



## Paramétrage des langages Méta modèle

### Sommaire

<b>LANGAGE DU META MODELE</b>	<b>4</b>
<hr/>	
<b>1. GENERALITES</b>	<b>4</b>
1.1 INTRODUCTION	4
1.2 LES COMMANDES MML	4
1.3 LES EXPRESSIONS MML	4
1.4 CONVENTIONS UTILISES	4
<b>2. OPERATEURS DU LANGAGE MML</b>	<b>5</b>
<hr/>	
2.1 INTRODUCTION	5
2.2 OPERATEURS ARITHMETIQUES	5
2.3 OPERATEURS LOGIQUES	5
2.4 OPERATEURS DE COMPARAISON	5
2.5 OPERATEURS DE CONCATENATION	5
<b>3. COMMANDES DU LANGAGE MML</b>	<b>6</b>
<hr/>	
3.1 INTRODUCTION	6
3.2 COMMANDE IF	6
3.3 COMMANDE FOREACH	6
3.4 COMMANDE FOREACH_ITEM	6
3.5 COMMANDE NODUPLICATE	7
3.6 COMMANDE NEWLINE	7
3.7 COMMANDE SETVALUE	7
3.8 COMMANDE LOG	7
3.9 COMMANDE WARNING	8
3.10 COMMANDE ERROR	8
<b>4. FONCTIONS DU LANGAGE MML</b>	<b>8</b>
<hr/>	
4.1 INTRODUCTION	8
4.2 FONCTION GETCOUNTITEM	8
4.3 FONCTION ISHEADITEM	9
4.4 FONCTION ISTAILITEM	9



4.5	FONCTION MAKELOWER	9
4.6	FONCTION MAKEUPPER	9
4.7	FONCTION MID	10
4.8	FONCTION LEFT	10
4.9	FONCTION RIGHT	10
4.10	FONCTION COMPARE	10
4.11	FONCTION COMPARENOCASE	11
<b>5.</b>	<b><u>EXPRESSIONS DU LANGAGE MML</u></b>	<b>11</b>
5.1	INTRODUCTION	11
5.2	ACCES A UN OBJET	11
5.3	ACCES A UNE CARACTERISTIQUE D'UN OBJET	11
5.4	ACCES A UNE LISTE D'OBJET	12
5.5	ACCES A UN PARAMETRE DU TEMPLATE COURANT	12
5.6	ACCES A UN AUTRE TEMPLATE	12
5.7	ACCES A UNE VARIABLE GLOBALE	12
5.8	ACCES A UNE VARIABLE LOCALE	12
5.9	ACCES A UNE VARIABLE UTILISATEUR	13
5.10	ACCES A UNE FONCTIONS	13
<b>6.</b>	<b><u>CONSTANTES DU LANGAGE MML</u></b>	<b>13</b>
6.1	INTRODUCTION	13
6.2	CONSTANTES	13
<b>7.</b>	<b><u>GRAMMAIRE DU LANGAGE MML</u></b>	<b>13</b>
	<b><u>MOTS CLES ASSOCIES AUX OBJETS MODELISES</u></b>	<b>16</b>
<b>8.</b>	<b><u>DIAGRAMME DE CLASSE</u></b>	<b>16</b>
8.1	OBJETS PRINCIPAUX	16
8.1.1.1	Package	16
8.1.2	CLASSE	17
8.1.3	INTERFACE	18
8.1.4	ATTRIBUT D'UNE CLASSE	18
8.1.5	METHODE D'UNE CLASSE	19
8.1.6	PARAMETRE D'UNE METHODE	20
8.1.7	ASSOCIATION	21
8.1.8	DEPENDANCE	22
8.1.9	COMMENTAIRE	23
8.1.10	CONTRAINTE (MODELE OBJET)	23
8.2	LIENS	24
8.2.1.1	Généralisation	24
8.2.2	LIEN AVEC UN COMMENTAIRE	25
8.2.3	LIEN AVEC UNE CONTRAINTE	25
8.2.4	LIEN AVEC UNE DEPENDANCE	26
8.2.5	LIEN AVEC UNE ASSOCIATION	27
8.2.6	LIEN D'INCLUSION DANS UNE CLASSE	27
8.2.7	LIEN ENTRE UNE ASSOCIATION ET UNE CLASSE ASSOCIATION	28





# Langage du méta modèle

---

## 1. Généralités

---

### 1.1 Introduction

Le langage qui permet de parcourir le méta modèle est appelé MML (Meta Model Language).

Le contenu d'un template est un mélange des trois éléments suivant :

- du texte,
- des commandes MML (if, foreach, etc.)
- des expressions MML à évaluer (§c:nom\$, §f:MakeLower(§c:Nom\$)\$, etc.)

---

### 1.2 Les commandes MML

Les lignes de commandes commencent toujours par le caractère `:`.

Si vous désirez commencer une ligne par les deux points sans qu'il soit interprété comme une commande, il suffit de le doubler.

exemple :

<code>:newline</code>	ligne de commande pour forcer un retour à la ligne
<code>::bonjour</code>	interprète la ligne comme du texte :bonjour

---

### 1.3 Les expressions MML

Les expressions MML sont des expressions qui sont évaluées aussi bien dans un texte que dans une commande.

Elles sont facilement identifiable par les marques `§`.

exemple :

<code>§c:nom\$</code>	évaluation de la caractéristique <b>nom</b> du méta modèle de l'objet courant.
-----------------------	--

<code>§t:montemplate\$</code>	évaluation du template <b>montemplate</b> .
-------------------------------	---

<code>§p:p1\$</code>	évaluation du premier paramètre du template courant.
----------------------	--

---

### 1.4 Conventions utilisés

Voici les conventions utilisées pour décrire la syntaxe des éléments.

<b>GRAS</b>	mot-clé du langage
<i>italique</i>	représente un élément du langage
noir	élément que vous devez saisir
[ ]	indique un élément optionnel
{ }	indique un élément optionnel qui se répète

---

## 2. Opérateurs du langage MML

---

### 2.1 Introduction

Les opérateurs sont utilisés dans les expressions et les conditions.  
Le langage MML comprend :

- des opérateurs arithmétiques
- des opérateurs logiques
- des opérateurs de comparaison
- des opérateurs de concaténation

---

### 2.2 Opérateurs arithmétiques

+	addition
-	soustraction
*	multiplication
/	division entière

---

### 2.3 Opérateurs logiques

<b>NOT</b>	négation
<b>AND</b>	et logique
<b>OR</b>	ou logique

---

### 2.4 Opérateurs de comparaison

=	égal
==	égal
≠	différent
≠	différent
<	inférieur
>	supérieur
≤	inférieur ou égal
≥	supérieur ou égal
≠	différent
≠	différent

---

### 2.5 Opérateurs de concaténation

+	concaténe deux chaînes de caractères
---	--------------------------------------

---

## 3. Commandes du langage MML

---

### 3.1 Introduction

Toute ligne de commande commence par le caractère :

---

### 3.2 Commande IF

La commande **IF** permet d'exécuter le template de manière conditionnel.

Syntaxe : **If** *condition* bloc { **Elseif** *condition* bloc } [ **Else** bloc ] **Endif**

Exemple :

```
:If (§c:visibilite§ = "#")
protected
:ElseIf (§c:visibilite§ = "-")
private
:Else
public
:Endif
```

Suivant la valeur de la caractéristique **visibilité**, le template retourne une valeur différente.

---

### 3.3 Commande FOREACH

La commande **ForEach** permet de parcourir les listes d'objets accessibles à l'objet courant de la pile.

Syntaxe : **ForEach** \$v:nom\_variable§ in \$l:nom\_liste§ [ **Reverse** ] bloc **Next**

Exemple :

```
:foreach $v:patte§ in $l:patte_association§
$t:Association:Declarations_h($v:patte§)§
:next
```

Pour chaque patte association, on évalue le template **Declaration** de l'association en passant en paramètre la patte courante.

---

### 3.4 Commande FOREACH\_ITEM

La commande **ForEach\_Item** permet de parcourir les listes d'objets accessibles à l'objet courant de la pile.

Cette commande fait la même chose que la commande **ForEach** sans passer par une variable.

Syntaxe : **ForEach\_item** in \$l:nom\_liste§ [ **Reverse** ] bloc **Next**

Exemple :

```
:ForEach_item in $l:generalisation§
:If (§o:parents§!=§o:outers§)
$c:parent.nom§
:Endif
:Next
```



Pour chaque généralisation, si l'objet parent est différent de l'objet dans la pile, le template retourne le nom du parent.

---

### 3.5 Commande NODUPLICATE

La commande **NoDuplicate** permet de s'assurer qu'il n'y aura pas deux lignes identiques au sein du bloc.

Syntaxe : **NoDuplicate** bloc **EndNoDuplicate**

Exemple :

```
:NoDuplicate
$t:include_h$
:EndNoDuplicate
```

Pour chaque généralisation, si l'objet parent est différent de l'objet dans la pile, le template retourne le nom du parent.

---

### 3.6 Commande NEWLINE

La commande **NewLine** permet de forcer l'écriture d'un retour à la ligne dans le résultat du template.

Syntaxe : **NewLine**

Exemple :

```
:NewLine
:If ($c:visibilite$ = "#") protected endif
```

Force un retour à la ligne avant une ligne de commande.

---

### 3.7 Commande SETVALUE

La commande **SetValue** permet de créer et d'initialiser une variable locale au template courant.

Syntaxe : **SetValue**( nom\_variable, *expression* )

Exemple :

```
:SetValue(filename,
$t:Owner_Package:GetDirectory$+$c:nom$+".h")
$v:filename$
```

Initialise la variable filename avec la concaténation de plusieurs chaînes de caractères.

Le template retourne ensuite cette valeur.

---

### 3.8 Commande LOG

La commande **Log** permet d'écrire avec la couleur noir dans la fenêtre résultat de Win'Design.



Syntaxe : **Log** *expression*

Exemple :

```
:log " génération objet : " + §c:nom§
```

Ecrit dans la fenêtre résultat "génération objet : " concaténé avec le nom de l'objet courant.

---

### 3.9 Commande WARNING

La commande **Warning** permet d'écrire avec la couleur bleu dans la fenêtre résultat de Win'Design.

Syntaxe : **Warning** *expression*

Exemple :

```
:warning "attention objet incorrect"
```

Ecrit dans la fenêtre résultat la chaîne "attention objet incorrect".

---

### 3.10 Commande ERROR

La commande **Error** permet d'écrire avec la couleur rouge dans la fenêtre résultat de Win'Design.

Syntaxe : **Error** *expression*

Exemple :

```
:error "erreur fatale"
```

Ecrit dans la fenêtre résultat la chaîne "erreur fatale".

---

## 4. Fonctions du langage MML

---

### 4.1 Introduction

Une fonction peut-être appelé aussi bien dans une expression d'une ligne de commande que dans un bloc de texte libre.

Toute fonction commence par **§f:** et se termine par **§**.

Syntaxe : **§f:** nom\_fonction ( [ *paramètre* { , *paramètre* } ] )**§**

Exemple :

```
§f:MakeLower ( §c:nom§ ) §
```

---

### 4.2 Fonction GetCountItem

La fonction **GetCountItem** retourne le nombre d'objets dans la liste courante lors d'un parcours

avec la commande **ForEach** ou dans la liste passée en paramètre.

Syntaxe : **GetCountItem**( [ \$!:nom\_liste\_objet\$ ] )

Exemple :

```
$f:GetCountItem($!:patte_association$) $  
  
:ForEach_item in $!:patte_association$  
$f:GetCountItem() $  
:Next
```

---

### 4.3 Fonction IsHeadItem

La fonction **IsHeadItem** retourne vrai si l'élément courant est en tête de la liste courante lors d'un parcours avec la commande **ForEach** ou de la liste passée en paramètre.

Syntaxe : **IsHeadItem**( [ \$!:nom\_liste\_objet\$ ] )

Exemple :

```
:ForEach_item in $!:patte_association$  
:if $f:IsHeadItem() $  
premier  
:endif  
:Next
```

---

### 4.4 Fonction IsTailItem

La fonction **IsTailItem** retourne vrai si l'élément courant est en fin de la liste courante lors d'un parcours avec la commande **ForEach** ou de la liste passée en paramètre.

Syntaxe : **IsTailItem**( [ \$!:nom\_liste\_objet\$ ] )

Exemple :

```
:ForEach_item in $!:patte_association$  
:if $f:IsTailItem() $  
dernier  
:endif  
:Next
```

---

### 4.5 Fonction MakeLower

La fonction **MakeLower** retourne le résultat de l'expression avec les caractères en minuscule.

Syntaxe : **MakeLower**( *expression* )

Exemple :

```
$f:MakeLower($c:nom$) $  
$f:MakeLower("Bonjour") $  
retourne bonjour
```

---

### 4.6 Fonction MakeUpper

La fonction **MakeUpper** retourne le résultat de l'expression avec les caractères en majuscule.



Syntaxe : **MakeUpper**( *expression* )

Exemple :

```
$f:MakeUpper( $c:nom$ )$  
$f:MakeUpper( "Bonjour" )$
```

 retourne **BONJOUR**

---

## 4.7 Fonction Mid

La fonction **Mid** extrait une chaîne de caractère de *nbCar* à partir de la position *début*.  
Si *nbCar* est omis, extrait le nombre de caractères restant.

Syntaxe : **Mid**( *expression*, début [ , *nbCar* ] )

Exemple :

```
$f:Mid( "abcdef", 2, 3 )$  
$f:Mid( "abcdef", 2 )$
```

 retourne **cde**  
retourne **cdef**

---

## 4.8 Fonction Left

La fonction **Left** extrait une chaîne de caractère de *nbCar* en partant de la gauche.

Syntaxe : **Left**( *expression*, *nbCar* )

Exemple :

```
$f:Left( "abcdef", 2 )$
```

 retourne **ab**

---

## 4.9 Fonction Right

La fonction **Right** extrait une chaîne de caractère de *nbCar* en partant de la droite.

Syntaxe : **Right**( *expression*, *nbCar* )

Exemple :

```
$f:Right( "abcdef", 2 )$
```

 retourne **ef**

---

## 4.10 Fonction Compare

La fonction **Compare** compare deux chaînes caractères sans ignorer la casse.

Retourne 0 si les chaînes sont identiques  
< 0 si *expression1* < *expression2*  
> 0 si *expression1* > *expression2*

Syntaxe : **Compare**( *expression1*, *expression2* )

Exemple :

```
$f:Compare( "abc", "abc" )$  
$f:Compare( "ABC", "abc" )$
```

 retourne **0**  
retourne **<0**

---

## 4.11 Fonction CompareNoCase

La fonction **CompareNoCase** compare deux chaînes caractères en ignorant la casse.

Retourne    0            si les chaînes sont identiques  
             < 0          si expression1 < expression2  
             > 0          si expression1 > expression2

Syntaxe :    **CompareNoCase**( *expression1*, *expression2* )

Exemple :

```
$f:CompareNoCase( "abc", "abc" )$            retourne 0  
$f:CompareNoCase( "ABC", "abc" )$            retourne 0
```

---

## 5. Expressions du langage MML

---

### 5.1 Introduction

Une expression spécifique MML peut-être appelée aussi bien dans une expression d'une ligne de commande que dans un bloc de texte libre.

Toute expression spécifique au langage commence par **\$:** et se termine par **\$**.

Syntaxe :    **\$type\_expression:** nom\_expression [ ( [ *expression* { , *expression* } ] ) ] **\$**

---

### 5.2 Accès à un objet

Le mot-clé **outer** permet d'accéder à l'objet en dehors du bloc des commandes **ForEach** et **ForEach\_Item**.

Syntaxe :    **\$o:{outer.}**{nom\_objet.}nom\_objet**\$**

Exemple :

```
$o:outer.outer.classe$  
$o:association$
```

---

### 5.3 Accès à une caractéristique d'un objet

Le mot-clé **outer** permet d'accéder à l'objet en dehors du bloc des commandes **ForEach** et **ForEach\_Item**.

Syntaxe :    **\$c:{outer.}**{nom\_objet.}caractéristique**\$**

Exemple :

```
$c:nom$  
$c:outer.nom$  
$c:outer.outer.nom$  
$c:association.nom$
```

---

## 5.4 Accès à une liste d'objet

Le mot-clé **outer** permet d'accéder à l'objet en dehors du bloc des commandes **ForEach** et **ForEach\_Item**.

Syntaxe : **\$l:{outer.}{nom\_objet.}nom\_liste\$**

Exemple :

```
$l:attribut_objt$  
$o:outer:patte_association$
```

---

## 5.5 Accès à un paramètre du template courant

Les paramètres du template sont passés lors de l'appel des templates.

Syntaxe : **\$p:pindex\$**

Exemple :

```
$p:p1$  
$p:p2$
```

---

## 5.6 Accès à un autre template

Le mot-clé **outer** permet d'accéder à l'objet en dehors du bloc des commandes **ForEach** et **ForEach\_Item**.

Syntaxe : **\$t:{outer.}{nom\_objet.} : ] nom\_template [ ( paramètre1 { , paramètreN } ) ] \$**

Exemple :

```
$t:include_h$  
$t:Attribut_h("#")$  
$t:Association:Attribut_h("#")$
```

---

## 5.7 Accès à une variable globale

Les variables globales sont définies pour tous les templates et initialisées au début de la génération.

Syntaxe : **\$g:nom\_variable\$**

Exemple :

```
$g:ordre_generation$
```

---

## 5.8 Accès à une variable locale

Les variables locales sont définies pour la commande **SetValue** et sont accessibles à l'intérieur du template courant.

Syntaxe : `$v:nom_variable$`

Exemple :

```
$v:maVar1$
```

---

## 5.9 Accès à une variable utilisateur

Les variables utilisateurs sont définies pour chaque type d'objet.

Syntaxe : `$u:nom_variable$`

Exemple :

```
$u:package_directory$
```

---

## 5.10 Accès à une fonctions

Les fonctions sont expliquées en détails dans la rubrique **Fonctions du langage MML**.

Syntaxe : `$f: nom_fonction ( [ paramètre { , paramètre } ] )$`

Exemple :

```
$f:MakeLower($c:nom$)$
```

---

# 6. Constantes du langage MML

---

## 6.1 Introduction

Les constantes sont utilisés dans les expressions et les conditions.

---

## 6.2 Constantes

<b>NULL</b>	valeur nulle pour le test d'un objet
<b>TRUE</b>	valeur vrai
<b>FALSE</b>	valeur faux

---

# 7. Grammaire du langage MML

D	[0-9]
L	[a-zA-Z_]
<i>nombre</i>	{D}+
<i>identifiant</i>	{D}+[.]{D}+
<i>texte</i>	{L}({L} {D})*
	tous caractères



MML_Language	: MML_ListElement   ;
MML_ListElement	: MML_Element   MML_ListElement MML_Element ;
MML_Element	: <i>texte</i>   MML_Expression   MML_Command ;
MML_Expression	: <b>\$o:</b> complex_id <b>\$</b>   <b>\$c:</b> complex_id <b>\$</b>   <b>\$t:</b> id_template <b>\$</b>   <b>\$t:</b> id_template ( lst_expr ) <b>\$</b>   <b>\$g:</b> id_simple <b>\$</b>   <b>\$u:</b> id_simple <b>\$</b>   <b>\$p:</b> id_simple <b>\$</b>   <b>\$f:</b> MML_Function <b>\$</b>   MML_ObjectList   MML_TemplateVar ;
MML_ObjectList	: <b>\$l:</b> complex_id <b>\$</b> ;
MML_TemplateVar	: <b>\$v:</b> id_simple <b>\$</b> ;
id_template	: id_simple   complex_id : id_simple ;
complex_id	: id_simple   complex_id . id_simple ;
id_simple	: <i>identifiant</i> ;
MML_Command	: <b>:If</b> MML_Command_IF   <b>:ForEach</b> Command_ForEach   <b>:ForEach_item</b> Command_ForEachItem   <b>:SetValue</b> ( id_simple , expr )   <b>:NewLine</b>   <b>:NoDuplicate</b> MML_ListElement <b>:EndNoDuplicate</b>   <b>:Log</b> expr   <b>:Warning</b> expr   <b>:Error</b> expr ;
Command_ForEach <b>Next</b>	: MML_TemplateVar <b>in</b> MML_ObjectList MML_ListElement <b>Next</b>   MML_TemplateVar <b>in</b> MML_ObjectList <b>Reverse</b>
MML_ListElement <b>Next</b>	;
Command_ForEachItem	: <b>in</b> MML_ObjectList MML_ListElement <b>Next</b>   <b>in</b> MML_ObjectList <b>Reverse</b> MML_ListElement <b>Next</b> ;



```
MML_Command_IF      : condition MML_ListElement Endif
                    | condition MML_ListElement Else MML_ListElement Endif
                    | condition MML_ListElement Elseif MML_Command_IF
                    ;

condition            : expr
                    | not condition
                    | condition and condition
                    | condition or condition
                    | ( condition )
                    ;

lst_expr             : expr
                    | lst_expr , expr
                    ;

expr                 : nombre
                    | MML_Expression
                    | MML_Function
                    | " texte "
                    | NULL
                    | TRUE
                    | FALSE
                    | - expr
                    | expr + expr
                    | expr - expr
                    | expr * expr
                    | expr / expr
                    | expr > expr
                    | expr < expr
                    | expr >= expr
                    | expr <= expr
                    | expr = expr          | expr == expr
                    | expr != expr        | expr <> expr
                    ;

MML_Function         : id_simple ( )
                    | id_simple ( lst_expr )
                    ;
```



## Mots clés associés aux objets modélisés

Ces mots clés sont utilisés pour la configuration des langages et pour le maquetage des états du documenteur RTF.

---

### 8. Diagramme de classe

---

#### 8.1 Objets principaux

---

##### 8.1.1.1 Package

Mot clé associé : **PACKAGE**

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
MAQUETTE	DATATYPE_BOOL
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL

Mots-clés des objets accessibles directement :

OWNER\_PACKAGE

Parcours des objets inclus :

Parcours des liens :

PATTE\_COMMENTAIRE

---

## 8.1.2 Classe

**Mot clé associé :** **CLASSE**

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
ABONNE	DATATYPE_BOOL
REFERENT	DATATYPE_BOOL
MAQUETTE	DATATYPE_BOOL
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL
TYPE_BASE_CLASSE	DATATYPE_STRING
ABSTRACT	DATATYPE_BOOL
CLASSE_ASSOCIATION	DATATYPE_STRING
PERSISTANT	DATATYPE_BOOL
TERMINAL	DATATYPE_BOOL
RACINE	DATATYPE_BOOL

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE

**Parcours des objets inclus :**

METHODE  
ATTRIBUT\_OBJ

**Parcours des liens :**

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ  
PATTE\_CLASSE\_ASSOCIATION  
PATTE\_ASSOCIATION  
GENERALISATION  
PATTE\_DEPENDANCE  
PATTE\_NESTED  
PATTE\_REALISE

### 8.1.3 Interface

Mot clé associé : [INTERFACE](#)

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL
TYPE_BASE_CLASSE	DATATYPE_STRING
ABSTRACT	DATATYPE_BOOL

Mots-clés des objets accessibles directement :

OWNER\_PACKAGE

Parcours des objets inclus :

METHODE  
ATTRIBUT\_OBJ

Parcours des liens :

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ  
PATTE\_CLASSE\_ASSOCIATION  
PATTE\_ASSOCIATION  
GENERALISATION  
PATTE\_DEPENDANCE  
PATTE\_NESTED  
PATTE\_REALISE

---

### 8.1.4 Attribut d'une classe

Mot clé associé : [ATTRIBUT\\_OBJ](#)

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD

TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL
IDENTIFIANT	DATATYPE_BOOL
OBLIGATOIRE	DATATYPE_BOOL
LONGUEUR	DATATYPE_WORD
PRECISION	DATATYPE_WORD
TYPE	DATATYPE_STRING
TYPE_LANGAGE	DATATYPE_STRING
USER_TYPE	DATATYPE_STRING
FORMAT	DATATYPE_STRING
STABLE	DATATYPE_BOOL
CALCULEE	DATATYPE_BOOL
VALEUR_MINI	DATATYPE_STRING
VALEUR_MAXI	DATATYPE_STRING
VALEUR_DEFAULT	DATATYPE_STRING
CASE_SENSITIVE	DATATYPE_BOOL
UPPERCASE	DATATYPE_BOOL
LOWERCASE	DATATYPE_BOOL
CHECK_VALUE	DATATYPE_STRING
VALEUR_ASSOCIEE	DATATYPE_STRING
REPETITION	DATATYPE_STRING
MULTIPLICITE	DATATYPE_STRING
STATIQUE	DATATYPE_BOOL
PERSISTANT	DATATYPE_BOOL
DERIVE	DATATYPE_BOOL

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
OWNER

**Parcours des objets inclus :**

**Parcours des liens :**

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ  
ATTRIBUT\_OBJ

---

### 8.1.5 Méthode d'une classe



**Mot clé associé :** [METHODE](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE

**Parcours des objets inclus :**

PARAMETER

**Parcours des liens :**

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ

---

### 8.1.6 Paramètre d'une méthode

**Mot clé associé :** [PARAMETER](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
CONTRAINTE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE

DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL
LONGUEUR	DATATYPE_WORD
PRECISION	DATATYPE_WORD
TYPE	DATATYPE_STRING
TYPE_LANGAGE	DATATYPE_STRING
USER_TYPE	DATATYPE_STRING
FORMAT	DATATYPE_STRING
STABLE	DATATYPE_BOOL
CALCULEE	DATATYPE_BOOL
VALEUR_MINI	DATATYPE_STRING
VALEUR_MAXI	DATATYPE_STRING
VALEUR_DEFAULT	DATATYPE_STRING
CASE_SENSITIVE	DATATYPE_BOOL
UPPERCASE	DATATYPE_BOOL
LOWERCASE	DATATYPE_BOOL
VALEUR_ASSOCIEE	DATATYPE_STRING
REPETITION	DATATYPE_STRING
TYPE_PARAM	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
OWNER  
METHODE

**Parcours des objets inclus :**

**Parcours des liens :**

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ  
PARAMETER

## 8.1.7 Association

**Mot clé associé :** [ASSOCIATION](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE



DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL
CLASSE_ASSOCIATION	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE

**Parcours des objets inclus :**

**Parcours des liens :**

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ  
PATTE\_CLASSE\_ASSOCIATION  
PATTE\_ASSOCIATION

---

## 8.1.8 Dépendance

**Mot clé associé :** [DEPENDANCE](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
ABONNE	DATATYPE_BOOL
REFERENT	DATATYPE_BOOL
MAQUETTE	DATATYPE_BOOL
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING
ROLE	DATATYPE_STRING
DEPENDANT	DATATYPE_BOOL
CARD_MIN	DATATYPE_STRING
CARD_MAX	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2  
DEPENDANCE

**Parcours des objets inclus :**

Parcours des liens :

---

### 8.1.9 Commentaire

Mot clé associé : [COMMENTAIRE](#)

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL

Mots-clés des objets accessibles directement :

OWNER\_PACKAGE

Parcours des objets inclus :

Parcours des liens :

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ

---

### 8.1.10 Contrainte (modèle objet)

Mot clé associé : [CONTRAINTE\\_OBJ](#)

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING



MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
LIBELLE	DATATYPE_STRING
ALIAS	DATATYPE_STRING
AUTEUR	DATATYPE_STRING
DATE_CREATION	DATATYPE_DATE
DATE_MODIFICATION	DATATYPE_DATE
DATE_VALIDATION	DATATYPE_DATE
DOMAINE	DATATYPE_STRING
CAR_LIBRE_1	DATATYPE_STRING
CAR_LIBRE_2	DATATYPE_STRING
CAR_LIBRE_3	DATATYPE_STRING
ACTUEL	DATATYPE_BOOL
FUTUR	DATATYPE_BOOL
VALIDE	DATATYPE_BOOL

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE

**Parcours des objets inclus :**

**Parcours des liens :**

PATTE\_COMMENTAIRE  
PATTE\_CONTRAINTE\_OBJ

---

## 8.2 Liens

---

### 8.2.1.1 Généralisation

**Mot clé associé :** [GENERALISATION](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
ABONNE	DATATYPE_BOOL
REFERENT	DATATYPE_BOOL
MAQUETTE	DATATYPE_BOOL
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING
DESCRIMINANT	DATATYPE_STRING
ALIAS	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE

LINKED\_OBJECT  
OBJECT1  
OBJECT2  
CHILD  
PARENT

**Parcours des objets inclus :**

**Parcours des liens :**

GENERALISATION

---

## 8.2.2 Lien avec un commentaire

**Mot clé associé :** [PATTE\\_COMMENTAIRE](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2  
COMMENTAIRE

**Parcours des objets inclus :**

**Parcours des liens :**

---

## 8.2.3 Lien avec une contrainte

**Mot clé associé :** [PATTE\\_CONTRAINTE\\_OBJ](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD



VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2  
COMMENTAIRE  
CONTRAINTE\_OBJ

**Parcours des objets inclus :**

**Parcours des liens :**

---

## 8.2.4 Lien avec une dépendance

**Mot clé associé :** [PATTE\\_DEPENDANCE](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING
ROLE	DATATYPE_STRING
DEPENDANT	DATATYPE_BOOL
CARD_MIN	DATATYPE_STRING
CARD_MAX	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2  
DEPENDANCE

**Parcours des objets inclus :**

**Parcours des liens :**

---

## 8.2.5 Lien avec une association

Mot clé associé : **PATTE\_ASSOCIATION**

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING
ROLE	DATATYPE_STRING
NAVIGABLE	DATATYPE_BOOL
ORDONNE	DATATYPE_STRING
AGREGATION	DATATYPE_STRING
CARD_MIN	DATATYPE_STRING
CARD_MAX	DATATYPE_STRING

Mots-clés des objets accessibles directement :

OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2

Parcours des objets inclus :

Parcours des liens :

---

## 8.2.6 Lien d'inclusion dans une classe

Mot clé associé : **PATTE\_NESTED**

Liste des caractéristiques :

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING

Mots-clés des objets accessibles directement :



OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2  
NESTED

**Parcours des objets inclus :**

**Parcours des liens :**

---

## 8.2.7 Lien entre une association et une classe association

**Mot clé associé :** [PATTE\\_CLASSE\\_ASSOCIATION](#)

**Liste des caractéristiques :**

NOM	DATATYPE_STRING
NUMERO_OBJET	DATATYPE_WORD
TYPE_OBJET	DATATYPE_STRING
CODE_INTERNE	DATATYPE_WORD
PROFONDEUR	DATATYPE_WORD
ANNOTATION	DATATYPE_NOTEPAD
TEXTE_LIBRE	DATATYPE_NOTEPAD
SOUS_MODELES	DATATYPE_NOTEPAD
VISIBILITE	DATATYPE_STRING
MODIFIABLE	DATATYPE_STRING
CONTRAINTE	DATATYPE_STRING
NOM_STEREOTYPE	DATATYPE_STRING
TEXTE	DATATYPE_STRING
CIBLE	DATATYPE_STRING

**Mots-clés des objets accessibles directement :**

OWNER\_PACKAGE  
LINKED\_OBJECT  
OBJECT1  
OBJECT2  
ASSOCIATION  
CLASSE\_ASSOCIATION

**Parcours des objets inclus :**

**Parcours des liens :**