

# naive-ebnf: L<sup>A</sup>T<sub>E</sub>X Package for EBNF in Plain Text<sup>\*</sup>

Yegor Bugayenko  
yegor256@gmail.com

2023-02-21, 0.0.5

## 1 Introduction

This package helps render an [Extended Backus-Naur Form](#) using plain text notation:

$\langle \lambda\text{-Expr} \rangle \rightarrow \langle \text{Var} \rangle$ $  "λ" \langle \text{Var} \rangle " ." \langle \text{Expr} \rangle$ $  "(" \langle \text{Expr} \rangle \langle \text{Expr} \rangle ")"$	1   \documentclass{minimal} 2   \usepackage{naive-ebnf} 3   \usepackage{mathtools} 4   \begin{document} 5   \begin{ebnf} 6   <\$\lambda\$-Expr> := <Var> \ 7      "\$\lambda\$-Expr" <Var> ".<Expr> \ 8      "\$\lparen\$ <Expr> <Expr> "\$\rparen\$" 9   \end{ebnf} 10   \end{document}
--	---

**ebnf** The `ebnf` environment *doesn't* add any formatting to the paragraph, but only replaces the plain text symbols, such as “`:=`” and “`<Var>`” with proper L<sup>A</sup>T<sub>E</sub>X commands. The following syntax is understood inside the `ebnf` environment:

- `:=` separates the left-hand side from the right-hand side of the production rule;
- `<...>` denotes a non-terminal (variable);
- `"..."` denotes a terminal symbol;
- `(...|...)` denotes a series of options to choose from;
- `[...]` denotes an optional substitution;
- `{...}` denotes a zero or more times repetition;
- `||` denotes an indented vertical bar at the beginning of the string.

**Attention:** The usage of some symbols is prohibited inside terminals. Instead, the following substitutions are recommended:

- `$\lparen` and `$\rparen` instead of “`(`” and “`)`” (from the [mathtools](#) package);

---

\*The sources are in GitHub at [yegor256/naive-ebnf](#)

- $\$\\langle$  and  $\$\\rangle$  instead of “<” and “>”;
- $\$\\{$  and  $\$\\}$  instead of “{” and “}” (also `mathtools`);
- $\$\\[$  and  $\$\\]$  instead of “[” and “]” (also `mathtools`);
- $\$\\|$  instead of “|”.

`width` There is an optional argument of `ebnf` environment, which sets the width of the left-hand side of each rule (the default width is `6em`):

This EBNF has a larger width of the left hand side than usual:  
 $\langle\text{VeryLongVariable}\rangle \rightarrow \langle X \rangle \mid \langle Y \rangle$   
 $\quad \langle X \rangle \rightarrow "X"$   
 $\quad \langle Y \rangle \rightarrow "Y"$

```

4 This EBNF has a larger width of \\
5 the left hand side than usual: \par
6 \begin{ebnf}[1.5in]
7 <VeryLongVariable> := <X> | <Y> \\
8 <X> := "X" \\
9 <Y> := "Y"
10 \end{ebnf}

```

`\terminal` Inside the text, terminals and non-terminals may be formatted using two supplementary commands:

The non-terminal `\Var` in  $\lambda$ -calculus  
may be equal to  $v_1, v_2, \dots$ . Application starts with “(“ and ends with “)”.

```

6 The non-terminal \nonterminal{Var}
7 in  $\$\\lambda$ -calculus may be equal
8 to  $\$v\_1, v\_2, \dots$ . Application
9 starts with \terminal{()} and ends
10 with \terminal{()}.


```

It's possible to use them in math-mode too, for example:

If “( $f_1(\lambda\text{-}\text{Var})$ )” is always true,  
then  $f_1$  is a tautology.

```

6 If \$\terminal{()} f_1
7 \nonterminal{\lambda-$\text{-}\text{Var}}
8 \terminal{()}\$ is always true, then
9 \$f_1\$ is a tautology.

```

## 2 Package Options

It's possible to configure the behavior of the package with the help of a few package options:

`bw` By default, some colors are used in the rendered grammar. However, the `bw` package option disables any colors and makes sure the grammar is black-and-white:

```
\usepackage[bw]{naive-ebnf}
```

`trail` The `ebnf` environment is doing pre-processing of the  $\text{\TeX}$  commands provided and then let  $\text{\LaTeX}$  render them. It may be useful to see the output generated by the pre-processing. The `trail` option (with a file name) asks the package to save the content of the environment after the pre-processing into the file:

```
\usepackage[trail=log.tex]{naive-ebnf}
```

### 3 Implementation

First, we process package options:

```
1 \RequirePackage{pgfopts}
2 \pgfkeys{
3   /ebnf/.cd,
4   bw/.store in=\ebnf@bw,
5   trail/.store in=\ebnf@trail,
6   trail/.default=naive-ebnf.tmp.tex,
7   trail
8 }
9 \ProcessPgfPackageOptions{/ebnf}
```

Then, we include a few packages, mostly to deal with  $\text{\LaTeX}^3$  expressions:

```
10 \RequirePackage{expl3}
```

`\ebnf@color` Then, we include `xcolor` to colorize the output a bit:

```
11 \makeatletter\ifdefined\ebnf@bw\else
12   \RequirePackage{xcolor}
13 \fi
14 \newcommand\ebnf@color[2]
15 { \ifdefined\ebnf@bw\#2\else\textcolor{\#1}{\#2}\fi}
16 \makeatother
```

`\terminal` Then, we a command to render a single terminal:

```
17 \makeatletter
18 \newcommand\terminal[1]{%
19   \relax\ifmmode\else\ttfamily\fi%
20   \ebnf@color{gray}{\relax\ifmmode\textsf{'}\else\sffamily'\fi}%
21   #1%
22   \ebnf@color{gray}{\relax\ifmmode\textsf{'}\else\sffamily'\fi}}%
23 \makeatother
```

`\nonterminal` Then, we a command to render a single non-terminal:

```
24 \makeatletter
25 \newcommand\nonterminal[1]{%
26   \ebnf@color{gray}{\relax\ifmmode\langle\else(\langle\fi}%
27   \relax\ifmmode\textsf{\#1}\else\sffamily\#1\fi%
28   \ebnf@color{gray}{\relax\ifmmode\rangle\else)\rangle\fi}}%
29 \makeatother
```

Then, we define supplementary commands:

```
30 \makeatletter
31 \newcommand\ebnf@optional[1]
32 { \ebnf@color{gray}{[\#1\ebnf@color{gray}{]}}}
33 \newcommand\ebnf@repetition[1]
34 { \ebnf@color{gray}{\{\#1\ebnf@color{gray}{\}}}}
35 \newcommand\ebnf@grouping[1]
36 { \ebnf@color{gray}{(\#1\ebnf@color{gray}{})}}
37 \ExplSyntaxOn
38 \newcommand\ebnf@terminal[1]{
39   \tl_set:Nn \l_ebnf_tl { }
40   \tl_set_rescan:Nno \l_ebnf_tl { } { #1 }
41   \terminal{\l_ebnf_tl}
```

```

42 }
43 \newcommand\ebnf@nonterminal[1]{
44   \tl_set:Nn \l_ebnf_tl { }
45   \tl_set_rescan:Nno \l_ebnf_tl { } { #1 }
46   \nonterminal{\l_ebnf_tl}
47 }
48 \ExplSyntaxOff
49 \newcommand\ebnf@to
50   {\ebnf@color{gray}{\(\to\)}}
51 \newcommand\ebnf@alternation
52   {\ebnf@color{gray}{\(\vee\)}}
53 \makeatother

```

**ebnf** Then, we define the ebnf environment:

```

54 \ExplSyntaxOn
55 \cs_generate_variant:Nn \tl_replace_all:Nnn {Nx}
56 \makeatletter
57 \NewDocumentEnvironment{ebnf}{O{4em}+b}
58   {\tl_set:Nn\l__ebnf_tmp_tl{#2}}
59   {%
60     \regex_replace_all:nnN { \{(.*?)\} }%
61       {\c{ebnf@repetition}{\1} \l__ebnf_tmp_t1}%
62     \regex_replace_all:nnN { \((.*?)\)}%
63       {\c{ebnf@grouping}{\1} \l__ebnf_tmp_t1}%
64     \regex_replace_all:nnN { \[(.*?)\]}%
65       {\c{ebnf@optional}{\1} \l__ebnf_tmp_t1}%
66     \regex_replace_all:nnN { <[^>]+?>\ :=) }%
67       {\c{makebox}{\1}[r]\l__ebnf_tmp_t1}%
68     \regex_replace_all:nnN { <(.*?)> }%
69       {\c{ebnf@nonterminal}{\1} \l__ebnf_tmp_t1}%
70     \regex_replace_all:nnN { "(.*?)" }%
71       {\c{ebnf@terminal}{\1} \l__ebnf_tmp_t1}%
72     \regex_replace_all:nnN { \|(\|) }%
73       {\c{makebox}{\1}[r]\l__ebnf_tmp_t1}%
74     \regex_replace_all:nnN { \|\ }%
75       {\c{ebnf@alternation}{} \l__ebnf_tmp_t1}%
76     \regex_replace_all:nnN { := }%
77       {\c{ebnf@to}{} \l__ebnf_tmp_t1}%
78     \tl_put_left:Nn \l__ebnf_tmp_t1 {\noindent}
79     \tl_put_right:Nn \l__ebnf_tmp_t1 {}
80     \ifdefined\ebnf@trail%
81       \newwrite\ebnf@write%
82       \immediate\openout\ebnf@write\ebnf@trail\relax%
83       \immediate\write\ebnf@write{\unexpanded\expandafter{\l__ebnf_tmp_t1}}%
84       \immediate\closeout\ebnf@write%
85     \message{naive-ebnf:\space pre-processed\space TeX}%
86       \space saved\space to\space "\ebnf@trail"^^J}%
87     \fi%
88     \l__ebnf_tmp_t1}
89 \makeatother
90 \ExplSyntaxOff
91 \endinput

```

## Change History

0.0.1	General: First draft.	3	0.0.3	\terminal: Quotes fixed in both text and math modes.	3
0.0.2	General: Proper parsing of grouping. Substitutions suggested for special symbols.	3	0.0.4	ebnf: Any symbols are allowed inside \nonterminal commands and inside the ebnf environment, where non-terminals are mentioned.	4
	\nonterminal: New command \nonterminal added, to enable rendering non-terminal symbols outside of the ebnf environment.	3	0.0.5	General: New package option trail added, to enable saving generated T <sub>E</sub> X content to a file, for debugging purposes.	3
	\terminal: New command \terminal added, to enable rendering terminal symbols outside of the ebnf environment.	3			

# Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in roman refer to the code lines where the entry is used.

Symbols		
\(	26, 28, 50, 52, 62	\ExplSyntaxOff ... 48, 90
\)	26, 28, 50, 52, 62	\ExplSyntaxOn ... 37, 54
\[	64	
\{	34, 60	\ifdefined ... 11, 15, 80
\}	34, 60	\ifmmode 19, 20, 22, 26, 27, 28
\_u	66	\immediate ... 82, 83, 84
\]	64	
\	72, 74	\l ... 39, 40, 41, 44, 45, 46, 58, 61, 63, 65, 67, 69, 71, 73, 75, 77, 78, 79, 83, 88
C		
\c	61, 63, 65, 67, 69, 71, 73, 75, 77	\langle ... 26
\closeout	84	
\cs	55	
E		\makeatletter ... ... 11, 17, 24, 30, 56
\ebnf	54	\makeatother ... ... 16, 23, 29, 53, 89
\ebnf@alternation	51	\message ... 85
\ebnf@bw	4, 11, 15	
\ebnf@color	... 11, 20, 22, 26, 28, 32, 34, 36, 50, 52	N
\ebnf@grouping	35	\newcommand ... ... 14, 18, 25, 31,
\ebnf@nonterminal	43	33, 35, 38, 43, 49, 51
\ebnf@optional	31	\NewDocumentEnvironment
\ebnf@repetition	33	... 57
\ebnf@terminal	38	\newwrite ... 81
\ebnf@to	49	\noindent ... 78
\ebnf@trail	5, 80, 82, 86	\nonterminal ... 24, 46
\ebnf@write	81, 82, 83, 84	O
\endinput	91	
\expandafter	83	\openout ... 82
		\vert ... 52
		W
		\write ... 83
P		
\pgfkeys	...	2
\ProcessPgfPackageOptions	...	9
R		
\rangle	...	28
\regex	60, 62, 64, 66, 68, 70, 72, 74, 76	
\relax	...	19, 20, 22, 26, 27, 28, 82
\RequirePackage	1, 10, 12	
S		
\sffamily	...	20, 22, 27
\space	...	85, 86
T		
\terminal	...	17, 41
\textcolor	...	15
\textsf	...	20, 22, 27
\tl	39, 40, 44, 45, 55, 58, 78, 79	
\to	...	50
\ttfamily	...	19
U		
\unexpanded	...	83
V		
\vert	...	52
W		