

1 Czym się zajmowałem

Zajmowałem się implementacją i rozwijaniem back-endu kompilatora, to jest klas Lexer i Parser.

2 Tokeny

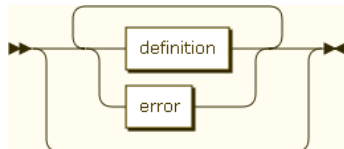
Przy wyborze tokenów staraliśmy się zachować zgodność ze sposobem zapisu atrybutów i operacji w UMLu (stosowanie oznaczeń '-', '+', etc. w przypadku widoczności) i ułatwić zrozumienie kodu (w przypadku ograniczeń: użycie nazw angielskich 'allow', 'deny').

3 Produkcje

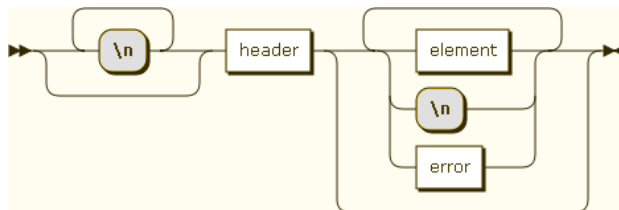
Na diagramach składni tokeny wyróżniono ciemniejszym kolorem. Elementy należące do biblioteki pyparsing wyróżniono czarnym kolorem.

3.1 Obiekty

- grammar:



- definition:



- header:



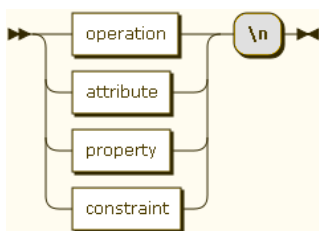
- parent_name:



- object_name:

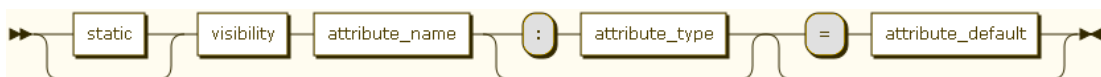


- element:



3.2 Atrybuty

- attribute:



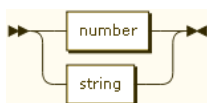
- attribute_name:



- attribute_type:



- attribute_value:



3.3 Operacje

- operation:



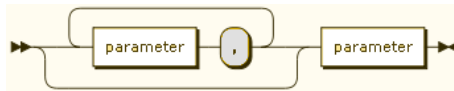
- operation_name:



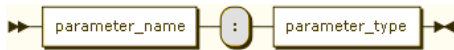
- return_type:



- parameters:



- parameter:



- parameter_name:

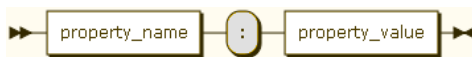


- parameter_type:



3.4 Właściwości

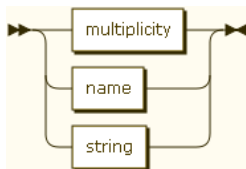
- property:



- property_name:



- property_value:



3.5 Ograniczenia

- constraint:



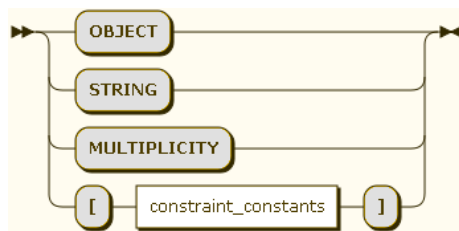
- constraint_type:



- constraint_key:



- constraint_value:



- constraints_constants:

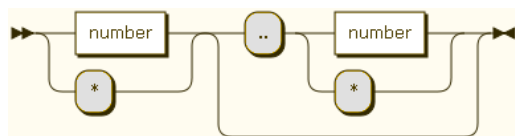


- constraint_constant:

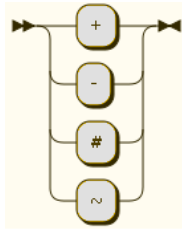


3.6 Generyczne

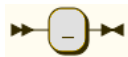
- multiplicity:



- visibility:



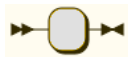
- static:



- prototype:



- error:



3.7 pyparsing

- number:



- name:



- string:

