

**1. Copyright.**

Copyright © Dave Bone 1998 - 2015

**2. *parallel\_parser\_phrase\_th* thread.**

Parse parallel-parser phrase.

Example of a parallel-parser construct to parse:

```

/*
file: pprsr.txt
Why: text example parallel parser construct.
*/
parallel-parser
(
  parallel-thread-function
  TH_angled_string
  ***
  parallel-la-boundary
  eolr - ||| - |+| - ">"
  ***
)

```

**3. Fsm *Cparallel\_parser\_ph\_th* class.****4. *Cparallel\_parser\_ph\_th* constructor directive.**

⟨*Cparallel\_parser\_ph\_th* constructor directive 4⟩ ≡  
*parallel\_parser\_phrase\_* = 0;

**5. *Cparallel\_parser\_ph\_th* op directive.**

⟨*Cparallel\_parser\_ph\_th* op directive 5⟩ ≡  
 if (*parallel\_parser\_phrase\_* ≠ 0) {  
 delete *parallel\_parser\_phrase\_;*  
*parallel\_parser\_phrase\_* = 0;  
 }  
*parallel\_parser\_phrase\_* = new *T\_parallel\_parser\_phrase;*  
*parallel\_parser\_phrase\_*→set\_rc(*\*parser\_*→start\_token\_, \_\_FILE\_\_, \_\_LINE\_\_);  
 AST \**t* = new AST(*\*parallel\_parser\_phrase\_*);  
*parallel\_parser\_phrase\_*→phrase\_tree(*t*);

**6. *Cparallel\_parser\_ph\_th* user-declaration directive.**

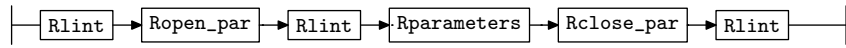
⟨*Cparallel\_parser\_ph\_th* user-declaration directive 6⟩ ≡  
**public:** *T\_parallel\_parser\_phrase* \* *parallel\_parser\_phrase\_;*

**7. *Cparallel\_parser\_ph\_th* user-prefix-declaration directive.**

⟨*Cparallel\_parser\_ph\_th* user-prefix-declaration directive 7⟩ ≡  
**#include** "lint\_balls.h"  
**#include** "identifier.h"  
**#include** "o2\_sdc.h"  
**#include** "o2\_code\_end.h"  
**#include** "la\_expr\_src.h"

8. *Rparallel\_parser\_phrase* rule.

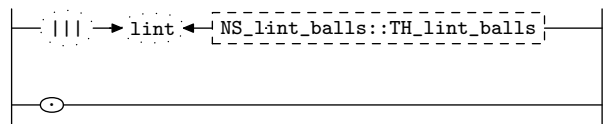
Rparallel\_parser\_phrase



⟨ Rparallel\_parser\_phrase subrule 1 op directive 8 ⟩ ≡  
*Cparallel\_parser\_ph\_th* \* *fsm* = ( *Cparallel\_parser\_ph\_th* \* ) *rule\_info...parser...fsm\_tbl...*;  
 RSVP(*fsm→parallel\_parser\_phrase\_*);  
*fsm→parallel\_parser\_phrase\_* = 0;

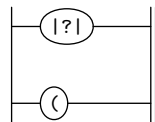
9. *Rlint* rule.

Rlint

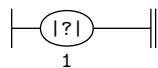


10. *Ropen\_par* rule.

Ropen\_par



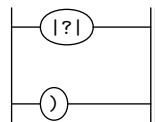
11. *Ropen\_par*'s subrule 1.



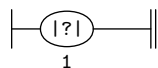
⟨ Ropen\_par subrule 1 op directive 11 ⟩ ≡  
*CAbs\_lr1\_sym* \* *sym* = **new** *Err\_no\_open\_parenthesis*;  
*sym→set\_rc*(\**sf→p1...*, *FILE...*, *LINE...*);  
 RSVP(*sym*);  
*rule\_info...parser...set\_stop\_parse*(*true*);

12. *Rclose\_par* rule.

Rclose\_par



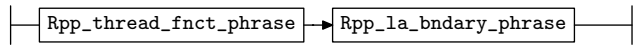
13. *Rclose\_par*'s subrule 1.



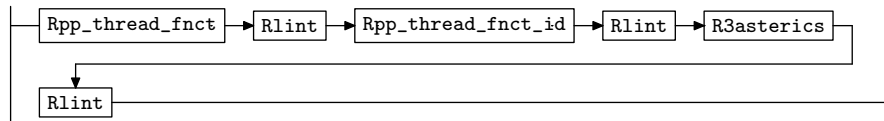
⟨ Rclose\_par subrule 1 op directive 13 ⟩ ≡  
*CAbs\_lr1\_sym* \* *sym* = **new** *Err\_no\_close\_parenthesis*;  
*sym→set\_rc*(\**sf→p1...*, *FILE...*, *LINE...*);  
 RSVP(*sym*);  
*rule\_info...parser...set\_stop\_parse*(*true*);

14. *Rparameters* rule.

Rparameters

15. *Rpp\_thread\_fnct\_phrase* rule.

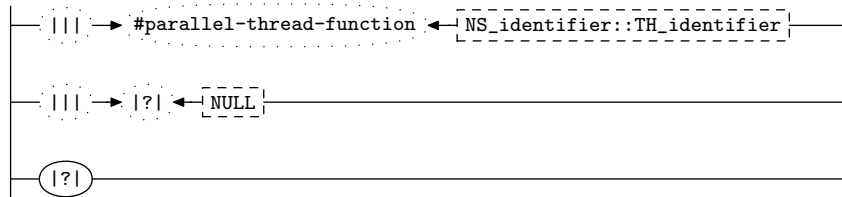
Rpp\_thread\_fnct\_phrase



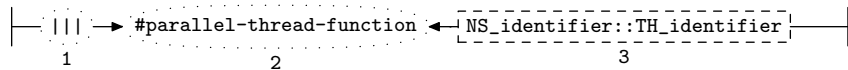
⟨Rpp\_thread\_fnct\_phrase subrule 1 op directive 15⟩ ≡  
*T\_parallel\_thread\_function* \* *pp\_fnct* = *sf-p1--pp\_fnct\_*;  
*pp\_fnct-identifier(sf-p3--id\_)*; *Cparallel\_parser\_ph\_th* \* *fsm* = ( *Cparallel\_parser\_ph\_th* \* )  
*rule\_info--parser--fsm\_tbl\_*;  
*fsm-parallel\_parser\_phrase--pp\_fnct(pp\_fnct)*;

16. *Rpp\_thread\_fnct* rule.

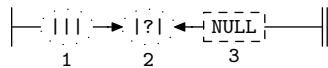
Rpp\_thread\_fnct

17. *Rpp\_thread\_fnct* user-declaration directive.

⟨Rpp\_thread\_fnct user-declaration directive 17⟩ ≡  
**public:** *T\_parallel\_thread\_function* \* *pp\_fnct\_*;

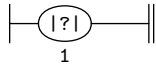
18. *Rpp\_thread\_fnct*'s subrule 1.

⟨Rpp\_thread\_fnct subrule 1 op directive 18⟩ ≡  
*pp\_fnct\_* = *sf-p2\_*;

19. *Rpp\_thread\_fnct*'s subrule 2.

⟨Rpp\_thread\_fnct subrule 2 op directive 19⟩ ≡  
*pp\_fnct\_* = 0;  
*sf-p2--set\_auto\_delete(true)*;  
*CAbs\_lr1\_sym* \* *sym* = **new** *Err\_no\_pp\_thread\_function\_present*;  
*sym-set\_rc(\*sf-p2--, \_\_FILE\_\_, \_\_LINE\_\_)*;  
*RSVP(sym)*;  
*rule\_info--parser--set\_stop\_parse(true)*;

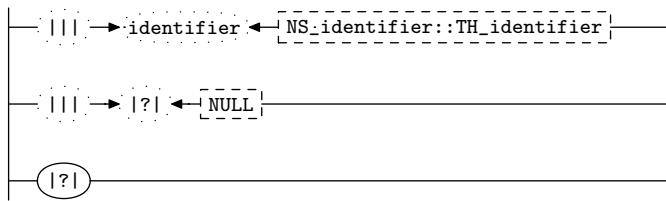
20. *Rpp\_thread\_fnct*'s subrule 3.



⟨Rpp\_thread\_fnct subrule 3 op directive 20⟩ ≡  
`pp_fnct_ = 0;`  
`CAbs_lr1_sym * sym = new Err_no_pp_thread_function_present;`  
`sym->set_rc(*rule_info_.parser->current_token(), __FILE__, __LINE__);`  
`RSVP(sym);`  
`rule_info_.parser->set_stop_parse(true);`

21. *Rpp\_thread\_fnct\_id* rule.

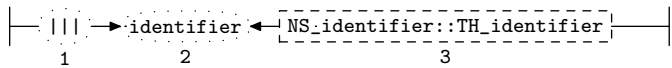
Rpp\_thread\_fnct\_id



22. *Rpp\_thread\_fnct\_id* user-declaration directive.

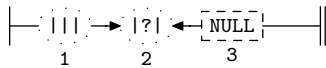
⟨Rpp\_thread\_fnct\_id user-declaration directive 22⟩ ≡  
**public:** *T\_identifier* \* *id\_;*

23. *Rpp\_thread\_fnct\_id*'s subrule 1.

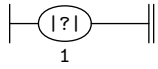


⟨Rpp\_thread\_fnct\_id subrule 1 op directive 23⟩ ≡  
`id_ = sf-p2_;`

24. *Rpp\_thread\_fnct\_id*'s subrule 2.



⟨Rpp\_thread\_fnct\_id subrule 2 op directive 24⟩ ≡  
`sf-p2->set_auto_delete(true);`  
`id_ = 0;`  
`CAbs_lr1_sym * sym = new Err_no_pp_fnct_id_present;`  
`sym->set_rc(*sf-p2_, __FILE__, __LINE__);`  
`RSVP(sym);`  
`rule_info_.parser->set_stop_parse(true);`

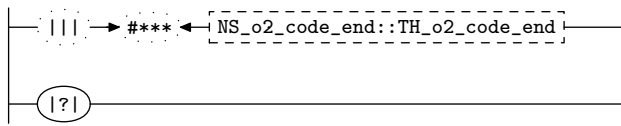
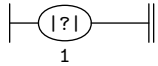
**25. *Rpp\_thread\_fnct\_id*'s subrule 3.**

⟨ *Rpp\_thread\_fnct\_id* subrule 3 op directive 25 ⟩ ≡

```
id_ = 0;
CAbs_lr1_sym * sym = new Err_no_pp_fnct_id_present;
sym->set_rc(*rule_info_.parser->current_token(), __FILE__, __LINE__);
RSVP(sym);
rule_info_.parser->set_stop_parse(true);
```

**26. *R3asterics* rule.**

*R3asterics*

**27. *R3asterics*'s subrule 2.**

⟨ *R3asterics* subrule 2 op directive 27 ⟩ ≡

```
CAbs_lr1_sym * sym = new Err_no_syntax_code_end_present;
sym->set_rc(*sf->p1_, __FILE__, __LINE__);
RSVP(sym);
rule_info_.parser->set_stop_parse(true);
```

**28. *Rpp\_la\_bndary\_phrase* rule.**

*Rpp\_la\_bndary\_phrase*

**29. *Rpp\_la\_bndary\_phrase* user-declaration directive.**

⟨ *Rpp\_la\_bndary\_phrase* user-declaration directive 29 ⟩ ≡

**public:** *T\_parallel\_la\_boundary* \* *pp\_la\_bndary\_*;

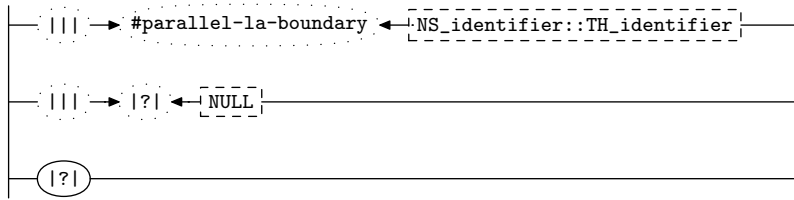
**30. *Rpp\_la\_bndary\_phrase*'s subrule 1.**

⟨ *Rpp\_la\_bndary\_phrase* subrule 1 op directive 30 ⟩ ≡

```
pp_la_bndary_ = sf->p1->pp_la_bndary_;
pp_la_bndary_-la_supplier(sf->p3_-la_exp_-la_tok_can());
sf->p3_-la_exp_-zero_la_tok_can(); Cparallel_parser_ph_th * fsm = ( Cparallel_parser_ph_th * )
rule_info_.parser->fsm_tbl_;
fsm->parallel_parser_phrase_-la_bndry(pp_la_bndary_);
```

**31. Rpp\_la\_bndary rule.**

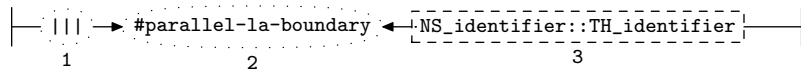
Rpp\_la\_bndary



**32. Rpp\_la\_bndary user-declaration directive.**

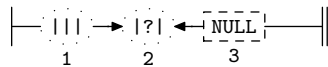
⟨Rpp\_la\_bndary user-declaration directive 32⟩ ≡  
**public:** T\_parallel\_la\_boundary \* pp\_la\_bndary\_;

**33. Rpp\_la\_bndary's subrule 1.**



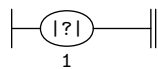
⟨Rpp\_la\_bndary subrule 1 op directive 33⟩ ≡  
 pp\_la\_bndary\_ = sf-p2\_;

**34. Rpp\_la\_bndary's subrule 2.**



⟨Rpp\_la\_bndary subrule 2 op directive 34⟩ ≡  
 pp\_la\_bndary\_ = 0;  
 sf-p2\_→set\_auto\_delete(true);  
 CAbs\_lr1\_sym \* sym = **new** Err\_no\_pp\_bndry\_present;  
 sym→set\_rc(\*sf-p2\_, \_\_FILE\_\_, \_\_LINE\_\_);  
 RSVP(sym);  
 rule\_info\_.parser\_→set\_stop\_parse(true);

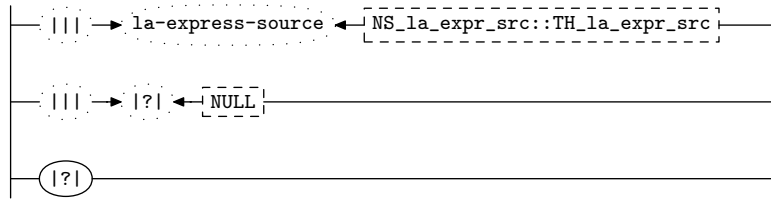
**35. Rpp\_la\_bndary's subrule 3.**



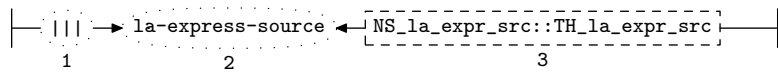
⟨Rpp\_la\_bndary subrule 3 op directive 35⟩ ≡  
 pp\_la\_bndary\_ = 0;  
 CAbs\_lr1\_sym \* sym = **new** Err\_no\_pp\_bndry\_present;  
 sym→set\_rc(\*sf-p1\_, \_\_FILE\_\_, \_\_LINE\_\_);  
 RSVP(sym);  
 rule\_info\_.parser\_→set\_stop\_parse(true);

**36. Rpp\_la\_bndary\_stc rule.**

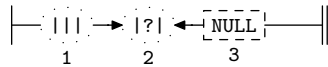
Rpp\_la\_bndary\_stc

**37. Rpp\_la\_bndary\_stc user-declaration directive.**

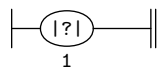
⟨Rpp\_la\_bndary\_stc user-declaration directive 37⟩ ≡  
**public:** *T\_la\_expr\_src* \* *la\_exp\_*;

**38. Rpp\_la\_bndary\_stc's subrule 1.**

⟨Rpp\_la\_bndary\_stc subrule 1 op directive 38⟩ ≡  
*la\_exp\_* = *sf-p2\_*;

**39. Rpp\_la\_bndary\_stc's subrule 2.**

⟨Rpp\_la\_bndary\_stc subrule 2 op directive 39⟩ ≡  
*la\_exp\_* = 0;  
 RSVP(*sf-p2\_*);  
*rule\_info\_*..*parser\_*--*set\_stop\_parse*(*true*);

**40. Rpp\_la\_bndary\_stc's subrule 3.**

⟨Rpp\_la\_bndary\_stc subrule 3 op directive 40⟩ ≡  
*la\_exp\_* = 0;  
*CAbs\_lr1\_sym* \* *sym* = **new** *Err\_no\_pp\_la\_bndary\_expr\_present*;  
*sym*--*set\_rc*(\**sf-p1\_*\_\_, \_\_FILE\_\_, \_\_LINE\_\_);  
 RSVP(*sym*);  
*rule\_info\_*..*parser\_*--*set\_stop\_parse*(*true*);



41. First Set Language for  $O_2^{linker}$ .

```
/*
  File: parallel_parser_ph_th.fsc
  Date and Time: Fri Jan  2 15:33:47 2015
*/
transitive      y
grammar-name    "parallel_parser_ph_th"
name-space     "NS_parallel_parser_ph_th"
thread-name     "TH_parallel_parser_ph_th"
monolithic     n
file-name      "parallel_parser_ph_th.fsc"
no-of-T        569
list-of-native-first-set-terminals 2
  raw_open_bracket
  LR1_questionable_shift_operator
end-list-of-native-first-set-terminals
list-of-transitive-threads 1
  NS_lint_balls::TH_lint_balls
end-list-of-transitive-threads
list-of-used-threads 4
  NS_identifier::TH_identifier
  NS_la_expr_src::TH_la_expr_src
  NS_lint_balls::TH_lint_balls
  NS_o2_code_end::TH_o2_code_end
end-list-of-used-threads
fsm-comments
"Parse grammar's parallel-parser construct."
```

## 42. Lr1 State Network.

$\Rightarrow$					State: 1 state type: $s/r$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
c	Rlint		2 2 1 $\epsilon$				1 0 1 1	
c	Rlint		2 1 1		lint NS_lint_balls::TH_lint_balls		1 2 3	
c	Rparallel_parser_phrase		1 1 1	Rlint	<u>Ropen.par</u>		1 4 11	
$\Rightarrow$		arbitration-code: $\epsilon$			State: 2 state type: $s$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
t	Rlint		2 1 2	lint			1 3 3	
$\Rightarrow$	lint				State: 3 state type: $r$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
t	Rlint		2 1 3				1 0 3 1	
$\Rightarrow$	Rlint				State: 4 state type: $s$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
c	Ropen.par		3 1 1	?			4 12 12	
c	Ropen.par		3 2 1	(			4 13 13	
t	Rparallel_parser_phrase		1 1 2	Ropen.par	<u>Rlint<math>^\epsilon</math> Rparameters</u>		1 5 11	
$\Rightarrow$	Ropen.par				State: 5 state type: $s/r$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
c	Rlint		2 2 1	$\epsilon$			5 0 5 2	
c	Rlint		2 1 1		lint NS_lint_balls::TH_lint_balls		5 2 3	
t	Rparallel_parser_phrase		1 1 3	Rlint	<u>Rparameters</u>		1 6 11	
$\Rightarrow$	Rlint				State: 6 state type: $s$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
c	Rpp_thread_fnct		7 3 1	?			6 14 14	
c	Rpp_thread_fnct		7 2 1		?  NULL		6 15 16	
c	Rpp_thread_fnct		7 1 1		# parallel-thread-function NS_identifier::TH_identifier		6 15 17	
t	Rparallel_parser_phrase		1 1 4	Rparameters	<u>Rclose.par</u>		1 7 11	
c	Rparameters		5 1 1	Rpp_thread_fnct_phrase	<u>Rpp_la_boundary_phrase</u>		6 18 23	
c	Rpp_thread_fnct_phrase		6 1 1	Rpp_thread_fnct	<u>Rlint<math>^\epsilon</math> Rpp_thread_fnct_id</u>		6 32 40	
$\Rightarrow$	Rparameters				State: 7 state type: $s$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
c	Rclose.par		4 1 1	?			7 8 8	
c	Rclose.par		4 2 1	)			7 9 9	
t	Rparallel_parser_phrase		1 1 5	Rclose.par	<u>Rlint<math>^\epsilon</math></u>		1 10 11	
$\Rightarrow$	?				State: 8 state type: $r$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
t	Rclose.par		4 1 2				7 0 8 1	
$\Rightarrow$	)				State: 9 state type: $r$			
$\leftarrow$	rule	$\rightarrow$	R# sr# Po	$\leftarrow$	subrule element	$\rightarrow$	Brn Gto Red LA	
t	Rclose.par		4 2 2				7 0 9 1	
$\Rightarrow$	Rclose.par				State: 10 state type: $s/r$			

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rlint			2	2	1	←	ε		10	0	10	1
c	Rlint			2	1	1	←	lint NS lint_balls::TH lint_balls		10	2	3	
t	Rparallel_parser_phrase			1	1	6	←	Rlint		1	11	11	
⇒ <i>Rlint</i> State: 11 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rparallel_parser_phrase			1	1	7	←			1	0	11	1
⇒ <i> ? </i> State: 12 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Ropen_par			3	1	2	←			4	0	12	2
⇒ <i>(</i> State: 13 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Ropen_par			3	2	2	←			4	0	13	2
⇒ <i> ? </i> State: 14 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_thread_funct			7	3	2	←			6	0	14	2
⇒ <i>    arbitration-code: ε</i> State: 15 state type: <i>s</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_thread_funct			7	2	2	←	?		6	16	16	
t	Rpp_thread_funct			7	1	2	←	# parallel-thread-function		6	17	17	
⇒ <i> ? </i> State: 16 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_thread_funct			7	2	3	←			6	0	16	2
⇒ <i>#parallel-thread-function</i> State: 17 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_thread_funct			7	1	3	←			6	0	17	2
⇒ <i>Rpp_thread_funct_phrase</i> State: 18 state type: <i>s</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rpp_la_boundary			11	3	1	←	?		18	19	19	
c	Rpp_la_boundary			11	1	1	←	# parallel-la-boundary NS identifier::TH identifier		18	20	22	
c	Rpp_la_boundary			11	2	1	←	?  NULL		18	20	21	
t	Rparameters			5	1	2	←	Rpp_la_boundary_phrase		6	23	23	
c	Rpp_la_boundary_phrase			10	1	1	←	Rpp_la_boundary <u>Rlint<sup>ε</sup> Rpp_la_boundary_stc</u>		18	24	31	
⇒ <i> ? </i> State: 19 state type: <i>r</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_la_boundary			11	3	2	←			18	0	19	2
⇒ <i>    arbitration-code: ε</i> State: 20 state type: <i>s</i>													
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_la_boundary			11	2	2	←	?		18	21	21	
t	Rpp_la_boundary			11	1	2	←	# parallel-la-boundary		18	22	22	
⇒ <i> ? </i> State: 21 state type: <i>r</i>													

←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary		11	2	3				18	0	21	2
⇒ #parallel-la-boundary												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary		11	1	3				18	0	22	2
⇒ Rpp_la_bndary_phrase												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rparameters		5	1	3				6	0	23	3
⇒ Rpp_la_bndary												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
c	Rlint		2	2	1	ε			24	0	24	2
c	Rlint		2	1	1	lint NS lint_balls::TH lint_balls			24	2	3	
t	Rpp_la_bndary_phrase		10	1	2	Rlint <u>Rpp_la_bndary_stc</u>			18	25	31	
⇒ Rlint												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
c	Rpp_la_bndary_stc		12	3	1	?			25	26	26	
c	Rpp_la_bndary_stc		12	1	1	la-express-source NS la_expr_src::TH la_expr_src			25	27	29	
c	Rpp_la_bndary_stc		12	2	1	?  NULL			25	27	28	
t	Rpp_la_bndary_phrase		10	1	3	Rpp_la_bndary_stc <u>Rlint</u> <sup>ε</sup>			18	30	31	
⇒  ?												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary_stc		12	3	2				25	0	26	4
⇒     arbitration-code: ε												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary_stc		12	2	2	?			25	28	28	
t	Rpp_la_bndary_stc		12	1	2	la-express-source			25	29	29	
⇒  ?												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary_stc		12	2	3				25	0	28	4
⇒ la-express-source												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary_stc		12	1	3				25	0	29	4
⇒ Rpp_la_bndary_stc												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
c	Rlint		2	2	1	ε			30	0	30	3
c	Rlint		2	1	1	lint NS lint_balls::TH lint_balls			30	2	3	
t	Rpp_la_bndary_phrase		10	1	4	Rlint			18	31	31	
⇒ Rlint												
←	<b>rule</b>	→	<b>R#</b>	<b>sr#</b>	<b>Po</b>	←	<b>subrule element</b>	→	<b>Brn</b>	<b>Gto</b>	<b>Red</b>	<b>LA</b>
t	Rpp_la_bndary_phrase		10	1	5				18	0	31	3
⇒ Rpp_thread_fnct												
State: 32 state type: <i>s/r</i>												

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rlint			2	2	1	←	ε		32	0	32	2
c	Rlint			2	1	1	←	lint NS lint_balls::TH lint_balls		32	2	3	
t	Rpp_thread_funct_phrase			6	1	2	←	Rlint <u>Rpp_thread_funct_id</u>		6	33	40	
⇒ <i>Rlint</i>										State: 33 state type: <i>s</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rpp_thread_funct_id			8	3	1	←	?		33	41	41	
c	Rpp_thread_funct_id			8	1	1	←	identifier NS identifier::TH identifier		33	42	44	
c	Rpp_thread_funct_id			8	2	1	←	?  NULL		33	42	43	
t	Rpp_thread_funct_phrase			6	1	3	←	Rpp_thread_funct_id <u>Rlint<sup>ε</sup> R3asterics</u>		6	34	40	
⇒ <u>Rpp_thread_funct_id</u>										State: 34 state type: <i>s/r</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rlint			2	2	1	←	ε		34	0	34	2
c	Rlint			2	1	1	←	lint NS lint_balls::TH lint_balls		34	2	3	
t	Rpp_thread_funct_phrase			6	1	4	←	Rlint <u>R3asterics</u>		6	35	40	
⇒ <i>Rlint</i>										State: 35 state type: <i>s</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	R3asterics			9	2	1	←	?		35	36	36	
c	R3asterics			9	1	1	←	#*** NS_o2_code_end::TH_o2_code_end		35	37	38	
t	Rpp_thread_funct_phrase			6	1	5	←	R3asterics <u>Rlint<sup>ε</sup></u>		6	39	40	
⇒ <u> ? </u>										State: 36 state type: <i>r</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	R3asterics			9	2	2	←			35	0	36	2
⇒ <u>    arbitration-code: ε</u>										State: 37 state type: <i>s</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	R3asterics			9	1	2	←	#***		35	38	38	
⇒ <u>#***</u>										State: 38 state type: <i>r</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	R3asterics			9	1	3	←			35	0	38	2
⇒ <u>R3asterics</u>										State: 39 state type: <i>s/r</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rlint			2	2	1	←	ε		39	0	39	2
c	Rlint			2	1	1	←	lint NS lint_balls::TH lint_balls		39	2	3	
t	Rpp_thread_funct_phrase			6	1	6	←	Rlint		6	40	40	
⇒ <i>Rlint</i>										State: 40 state type: <i>r</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_thread_funct_phrase			6	1	7	←			6	0	40	2
⇒ <u> ? </u>										State: 41 state type: <i>r</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpp_thread_funct_id			8	3	2	←			33	0	41	2
⇒ <u>    arbitration-code: ε</u>										State: 42 state type: <i>s</i>			
	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA

t Rpp_thread_funct_id	8	2	2	?
t Rpp_thread_funct_id	8	1	2	identifier

33 43 43

33 44 44

⇒|?|

←	rule	→	R#	sr#	Po	←
t Rpp_thread_funct_id			8	2	3	

State: 43 state type: *r*

subrule element

→ Brn Gto Red LA

33 0 43 2

⇒*identifier*

←	rule	→	R#	sr#	Po	←
t Rpp_thread_funct_id			8	1	3	

State: 44 state type: *r*

subrule element

→ Brn Gto Red LA

33 0 44 2

**43. Index.**

- $\epsilon$  : 9.
- # \*\*\*: 26.
- # parallel-la-boundary: 31.
- # parallel-thread-function: 16.
- |||: 9, 16, 21, 26, 31, 36.
- |?|: 10, 12, 16, 21, 26, 31, 36.
- \_\_FILE\_\_: 5, 11, 13, 19, 20, 24, 25, 27, 34, 35, 40.
- \_\_LINE\_\_: 5, 11, 13, 19, 20, 24, 25, 27, 34, 35, 40.
- AST: 5.
- CAbs\_lr1\_sym: 11, 13, 19, 20, 24, 25, 27, 34, 35, 40.
- Cparallel\_parser\_ph.th: 8, 15, 30.
- current\_token: 20, 25.
- Err\_no\_close\_parenthesis: 13.
- Err\_no\_open\_parenthesis: 11.
- Err\_no\_pp\_bndry\_present: 34, 35.
- Err\_no\_pp\_fnct\_id\_present: 24, 25.
- Err\_no\_pp\_la\_bndary\_expr\_present: 40.
- Err\_no\_pp\_thread\_function\_present: 19, 20.
- Err\_no\_syntax\_code\_end\_present: 27.
- fsm: 8, 15, 30.
- fsm\_tbl\_: 8, 15, 30.
- id\_: 15, 22, 23, 24, 25.
- identifier: 15.
- identifier: 21.
- la-express-source: 36.
- la\_bndry: 30.
- la\_exp\_: 30, 37, 38, 39, 40.
- la\_supplier: 30.
- la\_tok\_can: 30.
- lint: 9.
- NS\_identifier::TH\_identifier: 16, 21, 31.
- NS\_la\_expr\_src::TH\_la\_expr\_src: 36.
- NS\_lint\_balls::TH\_lint\_balls: 9.
- NS\_o2\_code\_end::TH\_o2\_code\_end: 26.
- NULL: 16, 21, 31, 36.
- parallel\_parser\_phrase\_: 4, 5, 6, 8, 15, 30.
- parallel\_parser\_phrase.th: 2.
- parser\_: 5, 8, 11, 13, 15, 19, 20, 24, 25, 27, 30, 34, 35, 39, 40.
- phrase\_tree: 5.
- pp\_fnct: 15.
- pp\_fnct\_: 15, 17, 18, 19, 20.
- pp\_funct: 15.
- pp\_la\_bndary\_: 29, 30, 32, 33, 34, 35.
- p1\_: 11, 13, 15, 27, 30, 35, 40.
- p2\_: 18, 19, 23, 24, 33, 34, 38, 39.
- p3\_: 15, 30.
- Rclose\_par: 8.
- Rclose\_par: 12, 13.
- Rlint: 9.
- Rlint: 8, 15, 28.
- Ropen\_par: 8.
- Ropen\_par: 10, 11.
- Rparallel\_parser\_phrase: 8.
- Rparameters: 14.
- Rparameters: 8.
- Rpp\_la\_bndary: 28.
- Rpp\_la\_bndary\_phrase: 14.
- Rpp\_la\_bndary\_stc: 28.
- Rpp\_thread\_fnct: 15.
- Rpp\_thread\_fnct\_id: 15.
- Rpp\_thread\_fnct\_phrase: 14.
- Rpp\_la\_bndary: 31, 33, 34, 35.
- Rpp\_la\_bndary\_phrase: 28, 30.
- Rpp\_la\_bndary\_stc: 36, 38, 39, 40.
- Rpp\_thread\_fnct: 16, 18, 19, 20.
- Rpp\_thread\_fnct\_id: 21, 23, 24, 25.
- Rpp\_thread\_fnct\_phrase: 15.
- RSVP: 8, 11, 13, 19, 20, 24, 25, 27, 34, 35, 39, 40.
- rule\_info\_: 8, 11, 13, 15, 19, 20, 24, 25, 27, 30, 34, 35, 39, 40.
- R3asterics: 15.
- R3asterics: 26, 27.
- set\_auto\_delete: 19, 24, 34.
- set\_rc: 5, 11, 13, 19, 20, 24, 25, 27, 34, 35, 40.
- set\_stop\_parse: 11, 13, 19, 20, 24, 25, 27, 34, 35, 39, 40.
- sf: 11, 13, 15, 18, 19, 23, 24, 27, 30, 33, 34, 35, 38, 39, 40.
- start\_token\_: 5.
- sym: 11, 13, 19, 20, 24, 25, 27, 34, 35, 40.
- T\_identifier: 22.
- T\_la\_expr\_src: 37.
- T\_parallel\_la\_boundary: 29, 32.
- T\_parallel\_parser\_phrase: 5, 6.
- T\_parallel\_thread\_function: 15, 17.
- true: 11, 13, 19, 20, 24, 25, 27, 34, 35, 39, 40.
- zero\_la\_tok\_can: 30.

< Cparallel\_parser\_ph.th constructor directive 4 >  
< Cparallel\_parser\_ph.th op directive 5 >  
< Cparallel\_parser\_ph.th user-declaration directive 6 >  
< Cparallel\_parser\_ph.th user-prefix-declaration directive 7 >  
< R3asterics subrule 2 op directive 27 >  
< Rclose\_par subrule 1 op directive 13 >  
< Ropen\_par subrule 1 op directive 11 >  
< Rparallel\_parser\_phrase subrule 1 op directive 8 >  
< Rpp\_la\_bndary subrule 1 op directive 33 >  
< Rpp\_la\_bndary subrule 2 op directive 34 >  
< Rpp\_la\_bndary subrule 3 op directive 35 >  
< Rpp\_la\_bndary user-declaration directive 32 >  
< Rpp\_la\_bndary\_phrase subrule 1 op directive 30 >  
< Rpp\_la\_bndary\_phrase user-declaration directive 29 >  
< Rpp\_la\_bndary\_stc subrule 1 op directive 38 >  
< Rpp\_la\_bndary\_stc subrule 2 op directive 39 >  
< Rpp\_la\_bndary\_stc subrule 3 op directive 40 >  
< Rpp\_la\_bndary\_stc user-declaration directive 37 >  
< Rpp\_thread\_funct subrule 1 op directive 18 >  
< Rpp\_thread\_funct subrule 2 op directive 19 >  
< Rpp\_thread\_funct subrule 3 op directive 20 >  
< Rpp\_thread\_funct user-declaration directive 17 >  
< Rpp\_thread\_funct\_id subrule 1 op directive 23 >  
< Rpp\_thread\_funct\_id subrule 2 op directive 24 >  
< Rpp\_thread\_funct\_id subrule 3 op directive 25 >  
< Rpp\_thread\_funct\_id user-declaration directive 22 >  
< Rpp\_thread\_funct\_phrase subrule 1 op directive 15 >



parallel\_parser\_ph\_th Grammar

Date: January 2, 2015 at 15:37

File: parallel\_parser\_ph\_th.lex Ns: NS\_parallel\_parser\_ph\_th

Version: 1.0

Debug: false

Grammar Comments:

Type: Thread

Parse grammar's parallel-parser construct.

1 element(s) in Lookahead Expression below

eolr

<i>parallel_parser_phrase_th</i> <b>thread</b> .....	2	2
Fsm Cparallel_parser_ph_th class .....	3	2
Cparallel_parser_ph_th constructor directive .....	4	2
Cparallel_parser_ph_th op directive .....	5	2
Cparallel_parser_ph_th user-declaration directive .....	6	2
Cparallel_parser_ph_th user-prefix-declaration directive .....	7	2
<i>Rparallel_parser_phrase</i> rule .....	8	3
<i>Rlint</i> rule .....	9	3
<i>Ropen_par</i> rule .....	10	3
<i>Ropen_par</i> 's subrule 1 .....	11	3
<i>Rclose_par</i> rule .....	12	3
<i>Rclose_par</i> 's subrule 1 .....	13	3
<i>Rparameters</i> rule .....	14	4
<i>Rpp_thread_funct_phrase</i> rule .....	15	4
<i>Rpp_thread_funct</i> rule .....	16	4
<i>Rpp_thread_funct</i> user-declaration directive .....	17	4
<i>Rpp_thread_funct</i> 's subrule 1 .....	18	4
<i>Rpp_thread_funct</i> 's subrule 2 .....	19	4
<i>Rpp_thread_funct</i> 's subrule 3 .....	20	5
<i>Rpp_thread_funct_id</i> rule .....	21	5
<i>Rpp_thread_funct_id</i> user-declaration directive .....	22	5
<i>Rpp_thread_funct_id</i> 's subrule 1 .....	23	5
<i>Rpp_thread_funct_id</i> 's subrule 2 .....	24	5
<i>Rpp_thread_funct_id</i> 's subrule 3 .....	25	6
<i>R3asterics</i> rule .....	26	6
<i>R3asterics</i> 's subrule 2 .....	27	6
<i>Rpp_la_bndary_phrase</i> rule .....	28	6
<i>Rpp_la_bndary_phrase</i> user-declaration directive .....	29	6
<i>Rpp_la_bndary_phrase</i> 's subrule 1 .....	30	6
<i>Rpp_la_bndary</i> rule .....	31	7
<i>Rpp_la_bndary</i> user-declaration directive .....	32	7
<i>Rpp_la_bndary</i> 's subrule 1 .....	33	7
<i>Rpp_la_bndary</i> 's subrule 2 .....	34	7
<i>Rpp_la_bndary</i> 's subrule 3 .....	35	7
<i>Rpp_la_bndary_stc</i> rule .....	36	8
<i>Rpp_la_bndary_stc</i> user-declaration directive .....	37	8
<i>Rpp_la_bndary_stc</i> 's subrule 1 .....	38	8
<i>Rpp_la_bndary_stc</i> 's subrule 2 .....	39	8
<i>Rpp_la_bndary_stc</i> 's subrule 3 .....	40	8
<b>First Set Language for <math>O_2^{linker}</math></b> .....	41	9
<b>Lr1 State Network</b> .....	42	10
<b>Index</b> .....	43	15