Specification: YaPPL

Here you find the BNF (Backus-Naur Form) of YaPPL.

YaPPL grammar BNF:

```
<policy> ::= '{' <id>' ','
             <preference> '}'
<id> ::= '_id' ':' <number>
<preference> ::= '"' 'preference' '"' ':' '[' <rule_list> ']'
<rule_list> ::= <rule> | <rule_list> <rule>
<rule> ::= '{'
           <purpose> ','
           <utilizer> ','
           <transformation> ','
           <valid_from> ','
           <exp_date>
           '}'
<purpose> ::= '"' 'purpose' '"' ':' '{'
                <permitted_purpose> ','
                <excluded_purpose>
                '}'
<utilizer> ::= '"' 'utilizer' '"' ':' '{'
                  <permitted_utilizer> ','
                  <excluded_utilizer>
                  '}'
<permitted_purpose> ::= '"' 'permitted' '"' ':'
                                  '[' <purpose_item_list> ']'
<excluded_purpose> ::= '"' 'excluded' '"' ':'
                                   '[' <purpose_item_list> ']'
<permitted_utilizer> ::= '"' 'permitted' '"' ':'
                                  '[' <utilizer_item_list> ']'
<excluded_utilizer> ::= '"' 'excluded' '"' ':'
                                  '[' <utilizer_item_list> ']'
```

```
<purpose_item_list> ::= <purpose_item> |
                            <purpose_item_list> <purpose_item>
<utilizer_item_list> ::= <utilizer_item> |
                          <utilizer_item_list> <utilizer_item>
<purpose_item> ::= '"' <string> '"'
<utilizer_item> ::= '"' <string> '"'
<transformation> ::= '"' 'transformation' '"' ':'
                  '[' <transformation_item_list> ']'
<transformation_item_list> ::= <transformation_item>
  | <transformation_item_list> <transformation_item>
<transformation_item> ::= '{'
                          "", 'attribute', '", ':'
                              <attribute> ','
                          "", "tr_func, "", ":,
                              <tr_func>
                          ι,
<attribute> ::= '"' <string> '"'
<tr_func> ::= '"' <string> '"'
<valid_from> ::= '"' 'valid_from' '"' ':' <datetime>
<exp_date> ::= '"' 'exp_date' '"' ':' <datetime>
<datetime> ::= '"','d','d','d','-','d','d','-','d','d','T'
               'd''d'':''d'''d'':''d'''d'''.'
               'd''d''Z' '"'
<number> ::= 'd' <number> | 'd'
<string> ::= 'l' <string> | 'd' <string> | 's' <string> |
             'l' | 'd' | 's'
where 'd' = [0..9] and 'l' = [a..z, A..Z] and
      's' = [<,>,_,-,(,)]
```