

**1. Copyright.**

Copyright © Dave Bone 1998 - 2015

**2. *la\_expr\_lexical* grammar.**

Verify the tokens “identifier”, “c-string”, and “c-literal” against the symbol table using their literal values as keys. The found vocabulary symbol is then put into the token container for *la\_expr* grammar digestion.

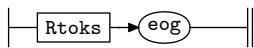
Remember, the “parallel-parser” construct containing the lookahead expression comes before the Grammar’s vocabulary definitions. These input token scraps are raw lexems without a symbol table translation that now need symbol table evaluation.

**3. Fsm *Cl\_expr\_lexical* class.****4. *Cl\_expr\_lexical* user-prefix-declaration directive.**

```
<Cl_expr_lexical user-prefix-declaration directive 4> ≡
#include "yacco2_stbl.h"
```

**5. *Rstripper* rule.**

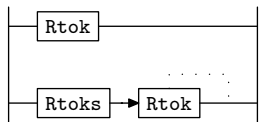
Rstripper

**6. *Rstripper* op directive.**

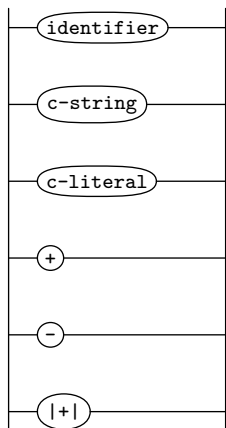
```
<Rstripper op directive 6> ≡
using namespace NS_yacco2_k_symbols;
ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2 :: PTR_LR1_eog_);
ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2 :: PTR_LR1_eog_);
```

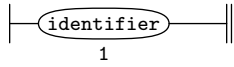
**7. *Rtoks* rule.**

Rtoks

**8. *Rtok* rule.**

Rtok



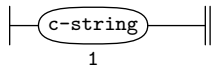
9. *Rtok's subrule 1.*

⟨Rtok subrule 1 op directive 9⟩ ≡

```

using namespace yacco2_stbl;
T_sym_tbl_report_card report_card;
find_sym_in_stbl(report_card, *sf→p1→identifier()→c_str());
if (report_card.action_ ≡ T_sym_tbl_report_card::not_fnd) {
    CAbs_lr1_sym * sym = new Err_bad_term_in_la_expr;
    sym→set_rc(*sf→p1→, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info→parser→set_stop_parse(true);
    return;
}
ADD_TOKEN_TO_PRODUCER_QUEUE(*report_card.tbl_entry→symbol_);

```

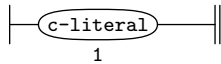
10. *Rtok's subrule 2.*

⟨Rtok subrule 2 op directive 10⟩ ≡

```

using namespace yacco2_stbl;
T_sym_tbl_report_card report_card;
find_sym_in_stbl(report_card, *sf→p1→c_string()→c_str());
if (report_card.action_ ≡ T_sym_tbl_report_card::not_fnd) {
    CAbs_lr1_sym * sym = new Err_bad_term_in_la_expr;
    sym→set_rc(*sf→p1→, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info→parser→set_stop_parse(true);
    return;
}
ADD_TOKEN_TO_PRODUCER_QUEUE(*report_card.tbl_entry→symbol_);

```

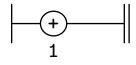
11. *Rtok's subrule 3.*

⟨Rtok subrule 3 op directive 11⟩ ≡

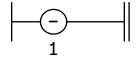
```

using namespace yacco2_stbl;
T_sym_tbl_report_card report_card;
find_sym_in_stbl(report_card, *sf→p1→c_literal()→c_str());
if (report_card.action_ ≡ T_sym_tbl_report_card::not_fnd) {
    CAbs_lr1_sym * sym = new Err_bad_term_in_la_expr;
    sym→set_rc(*sf→p1→, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info→parser→set_stop_parse(true);
    return;
}
ADD_TOKEN_TO_PRODUCER_QUEUE(*report_card.tbl_entry→symbol_);

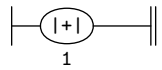
```

**12. *Rtok*'s subrule 4.**

⟨ Rtok subrule 4 op directive 12 ⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf-p1-);

**13. *Rtok*'s subrule 5.**

⟨ Rtok subrule 5 op directive 13 ⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf-p1-);

**14. *Rtok*'s subrule 6.**

Drain the swamp. Accept the keywords, LR k constants, and raw characters.  
 ⟨ Rtok subrule 6 op directive 14 ⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf-p1-);

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

```
/*
  File: la_expr_lexical.fsc
  Date and Time: Fri Jan  2 15:33:39 2015
*/
transitive      n
grammar-name    "la_expr_lexical"
name-space      "NS_la_expr_lexical"
thread-name     "Cla_expr_lexical"
monolithic      y
file-name       "la_expr_lexical.fsc"
no-of-T         569
list-of-native-first-set-terminals 6
  LR1_all_shift_operator
  raw_plus
  raw_minus
  T_c_literal
  T_c_string
  T_identifier
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
"Lexer: 2nd stage lexing for lookahead: stripper..."
```

## 16. Lr1 State Network.

$\Rightarrow$					State: 1 state type: $s$			
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
c	Rtok		3	6	1	+		1 2 2
c	Rtok		3	4	1	+		1 3 3
c	Rtok		3	5	1	-		1 4 4
c	Rtok		3	3	1	c-literal		1 5 5
c	Rtok		3	2	1	c-string		1 6 6
c	Rtok		3	1	1	identifier		1 7 7
c	Rtoks		2	2	1	Rtoks <u>Rtok</u>		1 8 10
c	Rstripper		1	1	1	Rtoks <u>eog</u>		1 8 9
c	Rtoks		2	1	1	Rtok		1 11 11
$\Rightarrow$	+						State: 2 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rtok		3	6	2			1 0 2 1
$\Rightarrow$	+						State: 3 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rtok		3	4	2			1 0 3 1
$\Rightarrow$	-						State: 4 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rtok		3	5	2			1 0 4 1
$\Rightarrow$	c-literal						State: 5 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rtok		3	3	2			1 0 5 1
$\Rightarrow$	c-string						State: 6 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rtok		3	2	2			1 0 6 1
$\Rightarrow$	identifier						State: 7 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rtok		3	1	2			1 0 7 1
$\Rightarrow$	Rtoks						State: 8 state type: $s$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rstripper		1	1	2	eog		1 9 9
c	Rtok		3	6	1	+		8 2 2
c	Rtok		3	4	1	+		8 3 3
c	Rtok		3	5	1	-		8 4 4
c	Rtok		3	3	1	c-literal		8 5 5
c	Rtok		3	2	1	c-string		8 6 6
c	Rtok		3	1	1	identifier		8 7 7
t	Rtoks		2	2	2	Rtok		1 10 10
$\Rightarrow$	eog						State: 9 state type: $r$	
$\leftarrow$	rule	$\rightarrow$	R#	sr#	Po	$\leftarrow$	subrule element	$\rightarrow$ Brn Gto Red LA
t	Rstripper		1	1	3			1 0 9 2

$\Rightarrow$  *Rtok*  
 $\leftarrow$  rule  $\rightarrow$  R# sr# Po  $\leftarrow$   
t Rtoks 2 2 3

State: 10 state type: *r*  
subrule element

$\rightarrow$  Brn Gto Red LA  
1 0 10 1

$\Rightarrow$  *Rtok*  
 $\leftarrow$  rule  $\rightarrow$  R# sr# Po  $\leftarrow$   
t Rtoks 2 1 2

State: 11 state type: *r*  
subrule element

$\rightarrow$  Brn Gto Red LA  
1 0 11 1

**17. Index.**

|+|: 8.  
\_\_FILE\_\_: 9, 10, 11.  
\_\_LINE\_\_: 9, 10, 11.  
*action\_*: 9, 10, 11.  
ADD\_TOKEN\_TO\_ERROR\_QUEUE: 9, 10, 11.  
ADD\_TOKEN\_TO\_PRODUCER\_QUEUE: 6, 9, 10, 11,  
12, 13, 14.  
c-literal: 8.  
c-string: 8.  
*c.literal*: 11.  
*c.str*: 9, 10, 11.  
*c.string*: 10.  
*CAbs\_lr1\_sym*: 9, 10, 11.  
eog: 5.  
*Err\_bad\_term\_in\_la\_expr*: 9, 10, 11.  
*find\_sym\_in\_stbl*: 9, 10, 11.  
identifier: 8.  
*identifier*: 9.  
*la\_expr*: 2.  
*la\_expr\_lexical*: 2.  
*not\_fnd*: 9, 10, 11.  
**NS\_yacco2\_k\_symbols**: 6.  
*parser\_*: 9, 10, 11.  
*PTR\_LR1\_eog\_*: 6.  
*p1\_*: 9, 10, 11, 12, 13, 14.  
*report\_card*: 9, 10, 11.  
*Rstripper*: 5.  
Rtok: 7.  
*Rtok*: 8, 9, 10, 11, 12, 13, 14.  
*Rtoks*: 7.  
Rtoks: 5, 7.  
*rule\_info\_*: 9, 10, 11.  
*set\_rc*: 9, 10, 11.  
*set\_stop\_parse*: 9, 10, 11.  
*sf*: 9, 10, 11, 12, 13, 14.  
*sym*: 9, 10, 11.  
*symbol\_*: 9, 10, 11.  
*T\_sym\_tbl\_report\_card*: 9, 10, 11.  
*tbl\_entry\_*: 9, 10, 11.  
*true*: 9, 10, 11.  
*yacco2*: 6.  
**yacco2\_stbl**: 9, 10, 11.



- ⟨ Cla\_expr\_lexical user-prefix-declaration directive 4 ⟩
- ⟨ Rstripper op directive 6 ⟩
- ⟨ Rtok subrule 1 op directive 9 ⟩
- ⟨ Rtok subrule 2 op directive 10 ⟩
- ⟨ Rtok subrule 3 op directive 11 ⟩
- ⟨ Rtok subrule 4 op directive 12 ⟩
- ⟨ Rtok subrule 5 op directive 13 ⟩
- ⟨ Rtok subrule 6 op directive 14 ⟩

# la\_expr\_lexical Grammar

Date: January 2, 2015 at 15:36

File: la\_expr\_lexical.lex

Ns: NS\_la\_expr\_lexical

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

Lexer: 2nd stage lexing for lookahead: stripper...

	Section	Page
<b>Copyright</b> .....	<a href="#">1</a>	1
<i>la_expr_lexical</i> <b>grammar</b> .....	<a href="#">2</a>	2
Fsm Cla_expr_lexical class .....	<a href="#">3</a>	2
Cla_expr_lexical user-prefix-declaration directive .....	<a href="#">4</a>	2
<i>Rstripper</i> rule .....	<a href="#">5</a>	2
Rstripper op directive .....	<a href="#">6</a>	2
<i>Rtoks</i> rule .....	<a href="#">7</a>	2
<i>Rtok</i> rule .....	<a href="#">8</a>	2
<i>Rtok</i> 's subrule 1 .....	<a href="#">9</a>	3
<i>Rtok</i> 's subrule 2 .....	<a href="#">10</a>	3
<i>Rtok</i> 's subrule 3 .....	<a href="#">11</a>	3
<i>Rtok</i> 's subrule 4 .....	<a href="#">12</a>	4
<i>Rtok</i> 's subrule 5 .....	<a href="#">13</a>	4
<i>Rtok</i> 's subrule 6 .....	<a href="#">14</a>	4
<b>First Set Language for <math>O_2^{linker}</math></b> .....	<a href="#">15</a>	5
<b>Lr1 State Network</b> .....	<a href="#">16</a>	6
<b>Index</b> .....	<a href="#">17</a>	8