

MASS Preprocessor

Introduction

Given a MASS program that contains an Agents or Places constructor, the preprocessor will modify the appropriate method calls to use predefined integer values instead of reflection. It will append a `callMethod` method to the end of the class. If no Agents or Places are recognized, the program will be output unaltered.

The following constructor arguments are recognized:

```
Places(int handle, String classname, Object argument, int... size)
```

```
Places(int handle, String primitiveType, String classname, Object  
argument, int...size)
```

The following methods are recognized and modified when called from a Places or Agents variable with an appropriate String parameter:

```
callAll()
```

```
callSome()
```

```
exchangeAll()
```

```
exchangeSome()
```

Executing the MASS preprocessor

From a Linux or Windows command prompt:

```
java MassPrePro <inputFile >outputFile
```

where `inputFile` is the MASS program coded using reflection and `outputFile` is the translated MASS program.

Building

Some tools are required to build the MASS preprocessor. JavaCC 5.0 and JJTree are both available from the JavaCC website:

<http://javacc.java.net>

Building the executable from the source can be a multi-step process depending on the changes that were made. Most files in the project have been auto-generated by JJTree or JavaCC. Do not attempt to change the auto-generated files directly. See below for a complete list of project files.

If changes have been made to Java1.1.jjt, the following steps must be taken to update the project:

1. Use "jjtree Java1.1.jjt" in Linux or "jjtree.bat Java1.1.jjt" in Windows to generate Java1.1.jj
2. Use "javacc Java1.1.jj" in Linux or "javacc.bat Java1.1.jj" in Windows to generate *.java files.
3. If new rules were added to Java1.1.jjt, the correspondingly methods must be added to UnparseVisitor.java. Rules that have been removed from Java1.1.jjt should be removed from UnparseVisitor.java.
4. Once the methods have been added to UnparseVisitor.java, they can be overridden in OptimizingVisitor.java if desired.
5. Remove existing *.class files and rebuild. (In Eclipse, select Project:Clean to rebuild the class files.)

If changes were only made to OptimizingVisitor, ScopeManager, MethodVar, MassVar, or MassPrePro, then only the classes that have been modified need to be rebuilt.

How it works

The preprocessor performs its optimizations by running the input code through a Java parser. The parser emits tokens in response to the input code. Actions act on specific tokens to check conditions, set flags, modify output, etc.

A grammar (Java1.1.jjt) defines a roughly correct version of Java. The grammar is then modified to create Abstract Syntax Trees. A parser is then generated (UnparseVisitor) which by default will output any input which matches the Java language as defined by the grammar.

The parser methods can be overridden with OptimizingVisitor to perform MASS optimizations. For example, when parsing a MethodDeclaration token, a flag will be set to indicate that a new method is being parsed and that a new scope must be placed on the stack. Subsequently, if a ResultType token is parsed while the MethodDeclaration flag remains set, the return type of the method being parsed can be recorded.

In general, flags are set in the various OptimizingParser.visit() methods. The logic to respond to those conditions occurs in the OptimizingParser.find(Token) method.

Limitations and issues

The preprocessor has been tested on a limited set of MASS programs that were manually converted from their existing format to use Java reflection.

MASS programs that are to be preprocessed should not contain a method named `callMethod` or use a trailing underscore (`_`) as part of a method name. The preprocessor will append an underscore to method names as part of the `callMethod()` generation. If the preprocessor encounters a naming conflict, it will report an error and halt.

All methods that are to be called as part of the MASS Agents or Places method must accept the same set of parameters in the same order as defined by the MASS constructor.

The preprocessor has been tested against a very limited set of Java programs. When the parser encounters a language construct that is not defined in its grammar file, it will crash. Examples of this include, but are not limited to, enumeration (`enum <ID> implements { }`) and annotations (`@override`).

More testing against valid Java programs should be done. Java programs for validating a Java compiler can be found in the openjdk project at <http://openjdk.java.net> (openjdk/langtools/test/tools/javac)

Files included in MassPrePro

Filename	Description
MassPrePro.java	Executable front-end that takes a MASS program from stdin and outputs an optimized version to stdout.
Java1.1.jjt	A grammar for the Java language. Originally based on 1.1 and extended to handle some newer Java syntax.
MassVar.java	Stores MASS variables (Agents, Places).
MethodData.java	Holds data related to the method currently being parsed.
OptimizingVisitor.java	Extends UnparseVisitor to provide MASS preprocessor functionality. Provides the logic at the parser token level. Controls output based on parser tokens.
ScopeManager.java	Store the variables associated with a particular scope. Adds and removes scopes from the scope-stack.
UnparseVisitor.java	Provides basic functionality for all parser tokens in the Java1.1.jj grammar.
SimpleNode.java	Implements Node
Java1.1.jj	Auto-generated by JJTree from Java1.1.jjt.
ASTAdditiveExpression.java	Auto-generated by JJTree
ASTAllocationExpression.java	Auto-generated by JJTree
ASTAndExpression.java	Auto-generated by JJTree
ASTArgumentList.java	Auto-generated by JJTree
ASTArguments.java	Auto-generated by JJTree
ASTArrayDimsAndInits.java	Auto-generated by JJTree
ASTArrayInitializer.java	Auto-generated by JJTree
ASTAssignmentOperator.java	Auto-generated by JJTree
ASTBlock.java	Auto-generated by JJTree
ASTBlockStatement.java	Auto-generated by JJTree
ASTBooleanLiteral.java	Auto-generated by JJTree
ASTBreakStatement.java	Auto-generated by JJTree
ASTCastExpression.java	Auto-generated by JJTree
ASTCastLookahead.java	Auto-generated by JJTree
ASTClassBody.java	Auto-generated by JJTree
ASTClassBodyDeclaration.java	Auto-generated by JJTree
ASTClassDeclaration.java	Auto-generated by JJTree
ASTCompilationUnit.java	Auto-generated by JJTree
ASTConditionalAndExpression.java	Auto-generated by JJTree
ASTConditionalExpression.java	Auto-generated by JJTree
ASTConditionalOrExpression.java	Auto-generated by JJTree
ASTConstructorDeclaration.java	Auto-generated by JJTree
ASTContinueStatement.java	Auto-generated by JJTree
ASTDoStatement.java	Auto-generated by JJTree
ASTEmptyStatement.java	Auto-generated by JJTree
ASTEqualityExpression.java	Auto-generated by JJTree

ASTExclusiveOrExpression.java	Auto-generated by JJTree
ASTExplicitConstructorInvocation.java	Auto-generated by JJTree
ASTExpression.java	Auto-generated by JJTree
ASTFieldDeclaration.java	Auto-generated by JJTree
ASTForEach.java	Auto-generated by JJTree
ASTForEachStatement.java	Auto-generated by JJTree
ASTForInit.java	Auto-generated by JJTree
ASTFormalParameter.java	Auto-generated by JJTree
ASTFormalParameters.java	Auto-generated by JJTree
ASTForStatement.java	Auto-generated by JJTree
ASTForTraditional.java	Auto-generated by JJTree
ASTForUpdate.java	Auto-generated by JJTree
ASTIdentifier.java	Auto-generated by JJTree
ASTIfStatement.java	Auto-generated by JJTree
ASTImportDeclaration.java	Auto-generated by JJTree
ASTInclusiveOrExpression.java	Auto-generated by JJTree
ASTInitializer.java	Auto-generated by JJTree
ASTInstanceOfExpression.java	Auto-generated by JJTree
ASTInterfaceDeclaration.java	Auto-generated by JJTree
ASTInterfaceMemberDeclaration.java	Auto-generated by JJTree
ASTLabeledStatement.java	Auto-generated by JJTree
ASTLiteral.java	Auto-generated by JJTree
ASTLocalVariableDeclaration.java	Auto-generated by JJTree
ASTMethodDeclaration.java	Auto-generated by JJTree
ASTMethodDeclarationLookahead.java	Auto-generated by JJTree
ASTMethodDeclarator.java	Auto-generated by JJTree
ASTMultiplicativeExpression.java	Auto-generated by JJTree
ASTName.java	Auto-generated by JJTree
ASTNameList.java	Auto-generated by JJTree
ASTNestedClassDeclaration.java	Auto-generated by JJTree
ASTNestedInterfaceDeclaration.java	Auto-generated by JJTree
ASTNullLiteral.java	Auto-generated by JJTree
ASTPackageDeclaration.java	Auto-generated by JJTree
ASTPostfixExpression.java	Auto-generated by JJTree
ASTPreDecrementExpression.java	Auto-generated by JJTree
ASTPreIncrementExpression.java	Auto-generated by JJTree
ASTPrimaryExpression.java	Auto-generated by JJTree
ASTPrimaryPrefix.java	Auto-generated by JJTree
ASTPrimarySuffix.java	Auto-generated by JJTree
ASTPrimitiveType.java	Auto-generated by JJTree
ASTRelationalExpression.java	Auto-generated by JJTree
ASTResultType.java	Auto-generated by JJTree
ASTReturnStatement.java	Auto-generated by JJTree
ASTShiftExpression.java	Auto-generated by JJTree
ASTStatement.java	Auto-generated by JJTree
ASTStatementExpression.java	Auto-generated by JJTree

ASTStatementExpressionList.java	Auto-generated by JJTree
ASTStringLiteral.java	Auto-generated by JJTree
ASTSwitchLabel.java	Auto-generated by JJTree
ASTSwitchStatement.java	Auto-generated by JJTree
ASTSynchronizedStatement.java	Auto-generated by JJTree
ASTThrowStatement.java	Auto-generated by JJTree
ASTTryStatement.java	Auto-generated by JJTree
ASTType.java	Auto-generated by JJTree
ASTTypeDeclaration.java	Auto-generated by JJTree
ASTUnaryExpression.java	Auto-generated by JJTree
ASTUnaryExpressionNotPlusMinus.java	Auto-generated by JJTree
ASTUnmodifiedClassDeclaration.java	Auto-generated by JJTree
ASTUnmodifiedInterfaceDeclaration.java	Auto-generated by JJTree
ASTVariableDeclarator.java	Auto-generated by JJTree
ASTVariableDeclaratorId.java	Auto-generated by JJTree
ASTVariableInitializer.java	Auto-generated by JJTree
ASTWhileStatement.java	Auto-generated by JJTree
JavaCharStream.java	Auto-generated by JavaCC
JavaParser.java	Auto-generated by JavaCC
JavaParserConstants.java	Auto-generated by JavaCC
JavaParserTokenManager.java	Auto-generated by JavaCC
JavaParserTreeConstants.java	Auto-generated by JavaCC
JavaParserVisitor.java	Auto-generated by JavaCC
JJTJavaParserState.java	Auto-generated by JavaCC
Node.java	Auto-generated by JJTree
ParseException.java	Auto-generated by JavaCC
Token.java	Auto-generated by JavaCC
TokenMgrError.java	Auto-generated by JavaCC