

1. Copyright.

Copyright © Dave Bone 1998 - 2015

2. *t_alphabet* grammar.

Parse Linker's t-alphabet language. It's just a list of terminal names. For the Yacco2 grammars, it looks like this:

```
T-alphabet
  LR1_questionable_shift_operator
  LR1_eog
  LR1_eolr
  ...
  T_file_of_T_alphabet
end-T-alphabet
```

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

⟨*Ct_alphabet* constructor directive 4⟩ ≡
enum_cnt_ = 0;

5. *Ct_alphabet* user-declaration directive.

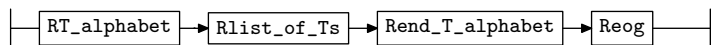
⟨*Ct_alphabet* user-declaration directive 5⟩ ≡
public: *std::map* < *int* , *std::string* > *enum_T_dictionary_;* *std::map* < *std::string* ,
int > *T_dictionary_;*
int *enum_cnt_;*

6. *Ct_alphabet* user-prefix-declaration directive.

⟨*Ct_alphabet* user-prefix-declaration directive 6⟩ ≡
include "yacco2_stbl.h" **extern** *std::vector* < *NS_yacco2_terminals::table_entry* * > *T_DICTIONARY;*
extern *std::vector* < *std::set* < *int* >> *T_THREAD_ID_LIST;*

7. *Rt_alphabet* rule.

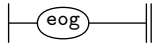
Rt_alphabet

**8. *Rt_alphabet* op directive.**

⟨*Rt_alphabet* op directive 8⟩ ≡
using namespace *NS_yacco2_k_symbols;*
*ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2::PTR_LR1_eog_);*
*ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2::PTR_LR1_eog_);*

9. Reog rule.

Reog



⟨ Reog subrule 1 op directive 9 ⟩ ≡

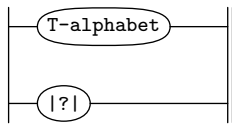
```

Ct_alphabet * fsm = ( Ct_alphabet * ) rule_info...parser--fsm_tbl_;
if ( fsm→T_dictionary.empty() ) {
  CAbs_lr1_sym * sym = new Err_no_terminals_present_in_T_alphabet;
  sym→set_rc(*sf→p1__, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info...parser--set_stop_parse(true);
}

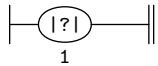
```

10. RT_alphabet rule.

RT_alphabet



11. RT_alphabet's subrule 2.



⟨ RT_alphabet subrule 2 op directive 11 ⟩ ≡

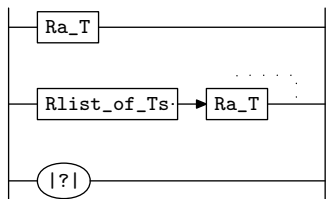
```

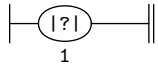
CAbs_lr1_sym * sym = new Err_T_alphabet_kw_not_present;
sym→set_rc(*sf→p1__, __FILE__, __LINE__);
ADD_TOKEN_TO_ERROR_QUEUE(*sym);
rule_info...parser--set_stop_parse(true);

```

12. Rlist_of_Ts rule.

Rlist_of_Ts



13. *Rlist_of_Ts*'s subrule 3.

no terminals in list:

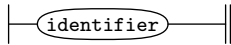
REMEMBER THERE IS NO ASSOCIATED CO-ORDINATES, USE CURRENT TOKEN.

\langle Rlist_of_Ts subrule 3 op directive 13 $\rangle \equiv$

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

14. *Ra_T* rule.

Ra_T

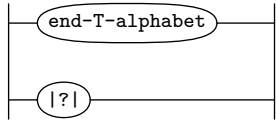


\langle Ra_T subrule 1 op directive 14 $\rangle \equiv$

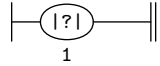
```
Ct_alphabet * fsm = ( Ct_alphabet * ) rule_info_.parser->fsm_tbl_;
T_identifier * k = sf->p1--; /* add to T dictionary */
std::map < std::string , int > ::iterator i = fsm->T_dictionary_.find(k->identifier()-c_str());
if (i != fsm->T_dictionary_.end()) {
    CAbs_lr1_sym * sym = new Err_dup_entry_in_sym_table;
    sym->set_rc(*sf->p1--, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*sym);
    rule_info_.parser->set_stop_parse(true);
    return;
}
fsm->T_dictionary_[k->identifier()-c_str()] = fsm->enum_cnt_;
string xxx(k->identifier()-c_str());
fsm->enum_T_dictionary_[fsm->enum_cnt_] = xxx;
using namespace yacco2_stbl;
T_sym_tbl_report_card report_card;
T_attributes * T_att = new T_attributes(k->identifier()-c_str(), fsm->enum_cnt_);
tth_in_stbl * t = new tth_in_stbl(T_att, k, rule_info_.parser_);
add_sym_to_stbl(report_card, *k->identifier()-c_str(), *t, table_entry::defed, table_entry::terminal);
if (report_card.status_ != T_sym_tbl_report_card::okay) {
    report_card.err_entry->set_rc(*sf->p1--, __FILE__, __LINE__);
    ADD_TOKEN_TO_ERROR_QUEUE(*report_card.err_entry_);
    rule_info_.parser->set_stop_parse(true);
    return;
}
t->stbl_idx(report_card.pos_);
T_DICTIONARY.push_back(report_card.tbl_entry_); T_THREAD_ID_LIST.push_back ( std::set < int > ( ) );
++fsm->enum_cnt_;
```

15. Rend_T_alphabet rule.

Rend_T_alphabet



16. Rend_T_alphabet's subrule 2.



Random characters or eog to err or not to ?

```

⟨ Rend_T_alphabet subrule 2 op directive 16 ⟩ ≡
  CAbs_lr1_sym * sym = new Err_end_T_alphabet_kw_not_present;
  sym->set_rc(*sf-p1--, __FILE__, __LINE__);
  ADD_TOKEN_TO_ERROR_QUEUE(*sym);
  rule_info--.parser-->set_stop_parse(true);

```

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

```
/*
  File: t_alphabet.fsc
  Date and Time: Fri Jan  2 15:33:57 2015
*/
transitive      n
grammar-name    "t_alphabet"
name-space      "NS_t_alphabet"
thread-name     "Ct_alphabet"
monolithic      y
file-name       "t_alphabet.fsc"
no-of-T         569
list-of-native-first-set-terminals 2
  LR1_questionable_shift_operator
  T_T_alphabet
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
"Parse Linker's t-alphabet language."
```

18. Lr1 State Network.

\Rightarrow					State: 1 state type: s		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
c	RT_alphabet		3 2 1	?			1 2 2
c	RT_alphabet		3 1 1	T_alphabet			1 3 3
c	Rt_alphabet		1 1 1	RT_alphabet	<u>Rlist_of_Ts</u>		1 4 11
\Rightarrow	?				State: 2 state type: r		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
t	RT_alphabet		3 2 2				1 0 2 1
\Rightarrow	T_alphabet				State: 3 state type: r		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
t	RT_alphabet		3 1 2				1 0 3 1
\Rightarrow	RT_alphabet				State: 4 state type: s		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
c	Rlist_of_Ts		4 3 1	?			4 12 12
c	Ra_T		5 1 1	identifier			4 7 7
t	Rt_alphabet		1 1 2	Rlist_of_Ts	<u>Rend_T_alphabet</u>		1 5 11
c	Rlist_of_Ts		4 2 1	Rlist_of_Ts	<u>Ra_T</u>		4 5 13
c	Rlist_of_Ts		4 1 1	Ra_T			4 14 14
\Rightarrow	Rlist_of_Ts				State: 5 state type: s		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
c	Rend_T_alphabet		6 2 1	?			5 6 6
c	Ra_T		5 1 1	identifier			5 7 7
c	Rend_T_alphabet		6 1 1	end-T_alphabet			5 8 8
t	Rlist_of_Ts		4 2 2	Ra_T			4 13 13
t	Rt_alphabet		1 1 3	Rend_T_alphabet	<u>Reog</u>		1 9 11
\Rightarrow	?				State: 6 state type: r		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
t	Rend_T_alphabet		6 2 2				5 0 6 2
\Rightarrow	identifier				State: 7 state type: r		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
t	Ra_T		5 1 2				5 0 7 3
\Rightarrow	end-T_alphabet				State: 8 state type: r		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
t	Rend_T_alphabet		6 1 2				5 0 8 2
\Rightarrow	Rend_T_alphabet				State: 9 state type: s		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
c	Reog		2 1 1	eog			9 10 10
t	Rt_alphabet		1 1 4	Reog			1 11 11
\Rightarrow	eog				State: 10 state type: r		
\leftarrow	rule	\rightarrow	R# sr# Po	\leftarrow	subrule element	\rightarrow	Brn Gto Red LA
t	Reog		2 1 2				9 0 10 4

\Rightarrow_{Reog}

←	rule	→	R#	sr#	Po	←
t	Rt.alphabet		1	1	5	

State: 11 state type: *r*
 subrule element

→	Brn	Gto	Red	LA
1	0	11	4	

 $\Rightarrow_{|?|}$

←	rule	→	R#	sr#	Po	←
t	Rlist_of.Ts		4	3	2	

State: 12 state type: *r*
 subrule element

→	Brn	Gto	Red	LA
4	0	12	3	

 \Rightarrow_{Ra_T}

←	rule	→	R#	sr#	Po	←
t	Rlist_of.Ts		4	2	3	

State: 13 state type: *r*
 subrule element

→	Brn	Gto	Red	LA
4	0	13	3	

 \Rightarrow_{Ra_T}

←	rule	→	R#	sr#	Po	←
t	Rlist_of.Ts		4	1	2	

State: 14 state type: *r*
 subrule element

→	Brn	Gto	Red	LA
4	0	14	3	

19. Index.

|?|: 10, 12, 15.
 __FILE__: 9, 11, 13, 14, 16.
 __LINE__: 9, 11, 13, 14, 16.
 add_sym_to_stbl: 14.
 ADD_TOKEN_TO_ERROR_QUEUE: 9, 11, 13, 14, 16.
 ADD_TOKEN_TO_PRODUCER_QUEUE: 8.
 c_str: 14.
 CAbs_lr1_sym: 9, 11, 13, 14, 16.
 Ct_alphabet: 9, 14.
 defed: 14.
 empty: 9.
 end: 14.
 end-T-alphabet: 15.
 enum_cnt: 4, 5, 14.
 enum_T_dictionary: 5, 14.
 eog: 9.
 Err_dup_entry_in_sym_table: 14.
 Err_end_T_alphabet_kw_not_present: 16.
 err_entry: 14.
 Err_no_terminals_in_T_alphabet_list: 13.
 Err_no_terminals_present_in_T_alphabet: 9.
 Err_T_alphabet_kw_not_present: 11.
 find: 14.
 fsm: 9, 14.
 fsm_tbl: 9, 14.
 identifier: 14.
 identifier: 14.
 iterator: 14.
 map: 5, 14.
 NS_yacco2_k_symbols: 8.
 NS_yacco2_terminals: 6.
 okay: 14.
 parser_: 9, 11, 13, 14, 16.
 pos: 14.
 PTR_LR1_eog_: 8.
 push_back: 14.
 p1_: 9, 11, 13, 14, 16.
 Ra_T: 12.
 Ra_T: 14.
 Rend_T_alphabet: 7.
 Rend_T_alphabet: 15, 16.
 Reog: 7.
 Reog: 9.
 report_card: 14.
 Rlist_of_Ts: 7, 12.
 Rlist_of_Ts: 12, 13.
 RT_alphabet: 7.
 Rt_alphabet: 7.
 RT_alphabet: 10, 11.
 rule_info_: 9, 11, 13, 14, 16.
 set: 6, 14.
 set_rc: 9, 11, 13, 14, 16.
 set_stop_parse: 9, 11, 13, 14, 16.
 sf: 9, 11, 13, 14, 16.
 status_: 14.
 stbl_idx: 14.
 std: 5, 6, 14.
 string: 5, 14.
 sym: 9, 11, 13, 14, 16.
 T-alphabet: 10.
 t_alphabet: 2.
 T_att: 14.
 T_attributes: 14.
 T_DICTIONARY: 6, 14.
 T_dictionary: 5, 9, 14.
 T_identifier: 14.
 T_sym_tbl_report_card: 14.
 T_THREAD_ID_LIST: 6, 14.
 table_entry: 6, 14.
 tbl_entry: 14.
 terminal: 14.
 true: 9, 11, 13, 14, 16.
 tth_in_stbl: 14.
 vector: 6.
 xxx: 14.
 yacco2: 8.
 yacco2_stbl: 14.

⟨ Ct_alphabet constructor directive 4 ⟩
⟨ Ct_alphabet user-declaration directive 5 ⟩
⟨ Ct_alphabet user-prefix-declaration directive 6 ⟩
⟨ RT_alphabet subrule 2 op directive 11 ⟩
⟨ Ra_T subrule 1 op directive 14 ⟩
⟨ Rend_T_alphabet subrule 2 op directive 16 ⟩
⟨ Reog subrule 1 op directive 9 ⟩
⟨ Rlist_of_Ts subrule 3 op directive 13 ⟩
⟨ Rt_alphabet op directive 8 ⟩

t_alphabet Grammar

Date: January 2, 2015 at 15:39

File: t_alphabet.lex

Ns: NS_t_alphabet

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

Parse Linker's t-alphabet language.

	Section	Page
Copyright	1	1
<i>t_alphabet</i> grammar	2	2
Fsm Ct_alphabet class	3	2
Ct_alphabet constructor directive	4	2
Ct_alphabet user-declaration directive	5	2
Ct_alphabet user-prefix-declaration directive	6	2
<i>Rt_alphabet</i> rule	7	2
Rt_alphabet op directive	8	2
<i>Reog</i> rule	9	3
<i>RT_alphabet</i> rule	10	3
<i>RT_alphabet</i> 's subrule 2	11	3
<i>Rlist_of_Ts</i> rule	12	3
<i>Rlist_of_Ts</i> 's subrule 3	13	4
<i>Ra_T</i> rule	14	4
<i>Rend_T_alphabet</i> rule	15	5
<i>Rend_T_alphabet</i> 's subrule 2	16	5
First Set Language for O_2^{linker}	17	6
Lr1 State Network	18	7
Index	19	9