

1. Copyright.

Copyright © Dave Bone 1998 - 2015

2. *rtn_component* Thread.

A T returned from a thread.

3. Fsm Crtn_component class.**4. Crtn_component user-declaration directive.**

⟨Crtn_component user-declaration directive 4⟩ ≡

```
public: void find_T_of_chr_str(const char *Str, yacco2 :: CAbs_lr1_sym * Tok);
```

5. Crtn_component user-implementation directive.

⟨Crtn_component user-implementation directive 5⟩ ≡

```
void Crtn_component :: find_T_of_chr_str(const char *Str, yacco2 :: CAbs_lr1_sym * Tok){
    T_sym.tbl_report_card report_card;
    using namespace yacco2_stbl;
    find_sym_in_stbl(report_card, *Str);
    if (report_card.action_ ≡ T_sym.tbl_report_card :: not_fnd) {
        CAbs_lr1_sym * sym = new Err_subrule_use_undefined_T;
        sym->set_rc(*parser->start_token_, __FILE__, __LINE__);
        RSVP_FSM(sym);
        parser->set_stop_parse(true);
        return;
    }
    if (report_card.tbl_entry->type_ ≠ table_entry :: terminal) {
        CAbs_lr1_sym * sym = new Err_not_T_for_rtned_token_from_th;
        sym->set_rc(*parser->start_token_, __FILE__, __LINE__);
        RSVP_FSM(sym);
        parser->set_stop_parse(true);
        return;
    }
    report_card.tbl_entry->used_ = true;
    CAbs_lr1_sym * sym = report_card.tbl_entry->symbol_; T_in_stbl * T = ( T_in_stbl * ) sym;
    refered_T * rT = new refered_T(*T);
    rT->set_rc(*Tok, __FILE__, __LINE__);
    T->add_T_into_xref(*rT);
    RSVP_FSM(T);
    parser->set_stop_parse(true); }
```

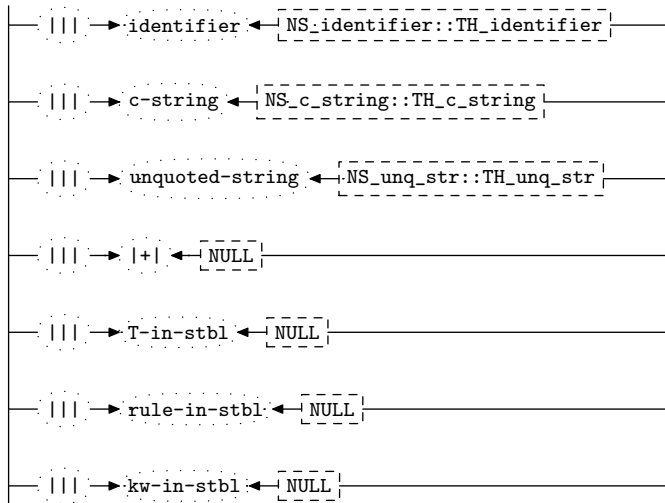
6. Crtn_component user-prefix-declaration directive.

⟨Crtn_component user-prefix-declaration directive 6⟩ ≡

```
#include "identifier.h"
#include "c_string.h"
#include "unq_str.h"
#include "yacco2_stbl.h"
```

7. *Rrtn_component* rule.

Rrtn_component



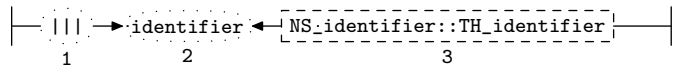
8. *Rrtn_component* arbitrator-code directive.

unquoted-string superset identifier and also includes other T like -> and cweb comments ie cweb-marker. Code only gets tried when # parms in accept queue > 1. Single entry is one of identifier or cweb.

<Rrtn_component arbitrator-code directive 8> ≡

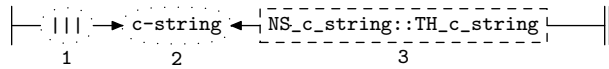
```
#include "war_begin_code.h"
using namespace NS_yacco2_T_enum;
for (i = 1; i ≤ ie; ++i) {
    if (Caller_pp_pp_accept_queue_[i].accept_token_→enumerated_id_ ≠
        NS_yacco2_T_enum::T_Enum::T_T_unquoted_string_) { /* bypass unq str
        */
        goto arbitrated_parameter;
    }
}
# include "war_end_code.h"
```

9. *Rrtn_component*'s subrule 1.



<Rrtn_component subrule 1 op directive 9> ≡ /* used rule and not defined, so create it */

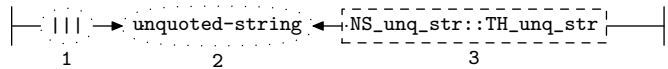
```
CAbs_lr1_sym * sym = new Err_not_T_for_rtned_token_from_th;
sym→set_rc(*rule_info→parser→start_token→, __FILE__, __LINE__);
RSVP(sym);
rule_info→parser→set_stop_parse(true);
```

10. *Rrtn_component*'s subrule 2.

```

⟨Rrtn_component subrule 2 op directive 10⟩ ≡
  Crtn_component * fsm = ( Crtn_component * ) rule_info...parser...fsm_tbl_;
  fsm-fnd_T_of_chr_str(sf-p2...c_string()-c_str(), sf-p2...); /* stop parsing in call */
  sf-p2...set_auto_delete(true);
  rule_info...parser...set_stop_parse(true);

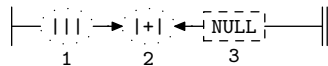
```

11. *Rrtn_component*'s subrule 3.

```

⟨Rrtn_component subrule 3 op directive 11⟩ ≡ /* only accept when no identifier */
  Crtn_component * fsm = ( Crtn_component * ) rule_info...parser...fsm_tbl_;
  fsm-fnd_T_of_chr_str(sf-p2...unquoted_string()-c_str(), sf-p2...); /* stop parsing in call */
  sf-p2...set_auto_delete(true);
  rule_info...parser...set_stop_parse(true);

```

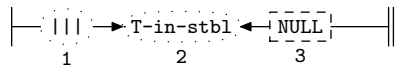
12. *Rrtn_component*'s subrule 4.

Forward the error.

```

⟨Rrtn_component subrule 4 op directive 12⟩ ≡
  RSVP(sf-p2...);
  rule_info...parser...set_stop_parse(true);

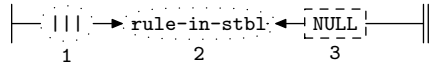
```

13. *Rrtn_component*'s subrule 5.

```

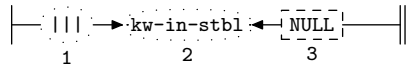
⟨Rrtn_component subrule 5 op directive 13⟩ ≡
  using namespace yacco2_stbl;
  T_sym_tbl_report_card report_card;
  find_sym_in_stbl(report_card, *sf-p2...t_def()-t_name()-c_str());
  if (report_card.action_ ≡ T_sym_tbl_report_card::not_fnd) {
    CAbs_lr1_sym * sym = new Err_T_not_in_stbl;
    sym-set_rc(*rule_info...parser...start_token_, __FILE__, __LINE__);
    RSVP(sym);
    rule_info...parser...set_stop_parse(true);
    return;
  }
  report_card.tbl_entry...used_ = true;
  refered_T * rt = new refered_T(*sf-p2...);
  rt-set_rc(*sf-p2..., __FILE__, __LINE__);
  sf-p2...add_T_into_xref(*rt);
  RSVP(sf-p2...);
  rule_info...parser...set_stop_parse(true);

```

14. *Rrtn_component's subrule 6.*

\langle Rrtn_component subrule 6 op directive 14 $\rangle \equiv$

```
CAbs_lr1_sym * sym = new Err_not_T_for_rtned_token_from_th;
sym->set_rc(*rule_info->parser->start_token--, __FILE__, __LINE__);
RSVP(sym);
rule_info->parser->set_stop_parse(true);
```

15. *Rrtn_component's subrule 7.*

\langle Rrtn_component subrule 7 op directive 15 $\rangle \equiv$

```
CAbs_lr1_sym * sym = new Err_not_T_for_rtned_token_from_th;
sym->set_rc(*rule_info->parser->start_token--, __FILE__, __LINE__);
RSVP(sym);
rule_info->parser->set_stop_parse(true);
return;
```

16. First Set Language for O_2^{linker} .

```
/*
  File: rtn_component.fsc
  Date and Time: Fri Jan  2 15:33:52 2015
*/
transitive      y
grammar-name    "rtn_component"
name-space     "NS_rtn_component"
thread-name    "TH_rtn_component"
monolithic     n
file-name      "rtn_component.fsc"
no-of-T        569
list-of-native-first-set-terminals 0
end-list-of-native-first-set-terminals
list-of-transitive-threads 3
  NS_unq_str::TH_unq_str
  NS_c_string::TH_c_string
  NS_identifier::TH_identifier
end-list-of-transitive-threads
list-of-used-threads 3
  NS_c_string::TH_c_string
  NS_identifier::TH_identifier
  NS_unq_str::TH_unq_str
end-list-of-used-threads
fsm-comments
"Recognizer of returned T from a thread call expression."
```

17. Lr1 State Network.

\Rightarrow					State: 1 state type: s			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
c	Rrtn_component		1 2 1		c-string NS.c_string::TH.c_string		1 2 4	
c	Rrtn_component		1 3 1		unquoted-string NS.unq_str::TH.unq_str		1 2 5	
c	Rrtn_component		1 5 1		T-in-stbl NULL		1 2 7	
c	Rrtn_component		1 1 1		identifier NS.identifier::TH.identifier		1 2 6	
c	Rrtn_component		1 4 1		+ NULL		1 2 3	
c	Rrtn_component		1 6 1		rule-in-stbl NULL		1 2 8	
c	Rrtn_component		1 7 1		kw-in-stbl NULL		1 2 9	
\Rightarrow	<i>arbitration-code: AR_Rrtn_component</i>				State: 2 state type: s			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 4 2	+			1 3 3	
t	Rrtn_component		1 2 2	c-string			1 4 4	
t	Rrtn_component		1 3 2	unquoted-string			1 5 5	
t	Rrtn_component		1 1 2	identifier			1 6 6	
t	Rrtn_component		1 5 2	T-in-stbl			1 7 7	
t	Rrtn_component		1 6 2	rule-in-stbl			1 8 8	
t	Rrtn_component		1 7 2	kw-in-stbl			1 9 9	
\Rightarrow	+				State: 3 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 4 3				1 0 3 1	
\Rightarrow	<i>c-string</i>				State: 4 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 2 3				1 0 4 1	
\Rightarrow	<i>unquoted-string</i>				State: 5 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 3 3				1 0 5 1	
\Rightarrow	<i>identifier</i>				State: 6 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 1 3				1 0 6 1	
\Rightarrow	<i>T-in-stbl</i>				State: 7 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 5 3				1 0 7 1	
\Rightarrow	<i>rule-in-stbl</i>				State: 8 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 6 3				1 0 8 1	
\Rightarrow	<i>kw-in-stbl</i>				State: 9 state type: r			
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA	
t	Rrtn_component		1 7 3				1 0 9 1	

18. Index.

|+|: 7.
 |||: 7.
 __FILE__: 5, 9, 13, 14, 15.
 __LINE__: 5, 9, 13, 14, 15.
 accept_token_: 8.
 action_: 5, 13.
 add_T_into_xref: 5, 13.
 AR_Rrtn_component: 17.
 arbitrated_parameter: 8.
 c-string: 7.
 c_str: 10, 11, 13.
 c_string: 10.
 CAbs_lr1_sym: 4, 5, 9, 13, 14, 15.
 Caller_pp: 8.
 Crtn_component: 5, 10, 11.
 enumerated_id_: 8.
 Err_not_T_for_rtned_token_from_th: 5, 9, 14, 15.
 Err_subrule_use_undefined_T: 5.
 Err_T_not_in_stbl: 13.
 find_sym_in_stbl: 5, 13.
 fnd_T_of_chr_str: 4, 5, 10, 11.
 fsm: 10, 11.
 fsm_tbl_: 10, 11.
 identifier: 7.
 ie: 8.
 kw-in-stbl: 7.
 not_fnd: 5, 13.
 NS_c_string::TH_c_string: 7.
 NS_identifier::TH_identifier: 7.
 NS_unq_str::TH_unq_str: 7.
 NS_yacco2_T_enum: 8.
 NULL: 7.
 parser_: 5, 9, 10, 11, 12, 13, 14, 15.
 pp_accept_queue_: 8.
 p2_: 10, 11, 12, 13.
 refered_T: 5, 13.
 report_card: 5, 13.
 Rrtn_component: 7, 9, 10, 11, 12, 13, 14, 15.
 RSVP: 9, 12, 13, 14, 15.
 RSVP_FSM: 5.
 rt: 13.
 rT: 5.
 rtn_component: 2.
 rule-in-stbl: 7.
 rule_info_: 9, 10, 11, 12, 13, 14, 15.
 set_auto_delete: 10, 11.
 set_rc: 5, 9, 13, 14, 15.
 set_stop_parse: 5, 9, 10, 11, 12, 13, 14, 15.
 sf: 10, 11, 12, 13.
 start_token_: 5, 9, 13, 14, 15.
 Str: 4, 5.
 sym: 5, 9, 13, 14, 15.
 symbol_: 5.
 T-in-stbl: 7.
 t_def: 13.
 T_Enum: 8.
 T_in_stbl: 5.
 t_name: 13.
 T_sym_tbl_report_card: 5, 13.
 T_T_unquoted_string_: 8.
 table_entry: 5.
 tbl_entry_: 5, 13.
 terminal: 5.
 Tok: 4, 5.
 true: 5, 9, 10, 11, 12, 13, 14, 15.
 type_: 5.
 unquoted-string: 7.
 unquoted_string: 11.
 used_: 5, 13.
 yacco2: 4, 5.
 yacco2_stbl: 5, 13.

⟨ Crtn_component user-declaration directive 4 ⟩
⟨ Crtn_component user-implementation directive 5 ⟩
⟨ Crtn_component user-prefix-declaration directive 6 ⟩
⟨ Rrtn_component arbitrator-code directive 8 ⟩
⟨ Rrtn_component subrule 1 op directive 9 ⟩
⟨ Rrtn_component subrule 2 op directive 10 ⟩
⟨ Rrtn_component subrule 3 op directive 11 ⟩
⟨ Rrtn_component subrule 4 op directive 12 ⟩
⟨ Rrtn_component subrule 5 op directive 13 ⟩
⟨ Rrtn_component subrule 6 op directive 14 ⟩
⟨ Rrtn_component subrule 7 op directive 15 ⟩

rtn_component Grammar

Date: January 2, 2015 at 15:38

File: rtn_component.lex Ns: NS_rtn_component

Version: 1.0 Debug: false

Grammar Comments: Type: Thread

Recognizer of returned T from a thread call expression.

1 element(s) in Lookahead Expression below

eolr

	Section	Page
Copyright	1	1
<i>rtn_component</i> Thread	2	2
Fsm Crtn_component class	3	2
Crtn_component user-declaration directive	4	2
Crtn_component user-implementation directive	5	2
Crtn_component user-prefix-declaration directive	6	2
<i>Rrtn_component</i> rule	7	3
Rrtn_component arbitrator-code directive	8	3
<i>Rrtn_component</i> 's subrule 1	9	3
<i>Rrtn_component</i> 's subrule 2	10	4
<i>Rrtn_component</i> 's subrule 3	11	4
<i>Rrtn_component</i> 's subrule 4	12	4
<i>Rrtn_component</i> 's subrule 5	13	4
<i>Rrtn_component</i> 's subrule 6	14	5
<i>Rrtn_component</i> 's subrule 7	15	5
First Set Language for O_2^{linker}	16	6
Lr1 State Network	17	7
Index	18	8