

**1. Copyright.**

Copyright © Dave Bone 1998 - 2015

**2. *c\_comments* Thread.**

Handles both flavours of c++ comments.

- 1) // single line type comment
- 2) /\* ... \*/ type comment

To speed things up, i'm using a finite-state approach to lower the pushdown deterministic rule push / pop cycles.

**3. Fsm Cc\_comments class.****4. Cc\_comments constructor directive.**

```
<Cc_comments constructor directive 4> ≡
    ddd_idx_ = 0;
    ddd_[ddd_idx_] = 0;
```

**5. Cc\_comments op directive.**

```
<Cc_comments op directive 5> ≡
    ddd_idx_ = 0;
    ddd_[ddd_idx_] = 0;
```

**6. Cc\_comments user-declaration directive.**

```
<Cc_comments user-declaration directive 6> ≡
public: char ddd_[1024 * 32];
int ddd_idx_;
void copy_str_into_buffer(std::string * Str);
void copy_kstr_into_buffer(const char *Str);
```

**7. Cc\_comments user-implementation directive.**

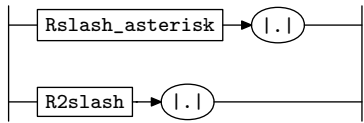
```
<Cc_comments user-implementation directive 7> ≡
void Cc_comments::copy_str_into_buffer(std::string * Str)
{
    const char *y = Str->c_str();
    int x(0);
    for ( ; y[x] ≠ 0; ++x, ++ddd_idx_) ddd_[ddd_idx_] = y[x];
    ddd_[ddd_idx_] = 0;
}
```

**8. *copy\_kstr\_into\_buffer*.**

```
<More code 8> ≡
void Cc_comments::copy_kstr_into_buffer(const char *Str)
{
    const char *y = Str;
    int x(0);
    for ( ; y[x] ≠ 0; ++x, ++ddd_idx_) ddd_[ddd_idx_] = y[x];
    ddd_[ddd_idx_] = 0;
}
```

9. *Rc\_comments* rule.

Rc\_comments

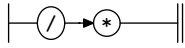
10. *Rc\_comments* op directive.

⟨ *Rc\_comments* op directive 10 ⟩ ≡

```

Cc_comments * fsm = ( Cc_comments * ) rule_info_.parser_--fsm_tbl_;
T_comment * com = new T_comment((const char *) &fsm_ddd_);
com->set_rc(*rule_info_.parser_--start_token_, __FILE__, __LINE__);
RSVP(com);

```

11. *Rslash\_asterisk* rule.**Rslash\_asterisk**

(*Rslash\_asterisk* subrule 1 op directive 11)  $\equiv$

```

Cc_comments * fsm = ( Cc_comments * ) rule_info__parser__fsm_tbl__
fsm->copy_kstr_into_buffer("/"); /* due to lex scanner */
fsm->copy_kstr_into_buffer("*");
loop:
switch (rule_info__parser__current_token()-enumerated_id__) {
case T_Enum::T_raw_lf_: goto other;
case T_Enum::T_raw_cr_: goto cr;
case T_Enum::T_LR1_eog_: goto overrun;
case T_Enum::T_raw_asteric_: goto aster;
default: goto other;
}
cr:
{ /* cr lf? */
fsm->copy_kstr_into_buffer(rule_info__parser__current_token()-id__);
rule_info__parser__get_next_token();
if (rule_info__parser__current_token()-enumerated_id__ != T_Enum::T_raw_lf_) goto loop;
/* not cr lf */
fsm->copy_kstr_into_buffer(rule_info__parser__current_token()-id__);
rule_info__parser__get_next_token();
goto loop;
}
;
aster:
{ /* end of k? */
fsm->copy_kstr_into_buffer(rule_info__parser__current_token()-id__);
rule_info__parser__get_next_token();
if (rule_info__parser__current_token()-enumerated_id__ != T_Enum::T_raw_slash_) goto loop;
/* false eok */
fsm->copy_kstr_into_buffer(rule_info__parser__current_token()-id__);
rule_info__parser__get_next_token();
return;
}
overrun:
{
CAbs_lr1_sym * sym = new Err_comment_overrun;
sym->set_rc(*rule_info__parser__start_token__, __FILE__, __LINE__);
RSVP(sym);
rule_info__parser__set_stop_parse(true);
return;
}
other:
{
fsm->copy_kstr_into_buffer(rule_info__parser__current_token()-id__);
rule_info__parser__get_next_token();
goto loop;
}

```

12. *R2slash* rule.

R2slash



⟨R2slash subrule 1 op directive 12⟩ ≡

```

Cc_comments * fsm = ( Cc_comments * ) rule_info__parser__fsm_tbl__;
fsm->copy_kstr_into_buffer("/");
fsm->copy_kstr_into_buffer("/");
for ( ; ; ) {
    switch (rule_info__parser__current_token()-enumerated_id_) {
        case T_Enum :: T_raw_lf_:
            {
                return;
            }
        case T_Enum :: T_raw_cr_:
            {
                return;
            }
        case T_Enum :: T_LR1_eog_: return;
    }
    fsm->copy_kstr_into_buffer(rule_info__parser__current_token()-id__);
    rule_info__parser__get_next_token();
}

```

**13. First Set Language for  $O_2^{linker}$ .**

```
/*
  File: c_comments.fsc
  Date and Time: Fri Jan  2 15:33:28 2015
*/
transitive    n
grammar-name  "c_comments"
name-space    "NS_c_comments"
thread-name   "TH_c_comments"
monolithic    n
file-name     "c_comments.fsc"
no-of-T       569
list-of-native-first-set-terminals 1
  raw_slash
end-list-of-native-first-set-terminals
list-of-transitive-threads 0
end-list-of-transitive-threads
list-of-used-threads 0
end-list-of-used-threads
fsm-comments
"C++ type comments lexer."
```

14. Lr1 State Network.

|                          |                 |   |                          |  |                               |                  |
|--------------------------|-----------------|---|--------------------------|--|-------------------------------|------------------|
| ⇒                        |                 |   |                          |  | State: 1 state type: <i>s</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| c                        | Rslash_asterisk |   | 2 1 1 /                  |  |                               | 1 2 3            |
| c                        | R2slash         |   | 3 1 1 /                  |  |                               | 1 2 4            |
| c                        | Rc_comments     |   | 1 1 1 Rslash_asterisk  . |  |                               | 1 5 6            |
| c                        | Rc_comments     |   | 1 2 1 R2slash  .         |  |                               | 1 7 8            |
| ⇒/                       |                 |   |                          |  | State: 2 state type: <i>s</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | Rslash_asterisk |   | 2 1 2 *                  |  |                               | 1 3 3            |
| t                        | R2slash         |   | 3 1 2 /                  |  |                               | 1 4 4            |
| ⇒*                       |                 |   |                          |  | State: 3 state type: <i>r</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | Rslash_asterisk |   | 2 1 3                    |  |                               | 1 0 3 1          |
| ⇒/                       |                 |   |                          |  | State: 4 state type: <i>r</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | R2slash         |   | 3 1 3                    |  |                               | 1 0 4 1          |
| ⇒ <i>Rslash_asterisk</i> |                 |   |                          |  | State: 5 state type: <i>s</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | Rc_comments     |   | 1 1 2  .                 |  |                               | 1 6 6            |
| ⇒ .                      |                 |   |                          |  | State: 6 state type: <i>r</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | Rc_comments     |   | 1 1 3                    |  |                               | 1 0 6 2          |
| ⇒ <i>R2slash</i>         |                 |   |                          |  | State: 7 state type: <i>s</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | Rc_comments     |   | 1 2 2  .                 |  |                               | 1 8 8            |
| ⇒ .                      |                 |   |                          |  | State: 8 state type: <i>r</i> |                  |
| ←                        | rule            | → | R# sr# Po ←              |  | subrule element               | → Brn Gto Red LA |
| t                        | Rc_comments     |   | 1 2 3                    |  |                               | 1 0 8 2          |

**15. Index.**

|. |: 9.  
\_\_FILE\_\_: 10, 11.  
\_\_LINE\_\_: 10, 11.  
aster: [11](#).  
c\_comments: 2.  
c\_str: 7.  
CAbs\_lr1\_sym: 11.  
Cc\_comments: [7](#), [8](#), 10, 11, 12.  
com: 10.  
copy\_kstr\_into\_buffer: [6](#), [8](#), 11, 12.  
copy\_str\_into\_buffer: [6](#), [7](#).  
cr: [11](#).  
current\_token: 11, 12.  
ddd: 4, 5, [6](#), 7, 8, 10.  
ddd\_idx: 4, 5, [6](#), 7, 8.  
enumerated\_id\_: 11, 12.  
Err\_comment\_ouerrun: 11.  
fsm: 10, 11, 12.  
fsm\_tbl\_: 10, 11, 12.  
get\_next\_token: 11, 12.  
id\_: 11, 12.  
loop: [11](#).  
other: [11](#).  
ouerrun: [11](#).  
parser\_: 10, 11, 12.  
Rc\_comments: [9](#).  
Rslash\_asterisk: 9.  
Rslash\_asterisk: [11](#).  
RSVP: 10, 11.  
rule\_info\_: 10, 11, 12.  
R2slash: [12](#).  
R2slash: 9.  
set\_rc: 10, 11.  
set\_stop\_parse: 11.  
start\_token\_: 10, 11.  
std: 6, 7.  
Str: [6](#), 7, [8](#).  
string: 6, 7.  
sym: 11.  
T\_comment: 10.  
T\_Enum: 11, 12.  
T\_LR1\_eog\_: 11, 12.  
T\_raw\_asteric\_: 11.  
T\_raw\_cr\_: 11, 12.  
T\_raw\_lf\_: 11, 12.  
T\_raw\_slash\_: 11.  
true: 11.  
x: [7](#), [8](#).  
y: [7](#), [8](#).



- ⟨ Cc\_comments constructor directive 4 ⟩
- ⟨ Cc\_comments op directive 5 ⟩
- ⟨ Cc\_comments user-declaration directive 6 ⟩
- ⟨ Cc\_comments user-implementation directive 7 ⟩
- ⟨ More code 8 ⟩
- ⟨ R2slash subrule 1 op directive 12 ⟩
- ⟨ Rc\_comments op directive 10 ⟩
- ⟨ Rslash\_asterisk subrule 1 op directive 11 ⟩

## c\_comments Grammar

Date: January 2, 2015 at 15:34

File: c\_comments.lex

Ns: NS\_c\_comments

Version: 1.0

Debug: false

Grammar Comments:

Type: Thread

C++ type comments lexer.

1 element(s) in Lookahead Expression below

eolr

|   | Section            | Page |
|---|--------------------|------|
| <b>Copyright</b> .....  | <a href="#">1</a>  | 1    |
| <i>c_comments</i> <b>Thread</b> .....                         | <a href="#">2</a>  | 2    |
| Fsm Cc_comments class .....                                   | <a href="#">3</a>  | 2    |
| Cc_comments constructor directive .....                       | <a href="#">4</a>  | 2    |
| Cc_comments op directive .....                                | <a href="#">5</a>  | 2    |
| Cc_comments user-declaration directive .....                  | <a href="#">6</a>  | 2    |
| Cc_comments user-implementation directive .....               | <a href="#">7</a>  | 2    |
| <i>copy_kstr_into_buffer</i> .....                            | <a href="#">8</a>  | 2    |
| <i>Rc_comments</i> rule .....                                 | <a href="#">9</a>  | 3    |
| Rc_comments op directive .....                                | <a href="#">10</a> | 3    |
| <i>Rslash_asterisk</i> rule .....                             | <a href="#">11</a> | 4    |
| <i>R2slash</i> rule .....                                     | <a href="#">12</a> | 5    |
| <b>First Set Language for <math>O_2^{linker}</math></b> ..... | <a href="#">13</a> | 6    |
| <b>Lr1 State Network</b> .....                                | <a href="#">14</a> | 7    |
| <b>Index</b> .....  | <a href="#">15</a> | 8    |