

1. Copyright.

Copyright © Dave Bone 1998 - 2015

2. *fsc.file* stand alone Grammar.

Parses the “first set control” files outputted from each grammar. This is the fodder of O_2 's linker to generate the first sets per thread. It puts the grammar's attributes into yacco2's symbol table and into a global dictionary of threads for consolidation processing. See O_2 linker documentation on the real lowdown.

Linker's control language.

```
file-of-T-alphabet "c:/yacco2/compiler/grammars/yacco2_T_enumeration.fsc"
emitfile "/yacco2/compiler/grammars/yacco2_fsc.cpp"
preamble
#include <yacco2.h>
#include <yacco2_T_enumeration.h>
#include <yacco2_err_symbols.h>
#include <yacco2_k_symbols.h>
#include <yacco2_terminals.h>
#include <yacco2_characters.h>
using namespace NS_yacco2_T_enum;
using namespace NS_yacco2_err_symbols;
using namespace NS_yacco2_k_symbols;
using namespace NS_yacco2_terminals;
using namespace NS_yacco2_characters;
end-preamble
"/yacco2/compiler/grammars/error_symbols_phrase.fsc"
"/yacco2/compiler/grammars/error_symbols_phrase_th.fsc"
"/yacco2/compiler/grammars/angled_string.fsc"
"/yacco2/compiler/grammars/bad_char_set.fsc"
"/yacco2/compiler/grammars/c_comments.fsc"
...
```

3. Fsm Cfsc.file class.**4. Cfsc.file user-declaration directive.**

\langle Cfsc.file user-declaration directive 4 $\rangle \equiv$

public: *std* :: *stringfully_qualified_th_name*;

char *transitive*;

char *monolithic*;

NS_yacco2_terminals :: *T_c_string* * *grammar_file_name*;

NS_yacco2_terminals :: *T_c_string* * *name_space_name*;

NS_yacco2_terminals :: *T_c_string* * *thread_name*;

NS_yacco2_terminals :: *T_c_string* * *file_name*;

NS_yacco2_terminals :: *T_c_string* * *fsm_comments*;

NS_yacco2_terminals :: *T_int_no* * *no_of_T*; *std* :: *vector* < **int** > *list_of_Ts*;

NS_yacco2_terminals :: *T_int_no* * *no_native_T*;

std :: *vector* < *NS_yacco2_terminals* :: *thread_attributes* * > *list_of_transitive_threads*;

std :: *vector* < *NS_yacco2_terminals* :: *thread_attributes* * > *list_of_used_threads*;

NS_yacco2_terminals :: *T_int_no* * *no_threads*;

NS_yacco2_terminals :: *T_int_no* * *used_no_threads*;

std :: *vector* < *std* :: *string* > **used_threads*;

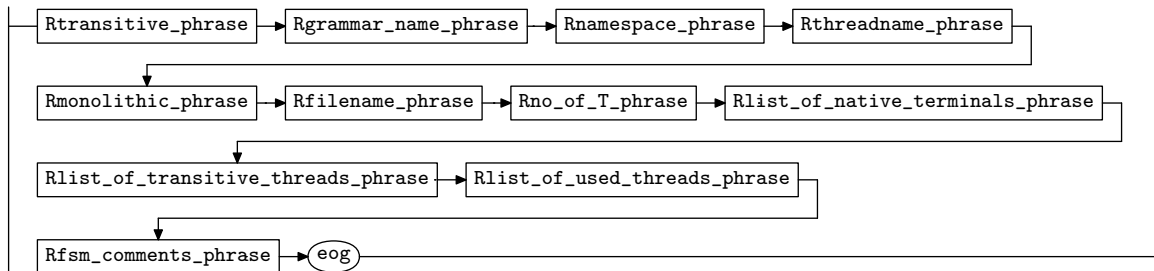
5. Cfsc_file user-prefix-declaration directive.

⟨ Cfsc_file user-prefix-declaration directive 5 ⟩ ≡

```
#include "yacco2_stbl.h"
extern std::vector < NS_yacco2_terminals::table_entry *> GRAMMAR_DICTIONARY;
extern std::vector < NS_yacco2_terminals::table_entry *> T_DICTIONARY;
extern std::map < std::string, std::vector < std::string >> USED_THREADS_LIST;
```

6. Rfsc_file rule.

Rfsc_file



7. Rfsc_file op directive.

```

⟨Rfsc_file op directive 7⟩ ≡
  Cfsc_file * fsm = ( Cfsc_file * ) rule_info_.parser_--fsm_tbl_;
  using namespace NS_yacco2_k_symbols;
  int no_T = fsm-no_native_T-no();
  int sze_T = fsm-list_of_Ts_.size();
  if (sze_T ≠ no_T) {
    CAbs_lr1_sym * sym = new Err_no_of_native_Ts_in_list_not_equal(sze_T);
    sym-set_rc(*fsm-no_native_T-, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    return;
  }
  int no_th = fsm-no_threads_-no();
  int sze_th = fsm-list_of_transitive_threads_.size();
  if (sze_th ≠ no_th) {
    CAbs_lr1_sym * sym = new Err_no_of_threads_in_list_not_equal(sze_th);
    sym-set_rc(*fsm-no_threads-, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    return;
  }
  if (fsm-monolithic_ ≡ 'n') { /* pass epsilon pass thru: thread grammar only */
    if ((no_T ≡ 0) ∧ (no_th ≡ 0)) {
      CAbs_lr1_sym * sym = new Err_epsilon_pass_thru;
      sym-set_rc(*fsm-no_native_T-, __FILE__, __LINE__);
      ADD_TOKEN_TO_ERROR_QUEUE(*sym);
      rule_info_.parser_--set_stop_parse(true);
      return;
    }
    if ((no_T ≡ 1) ∧ (no_th ≡ 0)) {
      int transience = fsm-list_of_Ts_[0];
      if (transience ≡ 7) { /* |t| operator */
        CAbs_lr1_sym * sym = new Err_epsilon_pass_thru_no_Ts;
        sym-set_rc(*fsm-no_native_T-, __FILE__, __LINE__);
        ADD_TOKEN_TO_ERROR_QUEUE(*sym);
        rule_info_.parser_--set_stop_parse(true);
        return;
      }
    }
  }
  using namespace yacco2_stbl;
  T_sym_tbl_report_card report_card;
  find_sym_in_stbl(report_card,
    *fsm-fully_qualified_th_name_.c_str()); if (report_card.action_ ≡ T_sym_tbl_report_card::fnd) {
    /* crtd by another fsc file */
  }
  if (report_card.tbl_entry_-defined_ ≡ true) { /* probably dup fsc file */
    CAbs_lr1_sym * sym = new Err_already_defined_in_fsc_file;
    sym-set_rc(*fsm-thread_name-, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info_.parser_--set_stop_parse(true);
    return;
  }

```

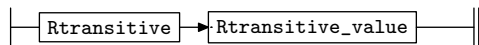
```

}
report_card.tbl_entry_defined_ = true; th_in_stbl * th_in_tbl = ( th_in_stbl * ) report_card.tbl_entry_symbol_;
thread_attributes * th_entry = th_in_tbl_thread_in_stbl();
th_entry_transitive_ = fsm_transitive_;
th_entry_grammar_file_name_ = fsm_grammar_file_name_;
th_entry_name_space_name_ = fsm_name_space_name_;
th_entry_thread_name_ = fsm_thread_name_;
th_entry_monolithic_ = fsm_monolithic_;
th_entry_file_name_ = fsm_file_name_;
th_entry_fsm_comments_ = fsm_fsm_comments_;
if (¬fsm_list_of_Ts_.empty())
    copy(fsm_list_of_Ts_.begin(), fsm_list_of_Ts_.end(), back_inserter(th_entry_list_of_Ts_));
if (¬fsm_list_of_transitive_threads_.empty()) copy(fsm_list_of_transitive_threads_.begin(),
    fsm_list_of_transitive_threads_.end(), back_inserter(th_entry_list_of_transitive_threads_));
return; } thread_attributes * th_id = new thread_attributes(fsm_fully_qualified_th_name_.c_str(),
    fsm_transitive_, fsm_grammar_file_name_, fsm_name_space_name_, fsm_thread_name_,
    fsm_monolithic_, fsm_file_name_, fsm_list_of_Ts_, fsm_list_of_transitive_threads_, fsm_fsm_comments_);
th_in_stbl * t = new th_in_stbl(th_id);
add_sym_to_stbl(report_card, *fsm_fully_qualified_th_name_.c_str(), *t, table_entry::defed,
    table_entry::thread);
if (report_card.status_ ≠ T_sym_tbl_report_card::okay) {
    t_stbl_idx(report_card.pos_);
    report_card.err_entry_set_rc(*th_id, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*report_card.err_entry_);
    rule_info__parser__set_stop_parse(true);
    return;
}
GRAMMAR_DICTIONARY.push_back(report_card.tbl_entry_);
th_id_th_enum_ = GRAMMAR_DICTIONARY.size() - 1;
ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2::PTR_LR1_eog_);
ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2::PTR_LR1_eog_);

```

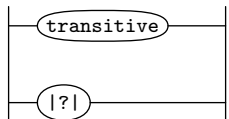
8. *Rtransitive_phrase* rule.

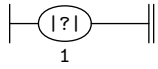
Rtransitive_phrase



9. *Rtransitive* rule.

Rtransitive

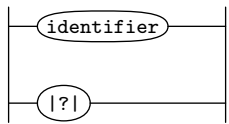
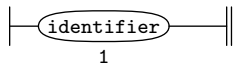


10. *Rtransitive's* subrule 2.

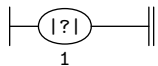
⟨Rtransitive subrule 2 op directive 10⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_transitive_kw_not_present*;
sym→*set_rc*(**sf*→*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__→*parser*__→*set_stop_parse*(*true*);

11. *Rtransitive_value* rule.

Rtransitive_value

12. *Rtransitive_value's* subrule 1.

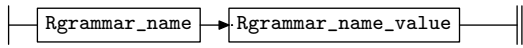
⟨Rtransitive_value subrule 1 op directive 12⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info*__→*parser*__→*fsm_tbl*__;
T_identifier * *k* = *sf*→*p1*__;
if (*k*→*identifier*()→*size*() ≠ 1) {
 CAbs_lr1_sym * *sym* = **new** *Err_transitive_value_bad*;
 sym→*set_rc*(**sf*→*p1*__, __FILE__, __LINE__);
 ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
 *rule_info*__→*parser*__→*set_stop_parse*(*true*);
 return;
}
char *c* = (**k*→*identifier*())[0];
if (*c* ≡ 'n' ∨ *c* ≡ 'y') {
 fsm→*transitive*_ = *c*;
 return;
}
CAbs_lr1_sym * *sym* = **new** *Err_transitive_value_bad*;
sym→*set_rc*(**sf*→*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__→*parser*__→*set_stop_parse*(*true*);

13. *Rtransitive_value's* subrule 2.

⟨Rtransitive_value subrule 2 op directive 13⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_transitive_value_bad*;
sym→*set_rc*(**sf*→*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__→*parser*__→*set_stop_parse*(*true*);

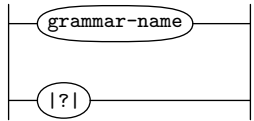
14. *Rgrammar_name_phrase* rule.

Rgrammar_name_phrase

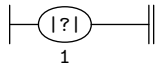


15. *Rgrammar_name* rule.

Rgrammar_name



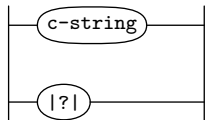
16. *Rgrammar_name*'s subrule 2.



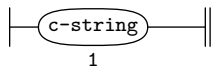
⟨Rgrammar_name subrule 2 op directive 16⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_grammar_name_kw_not_present*;
sym→*set_rc*(**sf*→*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__*parser*__→*set_stop_parse*(*true*);

17. *Rgrammar_name_value* rule.

Rgrammar_name_value

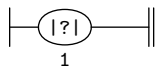


18. *Rgrammar_name_value*'s subrule 1.

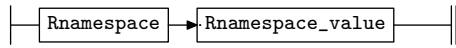
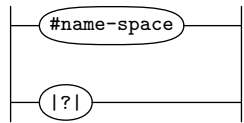
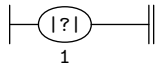


⟨Rgrammar_name_value subrule 1 op directive 18⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info*__*parser*__→*fsm_tbl*__;
fsm→*grammar_file_name*_ = *sf*→*p1*__;

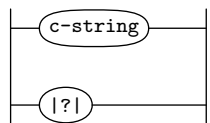
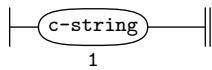
19. *Rgrammar_name_value*'s subrule 2.



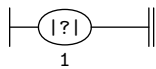
⟨Rgrammar_name_value subrule 2 op directive 19⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_grammar_name_value_not_present*;
sym→*set_rc*(**sf*→*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__*parser*__→*set_stop_parse*(*true*);

20. *Rnamespace_phrase* rule.*Rnamespace_phrase***21. *Rnamespace* rule.***Rnamespace***22. *Rnamespace*'s subrule 2.**

⟨ *Rnamespace* subrule 2 op directive 22 ⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_namespace_kw_not_present*;
sym-*set_rc*(**sf*-*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__*parser*__-*set_stop_parse*(*true*);

23. *Rnamespace_value* rule.*Rnamespace_value***24. *Rnamespace_value*'s subrule 1.**

⟨ *Rnamespace_value* subrule 1 op directive 24 ⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info*__*parser*__-*fsm_tbl*__;
fsm-*name_space_name*_ = *sf*-*p1*__;

25. *Rnamespace_value*'s subrule 2.

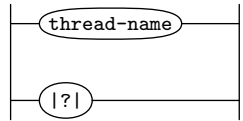
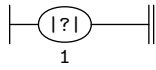
⟨ *Rnamespace_value* subrule 2 op directive 25 ⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_namespace_value_not_present*;
sym-*set_rc*(**sf*-*p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__*parser*__-*set_stop_parse*(*true*);

26. Rthreadname_phrase rule.

Rthreadname_phrase

**27. Rthreadname rule.**

Rthreadname

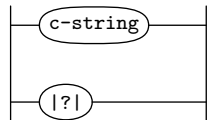
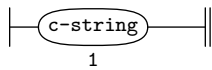
**28. Rthreadname's subrule 2.**

⟨ Rthreadname subrule 2 op directive 28 ⟩ ≡

```
CAbs_lr1_sym * sym = new Err_threadname_kw_not_present;
sym->set_rc(*sf->p1--, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(*sym);
rule_info_.parser--set_stop_parse(true);
```

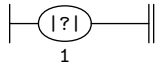
29. Rthreadname_value rule.

Rthreadname_value

**30. Rthreadname_value's subrule 1.**

⟨ Rthreadname_value subrule 1 op directive 30 ⟩ ≡

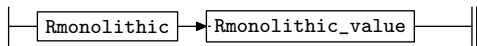
```
Cfsc_file * fsm = ( Cfsc_file * ) rule_info_.parser--fsm_tbl_;
fsm->thread_name_ = sf->p1--;
fsm->fully_qualified_th_name_ += fsm->name_space_name_->c_string()->c_str();
fsm->fully_qualified_th_name_ += " : ";
fsm->fully_qualified_th_name_ += fsm->thread_name_->c_string()->c_str();
T_identifier * th_id = new T_identifier(fsm->fully_qualified_th_name_->c_str());
th_id->set_rc(*fsm->thread_name_, __FILE__, __LINE__);
string gnm(sf->p1-->c_string()->c_str());
USED_THREADS_LIST[gnm] = std::vector < string > ();
std::map < std::string, std::vector < std::string >> ::iterator i = USED_THREADS_LIST.find(gnm);
fsm->used_threads_ = &i->second;
```

31. Rthreadname_value's subrule 2.

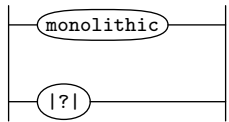
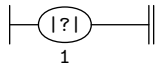
⟨Rthreadname_value subrule 2 op directive 31⟩ ≡
`CAbs_lr1_sym * sym = new Err_threadname_value_not_present;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info->parser--set_stop_parse(true);`

32. Rmonolithic_phrase rule.

Rmonolithic_phrase

**33. Rmonolithic rule.**

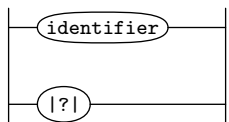
Rmonolithic

**34. Rmonolithic's subrule 2.**

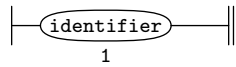
⟨Rmonolithic subrule 2 op directive 34⟩ ≡
`CAbs_lr1_sym * sym = new Err_monolithic_kw_not_present;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info->parser--set_stop_parse(true);`

35. Rmonolithic_value rule.

Rmonolithic_value



36. Rmonolithic_value's subrule 1.

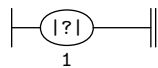


```

⟨Rmonolithic_value subrule 1 op directive 36⟩ ≡
  Cfsc_file * fsm = ( Cfsc_file * ) rule_info...parser...fsm_tbl...;
  T_identifier * k = sf-p1...;
  if (k-identifier()-size() ≠ 1) {
    CAbs_lr1_sym * sym = new Err_monolithic_value_bad;
    sym->set_rc(*sf-p1..., __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info...parser...set_stop_parse(true);
    return;
  }
  char c = (*k-identifier())[0];
  if (c ≡ 'n' ∨ c ≡ 'y') {
    fsm->monolithic_ = c;
    return;
  }
  CAbs_lr1_sym * sym = new Err_monolithic_value_bad;
  sym->set_rc(*sf-p1..., __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info...parser...set_stop_parse(true);

```

37. Rmonolithic_value's subrule 2.



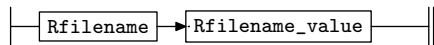
```

⟨Rmonolithic_value subrule 2 op directive 37⟩ ≡
  CAbs_lr1_sym * sym = new Err_monolithic_value_bad;
  sym->set_rc(*sf-p1..., __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info...parser...set_stop_parse(true);

```

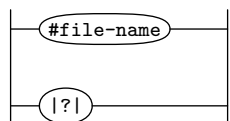
38. Rfilename_phrase rule.

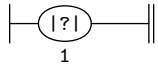
Rfilename_phrase



39. Rfilename rule.

Rfilename

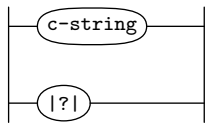
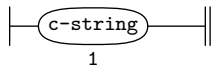


40. *Rfilename*'s subrule 2.

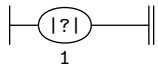
⟨ *Rfilename* subrule 2 op directive 40 ⟩ ≡
`CAbs_lr1_sym * sym = new Err_filename_kw_not_present;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info->parser->set_stop_parse(true);`

41. *Rfilename_value* rule.

Rfilename_value

42. *Rfilename_value*'s subrule 1.

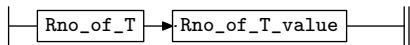
⟨ *Rfilename_value* subrule 1 op directive 42 ⟩ ≡
`Cfsc_file * fsm = (Cfsc_file *) rule_info->parser->fsm_tbl_;`
`fsm->file_name_ = sf->p1_;`

43. *Rfilename_value*'s subrule 2.

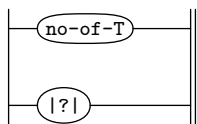
⟨ *Rfilename_value* subrule 2 op directive 43 ⟩ ≡
`CAbs_lr1_sym * sym = new Err_filename_value_not_present;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info->parser->set_stop_parse(true);`

44. *Rno_of_T_phrase* rule.

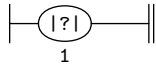
Rno_of_T_phrase

45. *Rno_of_T* rule.

Rno_of_T



46. *Rno_of_T*'s subrule 2.

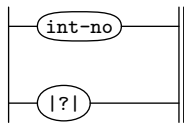


```

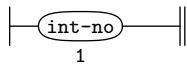
⟨ Rno_of_T subrule 2 op directive 46 ⟩ ≡
  CAbs_lr1_sym * sym = new Err_no_of_T_kw_not_present;
  sym->set_rc(*sf->p1--, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info->parser->set_stop_parse(true);
  
```

47. *Rno_of_T_value* rule.

Rno_of_T_value



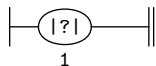
48. *Rno_of_T_value*'s subrule 1.



```

⟨ Rno_of_T_value subrule 1 op directive 48 ⟩ ≡
  Cfsc_file * fsm = ( Cfsc_file * ) rule_info->parser->fsm_tbl_;
  fsm->no_of_T_ = sf->p1--; /* not eq make as a warning but continue parsing */
  if (T_DICTIONARY.size() ≠ sf->p1-->no()) {
    CAbs_lr1_sym * sym = new Err_bad_T_alphabet;
    sym->set_rc(*sf->p1--, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info->parser->set_stop_parse(true);
  }
  
```

49. *Rno_of_T_value*'s subrule 2.

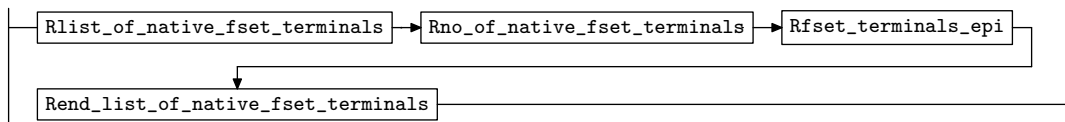


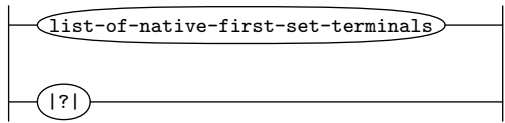
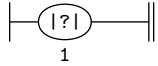
```

⟨ Rno_of_T_value subrule 2 op directive 49 ⟩ ≡
  CAbs_lr1_sym * sym = new Err_no_of_T_value_not_present;
  sym->set_rc(*sf->p1--, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info->parser->set_stop_parse(true);
  
```

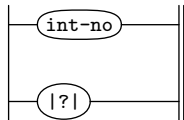
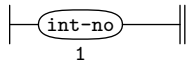
50. *Rlist_of_native_terminals_phrase* rule.

Rlist_of_native_terminals_phrase

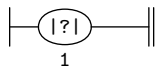


51. *Rlist_of_native_fset_terminals* rule.*Rlist_of_native_fset_terminals***52. *Rlist_of_native_fset_terminals*'s subrule 2.**

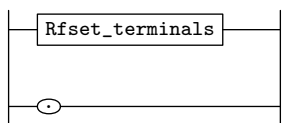
⟨*Rlist_of_native_fset_terminals* subrule 2 op directive 52⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_list_of_terminals_kw_not_present*;
sym-*set_rc*(**sf-p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__.*parser*__-*set_stop_parse*(*true*);

53. *Rno_of_native_fset_terminals* rule.*Rno_of_native_fset_terminals***54. *Rno_of_native_fset_terminals*'s subrule 1.**

⟨*Rno_of_native_fset_terminals* subrule 1 op directive 54⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info*__.*parser*__-*fsm_tbl*__;
fsm-no_native_T = *sf-p1*__;

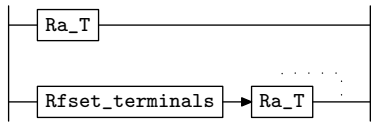
55. *Rno_of_native_fset_terminals*'s subrule 2.

⟨*Rno_of_native_fset_terminals* subrule 2 op directive 55⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_no_of_terminals_not_present*;
sym-*set_rc*(**sf-p1*__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info*__.*parser*__-*set_stop_parse*(*true*);

56. *Rfset_terminals_epi* rule.*Rfset_terminals_epi*

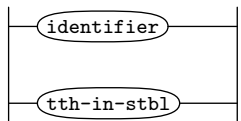
57. Rfset_terminals rule.

Rfset_terminals

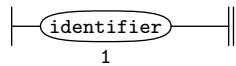


58. Ra_T rule.

Ra_T

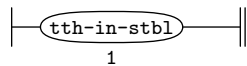


59. Ra_T's subrule 1.



⟨ Ra_T subrule 1 op directive 59 ⟩ ≡
T_identifier * *k* = *sf-p1*_;
CAbs_lr1_sym * *sym* = **new** *Err_bad_T_in_list*;
sym→*set_rc*(**k*, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
rule_info→*parser*→*set_stop_parse*(*true*);

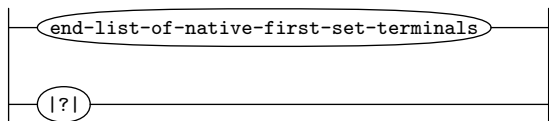
60. Ra_T's subrule 2.



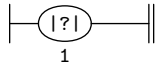
⟨ Ra_T subrule 2 op directive 60 ⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info*→*parser*→*fsm_tbl*_;
tth_in_stbl * *k* = *sf-p1*_;
T_attributes * *T_att* = *k-t_in_stbl*();
fsm-list_of_Ts→*push_back*(*T_att*→*T_enum*);

61. Rend_list_of_native_fset_terminals rule.

Rend_list_of_native_fset_terminals



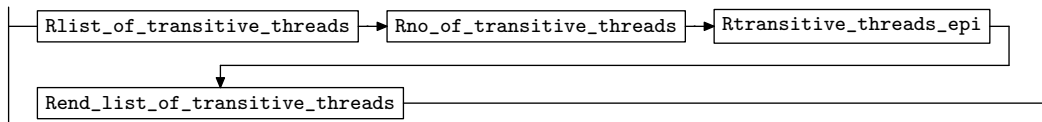
62. *Rend_list_of_native_fset_terminals's* subrule 2.



\langle *Rend_list_of_native_fset_terminals* subrule 2 op directive 62 \equiv
`CAbs_lr1_sym * sym = new Err_end_list_native_T_kw_not_present;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info...parser-->set_stop_parse(true);`

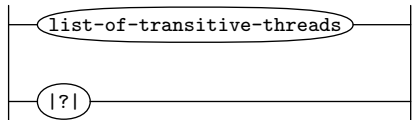
63. *Rlist_of_transitive_threads_phrase* rule.

Rlist_of_transitive_threads_phrase

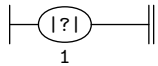


64. *Rlist_of_transitive_threads* rule.

Rlist_of_transitive_threads



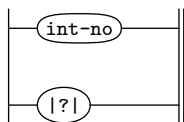
65. *Rlist_of_transitive_threads's* subrule 2.



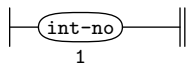
\langle *Rlist_of_transitive_threads* subrule 2 op directive 65 \equiv
`CAbs_lr1_sym * sym = new Err_no_list_of_trans_threads_kw;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info...parser-->set_stop_parse(true);`

66. *Rno_of_transitive_threads* rule.

Rno_of_transitive_threads

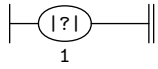


67. *Rno_of_transitive_threads's* subrule 1.



\langle *Rno_of_transitive_threads* subrule 1 op directive 67 \equiv
`Cfsc_file * fsm = (Cfsc_file *) rule_info...parser-->fsm_tbl_;`
`fsm->no_threads_ = sf->p1_;`

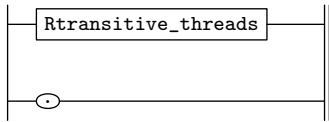
68. *Rno_of_transitive_threads's* subrule 2.



\langle Rno_of_transitive_threads subrule 2 op directive 68 $\rangle \equiv$
`CAbs_lr1_sym * sym = new Err_no_of_threads_not_present;`
`sym->set_rc(*sf->p1--, __FILE__, __LINE__);`
`ADD_TOKEN_TO_ERROR_QUEUE(*sym);`
`rule_info->parser->set_stop_parse(true);`

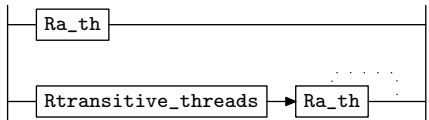
69. *Rtransitive_threads_epi* rule.

Rtransitive_threads_epi



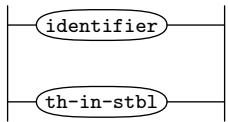
70. *Rtransitive_threads* rule.

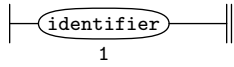
Rtransitive_threads



71. *Ra_th* rule.

Ra_th



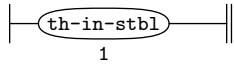
72. *Ra_th*'s subrule 1.

⟨*Ra_th* subrule 1 op directive 72⟩ ≡

```

Cfsc_file * fsm = ( Cfsc_file * ) rule_info...parser...fsm_tbl...;
T_identifier * k = sf→p1...;
using namespace yacco2_stbl;
T_sym_tbl_report_card report_card;
thread_attributes * th = new thread_attributes(k→identifier()→c_str());
th→set_rc(*k, __FILE__, __LINE__);
fsm→list_of_transitive_threads...push_back(th);
th→in_stbl * t = new th→in_stbl(th);
add_sym_to_stbl(report_card, *k→identifier()→c_str(), *t, table_entry::used, table_entry::thread);
if (report_card→status ≠ T_sym_tbl_report_card::okay) {
    t→stbl_idx(report_card→pos);
    report_card→err_entry→set_rc(*k, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*report_card→err_entry);
    rule_info...parser...→set_stop_parse(true);
    return;
}
GRAMMAR_DICTIONARY→push_back(report_card→tbl_entry);
th→th_enum = GRAMMAR_DICTIONARY→size() - 1;

```

73. Ra_th's subrule 2.

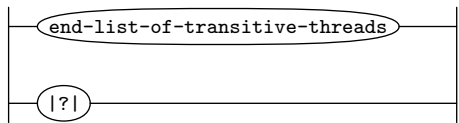
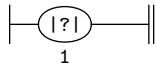
```

⟨Ra_th subrule 2 op directive 73⟩ ≡
  Cfsc_file * fsm = ( Cfsc_file * ) rule_info...parser...fsm.tbl_;
  th_in_stbl * k = sf-p1--;
  thread_attributes * th = k-thread_in_stbl();
  fsm-list_of_transitive_threads_.push_back(th);
using namespace yacco2_stbl;
  T_sym_tbl_report_card report_card;
  find_sym_in_stbl(report_card, *th-fully_qualified_th_name...c_str());
if (report_card.status_ ≠ T_sym_tbl_report_card::okay) {
  report_card.err_entry->set_rc(*k, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*report_card.err_entry);
  rule_info...parser...set_stop_parse(true);
  return;
}
if (report_card.action_ ≡ T_sym_tbl_report_card::fnd) { /* crted by another fsc file */
  report_card.tbl_entry->used_ = true;
  return;
}
if (report_card.action_ ≡ T_sym_tbl_report_card::not_fnd) {
  CAbs_lr1_sym * sym = new Err_no_sym_defs_present;
  sym->set_rc(*k, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info...parser...set_stop_parse(true);
  return;
}

```

74. Rend_list_of_transitive_threads rule.

Rend_list_of_transitive_threads

**75. Rend_list_of_transitive_threads's subrule 2.**

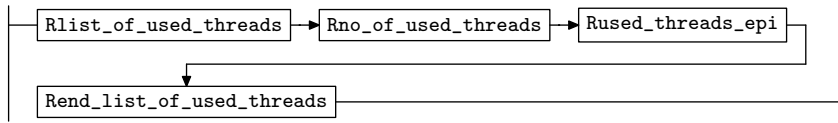
```

⟨Rend_list_of_transitive_threads subrule 2 op directive 75⟩ ≡
  CAbs_lr1_sym * sym = new Err_end_list_of_transitive_threads_kw_not_present;
  sym->set_rc(*sf-p1--, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info...parser...set_stop_parse(true);

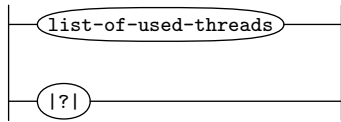
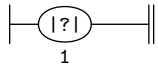
```

76. *Rlist_of_used_threads_phrase* rule.

Rlist_of_used_threads_phrase

**77. *Rlist_of_used_threads* rule.**

Rlist_of_used_threads

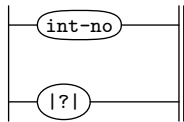
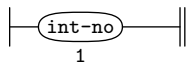
**78. *Rlist_of_used_threads*'s subrule 2.**

⟨Rlist_of_used_threads subrule 2 op directive 78⟩ ≡

```
CAbs_lr1_sym * sym = new Err_no_list_of_trans_threads_kw;
sym->set_rc(*sf-p1--, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(*sym);
rule_info_.parser--set_stop_parse(true);
```

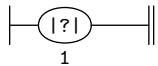
79. *Rno_of_used_threads* rule.

Rno_of_used_threads

**80. *Rno_of_used_threads*'s subrule 1.**

⟨Rno_of_used_threads subrule 1 op directive 80⟩ ≡

```
Cfsc_file * fsm = ( Cfsc_file * ) rule_info_.parser--fsm_tbl_;
fsm->used_no_threads_ = sf-p1--;
```

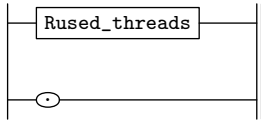
81. *Rno_of_used_threads*'s subrule 2.

⟨Rno_of_used_threads subrule 2 op directive 81⟩ ≡

```
CAbs_lr1_sym * sym = new Err_no_of_threads_not_present;
sym->set_rc(*sf-p1--, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(*sym);
rule_info_.parser--set_stop_parse(true);
```

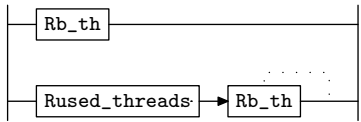
82. *Rused_threads_epi* rule.

Rused_threads_epi



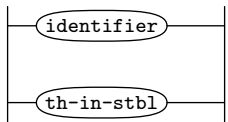
83. *Rused_threads* rule.

Rused_threads

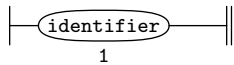


84. *Rb_th* rule.

Rb_th



85. *Rb_th*'s subrule 1.

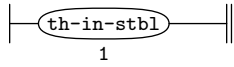


\langle Rb_th subrule 1 op directive 85 $\rangle \equiv$

```

Cfsc_file * fsm = ( Cfsc_file * ) rule_info...parser...fsm_tbl...;
T_identifier * k = sf-p1...;
using namespace yacco2_stbl;
T_sym_tbl_report_card report_card;
thread_attributes * th = new thread_attributes(k-identifier()-c_str());
th->set_rc(*k, __FILE__, __LINE__);
fsm-list_of_used_threads->push_back(th);
std::string::size_type x = th->fully_qualified_th_name->find("::");
string tnm;
string & fqn = th->fully_qualified_th_name;
for (int xx = x + 2; xx < fqn.size(); ++xx) {
    tnm += fqn[xx];
}
fsm-used_threads->push_back(tnm);

```

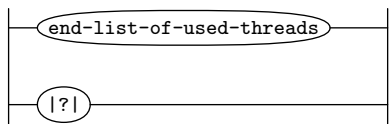
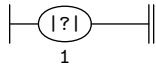
86. *Rb_th*'s subrule 2.

⟨*Rb_th* subrule 2 op directive 86⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info...**parser...**fsm_tbl...*;
th_in_stbl * *k* = *sf-p1...*;
thread_attributes * *th* = *k*→*thread_in_stbl*();
*fsm-list_of_used_threads...**push_back*(*th*);
std :: *string* :: *size_type* *x* = *th*→*fully_qualified_th_name...**find*(" : : ");
string *tnm*;
string & *fqn* = *th*→*fully_qualified_th_name...*;
for (**int** *xx* = *x* + 2; *xx* < *fqn.size*(); ++*xx*) {
 tnm += *fqn*[*xx*];
}

fsm→*used_threads...**push_back*(*tnm*);

87. *Rend_list_of_used_threads* rule.

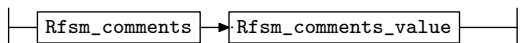
Rend_list_of_used_threads

**88. *Rend_list_of_used_threads*'s subrule 2.**

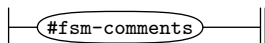
⟨*Rend_list_of_used_threads* subrule 2 op directive 88⟩ ≡
CAbs_lr1_sym * *sym* = **new** *Err_end_list_of_transitive_threads_kw_not_present*;
sym→*set_rc*(**sf-p1...*, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(**sym*);
*rule_info...**parser...*→*set_stop_parse*(*true*);

89. *Rfsm_comments_phrase* rule.

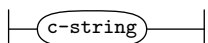
Rfsm_comments_phrase

**90. *Rfsm_comments* rule.**

Rfsm_comments

**91. *Rfsm_comments_value* rule.**

Rfsm_comments_value



⟨*Rfsm_comments_value* subrule 1 op directive 91⟩ ≡
Cfsc_file * *fsm* = (*Cfsc_file* *) *rule_info...**parser...**fsm_tbl...*;
fsm→*fsm_comments_* = *sf-p1...*;

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

```
/*
  File: fsc_file.fsc
  Date and Time: Fri Jan  2 15:33:36 2015
*/
transitive      n
grammar-name    "fsc_file"
name-space      "NS_fsc_file"
thread-name     "Cfsc_file"
monolithic      y
file-name       "fsc_file.fsc"
no-of-T         569
list-of-native-first-set-terminals 2
  T_transitive
  LR1_questionable_shift_operator
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
"\0linker's 'fsc' control file parser."
```

93. Lr1 State Network.

⇒					State: 1 state type: ^s			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rtransitive		3 2 1	?			1 2 2	
c	Rtransitive		3 1 1	transitive			1 3 3	
c	Rfsc_file		1 1 1	Rtransitive_phrase	<u>Rgrammar_name_phrase</u>		1 4 15	
c	Rtransitive_phrase		2 1 1	Rtransitive	<u>Rtransitive_value</u>		1 16 19	
⇒	?				State: 2 state type: ^r			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	Rtransitive		3 2 2				1 0 2 1	
⇒	transitive				State: 3 state type: ^r			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	Rtransitive		3 1 2				1 0 3 1	
⇒	Rtransitive_phrase				State: 4 state type: ^s			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rgrammar_name		6 2 1	?			4 20 20	
c	Rgrammar_name		6 1 1	grammar-name			4 21 21	
t	Rfsc_file		1 1 2	Rgrammar_name_phrase	<u>Rnamespace_phrase</u>		1 5 15	
c	Rgrammar_name_phrase		5 1 1	Rgrammar_name	<u>Rgrammar_name_value</u>		4 22 25	
⇒	Rgrammar_name_phrase				State: 5 state type: ^s			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rnamespace		9 2 1	?			5 26 26	
c	Rnamespace		9 1 1	# name-space			5 27 27	
t	Rfsc_file		1 1 3	Rnamespace_phrase	<u>Rthreadname_phrase</u>		1 6 15	
c	Rnamespace_phrase		8 1 1	Rnamespace	<u>Rnamespace_value</u>		5 28 31	
⇒	Rnamespace_phrase				State: 6 state type: ^s			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rthreadname		12 2 1	?			6 32 32	
c	Rthreadname		12 1 1	thread-name			6 33 33	
t	Rfsc_file		1 1 4	Rthreadname_phrase	<u>Rmonolithic_phrase</u>		1 7 15	
c	Rthreadname_phrase		11 1 1	Rthreadname	<u>Rthreadname_value</u>		6 34 37	
⇒	Rthreadname_phrase				State: 7 state type: ^s			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rmonolithic		15 2 1	?			7 38 38	
c	Rmonolithic		15 1 1	monolithic			7 39 39	
t	Rfsc_file		1 1 5	Rmonolithic_phrase	<u>Rfilename_phrase</u>		1 8 15	
c	Rmonolithic_phrase		14 1 1	Rmonolithic	<u>Rmonolithic_value</u>		7 40 43	
⇒	Rmonolithic_phrase				State: 8 state type: ^s			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rfilename		18 2 1	?			8 44 44	
c	Rfilename		18 1 1	# file-name			8 45 45	
t	Rfsc_file		1 1 6	Rfilename_phrase	<u>Rno_of_T_phrase</u>		1 9 15	
c	Rfilename_phrase		17 1 1	Rfilename	<u>Rfilename_value</u>		8 46 49	

⇒ *Rfilename_phrase*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c		Rno_of_T		21	2	1		?		9	50	50	
c		Rno_of_T		21	1	1		no-of-T		9	51	51	
t		Rfsc_file		1	1	7		Rno_of_T_phrase <u>Rlist_of_native_terminals_phrase</u>		1	10	15	
c		Rno_of_T_phrase		20	1	1		Rno_of_T <u>Rno_of_T_value</u>		9	52	55	

State: 9 state type: *s*

⇒ *Rno_of_T_phrase*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c		Rlist_of_native_fset_terminals		24	2	1		?		10	56	56	
c		Rlist_of_native_fset_terminals		24	1	1		list-of-native-first-set-terminals		10	57	57	
t		Rfsc_file		1	1	8		Rlist_of_native_terminals_phrase <u>Rlist_of_transitive_threads_phrase</u>		1	11	15	
c		Rlist_of_native_terminals_phrase		23	1	1		Rlist_of_native_fset_terminals <u>Rno_of_native_fset_terminals</u>		10	58	63	

State: 10 state type: *s*

⇒ *Rlist_of_native_terminals_phrase*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c		Rlist_of_transitive_threads		31	2	1		?		11	64	64	
c		Rlist_of_transitive_threads		31	1	1		list-of-transitive-threads		11	65	65	
t		Rfsc_file		1	1	9		Rlist_of_transitive_threads_phrase <u>Rlist_of_used_threads_phrase</u>		1	12	15	
c		Rlist_of_transitive_threads_phrase		30	1	1		Rlist_of_transitive_threads <u>Rno_of_transitive_threads</u>		11	66	71	

State: 11 state type: *s*

⇒ *Rlist_of_transitive_threads_phrase*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c		Rlist_of_used_threads		38	2	1		?		12	72	72	
c		Rlist_of_used_threads		38	1	1		list-of-used-threads		12	73	73	
t		Rfsc_file		1	1	10		Rlist_of_used_threads_phrase <u>Rfsm_comments_phrase</u>		1	13	15	
c		Rlist_of_used_threads_phrase		37	1	1		Rlist_of_used_threads <u>Rno_of_used_threads</u>		12	74	79	

State: 12 state type: *s*

⇒ *Rlist_of_used_threads_phrase*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c		Rfsm_comments		45	1	1		# fsm-comments		13	80	80	
t		Rfsc_file		1	1	11		Rfsm_comments_phrase <u>eog</u>		1	14	15	
c		Rfsm_comments_phrase		44	1	1		Rfsm_comments <u>Rfsm_comments_value</u>		13	81	83	

State: 13 state type: *s*

⇒ *Rfsm_comments_phrase*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t		Rfsc_file		1	1	12		eog		1	15	15	

State: 14 state type: *s*

⇒ *eog*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t		Rfsc_file		1	1	13				1	0	15	2

State: 15 state type: *r*

⇒ *Rtransitive*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c		Rtransitive_value		4	2	1		?		16	17	17	
c		Rtransitive_value		4	1	1		identifier		16	18	18	
t		Rtransitive_phrase		2	1	2		Rtransitive_value		1	19	19	

State: 16 state type: *s*

⇒ *|?|*

	←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t		Rtransitive_value		4	2	2				16	0	17	3

State: 17 state type: *r*

\Rightarrow <i>identifier</i>		State: 18 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rtransitive_value	4 1 2		16 0 18 3
\Rightarrow <i>Rtransitive_value</i>		State: 19 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rtransitive_phrase	2 1 3		1 0 19 3
\Rightarrow <i> ? </i>		State: 20 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rgrammar_name	6 2 2		4 0 20 4
\Rightarrow <i>grammar-name</i>		State: 21 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rgrammar_name	6 1 2		4 0 21 4
\Rightarrow <i>Rgrammar_name</i>		State: 22 state type: <i>s</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
c Rgrammar_name_value	7 2 1 ?		22 23 23
c Rgrammar_name_value	7 1 1 c-string		22 24 24
t Rgrammar_name_phrase	5 1 2 Rgrammar_name_value		4 25 25
\Rightarrow <i> ? </i>		State: 23 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rgrammar_name_value	7 2 2		22 0 23 5
\Rightarrow <i>c-string</i>		State: 24 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rgrammar_name_value	7 1 2		22 0 24 5
\Rightarrow <i>Rgrammar_name_value</i>		State: 25 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rgrammar_name_phrase	5 1 3		4 0 25 5
\Rightarrow <i> ? </i>		State: 26 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rnamespace	9 2 2		5 0 26 4
\Rightarrow <i>#name-space</i>		State: 27 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rnamespace	9 1 2		5 0 27 4
\Rightarrow <i>Rnamespace</i>		State: 28 state type: <i>s</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
c Rnamespace_value	10 2 1 ?		28 29 29
c Rnamespace_value	10 1 1 c-string		28 30 30
t Rnamespace_phrase	8 1 2 Rnamespace_value		5 31 31
\Rightarrow <i> ? </i>		State: 29 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rnamespace_value	10 2 2		28 0 29 6

\Rightarrow <i>c-string</i>		State: 30 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rnamespace_value	10 1 2		28 0 30 6
\Rightarrow <i>Rnamespace_value</i>		State: 31 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rnamespace_phrase	8 1 3		5 0 31 6
\Rightarrow <i> ? </i>		State: 32 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rthreadname	12 2 2		6 0 32 4
\Rightarrow <i>thread-name</i>		State: 33 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rthreadname	12 1 2		6 0 33 4
\Rightarrow <i>Rthreadname</i>		State: 34 state type: <i>s</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
c Rthreadname_value	13 2 1 ?		34 35 35
c Rthreadname_value	13 1 1 c-string		34 36 36
t Rthreadname_phrase	11 1 2 Rthreadname_value		6 37 37
\Rightarrow <i> ? </i>		State: 35 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rthreadname_value	13 2 2		34 0 35 7
\Rightarrow <i>c-string</i>		State: 36 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rthreadname_value	13 1 2		34 0 36 7
\Rightarrow <i>Rthreadname_value</i>		State: 37 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rthreadname_phrase	11 1 3		6 0 37 7
\Rightarrow <i> ? </i>		State: 38 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rmonolithic	15 2 2		7 0 38 1
\Rightarrow <i>monolithic</i>		State: 39 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rmonolithic	15 1 2		7 0 39 1
\Rightarrow <i>Rmonolithic</i>		State: 40 state type: <i>s</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
c Rmonolithic_value	16 2 1 ?		40 41 41
c Rmonolithic_value	16 1 1 identifier		40 42 42
t Rmonolithic_phrase	14 1 2 Rmonolithic_value		7 43 43
\Rightarrow <i> ? </i>		State: 41 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rmonolithic_value	16 2 2		40 0 41 8

\Rightarrow <i>identifier</i>		State: 42 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rmonolithic_value	16 1 2		40 0 42 8
\Rightarrow <i>Rmonolithic_value</i>		State: 43 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rmonolithic_phrase	14 1 3		7 0 43 8
\Rightarrow <i> ? </i>		State: 44 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rfilename	18 2 2		8 0 44 4
\Rightarrow <i>#file-name</i>		State: 45 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rfilename	18 1 2		8 0 45 4
\Rightarrow <i>Rfilename</i>		State: 46 state type: <i>s</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
c Rfilename_value	19 2 1 ?		46 47 47
c Rfilename_value	19 1 1 c-string		46 48 48
t Rfilename_phrase	17 1 2 Rfilename_value		8 49 49
\Rightarrow <i> ? </i>		State: 47 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rfilename_value	19 2 2		46 0 47 9
\Rightarrow <i>c-string</i>		State: 48 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rfilename_value	19 1 2		46 0 48 9
\Rightarrow <i>Rfilename_value</i>		State: 49 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rfilename_phrase	17 1 3		8 0 49 9
\Rightarrow <i> ? </i>		State: 50 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rno_of_T	21 2 2		9 0 50 10
\Rightarrow <i>no-of-T</i>		State: 51 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rno_of_T	21 1 2		9 0 51 10
\Rightarrow <i>Rno_of_T</i>		State: 52 state type: <i>s</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
c Rno_of_T_value	22 2 1 ?		52 53 53
c Rno_of_T_value	22 1 1 int-no		52 54 54
t Rno_of_T_phrase	20 1 2 Rno_of_T_value		9 55 55
\Rightarrow <i> ? </i>		State: 53 state type: <i>r</i>	
← rule	→ R# sr# Po ←	subrule element	→ Brn Gto Red LA
t Rno_of_T_value	22 2 2		52 0 53 11

\Rightarrow <i>int-no</i>		State: 54 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t Rno_of_T_value 22 1 2			52 0 54 11
\Rightarrow <i>Rno_of_T_value</i>		State: 55 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t Rno_of_T_phrase 20 1 3			9 0 55 11
\Rightarrow <i> ? </i>		State: 56 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t Rlist_of_native_fset_terminals 24 2 2			10 0 56 10
\Rightarrow <i>list-of-native-first-set-terminals</i>		State: 57 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t Rlist_of_native_fset_terminals 24 1 2			10 0 57 10
\Rightarrow <i>Rlist_of_native_fset_terminals</i>		State: 58 state type: <i>s</i>	
← rule → R# sr# Po ←		subrule element	→
c Rno_of_native_fset_terminals 25 2 1 <i> ? </i>			
c Rno_of_native_fset_terminals 25 1 1 <i>int-no</i>			
t Rlist_of_native_terminals_phrase 23 1 2		Rno_of_native_fset_terminals <u><i>Rfset_terminals.epi^c</i></u> <u><i>Rend_list_of_native_fset_terminals</i></u>	
\Rightarrow <i>Rno_of_native_fset_terminals</i>		State: 59 state type: <i>s/r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
c <i>Rfset_terminals.epi</i> 26 2 1 ϵ			59 0 59 12
c <i>Ra_T</i> 28 1 1 <i>identifier</i>			59 86 86
c <i>Ra_T</i> 28 2 1 <i>tth-in-stbl</i>			59 87 87
t <i>Rlist_of_native_terminals_phrase</i> 23 1 3		<i>Rfset_terminals.epi</i> <u><i>Rend_list_of_native_fset_terminals</i></u>	10 60 63
c <i>Rfset_terminals.epi</i> 26 1 1 <i>Rfset_terminals</i>			59 88 88
c <i>Rfset_terminals</i> 27 2 1 <i>Rfset_terminals</i> <u><i>Ra_T</i></u>			59 88 89
c <i>Rfset_terminals</i> 27 1 1 <i>Ra_T</i>			59 90 90
\Rightarrow <i>Rfset_terminals.epi</i>		State: 60 state type: <i>s</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
c <i>Rend_list_of_native_fset_terminals</i> 29 2 1 <i> ? </i>			60 61 61
c <i>Rend_list_of_native_fset_terminals</i> 29 1 1 <i>end-list-of-native-first-set-terminals</i>			60 62 62
t <i>Rlist_of_native_terminals_phrase</i> 23 1 4 <i>Rend_list_of_native_fset_terminals</i>			10 63 63
\Rightarrow <i> ? </i>		State: 61 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t <i>Rend_list_of_native_fset_terminals</i> 29 2 2			60 0 61 13
\Rightarrow <i>end-list-of-native-first-set-terminals</i>		State: 62 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t <i>Rend_list_of_native_fset_terminals</i> 29 1 2			60 0 62 13
\Rightarrow <i>Rend_list_of_native_fset_terminals</i>		State: 63 state type: <i>r</i>	
← rule → R# sr# Po ←		subrule element	→ Brn Gto Red LA
t <i>Rlist_of_native_terminals_phrase</i> 23 1 5			10 0 63 13
\Rightarrow <i> ? </i>		State: 64 state type: <i>r</i>	

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rlist_of_transitive_threads		31	2	2				11	0	64	10	
⇒	<i>list-of-transitive-threads</i>						State: 65 state type: <i>r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rlist_of_transitive_threads		31	1	2				11	0	65	10	
⇒	<i>Rlist_of_transitive_threads</i>						State: 66 state type: <i>s</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→					
c	Rno_of_transitive_threads		32	2	1	?							
c	Rno_of_transitive_threads		32	1	1	int-no							
t	Rlist_of_transitive_threads_phrase		30	1	2	Rno_of_transitive_threads	<i>Rtransitive_threads_epi</i> ^ε	<i>Rend_list_of_transitive_threads</i>					
⇒	<i>Rno_of_transitive_threads</i>						State: 67 state type: <i>s/r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
c	Rtransitive_threads_epi		33	2	1	ε			67	0	67	14	
c	Ra_th		35	1	1	identifier			67	93	93		
c	Ra_th		35	2	1	th-in-stbl			67	94	94		
t	Rlist_of_transitive_threads_phrase		30	1	3	Rtransitive_threads_epi	<i>Rend_list_of_transitive_threads</i>		11	68	71		
c	Rtransitive_threads_epi		33	1	1	Rtransitive_threads			67	95	95		
c	Rtransitive_threads		34	2	1	Rtransitive_threads	<i>Ra_th</i>		67	95	96		
c	Rtransitive_threads		34	1	1	Ra_th			67	97	97		
⇒	<i>Rtransitive_threads_epi</i>						State: 68 state type: <i>s</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
c	Rend_list_of_transitive_threads		36	2	1	?			68	69	69		
c	Rend_list_of_transitive_threads		36	1	1	end-list-of-transitive-threads			68	70	70		
t	Rlist_of_transitive_threads_phrase		30	1	4	Rend_list_of_transitive_threads			11	71	71		
⇒	?						State: 69 state type: <i>r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rend_list_of_transitive_threads		36	2	2				68	0	69	15	
⇒	<i>end-list-of-transitive-threads</i>						State: 70 state type: <i>r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rend_list_of_transitive_threads		36	1	2				68	0	70	15	
⇒	<i>Rend_list_of_transitive_threads</i>						State: 71 state type: <i>r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rlist_of_transitive_threads_phrase		30	1	5				11	0	71	15	
⇒	?						State: 72 state type: <i>r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rlist_of_used_threads		38	2	2				12	0	72	10	
⇒	<i>list-of-used-threads</i>						State: 73 state type: <i>r</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA	
t	Rlist_of_used_threads		38	1	2				12	0	73	10	
⇒	<i>Rlist_of_used_threads</i>						State: 74 state type: <i>s</i>						
←	rule	→	R#	sr#	Po	←	subrule element	→		Brn	Gto	Red	LA
c	Rno_of_used_threads		39	2	1	?			74	98	98		

c	Rno_of_used_threads	39	1	1	int-no					74	99	99
t	Rlist_of_used_threads_phrase	37	1	2	Rno_of_used_threads	<u>Rused_threads_epi</u> ^ε	<u>Rend_list_of_used_threads</u>			12	75	79
⇒ <i>Rno_of_used_threads</i> State: 75 state type: <i>s/r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rused_threads_epi		40	2	1	←	ε	→	75	0	75	16
c	Rb_th		42	1	1	←	identifier	→	75	100	100	
c	Rb_th		42	2	1	←	th-in-stbl	→	75	101	101	
t	Rlist_of_used_threads_phrase		37	1	3	←	Rused_threads_epi <u>Rend_list_of_used_threads</u>	→	12	76	79	
c	Rused_threads_epi		40	1	1	←	Rused_threads	→	75	102	102	
c	Rused_threads		41	2	1	←	Rused_threads <u>Rb_th</u>	→	75	102	103	
c	Rused_threads		41	1	1	←	Rb_th	→	75	104	104	
⇒ <i>Rused_threads_epi</i> State: 76 state type: <i>s</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rend_list_of_used_threads		43	2	1	←	?	→	76	77	77	
c	Rend_list_of_used_threads		43	1	1	←	end-list-of-used-threads	→	76	78	78	
t	Rlist_of_used_threads_phrase		37	1	4	←	Rend_list_of_used_threads	→	12	79	79	
⇒ <i> ? </i> State: 77 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rend_list_of_used_threads		43	2	2	←		→	76	0	77	17
⇒ <i>end-list-of-used-threads</i> State: 78 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rend_list_of_used_threads		43	1	2	←		→	76	0	78	17
⇒ <i>Rend_list_of_used_threads</i> State: 79 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rlist_of_used_threads_phrase		37	1	5	←		→	12	0	79	17
⇒ <i>#fsm-comments</i> State: 80 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rfsm_comments		45	1	2	←		→	13	0	80	18
⇒ <i>Rfsm_comments</i> State: 81 state type: <i>s</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rfsm_comments_value		46	1	1	←	c-string	→	81	82	82	
t	Rfsm_comments_phrase		44	1	2	←	Rfsm_comments_value	→	13	83	83	
⇒ <i>c-string</i> State: 82 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rfsm_comments_value		46	1	2	←		→	81	0	82	19
⇒ <i>Rfsm_comments_value</i> State: 83 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rfsm_comments_phrase		44	1	3	←		→	13	0	83	19
⇒ <i> ? </i> State: 84 state type: <i>r</i>												
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rno_of_native_fset_terminals		25	2	2	←		→	58	0	84	20

\Rightarrow <i>int-no</i>		State: 85 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rno.of.native.fset.terminals 25 1 2		58 0 85 20	
\Rightarrow <i>identifier</i>		State: 86 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Ra_T 28 1 2		59 0 86 20	
\Rightarrow <i>tth-in-stbl</i>		State: 87 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Ra_T 28 2 2		59 0 87 20	
\Rightarrow <i>Rfset_terminals</i>		State: 88 state type: <i>s/r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rfset.terminals.epi 26 1 2		59 0 88 12	
c Ra_T 28 1 1 identifier		88 86 86	
c Ra_T 28 2 1 tth-in-stbl		88 87 87	
t Rfset.terminals 27 2 2 Ra_T		59 89 89	
\Rightarrow <i>Ra_T</i>		State: 89 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rfset.terminals 27 2 3		59 0 89 20	
\Rightarrow <i>Ra_T</i>		State: 90 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rfset.terminals 27 1 2		59 0 90 20	
\Rightarrow <i> ? </i>		State: 91 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rno.of.transitive.threads 32 2 2		66 0 91 21	
\Rightarrow <i>int-no</i>		State: 92 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rno.of.transitive.threads 32 1 2		66 0 92 21	
\Rightarrow <i>identifier</i>		State: 93 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Ra.th 35 1 2		67 0 93 21	
\Rightarrow <i>th-in-stbl</i>		State: 94 state type: <i>r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Ra.th 35 2 2		67 0 94 21	
\Rightarrow <i>Rtransitive_threads</i>		State: 95 state type: <i>s/r</i>	
← rule → R# sr# Po ←	subrule element	→ Brn Gto Red LA	
t Rtransitive_threads.epi 33 1 2		67 0 95 14	
c Ra.th 35 1 1 identifier		95 93 93	
c Ra.th 35 2 1 th-in-stbl		95 94 94	
t Rtransitive_threads 34 2 2 Ra.th		67 96 96	
\Rightarrow <i>Ra.th</i>		State: 96 state type: <i>r</i>	

<p>← rule → R# sr# Po ← t Rtransitive_threads 34 2 3</p>	<p>subrule element</p>	<p>→ Brn Gto Red LA 67 0 96 21</p>
<p>⇒ <i>Ra_th</i></p>		
<p>← rule → R# sr# Po ← t Rtransitive_threads 34 1 2</p>	<p>State: 97 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 67 0 97 21</p>
<p>⇒ <i> ? </i></p>		
<p>← rule → R# sr# Po ← t Rno_of_used_threads 39 2 2</p>	<p>State: 98 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 74 0 98 22</p>
<p>⇒ <i>int-no</i></p>		
<p>← rule → R# sr# Po ← t Rno_of_used_threads 39 1 2</p>	<p>State: 99 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 74 0 99 22</p>
<p>⇒ <i>identifier</i></p>		
<p>← rule → R# sr# Po ← t Rb.th 42 1 2</p>	<p>State: 100 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 75 0 100 22</p>
<p>⇒ <i>th-in-stbl</i></p>		
<p>← rule → R# sr# Po ← t Rb.th 42 2 2</p>	<p>State: 101 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 75 0 101 22</p>
<p>⇒ <i>Rused_threads</i></p>		
<p>← rule → R# sr# Po ← t Rused_threads_epi 40 1 2 c Rb.th 42 1 1 identifier c Rb.th 42 2 1 th-in-stbl t Rused_threads 41 2 2 Rb.th</p>	<p>State: 102 state type: <i>s/r</i> subrule element</p>	<p>→ Brn Gto Red LA 75 0 102 16 102 100 100 102 101 101 75 103 103</p>
<p>⇒ <i>Rb_th</i></p>		
<p>← rule → R# sr# Po ← t Rused_threads 41 2 3</p>	<p>State: 103 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 75 0 103 22</p>
<p>⇒ <i>Rb_th</i></p>		
<p>← rule → R# sr# Po ← t Rused_threads 41 1 2</p>	<p>State: 104 state type: <i>r</i> subrule element</p>	<p>→ Brn Gto Red LA 75 0 104 22</p>

94. Index.

ϵ : 56, 69, 82.
 # file-name: 39.
 # fsm-comments: 90.
 # name-space: 21.
 |?: 9, 11, 15, 17, 21, 23, 27, 29, 33, 35, 39, 41,
 45, 47, 51, 53, 61, 64, 66, 74, 77, 79, 87.
 __FILE__: 7, 10, 12, 13, 16, 19, 22, 25, 28, 30, 31,
 34, 36, 37, 40, 43, 46, 48, 49, 52, 55, 59, 62,
 65, 68, 72, 73, 75, 78, 81, 85, 88.
 __LINE__: 7, 10, 12, 13, 16, 19, 22, 25, 28, 30, 31,
 34, 36, 37, 40, 43, 46, 48, 49, 52, 55, 59, 62,
 65, 68, 72, 73, 75, 78, 81, 85, 88.
 action_: 7, 73.
 add_sym_to_stbl: 7, 72.
 ADD_TOKEN_TO_ERROR_QUEUE: 7, 10, 12, 13, 16,
 19, 22, 25, 28, 31, 34, 36, 37, 40, 43, 46, 48, 49,
 52, 55, 59, 62, 65, 68, 72, 73, 75, 78, 81, 88.
 ADD_TOKEN_TO_PRODUCER_QUEUE: 7.
 back_inserter: 7.
 begin: 7.
 c: [12](#), [36](#).
 c-string: 17, 23, 29, 41, 91.
 c_str: 7, 30, 72, 73, 85.
 c_string: 30.
 CAbsLr1Sym: 7, 10, 12, 13, 16, 19, 22, 25, 28,
 31, 34, 36, 37, 40, 43, 46, 48, 49, 52, 55, 59,
 62, 65, 68, 73, 75, 78, 81, 88.
 Cfsc_file: 7, 12, 18, 24, 30, 36, 42, 48, 54, 60,
 67, 72, 73, 80, 85, 86, 91.
 copy: 7.
 defed: 7.
 defined_: 7.
 empty: 7.
 end: 7.
 end-list-of-native-first-set-terminals: 61.■
 end-list-of-transitive-threads: 74.
 end-list-of-used-threads: 87.
 eog: 6.
 Err_already_defined_in_fsc_file: 7.
 Err_bad_T_alphabet: 48.
 Err_bad_T_in_list: 59.
 Err_end_list_native_T_kw_not_present: 62.
 Err_end_list_of_transitive_threads_kw_not_present: 75.■
 88.
 err_entry_: 7, 72, 73.
 Err_epsilon_pass_thru: 7.
 Err_epsilon_pass_thru_no_Ts: 7.
 Err_filename_kw_not_present: 40.
 Err_filename_value_not_present: 43.
 Err_grammar_name_kw_not_present: 16.
 Err_grammar_name_value_not_present: 19.
 Err_list_of_terminals_kw_not_present: 52.
 Err_monolithic_kw_not_present: 34.
 Err_monolithic_value_bad: 36, 37.
 Err_namespace_kw_not_present: 22.
 Err_namespace_value_not_present: 25.
 Err_no_list_of_trans_threads_kw: 65, 78.
 Err_no_of_native_Ts_in_list_not_equal: 7.
 Err_no_of_T_kw_not_present: 46.
 Err_no_of_T_value_not_present: 49.
 Err_no_of_terminals_not_present: 55.
 Err_no_of_threads_in_list_not_equal: 7.
 Err_no_of_threads_not_present: 68, 81.
 Err_no_sym_defs_present: 73.
 Err_threadname_kw_not_present: 28.
 Err_threadname_value_not_present: 31.
 Err_transitive_kw_not_present: 10.
 Err_transitive_value_bad: 12, 13.
 file_name_: 4, 7, 42.
 find: 30, 85, 86.
 find_sym_in_stbl: 7, 73.
 fnd: 7, 73.
 fq_n: 85, 86.
 fsc_file: 2.
 fsm: 7, 12, 18, 24, 30, 36, 42, 48, 54, 60, 67,
 72, 73, 80, 85, 86, 91.
 fsm_comments_: 4, 7, 91.
 fsm_tbl_: 7, 12, 18, 24, 30, 36, 42, 48, 54, 60,
 67, 72, 73, 80, 85, 86, 91.
 fully_qualified_th_name_: 4, 7, 30, 73, 85, 86.
 gnm: 30.
 grammar_name: 15.
 GRAMMAR_DICTIONARY: 5, 7, 72.
 grammar_file_name_: 4, 7, 18.
 identifier: 11, 35, 58, 71, 84.
 identifier: 12, 36, 72, 85.
 int-no: 47, 53, 66, 79.
 iterator: 30.
 list-of-native-first-set-terminals: 51.
 list-of-transitive-threads: 64.
 list-of-used-threads: 77.
 list_of_transitive_threads_: 4, 7, 72, 73.
 list_of_Ts_: 4, 7, 60.
 list_of_used_threads_: 4, 85, 86.
 map: [5](#), 30.
 monolithic: 33.
 monolithic_: [4](#), 7, 36.
 name_space_name_: 4, 7, 24, 30.
 no: 7, 48.
 no-of-T: 45.
 no_native_T_: 4, 7, 54.
 no_of_T_: 4, 48.

- no_T*: [7](#).
- no_th*: [7](#).
- no_threads_*: [4](#), [7](#), [67](#).
- not_fnd*: [73](#).
- NS_yacco2_k_symbols**: [7](#).
- NS_yacco2_terminals*: [4](#), [5](#).
- okay*: [7](#), [72](#), [73](#).
- parser_*: [7](#), [10](#), [12](#), [13](#), [16](#), [18](#), [19](#), [22](#), [24](#), [25](#),
[28](#), [30](#), [31](#), [34](#), [36](#), [37](#), [40](#), [42](#), [43](#), [46](#), [48](#), [49](#),
[52](#), [54](#), [55](#), [59](#), [60](#), [62](#), [65](#), [67](#), [68](#), [72](#), [73](#), [75](#),
[78](#), [80](#), [81](#), [85](#), [86](#), [88](#), [91](#).
- pos_*: [7](#), [72](#).
- PTR_LR1_eog_*: [7](#).
- push_back*: [7](#), [60](#), [72](#), [73](#), [85](#), [86](#).
- p1_*: [10](#), [12](#), [13](#), [16](#), [18](#), [19](#), [22](#), [24](#), [25](#), [28](#), [30](#),
[31](#), [34](#), [36](#), [37](#), [40](#), [42](#), [43](#), [46](#), [48](#), [49](#), [52](#), [54](#),
[55](#), [59](#), [60](#), [62](#), [65](#), [67](#), [68](#), [72](#), [73](#), [75](#), [78](#), [80](#),
[81](#), [85](#), [86](#), [88](#), [91](#).
- Ra_T*: [57](#).
- Ra_th*: [70](#).
- Ra_T*: [58](#), [59](#), [60](#).
- Ra_th*: [71](#), [72](#), [73](#).
- Rb_th*: [83](#).
- Rb.th*: [84](#), [85](#), [86](#).
- Rend_list_of_native_fset_terminals*: [50](#).
- Rend_list_of_transitive_threads*: [63](#).
- Rend_list_of_used_threads*: [76](#).
- Rend_list_of_native_fset_terminals*: [61](#), [62](#).
- Rend_list_of_transitive_threads*: [74](#), [75](#).
- Rend_list_of_used_threads*: [87](#), [88](#).
- report_card*: [7](#), [72](#), [73](#), [85](#).
- Rfilename*: [39](#), [40](#).
- Rfilename*: [38](#).
- Rfilename_phrase*: [6](#).
- Rfilename_value*: [38](#).
- Rfilename_phrase*: [38](#).
- Rfilename_value*: [41](#), [42](#), [43](#).
- Rfsc_file*: [6](#).
- Rfset_terminals*: [56](#), [57](#).
- Rfset_terminals_epi*: [50](#).
- Rfset_terminals*: [57](#).
- Rfset_terminals_epi*: [56](#).
- Rfsm_comments*: [89](#).
- Rfsm_comments_phrase*: [6](#).
- Rfsm_comments_value*: [89](#).
- Rfsm_comments*: [90](#).
- Rfsm_comments_phrase*: [89](#).
- Rfsm_comments_value*: [91](#).
- Rgrammar_name*: [14](#).
- Rgrammar_name_phrase*: [6](#).
- Rgrammar_name_value*: [14](#).
- Rgrammar_name*: [15](#), [16](#).
- Rgrammar_name_phrase*: [14](#).
- Rgrammar_name_value*: [17](#), [18](#), [19](#).
- Rlist_of_native_fset_terminals*: [50](#).
- Rlist_of_native_terminals_phrase*: [6](#).
- Rlist_of_transitive_threads*: [63](#).
- Rlist_of_transitive_threads_phrase*: [6](#).
- Rlist_of_used_threads*: [76](#).
- Rlist_of_used_threads_phrase*: [6](#).
- Rlist_of_native_fset_terminals*: [51](#), [52](#).
- Rlist_of_native_terminals_phrase*: [50](#).
- Rlist_of_transitive_threads*: [64](#), [65](#).
- Rlist_of_transitive_threads_phrase*: [63](#).
- Rlist_of_used_threads*: [77](#), [78](#).
- Rlist_of_used_threads_phrase*: [76](#).
- Rmonolithic*: [33](#), [34](#).
- Rmonolithic*: [32](#).
- Rmonolithic_phrase*: [6](#).
- Rmonolithic_value*: [32](#).
- Rmonolithic_phrase*: [32](#).
- Rmonolithic_value*: [35](#), [36](#), [37](#).
- Rnamespace*: [20](#).
- Rnamespace*: [21](#), [22](#).
- Rnamespace_phrase*: [6](#).
- Rnamespace_value*: [20](#).
- Rnamespace_phrase*: [20](#).
- Rnamespace_value*: [23](#), [24](#), [25](#).
- Rno_of_native_fset_terminals*: [50](#).
- Rno_of_T*: [44](#).
- Rno_of_T_phrase*: [6](#).
- Rno_of_T_value*: [44](#).
- Rno_of_transitive_threads*: [63](#).
- Rno_of_used_threads*: [76](#).
- Rno_of_native_fset_terminals*: [53](#), [54](#), [55](#).
- Rno_of_T*: [45](#), [46](#).
- Rno_of_T_phrase*: [44](#).
- Rno_of_T_value*: [47](#), [48](#), [49](#).
- Rno_of_transitive_threads*: [66](#), [67](#), [68](#).
- Rno_of_used_threads*: [79](#), [80](#), [81](#).
- Rthreadname*: [27](#), [28](#).
- Rthreadname*: [26](#).
- Rthreadname_phrase*: [6](#).
- Rthreadname_value*: [26](#).
- Rthreadname_phrase*: [26](#).
- Rthreadname_value*: [29](#), [30](#), [31](#).
- Rtransitive*: [9](#), [10](#).
- Rtransitive*: [8](#).
- Rtransitive_phrase*: [6](#).
- Rtransitive_threads*: [69](#), [70](#).
- Rtransitive_threads_epi*: [63](#).
- Rtransitive_value*: [8](#).
- Rtransitive_phrase*: [8](#).
- Rtransitive_threads*: [70](#).

- Rtransitive_threads_epi*: [69](#).
Rtransitive_value: [11](#), [12](#), [13](#).
rule_info_: [7](#), [10](#), [12](#), [13](#), [16](#), [18](#), [19](#), [22](#), [24](#), [25](#),
[28](#), [30](#), [31](#), [34](#), [36](#), [37](#), [40](#), [42](#), [43](#), [46](#), [48](#), [49](#),
[52](#), [54](#), [55](#), [59](#), [60](#), [62](#), [65](#), [67](#), [68](#), [72](#), [73](#), [75](#),
[78](#), [80](#), [81](#), [85](#), [86](#), [88](#), [91](#).
Rused_threads: [82](#), [83](#).
Rused_threads_epi: [76](#).
Rused_threads: [83](#).
Rused_threads_epi: [82](#).
second: [30](#).
set_rc: [7](#), [10](#), [12](#), [13](#), [16](#), [19](#), [22](#), [25](#), [28](#), [30](#), [31](#), [34](#),
[36](#), [37](#), [40](#), [43](#), [46](#), [48](#), [49](#), [52](#), [55](#), [59](#), [62](#), [65](#),
[68](#), [72](#), [73](#), [75](#), [78](#), [81](#), [85](#), [88](#).
set_stop_parse: [7](#), [10](#), [12](#), [13](#), [16](#), [19](#), [22](#), [25](#), [28](#),
[31](#), [34](#), [36](#), [37](#), [40](#), [43](#), [46](#), [48](#), [49](#), [52](#), [55](#), [59](#),
[62](#), [65](#), [68](#), [72](#), [73](#), [75](#), [78](#), [81](#), [88](#).
sf: [10](#), [12](#), [13](#), [16](#), [18](#), [19](#), [22](#), [24](#), [25](#), [28](#), [30](#),
[31](#), [34](#), [36](#), [37](#), [40](#), [42](#), [43](#), [46](#), [48](#), [49](#), [52](#), [54](#),
[55](#), [59](#), [60](#), [62](#), [65](#), [67](#), [68](#), [72](#), [73](#), [75](#), [78](#), [80](#),
[81](#), [85](#), [86](#), [88](#), [91](#).
size: [7](#), [12](#), [36](#), [48](#), [72](#), [85](#), [86](#).
size_type: [85](#), [86](#).
status_: [7](#), [72](#), [73](#).
stbl_idx: [7](#), [72](#).
std: [4](#), [5](#), [30](#), [85](#), [86](#).
string: [4](#), [5](#), [30](#), [85](#), [86](#).
sym: [7](#), [10](#), [12](#), [13](#), [16](#), [19](#), [22](#), [25](#), [28](#), [31](#), [34](#),
[36](#), [37](#), [40](#), [43](#), [46](#), [48](#), [49](#), [52](#), [55](#), [59](#), [62](#), [65](#),
[68](#), [73](#), [75](#), [78](#), [81](#), [88](#).
symbol_: [7](#).
szc_T: [7](#).
szc_th: [7](#).
T_att: [60](#).
T_attributes: [60](#).
T_c_string: [4](#).
T_DICTIONARY: [5](#), [48](#).
T_enum_: [60](#).
T_identifier: [12](#), [30](#), [36](#), [59](#), [72](#), [85](#).
t_in_stbl: [60](#).
T_int_no: [4](#).
T_sym_tbl_report_card: [7](#), [72](#), [73](#), [85](#).
table_entry: [5](#), [7](#), [72](#).
tbl_entry_: [7](#), [72](#), [73](#).
th: [72](#), [73](#), [85](#), [86](#).
th-in-stbl: [71](#), [84](#).
th_entry: [7](#).
th_enum_: [7](#), [72](#).
th_id: [7](#), [30](#).
th_in_stbl: [7](#), [72](#), [73](#), [86](#).
th_in_tbl: [7](#).
thread: [7](#), [72](#).
thread-name: [27](#).
thread_attributes: [4](#), [7](#), [72](#), [73](#), [85](#), [86](#).
thread_in_stbl: [7](#), [73](#), [86](#).
thread_name_: [4](#), [7](#), [30](#).
tnm: [85](#), [86](#).
transience: [7](#).
transitive: [9](#).
transitive_: [4](#), [7](#), [12](#).
true: [7](#), [10](#), [12](#), [13](#), [16](#), [19](#), [22](#), [25](#), [28](#), [31](#), [34](#),
[36](#), [37](#), [40](#), [43](#), [46](#), [48](#), [49](#), [52](#), [55](#), [59](#), [62](#), [65](#),
[68](#), [72](#), [73](#), [75](#), [78](#), [81](#), [88](#).
tth-in-stbl: [58](#).
tth_in_stbl: [60](#).
used: [72](#).
used_: [73](#).
used_no_threads_: [4](#), [80](#).
used_threads_: [4](#), [30](#), [85](#), [86](#).
USED_THREADS_LIST: [5](#), [30](#).
vector: [4](#), [5](#), [30](#).
xx: [85](#), [86](#).
yacco2: [7](#).
yacco2-stbl: [7](#), [72](#), [73](#), [85](#).

< Cfsc_file user-declaration directive 4 >
< Cfsc_file user-prefix-declaration directive 5 >
< Ra_T subrule 1 op directive 59 >
< Ra_T subrule 2 op directive 60 >
< Ra_th subrule 1 op directive 72 >
< Ra_th subrule 2 op directive 73 >
< Rb_th subrule 1 op directive 85 >
< Rb_th subrule 2 op directive 86 >
< Rend_list_of_native_fset_terminals subrule 2 op directive 62 >
< Rend_list_of_transitive_threads subrule 2 op directive 75 >
< Rend_list_of_used_threads subrule 2 op directive 88 >
< Rfilename subrule 2 op directive 40 >
< Rfilename_value subrule 1 op directive 42 >
< Rfilename_value subrule 2 op directive 43 >
< Rfsc_file op directive 7 >
< Rfsm_comments_value subrule 1 op directive 91 >
< Rgrammar_name subrule 2 op directive 16 >
< Rgrammar_name_value subrule 1 op directive 18 >
< Rgrammar_name_value subrule 2 op directive 19 >
< Rlist_of_native_fset_terminals subrule 2 op directive 52 >
< Rlist_of_transitive_threads subrule 2 op directive 65 >
< Rlist_of_used_threads subrule 2 op directive 78 >
< Rmonolithic subrule 2 op directive 34 >
< Rmonolithic_value subrule 1 op directive 36 >
< Rmonolithic_value subrule 2 op directive 37 >
< Rnamespace subrule 2 op directive 22 >
< Rnamespace_value subrule 1 op directive 24 >
< Rnamespace_value subrule 2 op directive 25 >
< Rno_of_T subrule 2 op directive 46 >
< Rno_of_T_value subrule 1 op directive 48 >
< Rno_of_T_value subrule 2 op directive 49 >
< Rno_of_native_fset_terminals subrule 1 op directive 54 >
< Rno_of_native_fset_terminals subrule 2 op directive 55 >
< Rno_of_transitive_threads subrule 1 op directive 67 >
< Rno_of_transitive_threads subrule 2 op directive 68 >
< Rno_of_used_threads subrule 1 op directive 80 >
< Rno_of_used_threads subrule 2 op directive 81 >
< Rthreadname subrule 2 op directive 28 >
< Rthreadname_value subrule 1 op directive 30 >
< Rthreadname_value subrule 2 op directive 31 >
< Rtransitive subrule 2 op directive 10 >
< Rtransitive_value subrule 1 op directive 12 >
< Rtransitive_value subrule 2 op directive 13 >

fsc_file Grammar

Date: January 2, 2015 at 15:35

File: fsc_file.lex

Ns: NS_fsc_file

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

O₂linker's "fsc" control file parser.

<i>fsc_file</i> stand alone Grammar	2	2
Fsm Cfsc_file class	3	2
Cfsc_file user-declaration directive	4	2
Cfsc_file user-prefix-declaration directive	5	3
Rfsc_file rule	6	3
Rfsc_file op directive	7	4
Rtransitive_phrase rule	8	5
Rtransitive rule	9	5
Rtransitive's subrule 2	10	6
Rtransitive_value rule	11	6
Rtransitive_value's subrule 1	12	6
Rtransitive_value's subrule 2	13	6
Rgrammar_name_phrase rule	14	7
Rgrammar_name rule	15	7
Rgrammar_name's subrule 2	16	7
Rgrammar_name_value rule	17	7
Rgrammar_name_value's subrule 1	18	7
Rgrammar_name_value's subrule 2	19	7
Rnamespace_phrase rule	20	8
Rnamespace rule	21	8
Rnamespace's subrule 2	22	8
Rnamespace_value rule	23	8
Rnamespace_value's subrule 1	24	8
Rnamespace_value's subrule 2	25	8
Rthreadname_phrase rule	26	9
Rthreadname rule	27	9
Rthreadname's subrule 2	28	9
Rthreadname_value rule	29	9
Rthreadname_value's subrule 1	30	9
Rthreadname_value's subrule 2	31	10
Rmonolithic_phrase rule	32	10
Rmonolithic rule	33	10
Rmonolithic's subrule 2	34	10
Rmonolithic_value rule	35	10
Rmonolithic_value's subrule 1	36	11
Rmonolithic_value's subrule 2	37	11
Rfilename_phrase rule	38	11
Rfilename rule	39	11
Rfilename's subrule 2	40	12
Rfilename_value rule	41	12
Rfilename_value's subrule 1	42	12
Rfilename_value's subrule 2	43	12
Rno_of_T_phrase rule	44	12
Rno_of_T rule	45	12
Rno_of_T's subrule 2	46	13
Rno_of_T_value rule	47	13
Rno_of_T_value's subrule 1	48	13
Rno_of_T_value's subrule 2	49	13
Rlist_of_native_terminals_phrase rule	50	13
Rlist_of_native_fset_terminals rule	51	14
Rlist_of_native_fset_terminals's subrule 2	52	14
Rno_of_native_fset_terminals rule	53	14

<i>Rno_of_native_fset_terminals</i> 's subrule 1	54	14
<i>Rno_of_native_fset_terminals</i> 's subrule 2	55	14
<i>Rfset_terminals_epi</i> rule	56	14
<i>Rfset_terminals</i> rule	57	15
<i>Ra_T</i> rule	58	15
<i>Ra_T</i> 's subrule 1	59	15
<i>Ra_T</i> 's subrule 2	60	15
<i>Rend_list_of_native_fset_terminals</i> rule	61	15
<i>Rend_list_of_native_fset_terminals</i> 's subrule 2	62	16
<i>Rlist_of_transitive_threads_phrase</i> rule	63	16
<i>Rlist_of_transitive_threads</i> rule	64	16
<i>Rlist_of_transitive_threads</i> 's subrule 2	65	16
<i>Rno_of_transitive_threads</i> rule	66	16
<i>Rno_of_transitive_threads</i> 's subrule 1	67	16
<i>Rno_of_transitive_threads</i> 's subrule 2	68	17
<i>Rtransitive_threads_epi</i> rule	69	17
<i>Rtransitive_threads</i> rule	70	17
<i>Ra_th</i> rule	71	17
<i>Ra_th</i> 's subrule 1	72	18
<i>Ra_th</i> 's subrule 2	73	19
<i>Rend_list_of_transitive_threads</i> rule	74	19
<i>Rend_list_of_transitive_threads</i> 's subrule 2	75	19
<i>Rlist_of_used_threads_phrase</i> rule	76	20
<i>Rlist_of_used_threads</i> rule	77	20
<i>Rlist_of_used_threads</i> 's subrule 2	78	20
<i>Rno_of_used_threads</i> rule	79	20
<i>Rno_of_used_threads</i> 's subrule 1	80	20
<i>Rno_of_used_threads</i> 's subrule 2	81	20
<i>Rused_threads_epi</i> rule	82	21
<i>Rused_threads</i> rule	83	21
<i>Rb_th</i> rule	84	21
<i>Rb_th</i> 's subrule 1	85	21
<i>Rb_th</i> 's subrule 2	86	22
<i>Rend_list_of_used_threads</i> rule	87	22
<i>Rend_list_of_used_threads</i> 's subrule 2	88	22
<i>Rfsm_comments_phrase</i> rule	89	22
<i>Rfsm_comments</i> rule	90	22
<i>Rfsm_comments_value</i> rule	91	22
First Set Language for O_2^{linker}	92	23
Lr1 State Network	93	24
Index	94	34