```
<root> ::= "<root " <name> ">" <node> <parameter> <constraint> "</root>"
<node> ::= "<node " <name> ">" <node> <parameter> <constraint> "</node>"
          | <node> <node>

<parameter> ::= "<parameter " <name> <p_arg> <nb_instances> "/>"
              | <parameter> <parameter>
              | ε

<name> ::= "name="" <letter> <string> "" "
<nb_instances> ::= "nb_instances="" <integer> "" "
               | ε

<p_arg> ::= "type="boolean" "
          | "type="boolean" " <boolean_values>
          | "type="boolean" " <boolean_values> <boolean_weights>
          | "type="string" " <string_values>
          | "type="string" " <string_values_and_weights>
          | "type="integer" " <integer_min> <integer_max> <numerical_optional_attributes>
          | "type="real" " <real_min> <real_max> <numerical_optional_attributes>

<boolean_values> ::= "values="[True;False]" "
                   | "values="[False;True]" "
<boolean_weights> ::= "weights="[" <integer> ";" <integer> "]" "

<string_values> :== "values="[" <val> "]" "
<val> ::= <string>
        | <string> ";" <string>

<string_values_and_weights> ::= "values="[" <string> <vnw> <integer> "]" "
<vnw> ::= ";" <string> <vnw> <integer> ";"
        | "]" weights="["

<integer_min> ::= "min="" <integer> "" "
<integer_max> ::= "max="" <integer> "" "
<real_min> ::= "min="" <float> "" "
<real_max> ::= "max="" <float> "" "

<numerical_optional_attributes> ::= ε
                                  | "distribution="u" "
                                  | "distribution="n" "
                                  | "distribution="n" " <mean>
                                  | "distribution="n" " <variance>
                                  | "distribution="n" " <mean> <variance>
                                  | "distribution="i" " <interval>
                                  | "distribution="i" " <interval_and_weights>

<mean> ::= "mean="" <float> "" "
<variance> ::= "variance="" <float> "" "
<interval> ::= "ranges="" <inter_list> "" "
<inter_list> ::= <inter> ";" <inter>
```

```
                        | <inter>
<inter> ::= "[" <float> "," <float> "] "

<interval_and_weights> :== "ranges="" <inter> <inw> <integer> "]" "
<inw> ::= ";" <inter> <vnw> <integer> ";"
          |"" weights="["

<constraint> ::= "<constraint " <name> <type><expressions> <quantifiers_and_ranges> "/>"
                 |<constraint> <constraint>
                 | ε
<type> ::= "types = "forall""
          |"types = "exist""
          |"types = "unique""
          | ε

<quantifiers_and_ranges> ::= ε
                             |"quantifiers="[" <letter> <qnr> <range> "]" "
<qnr> ::= ";" <letter> <vnw> <range> ";"
          |"]" ranges="["
<range> ::= "[" <expr> "," <expr> "]"

expressions> ::= " expressions="" <expr> "" "
<expr> ::= ";" <expr>
           | ε

<expr> ::= "(" <expr> ")"
           |"AND(" <expr> ","<expr> ")"
           |"OR(" <expr> ","<expr> ")"
           |"NOT(" <expr> ")"
           |"IMPLIES(" <expr> ","<expr> ")"
           | <num_expr> <comparison_operator> <num_expr>

<num_expr> ::= <num_expr> "+" <num_expr>
               | <num_expr> "-" <num_expr>
               | <num_expr> "*" <num_expr>
               | <num_expr> "/" <num_expr>
               | <num_expr> "%" <num_expr>
               | <tree_path>
               | <integer>
               | <float>

<tree_path> ::= <tree_path_beginning><tree_path_element><tree_path_ending>

<tree_path_beginning> ::= ε
                          | "./"
                          | "../"
                          | <tree_path_beginning> <tree_path_beginning>

<tree_path_element> ::= <string> "[" <expr> "]"
                        | <tree_path_element> "/" <tree_path_element>
```

<tree_path_ending> ::= ε
               | ".nb_instances"
               | ".value"

<float> ::= <integer> "." <integer>
<integer> ::= <non_nul_digit><digit_string>

<digit_string> ::= <digit_string><digit_string>
             | <digit>
<string> ::= <string><string>
       | <letter>
       | <digit>
       | "_"
<non_nul_digit> ::= "1" | "2" | "3" …. "9"
<digit> ::= "0"
      | <non_nul_digit>

<letter> ::= "a" | "A" | "b" | "B" | "c" | "C" …. "y" | "Y" | "z" | "Z"
<comparison_operator> ::= "SUP" | "SUPEQ" | "INF" | "INFEQ" | "EQ" | "DIF"