MODULE KWALA-SYNTAX

Syntax

...

```
SYNTAX   #Id ::= object
               | main
               | null
SYNTAX   Variable ::= #Int
SYNTAX   TypeNameBase ::= #Id
                        | #Id / TypeNameBase
SYNTAX   TypeName ::= L TypeNameBase
                    | A TypeNameBase
SYNTAX   TypeReference ::= < #Id , TypeName >
SYNTAX   FieldReference ::= < #Id , TypeName , #Id , TypeReference >
SYNTAX   Params ::= List{Variable,","}
SYNTAX   NewInstructionBase ::= Variable =new TypeReference @ #Int
SYNTAX   NewInstruction ::= NewInstructionBase
                          | NewInstructionBase ( Params ) []i
SYNTAX   GetInstruction ::= Variable =getfield FieldReference  Variable
                          | Variable =getstatic FieldReference
SYNTAX   PutInstruction ::= putfield Variable = Variable  FieldReference
                          | putstatic Variable  FieldReference
SYNTAX   PhiInstruction ::= Variable =phi( Params )
SYNTAX   Instruction ::= NewInstruction
                       | GetInstruction
                       | PutInstruction
                       | PhiInstruction
SYNTAX   BBEdge ::= #Id -> #Id ;
SYNTAX   BlockBody ::= List{Instruction,";"}
SYNTAX   Block ::= #Id :{ BlockBody }
SYNTAX   TaskUnit ::= BBEdge
                    | Block
SYNTAX   Task ::= TaskUnit
                | Task  Task
```

END MODULE

MODULE KWALA
IMPORTS KWALA-SYNTAX

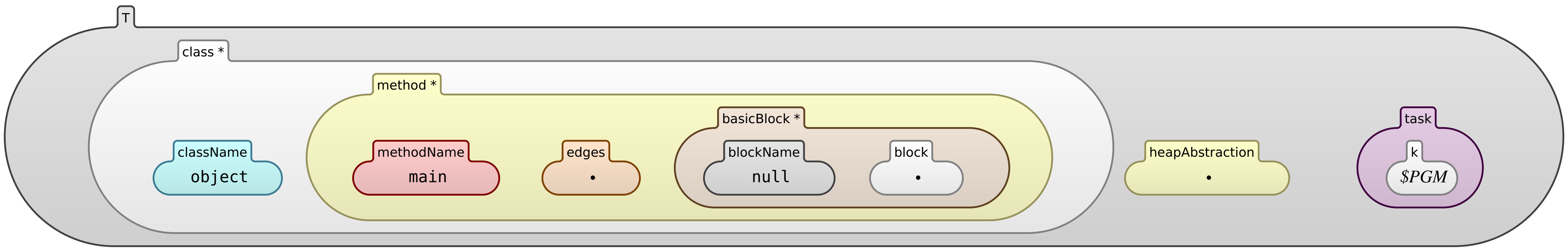
Semantics

...

Configuration

...

CONFIGURATION:



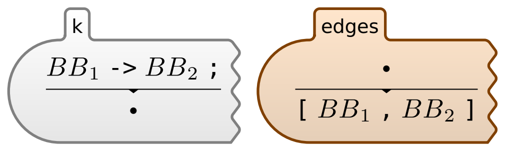
Processing Basic Blocks

...

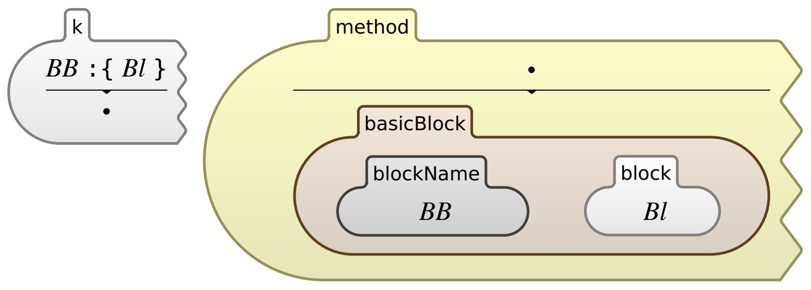
RULE $T_1 \ T_2 \Rightarrow T_1 \curvearrowright T_2$

SYNTAX *ListItem* ::= [*#Id* , *#Id*]

RULE



RULE



Results

...

END MODULE