

Ruby trunk - Feature #13686

Add states of scanner to tokens from Ripper.lex and Ripper::Filter#on_*

06/27/2017 12:58 PM - aycabta (aycabta .)

Status:	Closed
Priority:	Normal
Assignee:	nobu (Nobuyoshi Nakada)
Target version:	2.5

Description

I'm writing syntax analysis software by pure Ruby, for processing Ruby's source code and meta information what are classes, methods, constants, comments and others. I'm using Ripper for it. But the results of Ripper.sexp doesn't have comments, and the results of Ripper.lex doesn't have token states of scanner.

I think that the behavior of Ripper.sexp is correct because the position of comments in Ruby's syntax tree is blurred and unhandled.

On the other hand, Ripper.lex has comments token but Ripper.lex clearly drops the states of scanner for each tokens under "[finite-state scanner](#)". The states are very important for many Ripper use cases. EXPR_END is especially required to know the end of statement. If the states aren't provided, we must re-implement finite-state analyzer for tokens from Ripper. For example, borderlines of conditions, args, constants and others. It's just not realistic.

In Ripper.lex's behavior as of now:

```
require 'ripper'
require 'pp'

pp Ripper.lex(<<EOM)
def str?(v)
  String === v # check
end
EOM
#=> [[1, 0], :on_kw, "def" ],
# [[1, 3], :on_sp, " " ],
# [[1, 4], :on_ident, "str?" ],
# [[1, 8], :on_lparen, "(" ],
# [[1, 9], :on_ident, "v" ],
# [[1, 10], :on_rparen, ")" ],
# [[1, 11], :on_ignored_nl, "\n" ],
# [[2, 0], :on_sp, " " ],
# [[2, 2], :on_const, "String" ],
# [[2, 8], :on_sp, " " ],
# [[2, 9], :on_op, "===" ],
# [[2, 12], :on_sp, " " ],
# [[2, 13], :on_ident, "v" ],
# [[2, 14], :on_sp, " " ],
# [[2, 15], :on_comment, "# check\n"],
# [[3, 0], :on_kw, "end" ],
# [[3, 3], :on_nl, "\n" ]]
```

In Ripper.lex's behavior with attached patch:

```
require 'ripper'
require 'pp'

pp Ripper.lex(<<EOM)
def str?(v)
  String === v # check
end
EOM
#=> [[[1, 0], :on_kw, "def", Ripper::EXPR_FNAME ],
# [[1, 3], :on_sp, " ", Ripper::EXPR_FNAME ],
# [[1, 4], :on_ident, "str?", Ripper::EXPR_ENDFN ],
# [[1, 8], :on_lparen, "(", Ripper::EXPR_BEG | Ripper::EXPR_LABEL],
# [[1, 9], :on_ident, "v", Ripper::EXPR_ARG ],
```

```

# [[1, 10], :on_rparen,      ")",      Ripper::EXPR_ENDFN      ],
# [[1, 11], :on_ignored_nl, "\n",      Ripper::EXPR_BEG      ],
# [[2, 0], :on_sp,         " ",      Ripper::EXPR_BEG      ],
# [[2, 2], :on_const,     "String", Ripper::EXPR_CMDARG    ],
# [[2, 8], :on_sp,         " ",      Ripper::EXPR_CMDARG    ],
# [[2, 9], :on_op,         "===",    Ripper::EXPR_BEG      ],
# [[2, 12], :on_sp,        " ",      Ripper::EXPR_BEG      ],
# [[2, 13], :on_ident,    "v",      Ripper::EXPR_END | Ripper::EXPR_LABEL],
# [[2, 14], :on_sp,        " ",      Ripper::EXPR_END | Ripper::EXPR_LABEL],
# [[2, 15], :on_comment,   "# check\n", Ripper::EXPR_END | Ripper::EXPR_LABEL],
# [[3, 0], :on_kw,         "end",    Ripper::EXPR_END      ],
# [[3, 3], :on_nl,         "\n",     Ripper::EXPR_BEG      ]]

```

In Ripper::Filter#on_* with attached patch, you can use #state method:

```

require 'ripper'
require 'pp'

class MyFilter < Ripper::Filter
  def on_default(event, tok, data)
    data.push([event, tok, state])
  end
end

# Ripper::Filter#parse works like Enumerable#inject
pp MyFilter.new(<<EOM).parse([])
def str?(v)
  String === v # check
end
EOM
#=> [[[:on_kw,      "def",      Ripper::EXPR_FNAME      ],
# [:on_sp,         " ",      Ripper::EXPR_FNAME      ],
# [:on_ident,     "str?",    Ripper::EXPR_ENDFN     ],
# [:on_lparen,    "(",      Ripper::EXPR_BEG | Ripper::EXPR_LABEL],
# [:on_ident,     "v",      Ripper::EXPR_ARG       ],
# [:on_rparen,    ")",      Ripper::EXPR_ENDFN     ],
# [:on_ignored_nl, "\n",     Ripper::EXPR_BEG      ],
# [:on_sp,         " ",      Ripper::EXPR_BEG      ],
# [:on_const,     "String",  Ripper::EXPR_CMDARG    ],
# [:on_sp,         " ",      Ripper::EXPR_CMDARG    ],
# [:on_op,         "===",    Ripper::EXPR_BEG      ],
# [:on_sp,         " ",      Ripper::EXPR_BEG      ],
# [:on_ident,     "v",      Ripper::EXPR_END | Ripper::EXPR_LABEL],
# [:on_sp,         " ",      Ripper::EXPR_END | Ripper::EXPR_LABEL],
# [:on_comment,   "# check\n", Ripper::EXPR_END | Ripper::EXPR_LABEL],
# [:on_kw,         "end",    Ripper::EXPR_END      ],
# [:on_nl,         "\n",     Ripper::EXPR_BEG      ]]

```

Associated revisions

Revision 7df1e45b - 09/14/2017 10:53 AM - nobu (Nobuyoshi Nakada)

ripper: add states of scanner

- parse.y (ripper_state): add states of scanner to tokens from Ripper.lex and Ripper::Filter#on_*. based on the patch by aycabta (Code Ahss) at [ruby-core:81789]. [Feature #13686]
- ext/ripper/tools/preproc.rb (prelude, usercode): generate EXPR_* constants from enums.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@59896 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

Revision 59896 - 09/14/2017 10:53 AM - nobu (Nobuyoshi Nakada)

ripper: add states of scanner

- parse.y (ripper_state): add states of scanner to tokens from Ripper.lex and Ripper::Filter#on_*. based on the patch by

aycabta (Code Ahss) at [ruby-core:81789]. [Feature #13686]

- ext/ripper/tools/preproc.rb (prelude, usercode): generate EXPR_* constants from enums.

Revision 59896 - 09/14/2017 10:53 AM - nobu (Nobuyoshi Nakada)

ripper: add states of scanner

- parse.y (ripper_state): add states of scanner to tokens from Ripper.lex and Ripper::Filter#on_*. based on the patch by aycabta (Code Ahss) at [ruby-core:81789]. [Feature #13686]
- ext/ripper/tools/preproc.rb (prelude, usercode): generate EXPR_* constants from enums.

Revision 59896 - 09/14/2017 10:53 AM - nobu (Nobuyoshi Nakada)

ripper: add states of scanner

- parse.y (ripper_state): add states of scanner to tokens from Ripper.lex and Ripper::Filter#on_*. based on the patch by aycabta (Code Ahss) at [ruby-core:81789]. [Feature #13686]
- ext/ripper/tools/preproc.rb (prelude, usercode): generate EXPR_* constants from enums.

History

#1 - 08/31/2017 05:19 AM - hsb (Hiroshi SHIBATA)

- Target version set to 2.5
- Assignee set to nobu (Nobuyoshi Nakada)
- Status changed from Open to Assigned

#2 - 09/14/2017 10:53 AM - nobu (Nobuyoshi Nakada)

- Status changed from Assigned to Closed

Applied in changeset [trunk|r59896](#).

ripper: add states of scanner

- parse.y (ripper_state): add states of scanner to tokens from Ripper.lex and Ripper::Filter#on_*. based on the patch by aycabta (Code Ahss) at [ruby-core:81789]. [Feature [#13686](#)]
- ext/ripper/tools/preproc.rb (prelude, usercode): generate EXPR_* constants from enums.

Files

add-state-to-ripper-for-trunk.patch	16.1 KB	06/27/2017	aycabta (aycabta .)
-------------------------------------	---------	------------	---------------------