

**1. Copyright.**

Copyright © Dave Bone 1998 - 2015

**2. *link\_cleanser* grammar.**

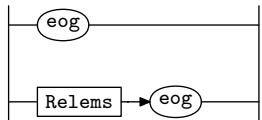
Lexical Phase for Linker. This is a front end to  $O_2$ 's linker. It throws out the chaffe and puts the balance into a holding container for parsing. This is my old style of parsing but now the 2 stages has been brought into 1 stage like *pass3*. It use to be *pass1..pass3* grammars.

**3. Fsm Clink\_cleanser class.****4. Clink\_cleanser user-prefix-declaration directive.**

```
<Clink_cleanser user-prefix-declaration directive 4> ≡
#include "linker_id.h"
#include "ws.h"
#include "c_comments.h"
#include "c_string.h"
#include "bad_char_set.h"
#include "eol.h"
#include "int_no.h"
```

**5. *Rpass3* rule.**

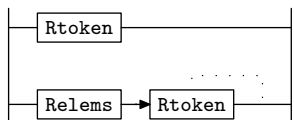
Rpass3

**6. Rpass3 op directive.**

```
<Rpass3 op directive 6> ≡
using namespace NS_yacco2_k_symbols;
ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2 :: PTR_LR1_eog_);
ADD_TOKEN_TO_PRODUCER_QUEUE(*yacco2 :: PTR_LR1_eog_);
```

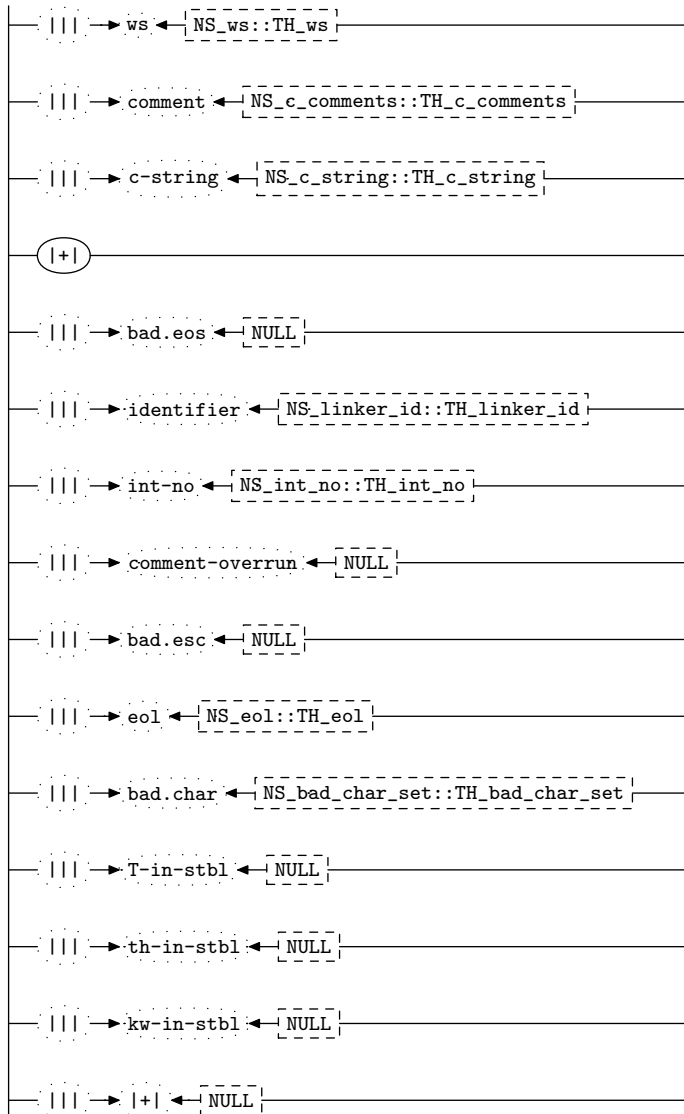
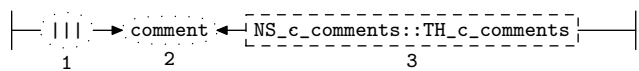
**7. *Relems* rule.**

Relems



8. *Rtoken* rule.

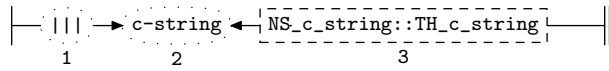
Rtoken

9. *Rtoken*'s subrule 2.

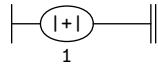
⟨Rtoken subrule 2 op directive 9⟩ ≡

*T\_comment* \* *k* = *sf-p2\_*;

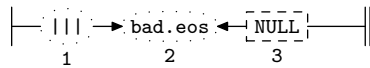
*k*→*set\_auto\_delete*(*true*);

10. *R*token's subrule 3.

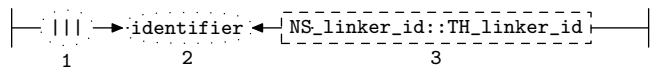
⟨Rtoken subrule 3 op directive 10⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf→p2\_);

11. *R*token's subrule 4.

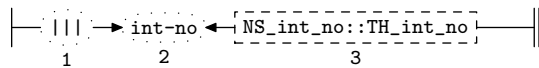
⟨Rtoken subrule 4 op directive 11⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf→p1\_);

12. *R*token's subrule 5.

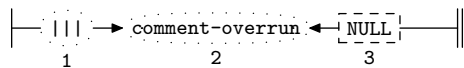
⟨Rtoken subrule 5 op directive 12⟩ ≡  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE(\*sf→p2\_);  
 rule\_info...parser\_→set\_abort\_parse(true);

13. *R*token's subrule 6.

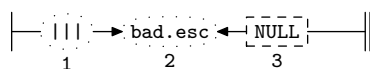
⟨Rtoken subrule 6 op directive 13⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf→p2\_);

14. *R*token's subrule 7.

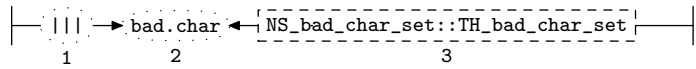
⟨Rtoken subrule 7 op directive 14⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\*sf→p2\_);

15. *R*token's subrule 8.

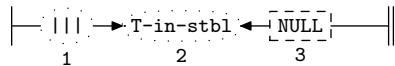
⟨Rtoken subrule 8 op directive 15⟩ ≡  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE(\*sf→p2\_);  
 rule\_info...parser\_→set\_abort\_parse(true);

16. *R*token's subrule 9.

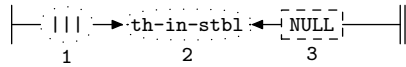
⟨Rtoken subrule 9 op directive 16⟩ ≡  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE(\*sf→p2\_);  
 rule\_info...parser\_→set\_abort\_parse(true);

**17. Rtoken's subrule 11.**

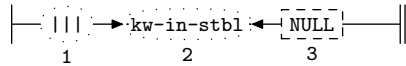
⟨ Rtoken subrule 11 op directive 17 ⟩ ≡  
*Err\_bad\_char* \* *k* = *sf-p2--*;  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE(\**k*);  
*rule\_info--parser--set\_abort\_parse*(*true*);

**18. Rtoken's subrule 12.**

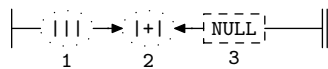
⟨ Rtoken subrule 12 op directive 18 ⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\**sf-p2--*);

**19. Rtoken's subrule 13.**

⟨ Rtoken subrule 13 op directive 19 ⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\**sf-p2--*);

**20. Rtoken's subrule 14.**

⟨ Rtoken subrule 14 op directive 20 ⟩ ≡  
*kw\_in\_stbl* \* *kw* = *sf-p2--*;  
*CAbs\_lr1\_sym* \* *sym* = *kw-keyword\_in\_stbl*();  
*sym-set\_rc*(\**rule\_info--parser--current\_token*()), *\_\_FILE\_\_*, *\_\_LINE\_\_*);  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\**sym*);

**21. Rtoken's subrule 15.**

⟨ Rtoken subrule 15 op directive 21 ⟩ ≡  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE(\**sf-p2--*);

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

```

/*
  File: link_cleanser.fsc
  Date and Time: Fri Jan  2 15:33:40 2015
*/
transitive      y
grammar-name    "link_cleanser"
name-space     "NS_link_cleanser"
thread-name    "Clink_cleanser"
monolithic     y
file-name      "link_cleanser.fsc"
no-of-T        569
list-of-native-first-set-terminals 2
  LR1_eog
  LR1_all_shift_operator
end-list-of-native-first-set-terminals
list-of-transitive-threads 7
  NS_linker_id::TH_linker_id
  NS_eol::TH_eol
  NS_bad_char_set::TH_bad_char_set
  NS_ws::TH_ws
  NS_c_comments::TH_c_comments
  NS_c_string::TH_c_string
  NS_int_no::TH_int_no
end-list-of-transitive-threads
list-of-used-threads 7
  NS_bad_char_set::TH_bad_char_set
  NS_c_comments::TH_c_comments
  NS_c_string::TH_c_string
  NS_eol::TH_eol
  NS_int_no::TH_int_no
  NS_linker_id::TH_linker_id
  NS_ws::TH_ws
end-list-of-used-threads
fsm-comments
"Lexer: \\olinker's cleanser from \n previous lexing to remove chaffe before parsing
stage."

```

## 23. Lr1 State Network.

⇒ State: 1 state type: <sup>s</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Rpass3		1	1	1		eog		1	2	2	
c	Rtoken		3	1	1		ws NS_ws::TH_ws		1	3	7	
c	Rtoken		3	3	1		c-string NS_c_string::TH_c_string		1	3	8	
c	Rtoken		3	6	1		identifier NS_linker_id::TH_linker_id		1	3	9	
c	Rtoken		3	12	1		T-in-stbl NULL		1	3	11	
c	Rtoken		3	14	1		kw-in-stbl NULL		1	3	13	
c	Rtoken		3	15	1		+  NULL		1	3	4	
c	Rtoken		3	2	1		comment NS_c_comments::TH_c_comments		1	3	6	
c	Rtoken		3	5	1		bad eos NULL		1	3	14	
c	Rtoken		3	7	1		int-no NS_int_no::TH_int_no		1	3	10	
c	Rtoken		3	8	1		comment-overflow NULL		1	3	16	
c	Rtoken		3	9	1		bad esc NULL		1	3	15	
c	Rtoken		3	10	1		eol NS_eol::TH_eol		1	3	5	
c	Rtoken		3	11	1		bad char NS_bad_char_set::TH_bad_char_set		1	3	17	
c	Rtoken		3	13	1		th-in-stbl NULL		1	3	12	
c	Rtoken		3	4	1	+			1	18	18	
c	Rpass3		1	2	1		Relems <u>eog</u>		1	19	20	
c	Relems		2	2	1		Relems <u>Rtoken</u>		1	19	21	
c	Relems		2	1	1		Rtoken		1	22	22	

⇒<sup>eog</sup> State: 2 state type: <sup>r</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpass3		1	1	2				1	0	2	1

⇒<sup>||| arbitration-code: ε</sup> State: 3 state type: <sup>s</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rtoken		3	15	2	+			1	4	4	
t	Rtoken		3	10	2	eol			1	5	5	
t	Rtoken		3	2	2	comment			1	6	6	
t	Rtoken		3	1	2	ws			1	7	7	
t	Rtoken		3	3	2	c-string			1	8	8	
t	Rtoken		3	6	2	identifier			1	9	9	
t	Rtoken		3	7	2	int-no			1	10	10	
t	Rtoken		3	12	2	T-in-stbl			1	11	11	
t	Rtoken		3	13	2	th-in-stbl			1	12	12	
t	Rtoken		3	14	2	kw-in-stbl			1	13	13	
t	Rtoken		3	5	2	bad eos			1	14	14	
t	Rtoken		3	9	2	bad esc			1	15	15	
t	Rtoken		3	8	2	comment-overflow			1	16	16	
t	Rtoken		3	11	2	bad char			1	17	17	

⇒<sup>|+|</sup> State: 4 state type: <sup>r</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rtoken		3	15	3				1	0	4	2

⇒<sup>eol</sup> State: 5 state type: <sup>r</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rtoken		3	10	3				1	0	5	2

$\Rightarrow$ comment					State: 6 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	2	3			1	0 6 2
$\Rightarrow$ ws					State: 7 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	1	3			1	0 7 2
$\Rightarrow$ c-string					State: 8 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	3	3			1	0 8 2
$\Rightarrow$ identifier					State: 9 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	6	3			1	0 9 2
$\Rightarrow$ int-no					State: 10 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	7	3			1	0 10 2
$\Rightarrow$ T-in-stbl					State: 11 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	12	3			1	0 11 2
$\Rightarrow$ th-in-stbl					State: 12 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	13	3			1	0 12 2
$\Rightarrow$ kw-in-stbl					State: 13 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	14	3			1	0 13 2
$\Rightarrow$ badeos					State: 14 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	5	3			1	0 14 2
$\Rightarrow$ badesc					State: 15 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	9	3			1	0 15 2
$\Rightarrow$ comment-overrun					State: 16 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	8	3			1	0 16 2
$\Rightarrow$ badchar					State: 17 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	11	3			1	0 17 2
$\Rightarrow$  +					State: 18 state type: $r$		
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto Red LA
t Rtoken	3	4	2			1	0 18 2



⇒ *Relems* State: 19 state type: <sup>s</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpass3		1	2	2		eog		1	20	20	
c	Rtoken		3	1	1		ws NS_ws::TH_ws		19	3	7	
c	Rtoken		3	3	1		c-string NS_c_string::TH_c_string		19	3	8	
c	Rtoken		3	6	1		identifier NS_linker_id::TH_linker_id		19	3	9	
c	Rtoken		3	12	1		T-in-stbl NULL		19	3	11	
c	Rtoken		3	14	1		kw-in-stbl NULL		19	3	13	
c	Rtoken		3	15	1		+  NULL		19	3	4	
c	Rtoken		3	2	1		comment NS_c_comments::TH_c_comments		19	3	6	
c	Rtoken		3	5	1		bad eos NULL		19	3	14	
c	Rtoken		3	7	1		int-no NS_int_no::TH_int_no		19	3	10	
c	Rtoken		3	8	1		comment-overrun NULL		19	3	16	
c	Rtoken		3	9	1		bad esc NULL		19	3	15	
c	Rtoken		3	10	1		eol NS_eol::TH_eol		19	3	5	
c	Rtoken		3	11	1		bad char NS_bad_char_set::TH_bad_char_set		19	3	17	
c	Rtoken		3	13	1		th-in-stbl NULL		19	3	12	
c	Rtoken		3	4	1	+			19	18	18	
t	Relems		2	2	2		Rtoken		1	21	21	

⇒ *eog* State: 20 state type: <sup>r</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rpass3		1	2	3				1	0	20	1

⇒ *Rtoken* State: 21 state type: <sup>r</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Relems		2	2	3				1	0	21	2

⇒ *Rtoken* State: 22 state type: <sup>r</sup>

←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Relems		2	1	2				1	0	22	2

**24. Index.**

|+|: 8.  
 |||: 8.  
 \_\_FILE\_\_: 20.  
 \_\_LINE\_\_: 20.  
 ADD\_TOKEN\_TO\_ERROR\_QUEUE: 12, 15, 16, 17.  
 ADD\_TOKEN\_TO\_PRODUCER\_QUEUE: 6, 10, 11, 13,  
     14, 18, 19, 20, 21.  
 bad char: 8.  
 bad eos: 8.  
 bad esc: 8.  
 c-string: 8.  
 CAbs\_lr1\_sym: 20.  
 comment: 8.  
 comment-overflow: 8.  
 current\_token: 20.  
 eog: 5.  
 eol: 8.  
 Err\_bad\_char: 17.  
 identifier: 8.  
 int-no: 8.  
 keyword\_in\_stbl: 20.  
 kw: 20.  
 kw-in-stbl: 8.  
 kw\_in\_stbl: 20.  
 link\_cleanser: 2.  
 NS\_bad\_char\_set::TH\_bad\_char\_set: 8.  
 NS\_c\_comments::TH\_c\_comments: 8.  
 NS\_c\_string::TH\_c\_string: 8.  
 NS\_eol::TH\_eol: 8.  
 NS\_int\_no::TH\_int\_no: 8.  
 NS\_linker\_id::TH\_linker\_id: 8.  
 NS\_ws::TH\_ws: 8.  
 NS\_yacco2\_k\_symbols: 6.  
 NULL: 8.  
 parser\_: 12, 15, 16, 17, 20.  
 pass1: 2.  
 pass3: 2.  
 PTR\_LR1\_eog\_: 6.  
 p1\_: 11.  
 p2\_: 9, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21.  
 Relems: 5, 7.  
 Relems: 7.  
 Rpass3: 5.  
 Rtoken: 7.  
 Rtoken: 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,  
     18, 19, 20, 21.  
 rule\_info\_: 12, 15, 16, 17, 20.  
 set\_abort\_parse: 12, 15, 16, 17.  
 set\_auto\_delete: 9.  
 set\_rc: 20.  
 sf: 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21.  
 sym: 20.  
 T-in-stbl: 8.  
 T\_comment: 9.  
 th-in-stbl: 8.  
 true: 9, 12, 15, 16, 17.  
 ws: 8.  
 yacco2: 6.

⟨ Clink\_cleanser user-prefix-declaration directive 4 ⟩  
⟨ Rpass3 op directive 6 ⟩  
⟨ Rtoken subrule 11 op directive 17 ⟩  
⟨ Rtoken subrule 12 op directive 18 ⟩  
⟨ Rtoken subrule 13 op directive 19 ⟩  
⟨ Rtoken subrule 14 op directive 20 ⟩  
⟨ Rtoken subrule 15 op directive 21 ⟩  
⟨ Rtoken subrule 2 op directive 9 ⟩  
⟨ Rtoken subrule 3 op directive 10 ⟩  
⟨ Rtoken subrule 4 op directive 11 ⟩  
⟨ Rtoken subrule 5 op directive 12 ⟩  
⟨ Rtoken subrule 6 op directive 13 ⟩  
⟨ Rtoken subrule 7 op directive 14 ⟩  
⟨ Rtoken subrule 8 op directive 15 ⟩  
⟨ Rtoken subrule 9 op directive 16 ⟩

# link\_cleanser Grammar

Date: January 2, 2015 at 15:36

File: link\_cleanser.lex

Ns: NS\_link\_cleanser

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

Lexer:  $O_2^{linker}$ 's cleanser from previous lexing to remove chaffe before parsing stage.

	Section	Page
<b>Copyright</b> .....	1	1
<i>link_cleanser</i> <b>grammar</b> .....	2	2
Fsm Clink_cleanser class .....	3	2
Clink_cleanser user-prefix-declaration directive .....	4	2
<i>Rpass3</i> rule .....	5	2
Rpass3 op directive .....	6	2
<i>Relems</i> rule .....	7	2
<i>Rtoken</i> rule .....	8	3
<i>Rtoken</i> 's subrule 2 .....	9	3
<i>Rtoken</i> 's subrule 3 .....	10	4
<i>Rtoken</i> 's subrule 4 .....	11	4
<i>Rtoken</i> 's subrule 5 .....	12	4
<i>Rtoken</i> 's subrule 6 .....	13	4
<i>Rtoken</i> 's subrule 7 .....	14	4
<i>Rtoken</i> 's subrule 8 .....	15	4
<i>Rtoken</i> 's subrule 9 .....	16	4
<i>Rtoken</i> 's subrule 11 .....	17	5
<i>Rtoken</i> 's subrule 12 .....	18	5
<i>Rtoken</i> 's subrule 13 .....	19	5
<i>Rtoken</i> 's subrule 14 .....	20	5
<i>Rtoken</i> 's subrule 15 .....	21	5
<b>First Set Language for <math>O_2^{linker}</math></b> .....	22	6
<b>Lr1 State Network</b> .....	23	7
<b>Index</b> .....	24	10