

**1. Copyright.**

Copyright © Dave Bone 1998 - 2014

**2. *T\_enum\_phrase* grammar.**

Dispatcher to parse T-enumeration construct.

Note: Use of `||| |?|` phrase to catch other returned terminals or errors from called threads. In this case, it is errors so its raison d'être instead of `|+|` as a “catch all”. One can be selective within various contexts:

- 1) within arbitration
- 2) sub-rule level

The symbol's enumerated id is the fodder to filter on.

**3. Fsm *CT\_enum\_phrase* class.****4. *CT\_enum\_phrase* constructor directive.**

```
<CT_enum_phrase constructor directive 4> ≡
  cweb_marker_ = 0;
```

**5. *CT\_enum\_phrase* op directive.**

```
<CT_enum_phrase op directive 5> ≡
  if (CWEB_MARKER ≠ 0) {
    cweb_marker_ = CWEB_MARKER;
    CWEB_MARKER = 0;
  }
```

**6. *CT\_enum\_phrase* user-declaration directive.**

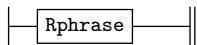
```
<CT_enum_phrase user-declaration directive 6> ≡
public: yacco2::AST * cweb_marker_;
```

**7. *CT\_enum\_phrase* user-prefix-declaration directive.**

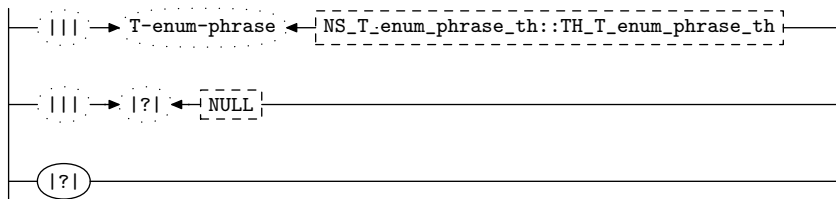
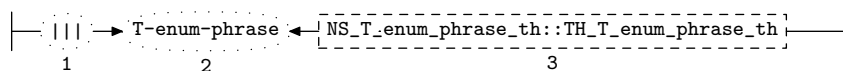
```
<CT_enum_phrase user-prefix-declaration directive 7> ≡
#include "T_enum_phrase_th.h"
#include "o2_externs.h"
```

**8. *RT\_enum\_phrase* rule.**

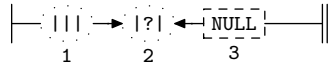
*RT\_enum\_phrase*

**9. *Rphrase* rule.**

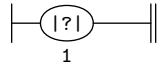
*Rphrase*

**10. *Rphrase*'s subrule 1.**

```
<Rphrase subrule 1 op directive 10> ≡
  ADD_TOKEN_TO_PRODUCER_QUEUE(*sf->p2_);
```

**11. Rphrase's subrule 2.**

⟨Rphrase subrule 2 op directive 11⟩ ≡  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE(\*sf→p2\_);

**12. Rphrase's subrule 3.**

⟨Rphrase subrule 3 op directive 12⟩ ≡  
 CAbs\_lr1\_sym \* sym = new Err\_no\_open\_parenthesis;  
 sym→set\_rc(\*rule\_info→parser→current\_token(), \_\_FILE\_\_, \_\_LINE\_\_);  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE(\*sym);

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

```
/*
  File: T_enum_phrase.fsc
  Date and Time: Sun May 11 09:59:33 2014
*/
transitive      y
grammar-name    "T_enum_phrase"
name-space      "NS_T_enum_phrase"
thread-name     "CT_enum_phrase"
monolithic      y
file-name       "T_enum_phrase.fsc"
no-of-T         569
list-of-native-first-set-terminals 1
  LR1_questionable_shift_operator
end-list-of-native-first-set-terminals
list-of-transitive-threads 1
  NS_T_enum_phrase_th::TH_T_enum_phrase_th
end-list-of-transitive-threads
list-of-used-threads 1
  NS_T_enum_phrase_th::TH_T_enum_phrase_th
end-list-of-used-threads
fsm-comments
"Dispatcher to parse T-enumeration construct."
```

14. Lr1 State Network.

⇒					State: 1 state type: <i>s</i>			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
c	Rphrase		2 3 1	?			1 2 2	
c	Rphrase		2 2 1	?  NULL			1 3 4	
c	Rphrase		2 1 1	T-enum-phrase NS_T_enum_phrase_th::TH_T_enum_phrase_th			1 3 5	
c	RT_enum_phrase		1 1 1	Rphrase			1 6 6	
⇒	?				State: 2 state type: <i>r</i>			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	Rphrase		2 3 2				1 0 2 1	
⇒					State: 3 state type: <i>s</i>			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	Rphrase		2 2 2	?			1 4 4	
t	Rphrase		2 1 2	T-enum-phrase			1 5 5	
⇒	?				State: 4 state type: <i>r</i>			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	Rphrase		2 2 3				1 0 4 1	
⇒	<i>T-enum-phrase</i>				State: 5 state type: <i>r</i>			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	Rphrase		2 1 3				1 0 5 1	
⇒	<i>Rphrase</i>				State: 6 state type: <i>r</i>			
←	rule	→	R# sr# Po	←	subrule element	→	Brn Gto Red LA	
t	RT_enum_phrase		1 1 2				1 0 6 1	

**15. Index.**

|||: 9.  
|?|: 9.  
\_\_FILE\_\_: 12.  
\_\_LINE\_\_: 12.  
ADD\_TOKEN\_TO\_ERROR\_QUEUE: 11, 12.  
ADD\_TOKEN\_TO\_PRODUCER\_QUEUE: 10.  
AST: 6.  
*CAbs\_lr1\_sym*: 12.  
*current\_token*: 12.  
CWEB\_MARKER: 5.  
*cweb\_marker\_*: 4, 5, 6.  
*Err\_no\_open\_parenthesis*: 12.  
NS\_T\_enum\_phrase\_th::TH\_T\_enum\_phrase\_th: 9.■  
NULL: 9.  
*parser\_*: 12.  
*p2\_*: 10, 11.  
*Rphrase*: 9, 10, 11, 12.  
Rphrase: 8.  
*RT\_enum\_phrase*: 8.  
*rule\_info\_*: 12.  
*set\_rc*: 12.  
*sf*: 10, 11.  
*sym*: 12.  
T-enum-phrase: 9.  
*T\_enum\_phrase*: 2.  
*yacco2*: 6.

- ⟨ CT\_enum\_phrase constructor directive 4 ⟩
- ⟨ CT\_enum\_phrase op directive 5 ⟩
- ⟨ CT\_enum\_phrase user-declaration directive 6 ⟩
- ⟨ CT\_enum\_phrase user-prefix-declaration directive 7 ⟩
- ⟨ Rphrase subrule 1 op directive 10 ⟩
- ⟨ Rphrase subrule 2 op directive 11 ⟩
- ⟨ Rphrase subrule 3 op directive 12 ⟩

# T\_enum\_phrase Grammar

Date: May 17, 2014 at 13:33

File: T\_enum\_phrase.lex

Ns: NS\_T\_enum\_phrase

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

Dispatcher to parse T-enumeration construct.

	Section	Page
<b>Copyright</b> .....	<a href="#">1</a>	1
<i>T_enum_phrase</i> <b>grammar</b> .....	<a href="#">2</a>	2
Fsm CT_enum_phrase class .....	<a href="#">3</a>	2
CT_enum_phrase constructor directive .....	<a href="#">4</a>	2
CT_enum_phrase op directive .....	<a href="#">5</a>	2
CT_enum_phrase user-declaration directive .....	<a href="#">6</a>	2
CT_enum_phrase user-prefix-declaration directive .....	<a href="#">7</a>	2
<i>RT_enum_phrase</i> rule .....	<a href="#">8</a>	2
<i>Rphrase</i> rule .....	<a href="#">9</a>	2
<i>Rphrase</i> 's subrule 1 .....	<a href="#">10</a>	2
<i>Rphrase</i> 's subrule 2 .....	<a href="#">11</a>	3
<i>Rphrase</i> 's subrule 3 .....	<a href="#">12</a>	3
<b>First Set Language for <math>O_2^{linker}</math></b> .....	<a href="#">13</a>	4
<b>Lr1 State Network</b> .....	<a href="#">14</a>	5
<b>Index</b> .....	<a href="#">15</a>	6