

## FeatureList xText EBNF Version 2.0

### • Core Features:

1. BNF front-end (Parser, Lexer, Linker, etc.)
2. translates a textual representation into an Ecore model
3. Editor - provides IDE features, e.g. auto-completion, syntax highlighting, etc.

### • Validation Rules: these rules will give warnings in the editor if they find find consistencies in the code

1. *checkUpdateGrammarConsistency*: checks, if the grammar is an updated grammar, for consistencies between the originals and the new grammar

status: not migrated

2. *!CheckReferencedOnlyOnce*: this should warn if there is a rule that is only referenced once, this is an inlining hint e.g.:

status: migrated

3. *checkPassthroughRule*: e.g.:this tests if a rule gets just passed through `a ::= b b ::= "literal" <=> a ::= "literal"` , which is better for EBNF

status: migrated

4. *checkEqualAlternative*: this tests if a rule got equal alternatives e.g.: `a ::= d | d` status: still some problems

status: migrated

5. *checkDuplicateRules*: checks if there is another rule in the EtsiBnf?, that got the same behaviour e.g.: `a ::= "literal" | b c ::= "literal" | b`

status: migrated

6. *checkSubruleDuplicates*:

status: not migrated

7. *check unused rule*: it checks if a rule is used (except for the rule with the number 1)

status: migrated

8. *checkNameIsUnique*: checks if the nam of a rule is already in use e.g.: `a ::= "end" a ::= "fin"`

status: migrated

### • Generation:

xsl-fo file for all bnf grammars (not for merge or deltagrammars) this can be Rendered to Pdf using apache-fop

- **Formatting:**

*old:* had some but they are got commented out in the main project

*new:* Line-break after every rule, white-spaces after all literals, possible update linebreak and double tab after all >|< bars

- **Refactoring & Quickfixing:**

1. *auto Numbering of rules*  
-not migrated
2. *remove unused rules*  
-migrated
3. *replacement of passthroughRules*  
-not migrated
4. *refactor uppercasetokenRules*  
-not migrated
5. *find references* , this one is standard in xtext2.\* -renaming, this is standard in xtext2.\*

\*Grammar composition:

-3 grammartypes:

grammar: normal grammar  
deltagrammar: different notation for a extension grammar,

adds new rules to a grammar (regular rules)and  
adds changes to rules of the extended  
grammar(extension rules) -can be automaticly  
created

mergegrammar: for merging extensions

-old version had an automatic composition Process, for merge grammars, with a small pop-up-menu.

- at the moment no support in the new version

- **Deploying:**

export project as Plugin: Plug-in Development/deployable plug-ins and fragments  
-rcp project with a minimal eclipse editor