

1. first_set_for_threads grammar.

Create a thread's first set by building a closure-only state. The difference between this algo and *first_set_for_rules* is due to the `|.` symbol. Now what does this symbol represent within the first set as it is not a terminal in the input token stream? `|.` acts like an epsilon subrule and therefore `u` must go through the `lr_goto` states out of the closure-only state deriving their concrete first sets. This is a recursive situation as if the `|.` is present in one of these shifted states `u` keep on being devine... If the State-rule has been completely derived (accepted) then the thread's "parallel-la-boundary" expression adds its booty to the first set. If the "eolr" is present in this booty, then the thread's first set is only this symbol as it represents "all terminals" situation.

A cheap way would have been to just make the first set contain the "eolr" symbol if the `|.` symbol was present. This would work but would be inefficient due to the false starts in firing up the thread to have it misfire and shut down: a bit of thread farting. So lets be a bit more efficient calculating what truly represents its thread's first set.

An example of the above situation is a thread whose Start-rule contains only rules that are all explicitly or implicitly epsilonable (`|.` is present).

Definition of First set:

Terminals that start all substrings generated by the rule's productions. The grammar tree is walked in prefix formation accepting only "rule-def" followed by its "subrule-def" terminals. Each rule within the grammar follows this pattern: ie, the start-rule is the first to be evaluated. Though it is never referenced in a subrule i still create its first set.

The Algorithm.

The grammar reads each individual rule-def and all its subrule-def(s). Using its bottom-up recognition, *Rsubrule_def* adds the 1st element of the subrule into the *fs_list_*. *Rrule* processes the *fs_list_* as a closure-only state generating the rule's first set. In generating the first set, the elements in *fs_list_* are consumed as they are evaluated by removal from the list. Referenced terminals are added to the rule's first set. For 1st time referenced rules, their subrules are added at the end of *fs_list_* for eventual consumption. The neat thing about this algorithm is the 1st element in the *fs_list_* is only visited! It's a singular point of evaluation that is throw out to be replaced by its next in line element: ahh the bank queue and the teller.

Due to *cweave* irregularities in formatting C++ code of this grammar, please see *o2externs* documentation where `GEN_FS_OF_START_RULE` is coded as an external to overcome this deficiency.

2. Fsm Cfirst_set_for_threads class.**3. Cfirst_set_for_threads op directive.**

```
< Cfirst_set_for_threads op directive 3 > ≡
  rule_def_ = 0;
  subrule_def_ = 0;
  elem_t_ = 0; ip_can_ = ( tok_can < AST *> * ) parser_->token_supplier_;
```

4. Cfirst_set_for_threads user-declaration directive.

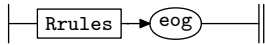
```
< Cfirst_set_for_threads user-declaration directive 4 > ≡
public: FS_ELEM_LIST_type fs_list_;
  RULES_IN_FS_LIST_type rules_in_fs_list_;
  rule_def * rule_def_;
  T_subrule_def * subrule_def_;
  AST * elem_t_;
  tok_can < AST *> *ip_can_;
```

5. Cfirst_set_for_threads user-prefix-declaration directive.

⟨ Cfirst_set_for_threads user-prefix-declaration directive 5 ⟩ ≡
`#include "o2_externs.h"`

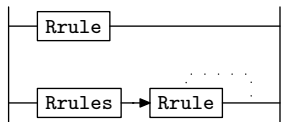
6. Rfirst_set_for_threads rule.

Rfirst_set_for_threads



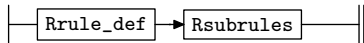
7. Rrules rule.

Rrules



8. Rrule rule.

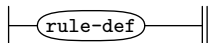
Rrule



⟨ Rrule subrule 1 op directive 8 ⟩ ≡
`Cfirst_set_for_threads * fsm = (Cfirst_set_for_threads *) rule_info_.parser_--fsm_tbl_;`
`GEN_FS_OF_START_RULE(fsm->fs_list_, fsm->rules_in_fs_list_, fsm->rule_def_);`

9. Rrule_def rule.

Rrule_def

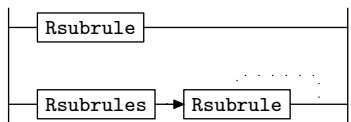


Initialize for its subrule findings.

⟨ Rrule_def subrule 1 op directive 9 ⟩ ≡
`Cfirst_set_for_threads * fsm = (Cfirst_set_for_threads *) rule_info_.parser_--fsm_tbl_;`
`fsm->rule_def_ = sf->p1_;`
`fsm->rules_in_fs_list_.clear();`
`fsm->fs_list_.clear();`

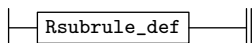
10. Rsubrules rule.

Rsubrules



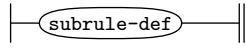
11. Rsubrule rule.

Rsubrule



12. Rsubrule_def rule.

Rsubrule_def



Create the entry within the *fs_list_*. Only the 1st element of eac subrule is evaluated.

⟨ Rsubrule_def subrule 1 op directive 12 ⟩ ≡

```
Cfirst_set_for_threads * fsm = ( Cfirst_set_for_threads * ) rule_info_..parser_--fsm_tbl_;
```

```
fsm→subrule_def_ = sf→p1_;
```

```
AST * sr_t = fsm→subrule_def_→subrule_s_tree();
```

```
AST * et = AST::get_spec_child(*sr_t, 1);
```

```
fsm→fs_list_.push_back(FS_ELEM_type(fsm→rule_def_, fsm→subrule_def_, et));
```

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

```
/*
  File: first_set_for_threads.fsc
  Date and Time: Sun Oct 30 13:39:16 2011
*/
transitive      n
grammar-name    "first_set_for_threads"
name-space     "NS_first_set_for_threads"
thread-name    "Cfirst_set_for_threads"
monolithic     y
file-name      "first_set_for_threads.fsc"
no-of-T        569
list-of-native-first-set-terminals 1
  rule_def
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
"Determine first set for thread."
```

14. Lr1 State Network.

\Rightarrow					State: 1 state type: s			
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	c	Rrule_def		4 1 1		rule_def		1 2 2
	c	Rrules		2 2 1		Rrules <u>Rrule</u>		1 3 5
	c	Rfirst_set_for_threads		1 1 1		Rrules <u>eog</u>		1 3 4
	c	Rrules		2 1 1		Rrule		1 12 12
	c	Rrule		3 1 1		Rrule_def <u>Rsubrules</u>		1 6 8
\Rightarrow	<i>rule-def</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rrule_def		4 1 2				1 0 2 1
\Rightarrow	<i>Rrules</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rfirst_set_for_threads		1 1 2		eog		1 4 4
	c	Rrule_def		4 1 1		rule_def		3 2 2
	t	Rrules		2 2 2		Rrule		1 5 5
	c	Rrule		3 1 1		Rrule_def <u>Rsubrules</u>		3 6 8
\Rightarrow	<i>eog</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rfirst_set_for_threads		1 1 3				1 0 4 2
\Rightarrow	<i>Rrule</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rrules		2 2 3				1 0 5 3
\Rightarrow	<i>Rrule_def</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	c	Rsubrule_def		7 1 1		subrule_def		6 7 7
	c	Rsubrules		5 2 1		Rsubrules <u>Rsubrule</u>		6 8 9
	t	Rrule		3 1 2		Rsubrules		3 8 8
	c	Rsubrules		5 1 1		Rsubrule		6 11 11
	c	Rsubrule		6 1 1		Rsubrule_def		6 10 10
\Rightarrow	<i>subrule-def</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rsubrule_def		7 1 2				6 0 7 4
\Rightarrow	<i>Rsubrules</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rrule		3 1 3				3 0 8 3
	c	Rsubrule_def		7 1 1		subrule_def		8 7 7
	t	Rsubrules		5 2 2		Rsubrule		6 9 9
	c	Rsubrule		6 1 1		Rsubrule_def		8 10 10
\Rightarrow	<i>Rsubrule</i>							
	\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
	t	Rsubrules		5 2 3				6 0 9 4

\Rightarrow *Rsubrule_def*

← rule → R# sr# Po ←
t Rsubrule 6 1 2

State: 10 state type: *r*
subrule element

→ Brn Gto Red LA
8 0 10 4

\Rightarrow *Rsubrule*

← rule → R# sr# Po ←
t Rsubrules 5 1 2

State: 11 state type: *r*
subrule element

→ Brn Gto Red LA
6 0 11 4

\Rightarrow *Rrule*

← rule → R# sr# Po ←
t Rrules 2 1 2

State: 12 state type: *r*
subrule element

→ Brn Gto Red LA
1 0 12 3

15. Index.

AST: 3, 4, 12.
Cfirst_set_for_threads: 8, 9, 12.
clear: 9.
cweave: 1.
elem_t: 3, 4.
eog: 6.
et: 12.
first_set_for_rules: 1.
first_set_for_threads: 1.
FS_ELEM_LIST_type: 4.
FS_ELEM_type: 12.
fs_list: 1, 4, 8, 9, 12.
fsm: 8, 9, 12.
fsm_tbl: 8, 9, 12.
 GEN_FS_OF_START_RULE: 1, 8.
get_spec_child: 12.
ip_can: 3, 4.
o2externs: 1.
parser: 3, 8, 9, 12.
push_back: 12.
p1: 9, 12.
Rfirst_set_for_threads: 6.
 Rrule: 7.
Rrule: 1, 8.
 Rrule_def: 8.
Rrule_def: 9.
 Rrules: 6, 7.
Rrules: 7.
 Rsubrule: 10.
Rsubrule: 11.
 Rsubrule_def: 11.
Rsubrule_def: 1, 12.
Rsubrules: 10.
 Rsubrules: 8, 10.
 rule-def: 9.
rule_def: 4.
rule_def: 3, 4, 8, 9, 12.
rule_info: 8, 9, 12.
rules_in_fs_list: 4, 8, 9.
RULES_IN_FS_LIST_type: 4.
sf: 9, 12.
sr_t: 12.
 subrule-def: 12.
subrule_def: 3, 4, 12.
subrule_s_tree: 12.
T_subrule_def: 4.
tok_can: 3, 4.
token_supplier: 3.

⟨ Cfirst_set_for_threads op directive 3 ⟩
⟨ Cfirst_set_for_threads user-declaration directive 4 ⟩
⟨ Cfirst_set_for_threads user-prefix-declaration directive 5 ⟩
⟨ Rrule subrule 1 op directive 8 ⟩
⟨ Rrule_def subrule 1 op directive 9 ⟩
⟨ Rsubrule_def subrule 1 op directive 12 ⟩

first_set_for_threads Grammar

Date: October 30, 2011 at 13:48

File: first_set_for_threads.lex

Ns: NS_first_set_for_threads

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

Determine first set for thread.

	Section	Page
<i>first_set_for_threads</i> grammar	1	1
Fsm Cfirst_set_for_threads class	2	1
Cfirst_set_for_threads op directive	3	1
Cfirst_set_for_threads user-declaration directive	4	1
Cfirst_set_for_threads user-prefix-declaration directive	5	2
<i>Rfirst_set_for_threads</i> rule	6	2
<i>Rrules</i> rule	7	2
<i>Rrule</i> rule	8	2
<i>Rrule_def</i> rule	9	2
<i>Rsubrules</i> rule	10	2
<i>Rsubrule</i> rule	11	2
<i>Rsubrule_def</i> rule	12	3
First Set Language for O_2^{linker}	13	4
Lr1 State Network	14	5
Index	15	7