



Type : SPEC	Projet RNTL MORSE	Réf : SPEC-LfP- MM_changes
Date : 08/07/2005		Sous-projet n°: 4
Version : 1.0		Tâche n°: 4.3
Auteurs : Antony Hubervic		
Relecteurs : Marc Richard-Foy		

Modification du méta-modèle LfP



Liste des révisions

	Paragraphe	Commentaire
<i>Version</i> : 1.0 <i>Date</i> : 08/07/05 <i>Auteur</i> : Antony Hubervic		Version initiale



Table des matières

1	INTRODUCTION	5
2	REMARQUES GENERALES	5
3	DIAGRAMME D'ARCHITECTURE	5
4	TYPES DE DONNEES	5
5	EXPRESSIONS	5
6	DIAGRAMME DE COMPORTEMENT	8
6.1	STRUCTURE	8
6.2	INSTRUCTIONS	9



1 Introduction

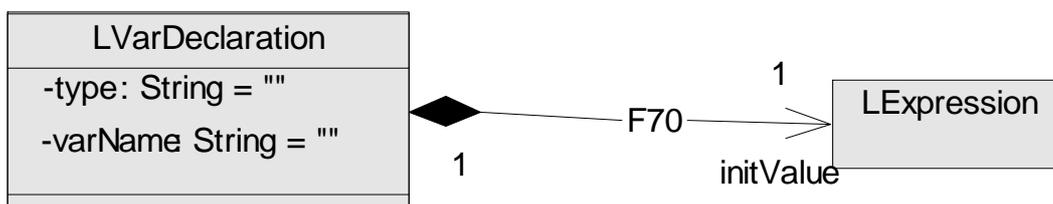
Ce document décrit les modifications apportées au méta-modèle LfP pour le rendre cohérent avec la grammaire textuelle. Ces modifications ont été nécessaires pour l'intégration de l'analyseur syntaxique LfP au générateur de code ACD.

Les chapitres suivants présentent les modifications en respectant la structure du méta-modèle. La documentation des classes est issue du modèle UML. Dans celui-ci, les noms des classes LfP sont préfixées par la lettre « L » pour éviter des collisions avec les méta-classes existantes.

2 Remarques générales

Les attributs de type complexe ont été remplacés par des associations vers ce type complexe (e.g. attribut **initValue** de type **Expression** de la classe **LVarDeclaration** remplacée par une association vers la classe **Expression**).

Les attributs de type complexe ne contenant qu'un attribut de type String ont été remplacés par des attributs de type String (e.g. attribut **varName** de type **Identifieur** de la classe **LVarDeclaration** remplacé par un attribut de type **String**).



3 Diagramme d'architecture

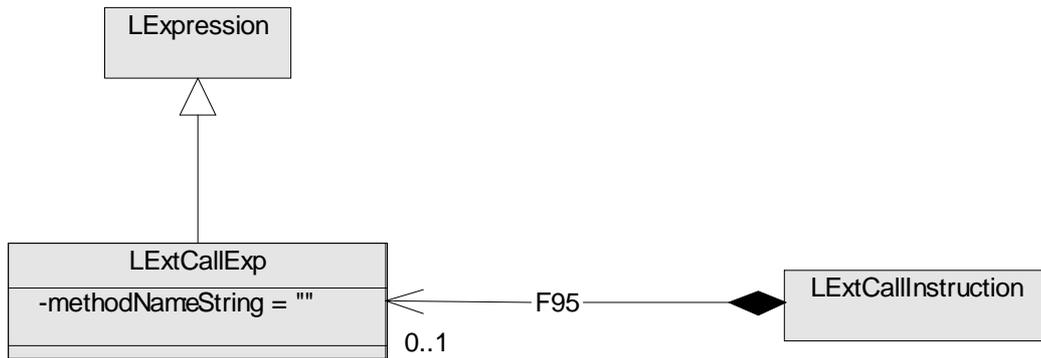
Pas de changements autres que ceux présentés dans les remarques générales.

4 Types de données

Pas de changements autres que ceux présentés dans les remarques générales.

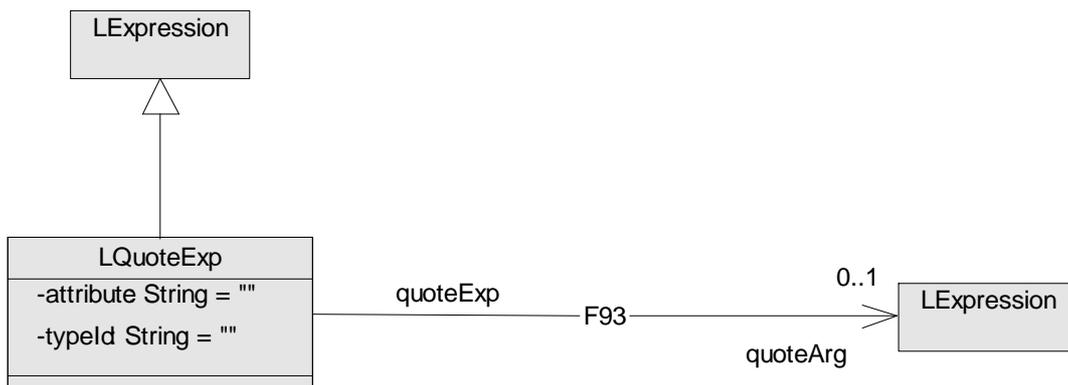
5 Expressions

Ajout de la classe ExtCallExp pour traiter les appels de méthode sur des types opaques (règle 4 de la grammaire LfP).

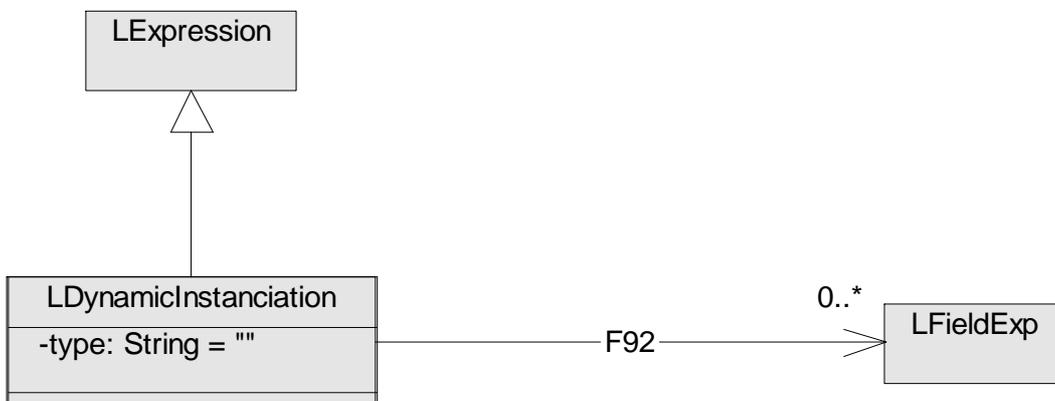


Ajout de la classe QuoteExp pour traiter les expressions utilisant l'opérateur « ' » (règle 3 de la grammaire LfP).

La règle 3 de la grammaire a été changée en :
 IDENTIFIER ' IDENTIFIER (expression)

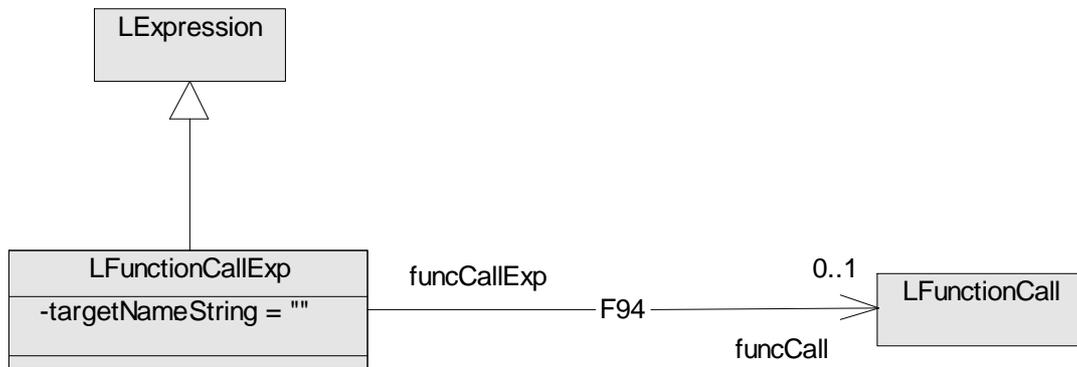


Ajout de la classe DynamicInstanciation pour traiter les expressions utilisant l'opérateur new.

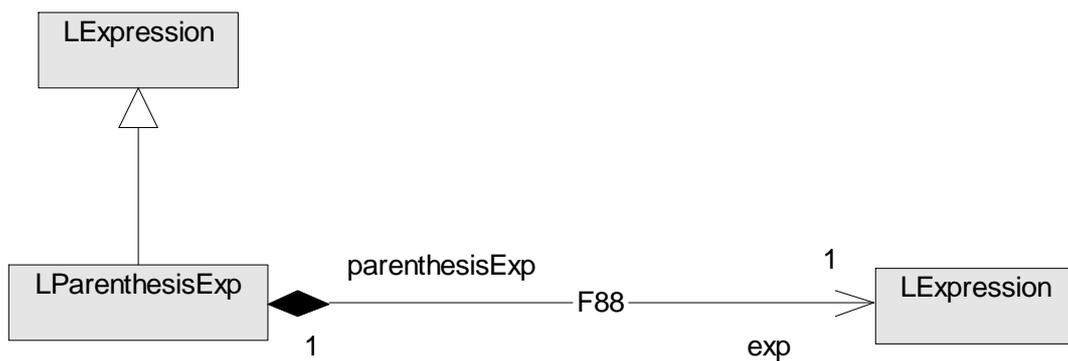




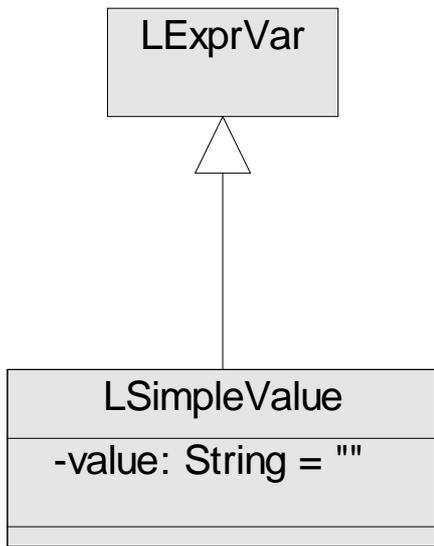
Ajout de la classe FunctionCallExp pour traiter les expressions d'appel de fonction (différents des instructions d'appel de fonction, règle 7 de la grammaire LfP).



Ajout de la classe ParenthesisExp pour traiter les expressions parenthésées (règle 2 de la grammaire LfP).



Ajout de la classe SimpleValue pour traiter les valeurs littérales.

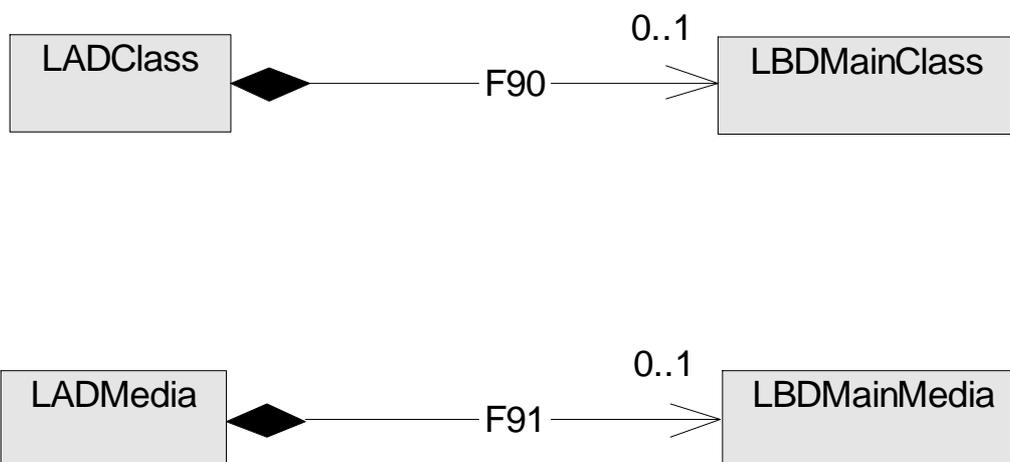


6 Diagramme de comportement

6.1 Structure

Ajout d'une association entre les méta-classes ADClass et BDMainClass permettant d'associer une classe à son diagramme de comportement.

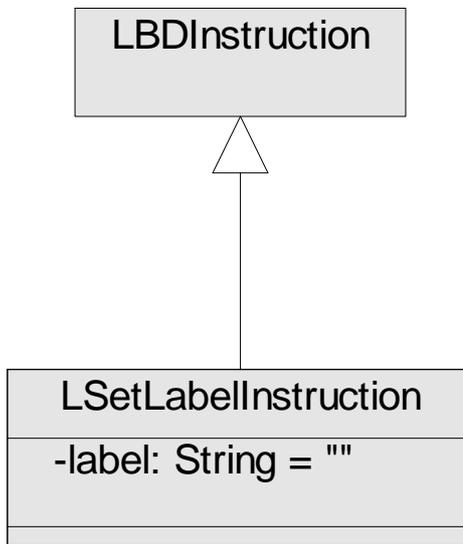
Ajout d'une association entre les méta-classes ADMedia et BDMainMedia permettant d'associer un média à son diagramme de comportement.



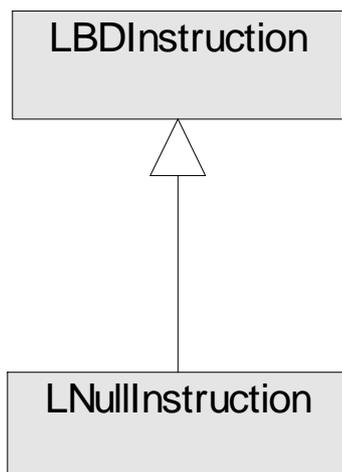


6.2 Instructions

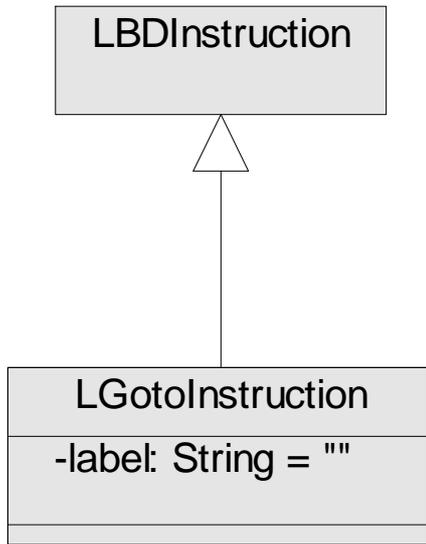
Ajout de la classe SetLabelInstruction pour traiter l'instruction de pose d'étiquette (règle 55 de la grammaire LfP).



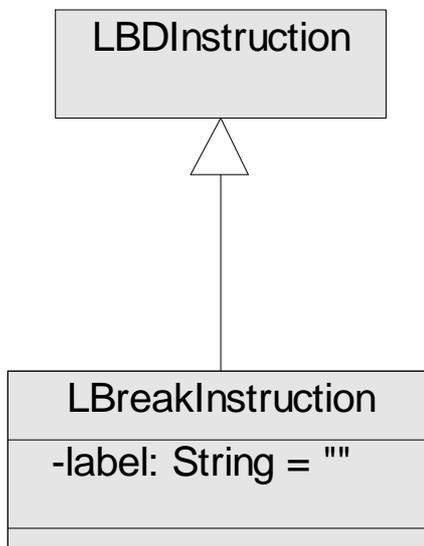
Ajout de la classe NullInstruction pour traiter l'instruction vide.



Ajout de la classe GotoIntruction pour traiter l'instruction de saut (règle 54 de la grammaire LfP).



Ajout de la class BreakInstruction pour traiter l’instruction de sortie de boucle (règle 52 de la grammaire LfP).



Ajout de la classe ExtCallInstruction pour traiter l’instruction d’appel de méthode sur un type opaque (règle 4 de la grammaire LfP).



Ajout de la classe MethodActivation pour traiter l’instruction d’attente d’activation de méthode (règles 58 et 59 de la grammaire LfP)

