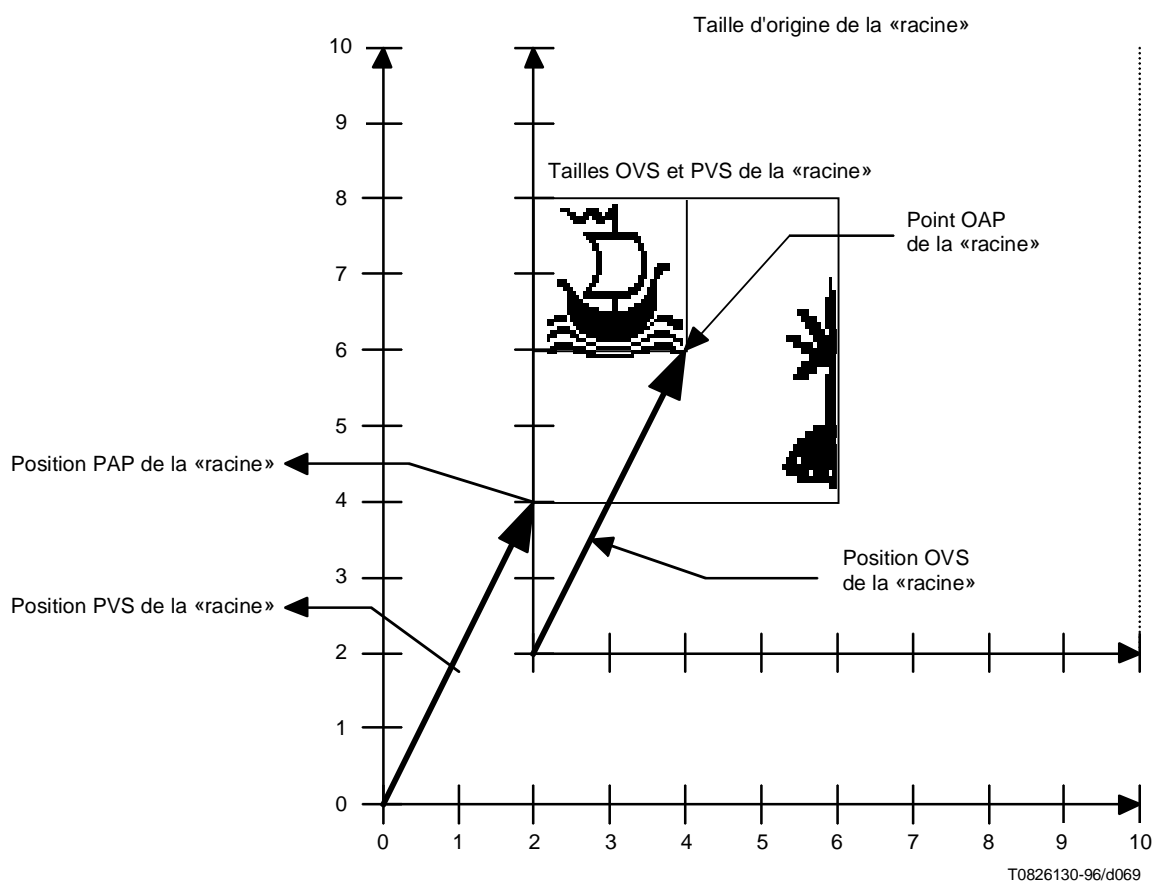


Figure E.3/T.171 – Détermination de la position PVS de la racine par rapport à l'espace CPS

### E.1.3 Défilement

La Figure E.4 montre l'effet de l'action suivante lorsqu'elle est appliquée au rt-composite «racine»:

- fixer la position OVS (racine, x: 3, y: 4).

L'effet de cette action est de faire défiler la taille OVS du rt-composite par rapport à sa taille d'origine. Les autres positions restent inchangées.

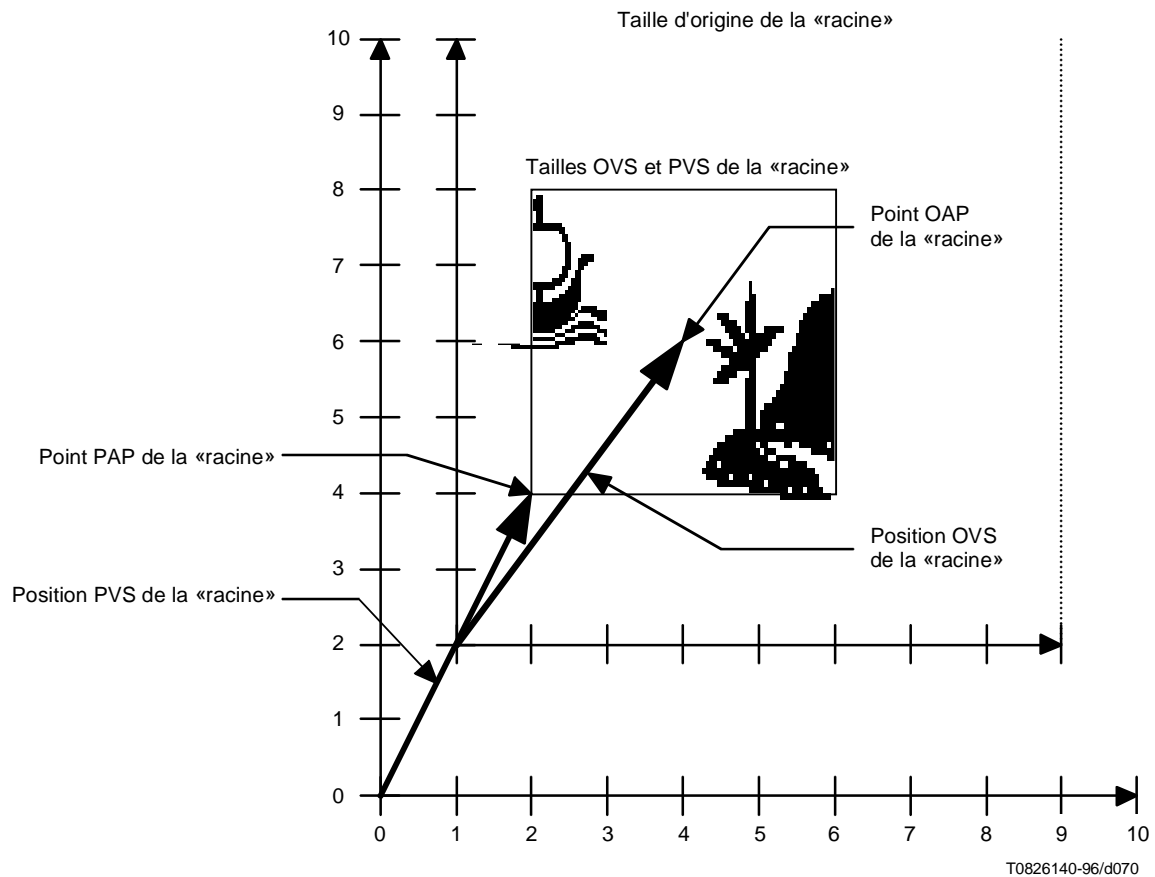


Figure E.4/T.171 – Défilement du rt-composite

### E.1.4 Changement d'échelle

La Figure E.5 montre l'effet des actions suivantes lorsqu'elles sont appliquées au rt-composite «racine»:

- fixer le point OAP (racine, x: 0, y: 0);
- fixer la position OVS (racine, x: 0, y: 0);
- fixer le facteur GSF (racine, x: 0,5, y: 0,5).

La taille PVS du rt-composite «racine» n'est pas touchée par l'action Set GSF (fixer facteur GSF) car l'attribut de la stratégie de projection OVS est mis à «fixe».

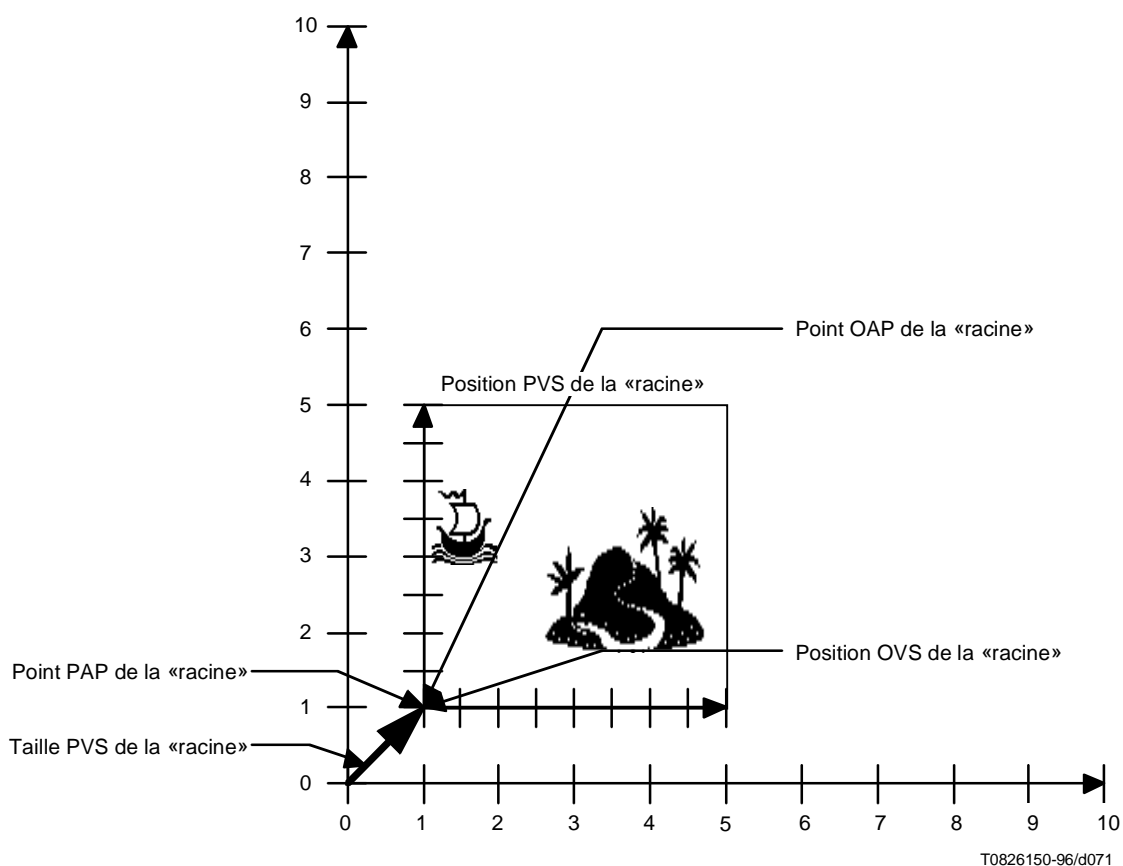


Figure E.5/T.171 – Changement d'échelle du rt-composite

### E.1.5 Mise à l'échelle des réceptacles

La Figure E.6 montre l'effet des actions suivantes lorsqu'elles sont appliquées au rt-contenu 1:

- fixer le facteur GSF (rt-contenu 1, x: 2,0 y: 2,0);
- fixer la position PVS (rt-contenu 1, x: 3, y: 0).

Etant donné que le point PAP avait été fixé à x: 1 et y: 0, il n'est pas touché par le changement du facteur GSF du rt-contenu 1. Si la position relative du PAP doit être maintenue comme spécifié ci-dessus indépendamment des variations du facteur GSF, la spécification de sa position devra s'effectuer de manière relative.

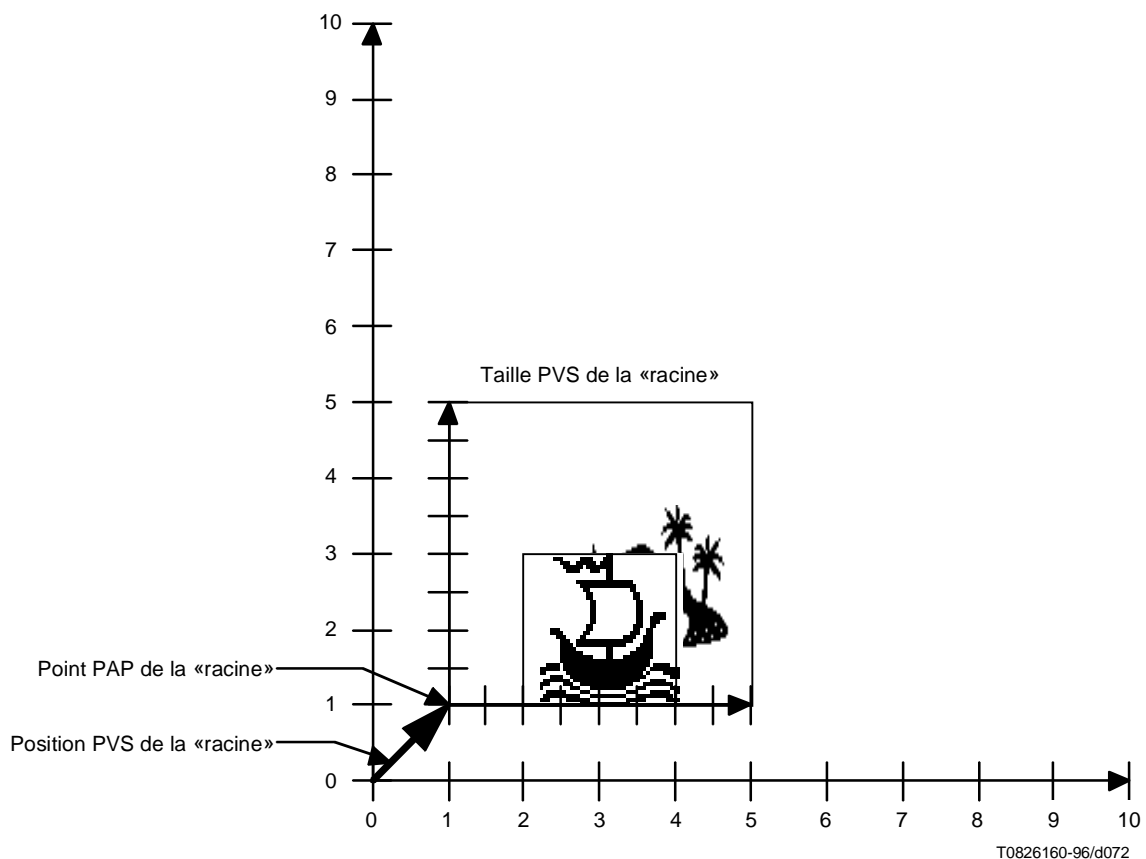


Figure E.6/T.171 – Changement d'échelle et déplacement du rt-contenu 1

## E.2 Exemple 2

Le présent exemple montre un menu déroulant type dans lequel un sous-menu est affecté directement à un canal. Cette caractéristique permet de contrôler directement un sous-menu sans qu'il soit nécessaire de présenter un élément dans l'espace RPS de son parent.

### E.2.1 Structure du rt-composite et valeurs d'origine

La Figure E.7 montre la structure de composition du «menu». Les rt-contenus «fichier» et «option» composent un menu principal. Les rt-contenus «ouvrir», «fermer» et «nouveau» composent un sous-menu de «fichier».

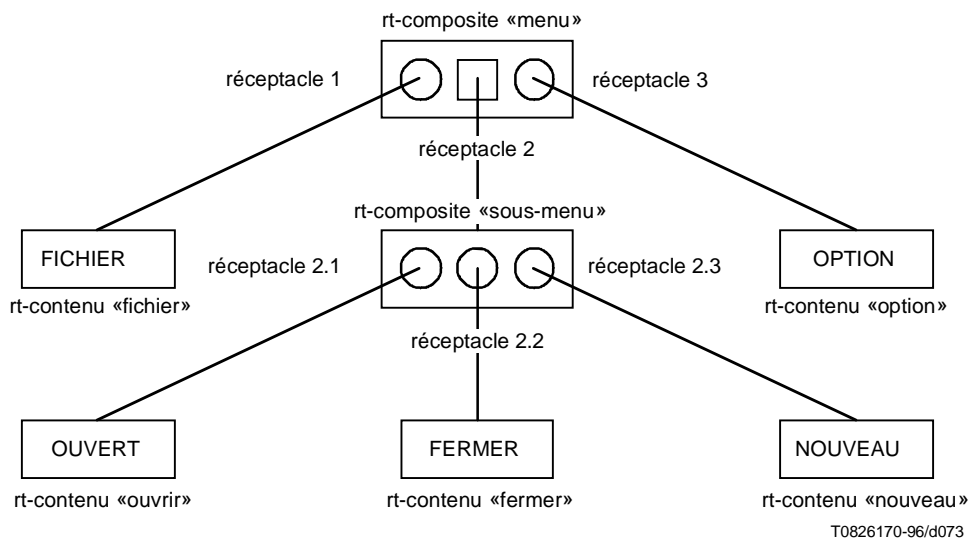


Figure E.7/T.171 – Structure de «menu»

### E.2.2 Présentation initiale dans l'espace CGS

La Figure E.8 montre la présentation initiale d'un menu dans un canal. Les rt-composites «menu» et «sous-menu» sont directement présentés dans le canal.

Les rt-contents «fichier» et «option» sont projetés dans le rt-composite «menu». La taille PVS du «menu» est positionnée dans le canal comme le montre la Figure E.8. Les rt-contents «ouvrir», «fermer» et «nouveau» sont projetés dans le rt-composite «sous-menu». La taille PVS du «sous-menu» est positionnée dans le canal comme le montre la Figure E.8.

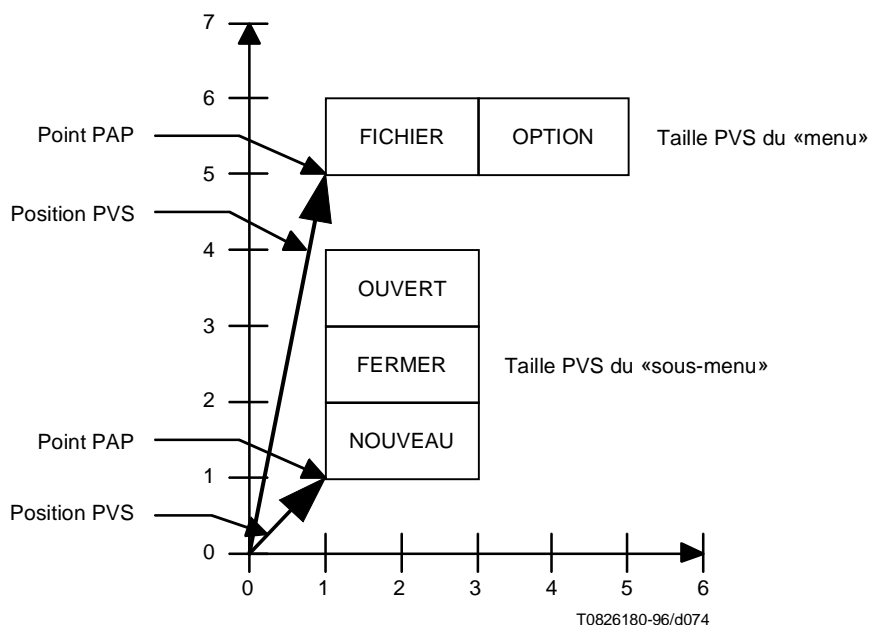


Figure E.8/T.171 – Présentation d'origine d'un menu

**Tableau E.2/T.171 – Etablissement des valeurs d'origine de l'exemple 2**

	Espace RPS	Stratégie de redimensionnement
rt-composite «menu»	CGS	minimum
rt-contenu «fichier»	parent	–
rt-contenu «option»	parent	–
rt-composite «sous-menu»	CGS	minimum
rt-composite «ouvrir»	parent	–
rt-composite «fermer»	parent	–
rt-composite «nouveau»	parent	–

## Annexe F

### Récapitulatif des identificateurs d'objets

Les identificateurs d'objet se présentent sous la forme {racine arc1 arc2 arc3} avec:

- **racine** = 2, norme commune ISO-UIT-T
- **arc1** = 19, identificateur de norme MHEG
- **arc2** = 1, version de la norme
- **arc3** = numéro de classe défini dans la classe *objets MH*.

**Tableau F.1/T.171 – Récapitulatif des identificateurs d'objet**

Nom du module ASN.1	Identificateur du module ASN.1	Identificateur de la classe MHEG	Mnémonique	Classe identifiée
ISOMHEG-mh	Néant	Néant	mh	Objet-MH
ISOMHEG-ac	{2 19 1 1}	{2 19 1 1}	ac	Action
ISOMHEG-lk	{2 19 1 2}	{2 19 1 2}	lk	Lien
ISOMHEG-mo	Néant	Néant	mo	Modèle
ISOMHEG-sc	{2 19 1 3}	{2 19 1 3}	sc	Script
ISOMHEG-cp	Néant	Néant	cp	Composant
ISOMHEG-ct	{2 19 1 4}	{2 19 1 4}	ct	Contenu
ISOMHEG-mu	{2 19 1 5}	{2 19 1 5}	mu	Contenu multiplexé
ISOMHEG-co	{2 19 1 6}	{2 19 1 6}	co	Composite
ISOMHEG-cr	{2 19 1 7}	{2 19 1 7}	cr	Contenant
ISOMHEG-de	{2 19 1 8}	{2 19 1 8}	de	Descripteur
ISOMHEG-ud	{2 19 1 9}	Néant	ud	Définition utile
ISOMHEG-ea	{2 19 1 10}	Néant	ea	Action élémentaire

## Annexe G

### Index

(Cet index correspond à la version anglaise de la Recommandation)

Action-Class.....	274	Comparison-Operator.....	276
Action-Object .....	275	Comparison-Value .....	276
Activation-Status.....	273	Comparison-Value-Constant .....	274
Add .....	284	Component-Class .....	258
Alias-Spec.....	283	Composite-Class .....	278
Alias-Spec-Macro .....	283	Composition-Element.....	279
Alias-Spec-Param .....	283	Condition.....	276
Alternative-Object .....	281	Constraint-Condition .....	275
Ancestor .....	273	Container-Class .....	280
Anchor .....	289	Container-Close-down .....	280
Anchor-Spec .....	289	Container-Element .....	280
Anchor-Spec-Macro.....	289	Container-Element-Reference .....	259
Anchor-Spec-Param.....	289	Container-Start-up.....	280
Aspect-Ratio .....	272	Content.....	277
Aspect-Ratio-Macro.....	287	Content-Class .....	277
Aspect-Ratio-Param.....	286	Content-Class-Information.....	281
Associated-Model.....	279	Content-Data .....	277
Attach-Anchor .....	289	Content-Hook.....	268
Availability-Close-down.....	279	Content-Object-Ref-Macro .....	283
Availability-Start-up .....	279	Content-Object-Ref-Param.....	283
Catalogued-Content-Encoding.....	268	Copy.....	284
Catalogued-Elementary-Action.....	283	CPS-Duration.....	289
Catalogued-Event.....	268	CPS-Initialisation .....	289
Catalogued-Extended-Elementary-Action .....	268	CPS-Initialisation-Macro .....	289
Catalogued-Media-Type .....	268	CPS-Initialisation-Param.....	289
Catalogued-Script-Classification .....	268	CPS-Size .....	290
Catalogued-Script-Encoding.....	268	Current-Point-Spec.....	270
Catalogued-Style.....	268	Current-Point-Spec-Macro .....	270
Catalogued-Style-Macro .....	289	Current-Point-Spec-Param .....	270
Catalogued-Style-Param .....	289	CV .....	271
Cat-Content-Classification.....	268	Data-Element .....	284
Cat-Ext-Attribute .....	268	Data-Element-Macro.....	284
Cat-Ext-Attribute-Macro.....	274	Data-Element-Param .....	284
Cat-Ext-Attribute-Param.....	274	Data-Inclusion.....	277
Cat-Ext-Elementary-Action-Macro.....	284	Data-Reference.....	259
Cat-Ext-Elementary-Action-Param.....	283	Delay .....	283
Channel-Availability-Status .....	271	Description.....	258
Channel-Information .....	281	Descriptor-Class.....	281
Channel-Perceptability.....	270	Elementary-Action .....	290
Channel-Perceptability-Macro .....	289	EmptyChild .....	273
Channel-Perceptability-Param .....	289	Evaluated-Boolean.....	267
Channel-Reference.....	261	Evaluated-Integer .....	267
Channel-Target .....	263	Evaluated-List .....	268
Channel-Target-Macro.....	263	Evaluated-Numeric .....	267
Channel-Target-Param.....	263	Evaluated-Ratio.....	267
Child .....	273	Evaluated-Reference .....	268
Class-ID .....	258	Evaluated-String.....	268
Class-Specific-Information .....	281	Evaluated-Value.....	267
Comparison-Operation.....	276	Event-Mapping.....	282



Expected-Axis-Result-Param .....	272	Get-OVS .....	266
Expected-OVD-Result .....	272	Get-OVS-Position .....	266
Expected-OVD-Result-Macro .....	272	Get-PAP .....	266
Expected-OVD-Result-Param .....	273	Get-POS .....	266
Expected-PVD-Result .....	272	Get-PVD .....	266
Expected-PVD-Result-Macro .....	272	Get-PVS .....	266
Expected-PVD-Result-Param .....	272	Get-PVS-Position .....	265
External-ID .....	259	Get-Reference .....	267
Generic-Boolean .....	264	Get-Rt-Composite-Address .....	266
Generic-Boolean-Macro .....	264	Get-Stream-Chosen-State .....	265
Generic-Boolean-Param .....	263	Get-User-Spatial-Control .....	265
Generic-Integer .....	264	GF .....	269
Generic-Integer-Macro .....	264	GF-Macro .....	269
Generic-Integer-Param .....	264	GF-Param .....	269
Generic-List .....	265	GSF .....	272
Generic-List-Elt-ID-Macro .....	265	GVF-Target .....	288
Generic-List-Elt-ID-Param .....	265	Initial-Point-Spec-Macro .....	270
Generic-List-Macro .....	265	Initial-Point-Spec-Param .....	270
Generic-List-Param .....	265	Interaction-Status .....	271
Generic-Numeric .....	264	Interaction-Status-Macro .....	288
Generic-Numeric-Macro .....	264	Interaction-Status-Param .....	288
Generic-Numeric-Param .....	264	Interaction-Type .....	271
Generic-Ratio .....	264	Interaction-Type-Macro .....	271
Generic-Ratio-Macro .....	264	Interaction-Type-Param .....	271
Generic-Ratio-Param .....	264	Length-Spec .....	269
Generic-Reference .....	265	Length-Spec-Macro .....	269
Generic-Reference-Macro .....	265	Length-Spec-Param .....	269
Generic-Reference-Param .....	264	Lengths-Spec .....	269
Generic-String .....	264	Link-Class .....	275
Generic-String-Macro .....	264	Link-Condition .....	275
Generic-String-Param .....	264	Link-Effect .....	276
Generic-Value .....	263	Link-Object .....	279
Generic-Value-Macro .....	263	Logical-Combination .....	276
Generic-Value-Param .....	263	Logical-Operator .....	276
Get-Any .....	267	Macro-Def-ID .....	259
Get-Catalogued-Attribute .....	266	Macro-Parameter-Resolution .....	276
Get-Data .....	266	Max-Interact-Required-Macro .....	288
Get-GSF .....	265	Max-Interact-Required-Param .....	288
Get-GVF .....	265	Mheg-ID .....	259
Get-Integer .....	266	Mh-object-Class .....	258
Get-Interaction-Ability .....	265	Mh-Reference .....	259
Get-Interaction-Status .....	265	Mh-Target .....	261
Get-List .....	267	Mh-Target-Macro .....	261
Get-Max-Interact-Required .....	265	Mh-Target-Param .....	261
Get-Min-Interact-Required .....	265	Min-Interact-Required-Macro .....	288
Get-Number-Of-Interacted-Sockets .....	265	Min-Interact-Required-Param .....	288
Get-OAP .....	266	Model-Class .....	258
Get-OS .....	266	Modifiability .....	271
Get-OVD .....	266	Modification-Status .....	271

Multiplexed-Content-Class .....	278	Root-Rt-Content-ID-Ref .....	260
Multiplexed-Stream .....	278	Root-Rt-Content-Reference.....	260
Mux-Content-Class-Info .....	281	Root-Rt-ID-Reference .....	260
Navigation-Command .....	273	Root-Rt-Mux-ID-Reference .....	260
Navigation-Command-Macro .....	273	Root-Rt-Mux-Reference.....	260
Navigation-Command-Param.....	273	Root-Rt-Reference .....	260
Number-Of-Performances.....	285	RPS-Assignment .....	273
Number-Of-Performances-Macro .....	285	RPS-Assignment-Macro.....	285
Number-Of-Performances-Param .....	284	RPS-Assignment-Param.....	285
Number-Of-Repetitions .....	286	Rt-Availability-Start-up.....	279
Object-Information .....	281	Rt-Availability-Status.....	273
OD .....	273	Rt-Component-Channel-Ref.....	261
OPS-Initialisation.....	268	Rt-Component-Channel-Tg.....	263
OV .....	271	Rt-Component-Channel-Tg-Macro .....	263
OVS-Proj-Strategy.....	272	Rt-Component-Channel-Tg-Param .....	263
OVS-Proj-Strategy-Macro .....	287	Rt-Component-Reference.....	260
OVS-Proj-Strategy-Param .....	287	Rt-Component-Target .....	262
Passing.....	285	Rt-Component-Target-Macro.....	262
Passing-Macro .....	285	Rt-Component-Target-Param .....	262
Passing-Param.....	285	Rt-Composite-Reference .....	261
PCV .....	271	Rt-Composite-Target.....	263
Plug.....	285	Rt-Composite-Target-Macro .....	262
Plug-In .....	285	Rt-Composite-Target-Param .....	262
Plug-In-Macro.....	285	Rt-Content-Reference.....	260
Plug-In-Param.....	285	Rt-Content-Target .....	262
Point.....	269	Rt-Content-Target-Macro.....	262
Point-Spec.....	270	Rt-Content-Target-Param.....	262
Point-Spec-Macro .....	270	Rt-Mux-Reference.....	261
Point-Spec-Param .....	270	Rt-Mux-Target .....	262
Points-Spec .....	270	Rt-Mux-Target-Macro.....	262
Point-Type-Param.....	272	Rt-Mux-Target-Param.....	262
Preparation-Status.....	274	Rt-Number-Reference .....	260
Presentation-Priority .....	273	Rt-Reference .....	260
Presentation-Priority-Macro .....	286	Rt-Script-ID-Reference .....	260
Presentation-Priority-Param.....	285	Rt-Script-Reference.....	260
Process-Indicator .....	284	Rt-Script-Target .....	262
Prop-Cat-Entry-ID .....	268	Rt-Script-Target-Macro.....	262
Ratio .....	264	Rt-Script-Target-Param.....	262
Reference .....	259	Rt-Target .....	262
Related-Object .....	281	Rt-Target-Macro .....	262
Resizing-Strategy.....	272	Rt-Target-Param.....	262
Resizing-Strategy-Macro .....	287	Run.....	284
Resizing-Strategy-Param.....	287	Running-Status .....	273
Return .....	283	Script-Class .....	277
Return-Target-Param .....	263	Script-Class-Information .....	281
Root-Rt-Component-ID-Ref .....	260	Script-Data .....	277
Root-Rt-Component-Reference .....	260	Script-Hook.....	268
Root-Rt-Composite-ID-Ref.....	261	Script-Inclusion.....	277
Root-Rt-Composite-Reference.....	261	Selectability.....	271

Selection-Status .....	271	Spatial-Control-Param.....	272
Set-Alias .....	283	Spatial-Length .....	269
Set-Aspect-Ratio .....	286	Spatial-Position .....	269
Set-Catalogued-Attribute .....	284	Spatial-Position-Spec .....	270
Set-Channel-Perceptability .....	289	Spatial-Position-Spec-Macro .....	270
Set-CPS.....	289	Spatial-Position-Spec-Param.....	269
Set-CTP .....	286	Stream-Chosen-State.....	271
Set-CV .....	288	Stream-Identification.....	271
Set-Data .....	284	Stream-Identification-Macro .....	271
Set-Event.....	290	Stream-Identification-Param .....	271
Set-GSF .....	287	Stream-Information .....	281
Set-GTF .....	286	Stream-Spec .....	288
Set-GVF.....	288	Stream-Spec-Macro.....	288
Set-Interaction-Ability .....	288	Stream-Spec-Param.....	288
Set-Interaction-Status.....	288	Substitution-Indicator.....	284
Set-OAP .....	287	Substitution-Indicator-Macro .....	284
Set-OVD .....	286	Substitution-Indicator-Param .....	284
Set-OVS.....	287	Substract.....	284
Set-OVS-Position.....	287	Synchro-Indicator.....	275
Set-OVS-Proj-Strategy .....	287	Synchro-Indicator-Macro .....	275
Set-PAP .....	287	Synchro-Indicator-Param .....	275
Set-Parameters .....	285	Synchronised-Action.....	275
Set-Perceptability.....	285	System-Readable-Material .....	281
Set-Presentation-Priority.....	285	Tail-Complement .....	259
Set-PVD-Position .....	286	Tail-Macro .....	259
Set-PVS-Position .....	287	Tail-Param.....	259
Set-Resizing-Strategy.....	287	Tail-Reference.....	259
Set-RPS-Assignment.....	285	Target-Macro .....	261
Set-Stream-Choice .....	288	Target-Param.....	261
Set-Style.....	289	Temporal-Termination .....	273
Set-Temporal-Termination .....	286	Temporal-Termination-Macro .....	286
Set-Timestones.....	286	Temporal-Termination-Param.....	286
Set-User-Spatial-Control.....	287	Temporal-Unit-Ref.....	283
Size .....	269	Temporal-Unit-Ref-Macro .....	283
Size-Spec .....	269	Temporal-Unit-Ref-Param .....	283
Size-Spec-Macro.....	269	Terminal-Point-Spec-Macro.....	270
Size-Spec-Param.....	269	Terminal-Point-Spec-Param.....	270
Socket-ID.....	259	Termination-Status .....	273
Socket-Identification.....	259	Timestone.....	286
Socket-ID-Reference.....	261	Timestone-Spec.....	286
Socket-Reference .....	261	Timestone-Spec-Macro .....	286
Socket-Tail-Complement .....	261	Timestone-Spec-Param .....	286
Socket-Tail-Reference .....	261	Trigger-Condition .....	275
Socket-Target.....	263	Update-Command .....	274
Socket-Target-Macro.....	263	User-Spatial-Control .....	272
Socket-Target-Param .....	263	User-Spatial-Control-Macro.....	287
Spatial-Control.....	272	User-Spatial-Control-Param.....	287
Spatial-Control-Macro .....	272	Value .....	263

## Index des comportements

Comportements	Etat et Attributs	Actions élémentaires	Actions Get (extraire)
Postpone behaviour		<b>Delay</b> (Temporal Unit Ref Param, ...)	
Returnability behaviour		<b>Return</b> (Return Target Param+, ...)	
Alias behaviour		<b>Set Alias</b> (Target Param+, ...)	
Extensibility behaviour	Catalogued Attribute@Cat Ext Attribute Param@Cat Ext Attribute Macro@	<b>Catalogued Elementary Action</b> ( Target Param+, ...) <b>Set Catalogued Attribute</b> ( Target Param+, ...)	Get Catalogued Attribute@
Mheg objects availability behaviour	Preparation Status@	<b>Prepare</b> (MH-Target Param+) <b>Destroy</b> (MH-Target Param+)	Get Preparation Status@
Link object activation behaviour	Activation Status@	<b>Activate</b> (Link Target Param+) <b>Deactivate</b> (Link Target Param+)	Get Activation Status@
Link object abort behaviour		<b>Link Abort</b> (Link Target Param+)	
Content class generic value storage behaviour	Data@	<b>Set Data</b> (Content Target Param+, ...) <b>Add</b> (Content Target Param+, ...) <b>Subtract</b> (Content Target Param+, ...)	Get Data@
Content class copy behaviour		<b>Copy</b> (Content Target Param, ...)	
Rt-objects availability behaviour	Rt-Availability Status@	<b>New</b> (Rt-Target Param+) <b>Delete</b> (Rt-Target Param+)	Get Rt-Availability Status@
Rt-objects running behaviour	Running Status@	<b>Run</b> (Rt-Target Param+, ...) <b>Stop</b> (Rt-Target Param+)	Get Running Status@
Rt-script passing parameter behaviour		<b>Set Parameters</b> (Rt-Script Target Param+, ...)	
Rt-scripts termination behaviour	Termination Status@		Get Termination Status@
Sockets presentation and structural dynamism behaviour		<b>Plug</b> (Socket Target Param+, ...)	
Rt-composite navigation behaviour	Rt-Composite Address@Navigation Command Param@Navigation Command Macro@Navigation Command@ Child@Empty Child@Sibling@ Ancestor@		Get Rt-Composite Address@
Rt-components rps assignment behaviour	RPS Assignment@	<b>Set RPS Assignment</b> (Rt-Component Target Param+, ...)	Get RPS Assignment@
Rt-components perceptability behaviour	Perceptability@Presentation Priority@	<b>Set Perceptability</b> (Rt-Component Target Param+, ...) <b>Set Presentation Priority</b> (Rt-Component Target Param+, ...)	Get Perceptability Get Presentation Priority@

### Index des comportements (suite)

Comportements	Etat et Attributs	Actions élémentaires	Actions Get (extraire)
Rt-components temporal behaviour	OD@POD@OVD@PVD@ Temporal Termination@PVD Position@CTP@GTF@Timestone Status@Timestone ID@Expected OVD Result Param@Expected OVD Result Macro@Expected OVD Result@Expected PVD Result Param@Expected PVD Result Macro@Expected PVD Result@	<b>Set OVD</b> (Rt-Component Target Param+, ...) <b>Set CTP</b> (Rt-Component Target Param+, ...) <b>Set Temporal Termination</b> (Rt-Target Param+, ...) <b>Set PVD Position</b> (Socket Target Param+, ...) <b>Set GTF</b> (Rt-Component Target Param+, ...) <b>Set Timestones</b> (Rt- Component Target Param+, ...)	Get OD Get POD Get OVD Get PVD Get CTP Get Temporal Termination Get PVD Position Get GTF Get Timestone Status@
Rt-components spatial behaviour	OS@POS@Aspect Ratio@Resizing Strategy@OVS@OAP@OVS Position@PVS@OVS Proj Strategy@PAP@PVS Position@GSF@Spatial Control@User Spatial Control@Expected Axis Result Param@Spatial Control Param@Spatial Control Macro@Point Type Param@	<b>Set Aspect Ratio</b> (Rt- Component Target Param+, ...) <b>Set Resizing Strategy</b> (Rt-Composite Target Param+, ...) <b>Set OVS</b> (Rt-Component Target Param+, ...) <b>Set OAP</b> (Rt-Component Target Param+, ...) <b>Set OVS Position</b> (Rt- Component Target Param+, ...) <b>Set PAP</b> (Rt-Component Target Param+, ...) <b>Set PVS Position</b> (Rt- Component Target Param+, ...) <b>Set GSF</b> (Rt-Component Target Param+, ...) <b>Set User Spatial Control</b> (Rt-Component Target Param+, ...)	Get OS Get POS Get Aspect Ratio Get Resizing Strategy Get OVS Proj Strategy Get OVS Get OAP Get OVS Position Get PVS Get PAP Get PVS Position Get GSF Get User Spatial Control@
Rt-components audible behaviour	OV@CV@PCV@GVF@	<b>Set CV</b> (Rt-Content Target Param+, ...) <b>Set GVF</b> (GVF Target, ...)	Get OV Get CV Get PCV Get GVF@
Rt-mux stream choice behaviour	Stream Choice@Stream Chosen State@Stream Identification Param@Stream Identification Macro@Stream Identification@	<b>Set Stream Choice</b> (Rt- Mux Target Param+, ...)	Get Stream Choice Get Stream Chosen State@
Interaction behaviour	Interaction Type Param@Interaction Type Macro@Interaction Type@Interaction Status@Selection Status@Modification Status@Interaction Ability@Selectability@Modifiability @Min Interact Required@Max Interact Required@Number Of Interacted Sockets@	<b>Set Interaction Ability</b> (Rt-Component Target Param+, ...) <b>Set Interaction Status</b> (Rt-Component Target Param+, ...)	Get Interaction Ability Get Min Interact Required Get Max Interact Required Get Interaction Status Get Number Of Interacted Sockets@
Rt-components style behaviour	Style	<b>Set Style</b> (Rt-Component Target Param+, ...) <b>Get Style</b> (Rt-Component Target Param)	
Rt-contents anchor behaviour		<b>Attach Anchor</b> (Rt- Content Target Param+, ...)	
Channel availability behaviour	Channel Availability Status	<b>New Channel</b> (Channel Target Param+) <b>Delete Channel</b> (Channel Target Param+)	Get Channel Availability Status

**Index des comportements (fin)**

<b>Comportements</b>	<b>Etat et Attributs</b>	<b>Actions élémentaires</b>	<b>Actions Get (extraire)</b>
Channel perceptability behaviour	Channel Perceptability	<b>Set Channel Perceptability</b> (Channel Target Param+, ...)	Get Channel Perceptability
Channel presentation space behaviour		<b>Set CPS</b> (Channel Target Param+, ...)	
Channels and rt-components events behaviour	Event Event Data	<b>Set Event</b> (Rt-Component Channel Tg Param+, ...)	Get Event Get Event Data

## SÉRIES DES RECOMMANDATIONS UIT-T

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