

1. Copyright.

Copyright © Dave Bone 1998 - 2015

2. *rules_phrase_th* Thread.

Parse all of the grammar's rules. The *rule_def_phrase* adds each rule defined to O_2 's global symbol table.

3. Fsm Crules_phrase_th class.**4. Crules_phrase_th constructor directive.**

```
< Crules_phrase_th constructor directive 4 > ≡
  rules_phrase_ = 0;
  end_rules_t_ = 0;
```

5. Crules_phrase_th op directive.

```
< Crules_phrase_th op directive 5 > ≡
  if (rules_phrase_ ≠ 0) {
    delete rules_phrase_;
    rules_phrase_ = 0;
    end_rules_t_ = 0;
  }
  rules_phrase_ = new T_rules_phrase;
  rules_phrase_→set_rc(*parser_→start_token_, __FILE__, __LINE__);
  AST *t = new AST(*rules_phrase_);
  rules_phrase_→phrase_tree(t);
```

6. Crules_phrase_th user-declaration directive.

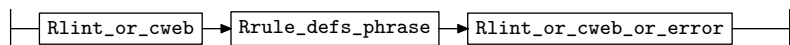
```
< Crules_phrase_th user-declaration directive 6 > ≡
public: T_rules_phrase * rules_phrase_;
  AST * end_rules_t_;
```

7. Crules_phrase_th user-prefix-declaration directive.

```
< Crules_phrase_th user-prefix-declaration directive 7 > ≡
#include "ws.h"
#include "eol.h"
#include "c_comments.h"
#include "cweb_or_c_k.h"
#include "rule_def_phrase.h"
#include "o2_externs.h"
```

8. Rrules_phrase rule.

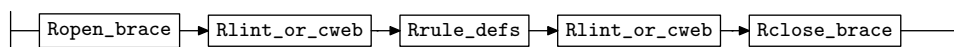
Rrules_phrase



```
< Rrules_phrase subrule 1 op directive 8 > ≡
  Crules_phrase_th * fsm = ( Crules_phrase_th * ) rule_info_→parser_→fsm_tbl_;
  RSVP(fsm→rules_phrase_);
  fsm→rules_phrase_ = 0;
```

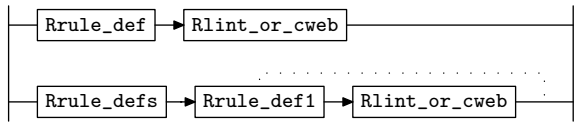
9. Rrule_defs_phrase rule.

Rrule_defs_phrase



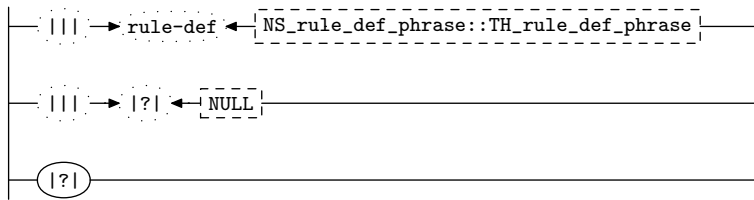
10. *Rrule_defs* rule.

Rrule_defs

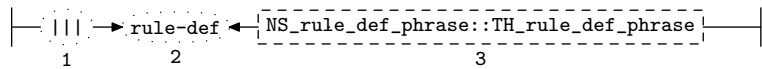


11. *Rrule_def* rule.

Rrule_def



12. *Rrule_def*'s subrule 1.

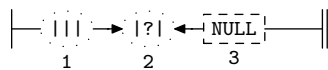


⟨Rrule_def subrule 1 op directive 12⟩ ≡

```

Crules_phrase_th * fsm = ( Crules_phrase_th * ) rule_info...parser...fsm_tbl...;
CAbs_lr1_sym * r = fsm->rules_phrase->add_r_to_alphabet(sf-p2..., rule_info...parser...);
if (r ≠ 0) {
  RSVP(r);
  rule_info...parser...->set_stop_parse(true);
  return;
}
AST * rt = sf-p2...->rule_s_tree(); /* ms has a bug so use temporary */
AST * t = fsm->rules_phrase->phrase_tree();
if (fsm->end_rules_t_ ≡ 0) {
  AST::join_pts(*t, *rt);
}
else {
  AST::join_sts(*fsm->end_rules_t_, *rt);
}
fsm->end_rules_t_ = rt;
  
```

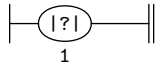
13. *Rrule_def*'s subrule 2.



⟨Rrule_def subrule 2 op directive 13⟩ ≡

```

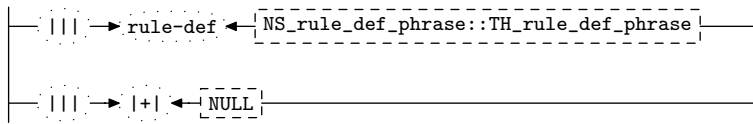
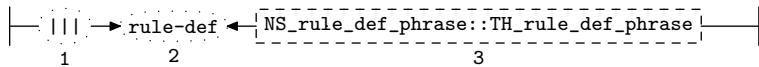
RSVP(sf-p2...);
rule_info...parser...->set_stop_parse(true);
  
```

14. *Rrule_def*'s subrule 3.

⟨*Rrule_def* subrule 3 op directive 14⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_no_rule_name_present*;
sym→*set_rc*(**rule_info*→*parser*→*current_token*(), *__FILE__*, *__LINE__*);
RSVP(*sym*);
rule_info→*parser*→*set_stop_parse*(*true*);

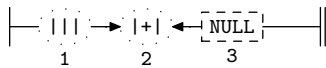
15. *Rrule_def1* rule.

Rrule_def1

16. *Rrule_def1*'s subrule 1.

⟨*Rrule_def1* subrule 1 op directive 16⟩ ≡
Crules_phrase_th * *fsm* = (*Crules_phrase_th* *) *rule_info*→*parser*→*fsm_tbl*→;
CAbs_lr1_sym * *r* = *fsm*→*rules_phrase*→*add_r_to_alphabet*(*sf*→*p2*→, *rule_info*→*parser*→);
if (*r* ≠ 0) {
 RSVP(*r*);
 rule_info→*parser*→*set_stop_parse*(*true*);
 return;
}

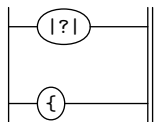
AST * *rt* = *sf*→*p2*→*rule_s_tree*();
AST::*join_sts*(**fsm*→*end_rules_t*→, **rt*);
fsm→*end_rules_t* = *rt*;

17. *Rrule_def1*'s subrule 2.

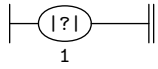
⟨*Rrule_def1* subrule 2 op directive 17⟩ ≡
RSVP(*sf*→*p2*→);
rule_info→*parser*→*set_stop_parse*(*true*);

18. *Ropen_brace* rule.

Ropen_brace



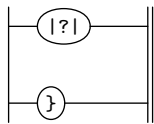
19. Ropen_brace's subrule 1.



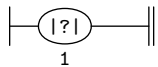
⟨ Ropen_brace subrule 1 op directive 19 ⟩ ≡
`CAbs_lr1_sym * sym = new Err_no_open_brace;`
`sym->set_rc(*rule_info...parser->current_token(), __FILE__, __LINE__);`
`RSVP(sym);`
`rule_info...parser->set_stop_parse(true);`

20. Rclose_brace rule.

Rclose_brace



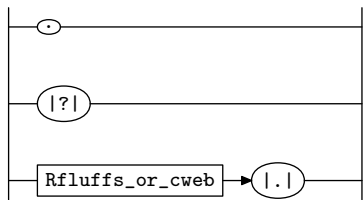
21. Rclose_brace's subrule 1.



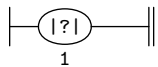
⟨ Rclose_brace subrule 1 op directive 21 ⟩ ≡
`CAbs_lr1_sym * sym = new Err_no_close_brace_ending_rules_defs;`
`sym->set_rc(*rule_info...parser->current_token(), __FILE__, __LINE__);`
`RSVP(sym);`
`rule_info...parser->set_stop_parse(true);`

22. Rlint_or_cweb_or_error rule.

Rlint_or_cweb_or_error



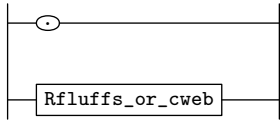
23. Rlint_or_cweb_or_error's subrule 2.



⟨ Rlint_or_cweb_or_error subrule 2 op directive 23 ⟩ ≡
`CAbs_lr1_sym * sym = new Err_improper_closing_of_rules;`
`sym->set_rc(*rule_info...parser->current_token(), __FILE__, __LINE__);`
`RSVP(sym);`
`rule_info...parser->set_stop_parse(true);`

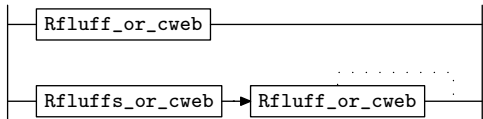
24. *Rlint_or_cweb* rule.

Rlint_or_cweb



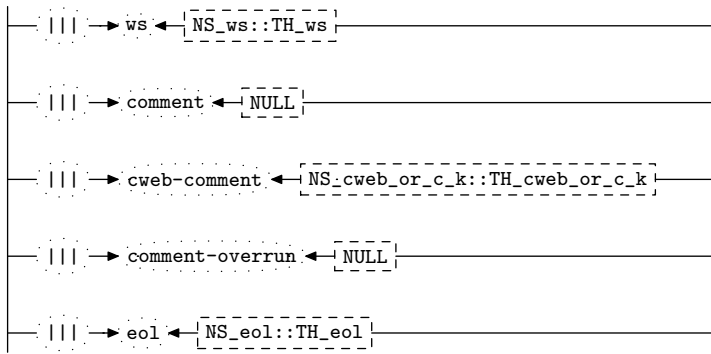
25. *Rfluffs_or_cweb* rule.

Rfluffs_or_cweb

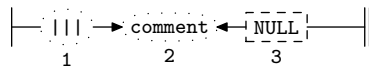


26. *Rfluff_or_cweb* rule.

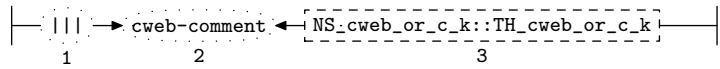
Rfluff_or_cweb



27. *Rfluff_or_cweb*'s subrule 2.



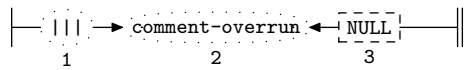
$\langle \text{Rfluff_or_cweb subrule 2 op directive 27} \rangle \equiv$
 $sf \rightarrow p2 \rightarrow set_auto_delete(true);$

28. Rfluff_or_cweb's subrule 3.

```

⟨ Rfluff_or_cweb subrule 3 op directive 28 ⟩ ≡
  Crules_phrase_th * fsm = ( Crules_phrase_th * ) rule_info__parser__fsm_tbl__;
  AST * t = fsm_rules_phrase_phrase_tree();
  T_cweb_comment * k = sf_p2__;
  AST * cwebk_t_ = new AST(*k);
  AST * rt = new AST();
  T_cweb_marker * cw = new T_cweb_marker(rt);
  cw_set_rc(*k, __FILE__, __LINE__);
  AST::set_content(*rt, *cw);
  AST::join_pts(*rt, *cwebk_t_);
  if (fsm_end_rules_t_ ≡ 0) {
    AST::join_pts(*t, *rt);
  }
  else {
    AST::join_sts(*fsm_end_rules_t_, *rt);
  }
  fsm_end_rules_t_ = rt;

```

29. Rfluff_or_cweb's subrule 4.

```

⟨ Rfluff_or_cweb subrule 4 op directive 29 ⟩ ≡
  RSVP(sf_p2__);
  rule_info__parser__set_stop_parse(true);

```

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

```
/*
  File: rules_phrase_th.fsc
  Date and Time: Fri Jan  2 15:33:54 2015
*/
transitive      y
grammar-name    "rules_phrase_th"
name-space     "NS_rules_phrase_th"
thread-name    "TH_rules_phrase_th"
monolithic     n
file-name      "rules_phrase_th.fsc"
no-of-T        569
list-of-native-first-set-terminals 2
  LR1_questionable_shift_operator
  raw_open_brace
end-list-of-native-first-set-terminals
list-of-transitive-threads 3
  NS_eol::TH_eol
  NS_ws::TH_ws
  NS_cweb_or_c_k::TH_cweb_or_c_k
end-list-of-transitive-threads
list-of-used-threads 4
  NS_cweb_or_c_k::TH_cweb_or_c_k
  NS_eol::TH_eol
  NS_rule_def_phrase::TH_rule_def_phrase
  NS_ws::TH_ws
end-list-of-used-threads
fsm-comments
"Parse all of the grammar's rules."
```


31. Lr1 State Network.

⇒					State: 1 state type: s/r				
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
c	Rlint_or_cweb		9	1	1	←	ε		1 0 1 1
c	Rfluff_or_cweb		11	1	1	←	ws NS_ws::TH_ws		1 2 5
c	Rfluff_or_cweb		11	3	1	←	cweb-comment NS_cweb_or_c_k::TH_cweb_or_c_k		1 2 6
c	Rfluff_or_cweb		11	2	1	←	comment NULL		1 2 4
c	Rfluff_or_cweb		11	4	1	←	comment-overrun NULL		1 2 7
c	Rfluff_or_cweb		11	5	1	←	eol NS_eol::TH_eol		1 2 3
c	Rrules_phrase		1	1	1	←	Rlint_or_cweb <u>Rrule_defs_phrase</u>		1 8 13
c	Rfluffs_or_cweb		10	2	1	←	Rfluffs_or_cweb <u>Rfluff_or_cweb</u>		1 28 16
c	Rlint_or_cweb		9	2	1	←	Rfluffs_or_cweb		1 28 28
c	Rfluffs_or_cweb		10	1	1	←	Rfluff_or_cweb		1 17 17
⇒	arbitration-code: ε						State: 2 state type: s		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
t	Rfluff_or_cweb		11	5	2	←	eol		1 3 3
t	Rfluff_or_cweb		11	2	2	←	comment		1 4 4
t	Rfluff_or_cweb		11	1	2	←	ws		1 5 5
t	Rfluff_or_cweb		11	3	2	←	cweb-comment		1 6 6
t	Rfluff_or_cweb		11	4	2	←	comment-overrun		1 7 7
⇒	eol						State: 3 state type: r		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
t	Rfluff_or_cweb		11	5	3	←			1 0 3 2
⇒	comment						State: 4 state type: r		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
t	Rfluff_or_cweb		11	2	3	←			1 0 4 2
⇒	ws						State: 5 state type: r		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
t	Rfluff_or_cweb		11	1	3	←			1 0 5 2
⇒	cweb-comment						State: 6 state type: r		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
t	Rfluff_or_cweb		11	3	3	←			1 0 6 2
⇒	comment-overrun						State: 7 state type: r		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
t	Rfluff_or_cweb		11	4	3	←			1 0 7 2
⇒	Rlint_or_cweb						State: 8 state type: s		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA
c	Ropen_brace		6	1	1	←	?		8 9 9
c	Ropen_brace		6	2	1	←	{		8 10 10
t	Rrules_phrase		1	1	2	←	Rrule_defs_phrase <u>Rlint_or_cweb_or_error^ε</u>		1 11 13
c	Rrule_defs_phrase		2	1	1	←	Ropen_brace <u>Rlint_or_cweb^ε</u> <u>Rrule_defs</u>		8 18 27
⇒	?						State: 9 state type: r		
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn Gto Red LA

t Ropen_brace	6	1	2					8	0	9	3
$\Rightarrow \{$											
← rule	→ R#	sr#	Po	←		State: 10 state type: r					
						subrule element		→ Brn	Gto	Red	LA
t Ropen_brace	6	2	2					8	0	10	3
$\Rightarrow Rrule_defs_phrase$											
← rule	→ R#	sr#	Po	←		State: 11 state type: s/r					
						subrule element		→ Brn	Gto	Red	LA
c Rlint_or_cweb_or_error	8	1	1	ε				11	0	11	4
c Rlint_or_cweb_or_error	8	2	1	?				11	12	12	
c Rfluff_or_cweb	11	1	1	ws NS_ws::TH_ws				11	2	5	
c Rfluff_or_cweb	11	3	1	cweb-comment NS_cweb_or_c_k::TH_cweb_or_c_k				11	2	6	
c Rfluff_or_cweb	11	2	1	comment NULL				11	2	4	
c Rfluff_or_cweb	11	4	1	comment-overrun NULL				11	2	7	
c Rfluff_or_cweb	11	5	1	eol NS_eol::TH_eol				11	2	3	
t Rrules_phrase	1	1	3	Rlint_or_cweb_or_error				1	13	13	
c Rlint_or_cweb_or_error	8	3	1	Rfluffs_or_cweb .				11	14	15	
c Rfluffs_or_cweb	10	2	1	Rfluffs_or_cweb <u>Rfluff_or_cweb</u>				11	14	16	
c Rfluffs_or_cweb	10	1	1	Rfluff_or_cweb				11	17	17	
$\Rightarrow ? $											
← rule	→ R#	sr#	Po	←		State: 12 state type: r					
						subrule element		→ Brn	Gto	Red	LA
t Rlint_or_cweb_or_error	8	2	2					11	0	12	4
$\Rightarrow Rlint_or_cweb_or_error$											
← rule	→ R#	sr#	Po	←		State: 13 state type: r					
						subrule element		→ Brn	Gto	Red	LA
t Rrules_phrase	1	1	4					1	0	13	4
$\Rightarrow Rfluffs_or_cweb$											
← rule	→ R#	sr#	Po	←		State: 14 state type: s					
						subrule element		→ Brn	Gto	Red	LA
c Rfluff_or_cweb	11	1	1	ws NS_ws::TH_ws				14	2	5	
c Rfluff_or_cweb	11	3	1	cweb-comment NS_cweb_or_c_k::TH_cweb_or_c_k				14	2	6	
c Rfluff_or_cweb	11	2	1	comment NULL				14	2	4	
c Rfluff_or_cweb	11	4	1	comment-overrun NULL				14	2	7	
c Rfluff_or_cweb	11	5	1	eol NS_eol::TH_eol				14	2	3	
t Rlint_or_cweb_or_error	8	3	2	.				11	15	15	
t Rfluffs_or_cweb	10	2	2	Rfluff_or_cweb				11	16	16	
$\Rightarrow . $											
← rule	→ R#	sr#	Po	←		State: 15 state type: r					
						subrule element		→ Brn	Gto	Red	LA
t Rlint_or_cweb_or_error	8	3	3					11	0	15	4
$\Rightarrow Rfluff_or_cweb$											
← rule	→ R#	sr#	Po	←		State: 16 state type: r					
						subrule element		→ Brn	Gto	Red	LA
t Rfluffs_or_cweb	10	2	3					11	0	16	2
$\Rightarrow Rfluff_or_cweb$											
← rule	→ R#	sr#	Po	←		State: 17 state type: r					
						subrule element		→ Brn	Gto	Red	LA
t Rfluffs_or_cweb	10	1	2					11	0	17	2
$\Rightarrow Ropen_brace$											
← rule	→ R#	sr#	Po	←		State: 18 state type: s/r					
						subrule element		→ Brn	Gto	Red	LA

c	Rlint_or_cweb	9	1	1	ε			18	0	18	3
c	Rfluff_or_cweb	11	1	1	ws NS_ws::TH_ws			18	2	5	
c	Rfluff_or_cweb	11	3	1	cweb-comment NS_cweb_or_c.k::TH_cweb_or_c.k			18	2	6	
c	Rfluff_or_cweb	11	2	1	comment NULL			18	2	4	
c	Rfluff_or_cweb	11	4	1	comment-overflow NULL			18	2	7	
c	Rfluff_or_cweb	11	5	1	eol NS_eol::TH_eol			18	2	3	
t	Rrule_defs_phrase	2	1	2	Rlint_or_cweb <u>Rrule_defs</u>			8	19	27	
c	Rfluffs_or_cweb	10	2	1	Rfluffs_or_cweb <u>Rfluff_or_cweb</u>			18	28	16	
c	Rlint_or_cweb	9	2	1	Rfluffs_or_cweb			18	28	28	
c	Rfluffs_or_cweb	10	1	1	Rfluff_or_cweb			18	17	17	

⇒ *Rlint_or_cweb*State: 19 state type: ^s

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rrule_def		4	3	1		rule-def NS_rule_def_phrase::TH_rule_def_phrase		19	29	29	
c	Rrule_def		4	1	1		rule-def NS_rule_def_phrase::TH_rule_def_phrase		19	30	32	
c	Rrule_def		4	2	1		? NULL		19	30	31	
t	Rrule_defs_phrase		2	1	3	Rrule_defs	<u>Rlint_or_cweb^ε</u> <u>Rclose_brace</u>		8	20	27	
c	Rrule_defs		3	2	1	Rrule_defs	<u>Rrule_def1</u>		19	20	34	
c	Rrule_defs		3	1	1	Rrule_def	<u>Rlint_or_cweb^ε</u>		19	35	36	

⇒ *Rrule_defs*State: 20 state type: ^{s/r}

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rlint_or_cweb		9	1	1	ε			20	0	20	5
c	Rfluff_or_cweb		11	1	1	ws NS_ws::TH_ws			20	21	5	
c	Rfluff_or_cweb		11	3	1	cweb-comment NS_cweb_or_c.k::TH_cweb_or_c.k			20	21	6	
c	Rrule_def1		5	1	1	rule-def NS_rule_def_phrase::TH_rule_def_phrase			20	21	23	
c	Rrule_def1		5	2	1		+ NULL		20	21	22	
c	Rfluff_or_cweb		11	2	1	comment NULL			20	21	4	
c	Rfluff_or_cweb		11	4	1	comment-overflow NULL			20	21	7	
c	Rfluff_or_cweb		11	5	1	eol NS_eol::TH_eol			20	21	3	
t	Rrule_defs		3	2	2	Rrule_def1	<u>Rlint_or_cweb^ε</u>		19	33	34	
t	Rrule_defs_phrase		2	1	4	Rlint_or_cweb	<u>Rclose_brace</u>		8	24	27	
c	Rfluffs_or_cweb		10	2	1	Rfluffs_or_cweb	<u>Rfluff_or_cweb</u>		20	28	16	
c	Rlint_or_cweb		9	2	1	Rfluffs_or_cweb			20	28	28	
c	Rfluffs_or_cweb		10	1	1	Rfluff_or_cweb			20	17	17	

⇒ *||| arbitration-code: ε*State: 21 state type: ^s

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rrule_def1		5	2	2		+		20	22	22	
t	Rfluff_or_cweb		11	5	2	eol			20	3	3	
t	Rfluff_or_cweb		11	2	2	comment			20	4	4	
t	Rfluff_or_cweb		11	1	2	ws			20	5	5	
t	Rrule_def1		5	1	2	rule-def			20	23	23	
t	Rfluff_or_cweb		11	3	2	cweb-comment			20	6	6	
t	Rfluff_or_cweb		11	4	2	comment-overflow			20	7	7	

⇒ *|+|*State: 22 state type: ^r

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rrule_def1		5	2	3				20	0	22	6

⇒ *rule-def*State: 23 state type: ^r

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
---	------	---	----	-----	----	---	-----------------	---	-----	-----	-----	----

t Rrule_def1	5	1	3					20	0	23	6
\Rightarrow <i>Rlint_or_cweb</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
c Rclose_brace	7	1	1	?				24	25	25	
c Rclose_brace	7	2	1	}				24	26	26	
t Rrule_defs_phrase	2	1	5	Rclose_brace				8	27	27	
\Rightarrow <i> ? </i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rclose_brace	7	1	2					24	0	25	4
\Rightarrow <i>}</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rclose_brace	7	2	2					24	0	26	4
\Rightarrow <i>Rclose_brace</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rrule_defs_phrase	2	1	6					8	0	27	4
\Rightarrow <i>Rfluffs_or_cweb</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rlint_or_cweb	9	2	2					20	0	28	5
c Rfluff_or_cweb	11	1	1	ws NS_ws::TH_ws				28	2	5	
c Rfluff_or_cweb	11	3	1	cweb-comment NS_cweb_or_c_k::TH_cweb_or_c_k				28	2	6	
c Rfluff_or_cweb	11	2	1	comment NULL				28	2	4	
c Rfluff_or_cweb	11	4	1	comment-overrun NULL				28	2	7	
c Rfluff_or_cweb	11	5	1	eol NS_eol::TH_eol				28	2	3	
t Rfluffs_or_cweb	10	2	2	Rfluff_or_cweb				20	16	16	
\Rightarrow <i> ? </i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rrule_def	4	3	2					19	0	29	6
\Rightarrow <i> arbitration-code: ε</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rrule_def	4	2	2	?				19	31	31	
t Rrule_def	4	1	2	rule-def				19	32	32	
\Rightarrow <i> ? </i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rrule_def	4	2	3					19	0	31	6
\Rightarrow <i>rule-def</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
t Rrule_def	4	1	3					19	0	32	6
\Rightarrow <i>Rrule_def1</i>											
← rule	→ R#	sr#	Po	←				→ Brn	Gto	Red	LA
c Rlint_or_cweb	9	1	1	ε				33	0	33	6
c Rfluff_or_cweb	11	1	1	ws NS_ws::TH_ws				33	2	5	
c Rfluff_or_cweb	11	3	1	cweb-comment NS_cweb_or_c_k::TH_cweb_or_c_k				33	2	6	

c	Rfluff_or_cweb	11	2	1	comment NULL	33	2	4
c	Rfluff_or_cweb	11	4	1	comment-overflow NULL	33	2	7
c	Rfluff_or_cweb	11	5	1	eol NS_eol::TH_eol	33	2	3
t	Rrule.defs	3	2	3	Rlint_or_cweb	19	34	34
c	Rfluffs_or_cweb	10	2	1	Rfluffs_or_cweb <u>Rfluff_or_cweb</u>	33	28	16
c	Rlint_or_cweb	9	2	1	Rfluffs_or_cweb	33	28	28
c	Rfluffs_or_cweb	10	1	1	Rfluff_or_cweb	33	17	17

⇒ *Rlint_or_cweb*

State: 34 state type: *r*

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rrule.defs		3	2	4				19	0	34	6

⇒ *Rrule_def*

State: 35 state type: *s/r*

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rlint_or_cweb		9	1	1	ε			35	0	35	6
c	Rfluff_or_cweb		11	1	1	ws NS_ws::TH_ws			35	2	5	
c	Rfluff_or_cweb		11	3	1	cweb-comment NS_cweb_or_c.k::TH_cweb_or_c.k			35	2	6	
c	Rfluff_or_cweb		11	2	1	comment NULL			35	2	4	
c	Rfluff_or_cweb		11	4	1	comment-overflow NULL			35	2	7	
c	Rfluff_or_cweb		11	5	1	eol NS_eol::TH_eol			35	2	3	
t	Rrule.defs		3	1	2	Rlint_or_cweb			19	36	36	
c	Rfluffs_or_cweb		10	2	1	Rfluffs_or_cweb <u>Rfluff_or_cweb</u>			35	28	16	
c	Rlint_or_cweb		9	2	1	Rfluffs_or_cweb			35	28	28	
c	Rfluffs_or_cweb		10	1	1	Rfluff_or_cweb			35	17	17	

⇒ *Rlint_or_cweb*

State: 36 state type: *r*

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rrule.defs		3	1	3				19	0	36	6

32. Index.

|+|: 15.
 € : 22, 24.
 |.|: 22.
 |||: 11, 15, 26.
 |?|: 11, 18, 20, 22.
 __FILE__: 5, 14, 19, 21, 23, 28.
 __LINE__: 5, 14, 19, 21, 23, 28.
 add_r_to_alphabet: 12, 16.
 AST: 5, 6, 12, 16, 28.
 CAbs_lr1_sym: 12, 14, 16, 19, 21, 23.
 comment: 26.
 comment-overrun: 26.
 Crules_phrase_th: 8, 12, 16, 28.
 current_token: 14, 19, 21, 23.
 cw: 28.
 cweb-comment: 26.
 cwebk_t: 28.
 end_rules_t: 4, 5, 6, 12, 16, 28.
 eol: 26.
 Err_improper_closing_of_rules: 23.
 Err_no_close_brace_ending_rules_defs: 21.
 Err_no_open_brace: 19.
 Err_no_rule_name_present: 14.
 fsm: 8, 12, 16, 28.
 fsm_tbl_: 8, 12, 16, 28.
 join_pts: 12, 28.
 join_sts: 12, 16, 28.
 NS_cweb_or_c_k::TH_cweb_or_c_k: 26.
 NS_eol::TH_eol: 26.
 NS_rule_def_phrase::TH_rule_def_phrase: 11, 15.
 NS_ws::TH_ws: 26.
 NULL: 11, 15, 26.
 parser_: 5, 8, 12, 13, 14, 16, 17, 19, 21, 23, 28, 29.
 phrase_tree: 5, 12, 28.
 p2_: 12, 13, 16, 17, 27, 28, 29.
 Rclose_brace: 9.
 Rclose_brace: 20, 21.
 Rfluff_or_cweb: 25.
 Rfluff_or_cweb: 26, 27, 28, 29.
 Rfluffs_or_cweb: 22, 24, 25.
 Rfluffs_or_cweb: 25.
 Rlint_or_cweb: 8, 9, 10.
 Rlint_or_cweb_or_error: 8.
 Rlint_or_cweb: 24.
 Rlint_or_cweb_or_error: 22, 23.
 Ropen_brace: 9.
 Ropen_brace: 18, 19.
 Rrule_def: 10.
 Rrule_defs: 9, 10.
 Rrule_defs_phrase: 8.
 Rrule_def1: 10.
 Rrule_def: 11, 12, 13, 14.
 Rrule_defs: 10.
 Rrule_defs_phrase: 9.
 Rrule_def1: 15, 16, 17.
 Rrules_phrase: 8.
 RSVP: 8, 12, 13, 14, 16, 17, 19, 21, 23, 29.
 rt: 12, 16, 28.
 rule-def: 11, 15.
 rule_def_phrase: 2.
 rule_info_: 8, 12, 13, 14, 16, 17, 19, 21, 23, 28, 29.
 rule_s_tree: 12, 16.
 rules_phrase_: 4, 5, 6, 8, 12, 16, 28.
 rules_phrase_th: 2.
 set_auto_delete: 27.
 set_content: 28.
 set_rc: 5, 14, 19, 21, 23, 28.
 set_stop_parse: 12, 13, 14, 16, 17, 19, 21, 23, 29.
 sf: 12, 13, 16, 17, 27, 28, 29.
 start_token_: 5.
 sym: 14, 19, 21, 23.
 T_cweb_comment: 28.
 T_cweb_marker: 28.
 T_rules_phrase: 5, 6.
 true: 12, 13, 14, 16, 17, 19, 21, 23, 27, 29.
 ws: 26.

⟨ Crules_phrase.th constructor directive 4 ⟩
⟨ Crules_phrase.th op directive 5 ⟩
⟨ Crules_phrase.th user-declaration directive 6 ⟩
⟨ Crules_phrase.th user-prefix-declaration directive 7 ⟩
⟨ Rclose_brace subrule 1 op directive 21 ⟩
⟨ Rfluff_or_cweb subrule 2 op directive 27 ⟩
⟨ Rfluff_or_cweb subrule 3 op directive 28 ⟩
⟨ Rfluff_or_cweb subrule 4 op directive 29 ⟩
⟨ Rlint_or_cweb_or_error subrule 2 op directive 23 ⟩
⟨ Ropen_brace subrule 1 op directive 19 ⟩
⟨ Rrule_def subrule 1 op directive 12 ⟩
⟨ Rrule_def subrule 2 op directive 13 ⟩
⟨ Rrule_def subrule 3 op directive 14 ⟩
⟨ Rrule_def1 subrule 1 op directive 16 ⟩
⟨ Rrule_def1 subrule 2 op directive 17 ⟩
⟨ Rrules_phrase subrule 1 op directive 8 ⟩

rules_phrase_th Grammar

Date: January 2, 2015 at 15:39

File: rules_phrase_th.lex Ns: NS_rules_phrase_th

Version: 1.0

Debug: false

Grammar Comments:

Type: Thread

Parse all of the grammar's rules.

1 element(s) in Lookahead Expression below

colr

	Section	Page
Copyright	1	1
<i>rules_phrase_th</i> Thread	2	2
Fsm Crules_phrase_th class	3	2
Crules_phrase_th constructor directive	4	2
Crules_phrase_th op directive	5	2
Crules_phrase_th user-declaration directive	6	2
Crules_phrase_th user-prefix-declaration directive	7	2
<i>Rrules_phrase</i> rule	8	2
<i>Rrule_defs_phrase</i> rule	9	2
<i>Rrule_defs</i> rule	10	3
<i>Rrule_def</i> rule	11	3
<i>Rrule_def</i> 's subrule 1	12	3
<i>Rrule_def</i> 's subrule 2	13	3
<i>Rrule_def</i> 's subrule 3	14	4
<i>Rrule_def1</i> rule	15	4
<i>Rrule_def1</i> 's subrule 1	16	4
<i>Rrule_def1</i> 's subrule 2	17	4
<i>Ropen_brace</i> rule	18	4
<i>Ropen_brace</i> 's subrule 1	19	5
<i>Rclose_brace</i> rule	20	5
<i>Rclose_brace</i> 's subrule 1	21	5
<i>Rlint_or_cweb_or_error</i> rule	22	5
<i>Rlint_or_cweb_or_error</i> 's subrule 2	23	5
<i>Rlint_or_cweb</i> rule	24	6
<i>Rfluff_or_cweb</i> rule	25	6
<i>Rfluff_or_cweb</i> rule	26	6
<i>Rfluff_or_cweb</i> 's subrule 2	27	6
<i>Rfluff_or_cweb</i> 's subrule 3	28	7
<i>Rfluff_or_cweb</i> 's subrule 4	29	7
First Set Language for O_2^{linker}	30	8
Lr1 State Network	31	9
Index	32	14