

# The Meta-Environment

Márcio Coelho PG16046  
André Rocha PG12345

- ASF - Algebraic Specification Formalism
- SDF - Syntax Definition Formalism
- Rascal

# ASF

- Algebraic Specification Formalism
- Permite rescrita de termos e travessia de arvores genéricas

# SDF

- Syntax Definition Formalism
- SDF é mais rica que BNF

```
module basic/Booleans
```

```
exports
```

```
  sorts Boolean
```

```
  context-free start-symbols Boolean
```

```
context-free syntax
```

```
  "true"                                -> Boolean
```

```
  "false"                               -> Boolean
```

```
  lhs:Boolean "/" rhs:Boolean -> Boolean {left}
```

```
  lhs:Boolean "&" rhs:Boolean -> Boolean {left}
```

```
  "not" "(" Boolean ")"                -> Boolean
```

```
  "(" Boolean ")"                      -> Boolean
```

```
context-free priorities
```

```
  Boolean "&" Boolean -> Boolean >
```

```
  Boolean "/" Boolean -> Boolean
```

# Operações

- Parser
- Análise de código
- Transformação de código
- Criação, implementação e compilação de DSL

# IDE

The screenshot displays the ASF+SDF Meta-Environment IDE interface. The main window, titled 'import-graph', shows a dependency graph with nodes such as Pico-eval, Value-environments, Pico-Values, Strings, Bytes, Integers, Table, List, Pico-Syntax, Pico-Identifiers, Pico-Types, NatCon, StrCon, BytesCon, Comments, Whitespace, Booleans, and BoolCon. The graph illustrates the relationships between these modules, with arrows indicating dependencies.

On the left side, there are two tool windows:

- Progress**: A table showing the status of various modules for 'sdf' and 'asf'.
- Moduledetails**: A table showing details for the selected module.

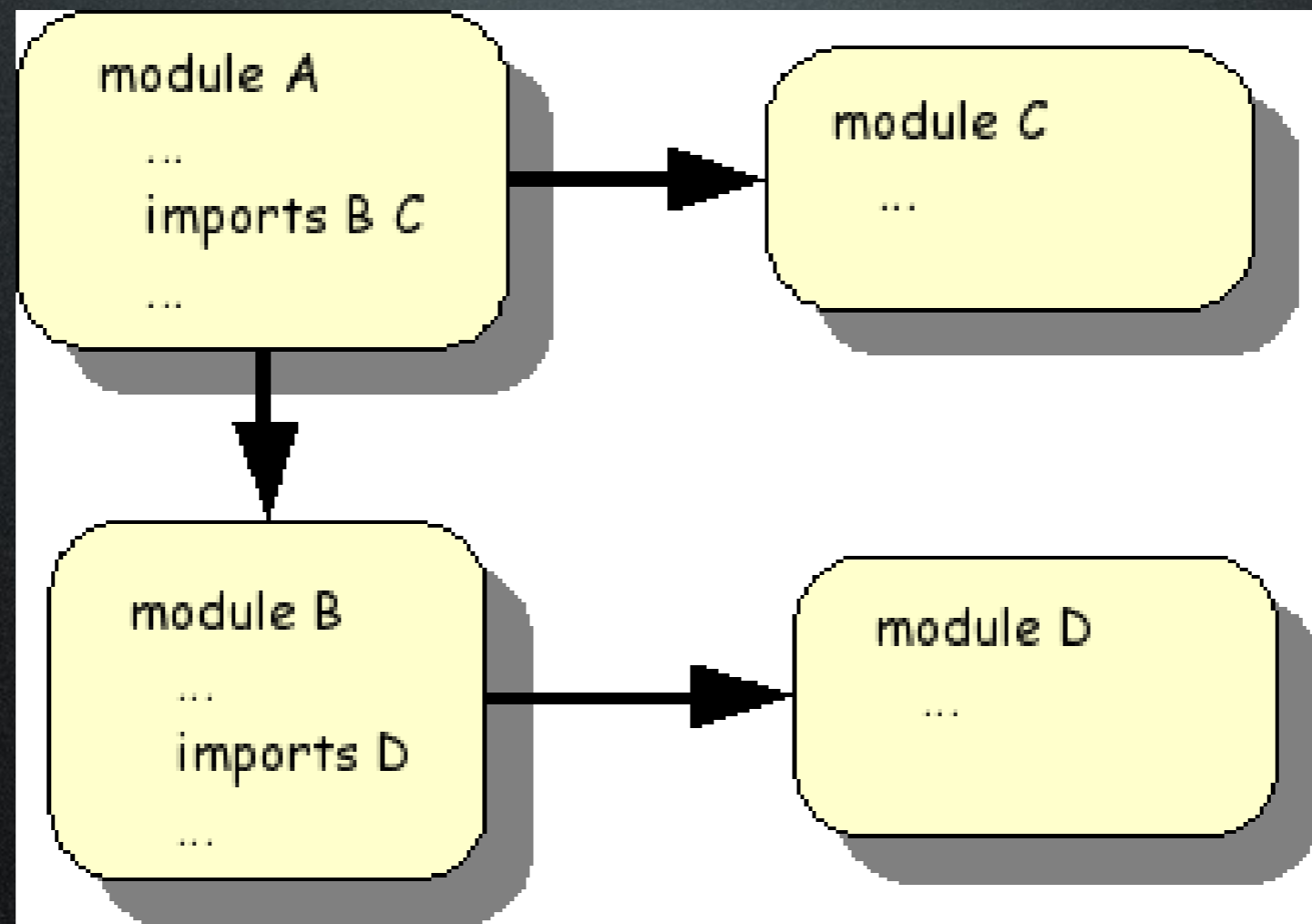
At the bottom right, the **Error Viewer** displays the following messages:

- illegal use of reserved sort CHAR in production
- character class not allowed in context-free syntax
- character class not allowed in context-free syntax

Module	sdf	asf
languages/pic...	complete	parsed
languages/pic...	complete	unavailable
languages/pic...	complete	unavailable
containers/Table	complete	parsed
languages/pic...	complete	unavailable
languages/pic...	complete	unavailable
basic/StrCon	complete	unavailable
basic/NatCon	complete	unavailable
languages/pic...	complete	unavailable
containers/List	complete	parsed
basic/Booleans	complete	parsed
basic/Comments	complete	unavailable
basic/Strings	complete	parsed
basic/Integers	complete	parsed
basic/Whitesp...	complete	unavailable
basic/BoolCon	complete	unavailable
basic/Bytes	complete	parsed
basic/BytesCon	complete	unavailable

Namespace	Key	Value
"asf"	"status"	parsed
"asf"	"path"	path("/export/...
"asf"	"name"	"languages/pic...
"graph"	"file"	file(file(relative...
"graph"	"label"	label("Pico-eval")
"graph"	"bgcolor"	fill-color(rgb(2...
"sdf"	"status"	complete
"sdf"	"path"	path("/export/...
"sdf"	"name"	"languages/pic...
"sdf"	"editable"	false
"asf"	"editable"	false
"graph"	"shape"	shape(ellipse)

# Anatomy





# Estrutura de um Modulo

```
module ModuleName  
    ImportSection*  
    ExportOrHiddenSection*  
equations  
    ConditionalEquation*
```

Exemplo ASF+SDF

```
Import MyBooleans  
imports basic/Whitespace  
imports basic/Booleans
```

```
exports  
  context-free syntax  
  Boolean "xor" Boolean --> Boolean
```

```
hiddens  
  context-free start-symbols Boolean
```

## Equations

[xor-1] true xor true = false

[xor-2] true xor false = true

[xor-3] false xor true = true

[xor-4] false xor false =  
false

CMinus

module Desktop/Cminus/Types

exports

sorts TYPE

context-free syntax

"char" -> TYPE

"int" -> TYPE

"nil-type" -> TYPE

module Desktop/Cminus/Iden

exports

sorts ID

lexical syntax

[a-z][a-zA-Z0-9]\* -> ID

context-free restrictions

ID -/- [a-zA-Z0-9]

context-free syntax

DECLARATION+           -> PROGRAM

context-free syntax

VARIABLE               -> DECLARATION

FUNCTION               -> DECLARATION

context-free syntax

TYPE ID ";"           -> VARIABLE

context-free syntax

TYPE ID

"(" ( FORMALPARAMETER ("," FORMALPARAMETER)\* )? ")"

BLOCK               -> FUNCTION

context-free syntax

TYPE ID               -> FORMALPARAMETER

context-free syntax

"{" VARIABLE\* STAT\* "}"           -> BLOCK