

**1. Copyright.**

Copyright © Dave Bone 1998 - 2015

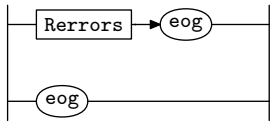
2. *o2\_err\_hdlr* grammar.

Grammar that processes for printing posted errors from yacc2.

3. Fsm Co2\_err\_hdlr class.

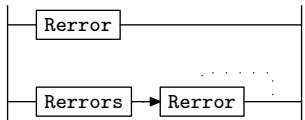
4. *Ro2\_err\_hdlr* rule.

Ro2\_err\_hdlr



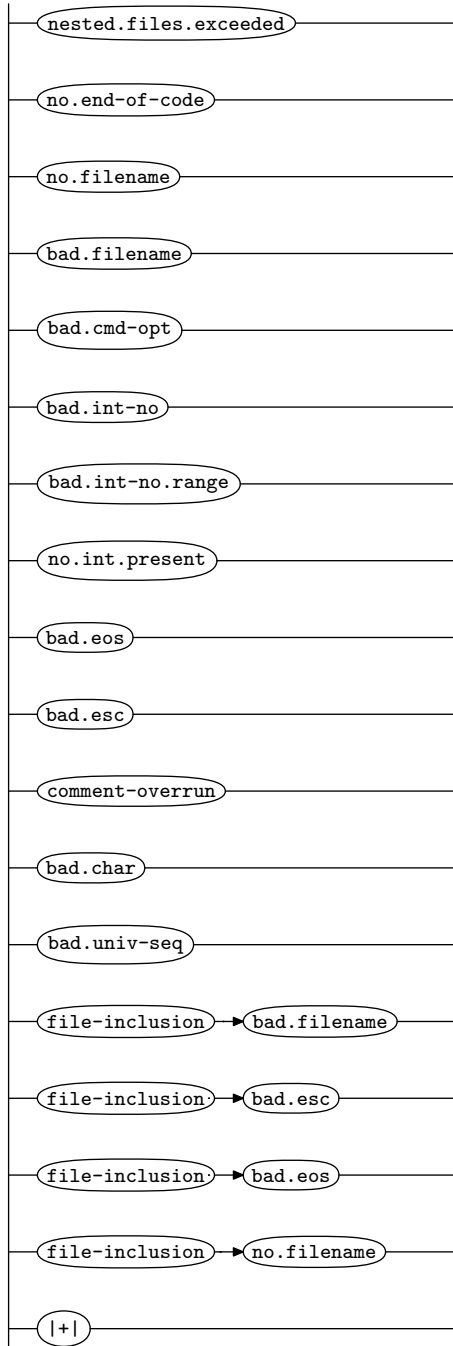
5. *Rerrors* rule.

Rerrors



**6. *Rerror* rule.**

Rerror



## 7. Rerror user-declaration directive.

```

⟨Rerror user-declaration directive 7⟩ ≡
void error_where(CAbs_lr1_sym * E_sym)
{
    std::string & ext_fle = yacco2::FILE_TBL__[E_sym-tok_co_ords...external_file_id_];
    std::string line_of_data;
    std::ifstream ifile;
    ifile.open(ext_fle.c_str());
    if (ifile.good()) {
        yacco2::UINT lno(1);
        yacco2::UINT dlno(E_sym-tok_co_ords...line_no_);
        for (; lno ≤ dlno; ++lno) {
            getline(ifile, line_of_data);
            if (lno ≡ dlno) break;
            line_of_data.clear();
        }
    }
    std::string space("\u");
    std::string::size_type f = line_of_data.find_first_of('\t');
    for (; f ≠ std::string::npos; ) {
        line_of_data.replace(f, 1, space);
        f = line_of_data.find_first_of('\t');
    }
    yacco2::lrclog << "Error\uin\ufile#:\u" << E_sym-tok_co_ords...external_file_id_ << "\u\" <<
        ext_fle.c_str() << "\u\" << std::endl;
    yacco2::lrclog << line_of_data.c_str() << std::endl;
    std::cout << "Error\uin\ufile#:\u" << E_sym-tok_co_ords...external_file_id_ << "\u\" <<
        ext_fle.c_str() << "\u\" << std::endl;
    std::cout << line_of_data.c_str() << std::endl;
    for (int pos = 1; pos < E_sym-tok_co_ords...pos_in_line_; ++pos) {
        yacco2::lrclog << '\u';
        std::cout << '\u';
    }
    yacco2::lrclog << '^' << std::endl;
    yacco2::lrclog << "\tfpos:\u" << E_sym-tok_co_ords...rc_pos_ << "\u" <<
        E_sym-tok_co_ords...line_no_ << "\ucpos:\u" << E_sym-tok_co_ords...pos_in_line_ << std::endl;
    std::cout << '^' << std::endl;
    std::cout << "\tfpos:\u" << E_sym-tok_co_ords...rc_pos_ << "\u" <<
        E_sym-tok_co_ords...line_no_ << "\ucpos:\u" << E_sym-tok_co_ords...pos_in_line_ << std::endl;
    if (E_sym-tok_co_ords...who_file_ ≠ 0) {
        yacco2::lrclog << "\twho\uthru\uit:\u" << E_sym-tok_co_ords...who_file_ << "\u" <<
            E_sym-tok_co_ords...who_line_no_ << std::endl;
        std::cout << "\twho\uthru\uit:\u" << E_sym-tok_co_ords...who_file_ << "\u" <<
            E_sym-tok_co_ords...who_line_no_ << std::endl;
    }
    ifile.close();
}
;

```

**8. Error's subrule 1.**

| nested.files.exceeded |  
1

⟨ Error subrule 1 op directive 8 ⟩ ≡

```
error_where(sf-p1--);
yacco2 :: lrclg << "\t" << sf-p1--id_ << "_nested_" << sf-p1--nested_cnt() << std :: endl;
std :: cout << "\t" << sf-p1--id_ << "_nested_" << sf-p1--nested_cnt() << std :: endl;
```

**9. Error's subrule 2.**

| no.end-of-code |  
1

⟨ Error subrule 2 op directive 9 ⟩ ≡

```
error_where(sf-p1--);
yacco2 :: lrclg << "\t" << sf-p1--id_ << std :: endl;
std :: cout << "\t" << sf-p1--id_ << std :: endl;
```

**10. Error's subrule 3.**

| no.filename |  
1

⟨ Error subrule 3 op directive 10 ⟩ ≡

```
error_where(sf-p1--);
yacco2 :: lrclg << "\t" << sf-p1--id_ << std :: endl;
std :: cout << "\t" << sf-p1--id_ << std :: endl;
```

**11. Error's subrule 4.**

| bad.filename |  
1

⟨ Error subrule 4 op directive 11 ⟩ ≡

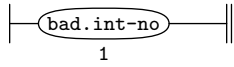
```
error_where(sf-p1--);
yacco2 :: lrclg << "\t" << sf-p1--id_ << "_filename:" << sf-p1--file_name()-c_str() <<
"\_does\_not\_exist" << std :: endl;
std :: cout << "\t" << sf-p1--id_ << "_filename:" << sf-p1--file_name()-c_str() <<
"\_does\_not\_exist" << std :: endl;
```

**12. Error's subrule 5.**

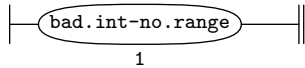
| bad.cmd-opt |  
1

⟨ Error subrule 5 op directive 12 ⟩ ≡

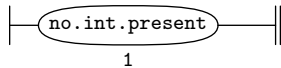
```
error_where(sf-p1--);
yacco2 :: lrclg << "\t" << sf-p1--id_ << std :: endl;
std :: cout << "\t" << sf-p1--id_ << std :: endl;
```

**13. *Error's subrule 6.***

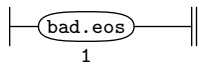
⟨ Error subrule 6 op directive 13 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrclg << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**14. *Error's subrule 7.***

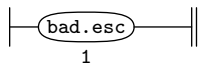
⟨ Error subrule 7 op directive 14 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrclg << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**15. *Error's subrule 8.***

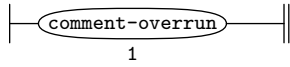
⟨ Error subrule 8 op directive 15 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrclg << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**16. *Error's subrule 9.***

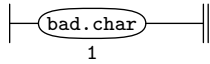
⟨ Error subrule 9 op directive 16 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrclg << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**17. *Error's subrule 10.***

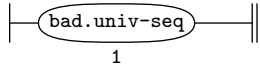
⟨ Error subrule 10 op directive 17 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrclg << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**18. Rerror's subrule 11.**

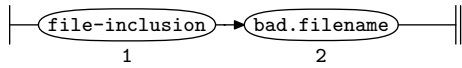
⟨ Rerror subrule 11 op directive 18 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrlog << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**19. Rerror's subrule 12.**

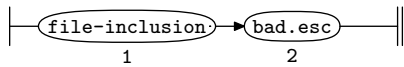
⟨ Rerror subrule 12 op directive 19 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrlog << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**20. Rerror's subrule 13.**

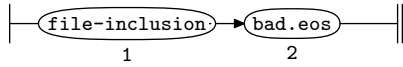
⟨ Rerror subrule 13 op directive 20 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrlog << "\t" << sf-p1--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_ << std :: endl;*

**21. Rerror's subrule 14.**

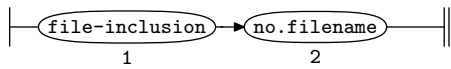
⟨ Rerror subrule 14 op directive 21 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrlog << "\t" << sf-p1--id\_;*  
*yacco2 :: lrlog << "□" << sf-p2--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_;*  
*std :: cout << "□" << sf-p2--id\_ << std :: endl;*

**22. Rerror's subrule 15.**

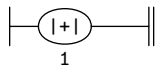
⟨ Rerror subrule 15 op directive 22 ⟩ ≡  
*error\_where(sf-p1\_);*  
*yacco2 :: lrlog << "\t" << sf-p1--id\_;*  
*yacco2 :: lrlog << "□" << sf-p2--id\_ << std :: endl;*  
*std :: cout << "\t" << sf-p1--id\_;*  
*std :: cout << "□" << sf-p2--id\_ << std :: endl;*

**23. *Rerror's* subrule 16.**

⟨ Rerror subrule 16 op directive 23 ⟩ ≡  
*error\_where*(*sf-p1--*);  
*yacco2 :: lrclg* << "\t" << *sf-p1--id--*;  
*yacco2 :: lrclg* << "□" << *sf-p2--id--* << *std :: endl*;  
*std :: cout* << "\t" << *sf-p1--id--*;  
*std :: cout* << "□" << *sf-p2--id--* << *std :: endl*;

**24. *Rerror's* subrule 17.**

⟨ Rerror subrule 17 op directive 24 ⟩ ≡  
*error\_where*(*sf-p1--*);  
*yacco2 :: lrclg* << "\t" << *sf-p1--id--*;  
*yacco2 :: lrclg* << "□" << *sf-p2--id--* << *std :: endl*;  
*std :: cout* << "\t" << *sf-p1--id--*;  
*std :: cout* << "□" << *sf-p2--id--* << *std :: endl*;

**25. *Rerror's* subrule 18.**

⟨ Rerror subrule 18 op directive 25 ⟩ ≡  
*error\_where*(*sf-p1--*);  
*yacco2 :: lrclg* << "\t" << *sf-p1--id--* << *std :: endl*;  
*std :: cout* << "\t" << *sf-p1--id--* << *std :: endl*;



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

```
/*
  File: o2_err_hdlr.fsc
  Date and Time: Fri Jan  2 15:33:44 2015
*/
transitive      n
grammar-name    "o2_err_hdlr"
name-space      "NS_o2_err_hdlr"
thread-name     "Co2_err_hdlr"
monolithic      y
file-name       "o2_err_hdlr.fsc"
no-of-T         569
list-of-native-first-set-terminals 16
  LR1_all_shift_operator
  LR1_eog
  T_file_inclusion
  Err_nested_files_exceeded
  Err_no_end_of_code
  Err_no_filename
  Err_bad_filename
  Err_bad_cmd_lne_opt
  Err_bad_int_no
  Err_bad_int_no_range
  Err_no_int_present
  Err_bad_eos
  Err_bad_esc
  Err_comment_overrun
  Err_bad_char
  Err_bad_univ_seq
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
"Logic sequencer: Print out errors from \\02."
```

## 27. Lr1 State Network.

⇒				State: 1 state type: <sup>s</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
c	Ro2_err_hdlr		1	2	1	←	eog		1	2	2	
c	Rerror		3	18	1	←	+		1	3	3	
c	Rerror		3	15	1	←	file-inclusion		1	4	8	
c	Rerror		3	14	1	←	file-inclusion		1	4	6	
c	Rerror		3	16	1	←	file-inclusion		1	4	7	
c	Rerror		3	17	1	←	file-inclusion		1	4	5	
c	Rerror		3	1	1	←	nested files exceeded		1	9	9	
c	Rerror		3	2	1	←	no end-of-code		1	10	10	
c	Rerror		3	3	1	←	no filename		1	11	11	
c	Rerror		3	4	1	←	bad filename		1	12	12	
c	Rerror		3	5	1	←	bad cmd-opt		1	13	13	
c	Rerror		3	6	1	←	bad int-no		1	14	14	
c	Rerror		3	7	1	←	bad int-no range		1	15	15	
c	Rerror		3	8	1	←	no int present		1	16	16	
c	Rerror		3	9	1	←	bad eos		1	17	17	
c	Rerror		3	10	1	←	bad esc		1	18	18	
c	Rerror		3	11	1	←	comment-overrun		1	19	19	
c	Rerror		3	12	1	←	bad char		1	20	20	
c	Rerror		3	13	1	←	bad univ-seq		1	21	21	
c	Rerrors		2	2	1	←	Rerrors <i>Rerror</i>		1	22	24	
c	Ro2_err_hdlr		1	1	1	←	Rerrors <i>eog</i>		1	22	23	
c	Rerrors		2	1	1	←	Rerror		1	25	25	
⇒ <i>eog</i>				State: 2 state type: <sup>r</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Ro2_err_hdlr		1	2	2				1	0	2	1
⇒  +				State: 3 state type: <sup>r</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rerror		3	18	2				1	0	3	2
⇒ <i>file-inclusion</i>				State: 4 state type: <sup>s</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rerror		3	17	2	←	no filename		1	5	5	
t	Rerror		3	14	2	←	bad filename		1	6	6	
t	Rerror		3	16	2	←	bad eos		1	7	7	
t	Rerror		3	15	2	←	bad esc		1	8	8	
⇒ <i>nofilename</i>				State: 5 state type: <sup>r</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rerror		3	17	3				1	0	5	2
⇒ <i>badfilename</i>				State: 6 state type: <sup>r</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA
t	Rerror		3	14	3				1	0	6	2
⇒ <i>badeos</i>				State: 7 state type: <sup>r</sup>								
←	rule	→	R#	sr#	Po	←	subrule element	→	Brn	Gto	Red	LA

t Rerror	3	16	3		1	0	7	2
⇒ <i>badesc</i>				State: 8 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	15	3			1	0	8 2
⇒ <i>nestedfilesexceeded</i>				State: 9 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	1	2			1	0	9 2
⇒ <i>noend-of-code</i>				State: 10 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	2	2			1	0	10 2
⇒ <i>nofilename</i>				State: 11 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	3	2			1	0	11 2
⇒ <i>badfilename</i>				State: 12 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	4	2			1	0	12 2
⇒ <i>badcmd-opt</i>				State: 13 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	5	2			1	0	13 2
⇒ <i>badint-no</i>				State: 14 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	6	2			1	0	14 2
⇒ <i>badint-norange</i>				State: 15 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	7	2			1	0	15 2
⇒ <i>nointpresent</i>				State: 16 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	8	2			1	0	16 2
⇒ <i>badeos</i>				State: 17 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	9	2			1	0	17 2
⇒ <i>badesc</i>				State: 18 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	10	2			1	0	18 2
⇒ <i>comment-override</i>				State: 19 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA
t Rerror	3	11	2			1	0	19 2
⇒ <i>badchar</i>				State: 20 state type: <i>r</i>				
← rule	→ R#	sr#	Po	←	subrule element	→ Brn	Gto	Red LA

t	Error		3	12	2				1	0	20	2
$\Rightarrow$ <i>baduniv-seq</i>												
←	rule	→	R#	sr#	Po	←	State: 21 state type: <i>r</i>		→	Brn	Gto	Red LA
							subrule element					
t	Error		3	13	2				1	0	21	2
$\Rightarrow$ <i>Rerrors</i>												
←	rule	→	R#	sr#	Po	←	State: 22 state type: <i>s</i>		→	Brn	Gto	Red LA
							subrule element					
t	Ro2_err_hdlr		1	1	2	eog			1	23	23	
c	Error		3	18	1	+			22	3	3	
c	Error		3	15	1	file-inclusion			22	4	8	
c	Error		3	14	1	file-inclusion			22	4	6	
c	Error		3	16	1	file-inclusion			22	4	7	
c	Error		3	17	1	file-inclusion			22	4	5	
c	Error		3	1	1	nested files exceeded			22	9	9	
c	Error		3	2	1	no end-of-code			22	10	10	
c	Error		3	3	1	no filename			22	11	11	
c	Error		3	4	1	bad filename			22	12	12	
c	Error		3	5	1	bad cmd-opt			22	13	13	
c	Error		3	6	1	bad int-no			22	14	14	
c	Error		3	7	1	bad int-no range			22	15	15	
c	Error		3	8	1	no int present			22	16	16	
c	Error		3	9	1	bad eos			22	17	17	
c	Error		3	10	1	bad esc			22	18	18	
c	Error		3	11	1	comment-overflow			22	19	19	
c	Error		3	12	1	bad char			22	20	20	
c	Error		3	13	1	bad univ-seq			22	21	21	
t	Rerrors		2	2	2	Error			1	24	24	
$\Rightarrow$ <i>eog</i>												
←	rule	→	R#	sr#	Po	←	State: 23 state type: <i>r</i>		→	Brn	Gto	Red LA
							subrule element					
t	Ro2_err_hdlr		1	1	3				1	0	23	1
$\Rightarrow$ <i>Rerror</i>												
←	rule	→	R#	sr#	Po	←	State: 24 state type: <i>r</i>		→	Brn	Gto	Red LA
							subrule element					
t	Rerrors		2	2	3				1	0	24	2
$\Rightarrow$ <i>Rerror</i>												
←	rule	→	R#	sr#	Po	←	State: 25 state type: <i>r</i>		→	Brn	Gto	Red LA
							subrule element					
t	Rerrors		2	1	2				1	0	25	2

**28. Index.**

|+|: 6.  
 bad char: 6.  
 bad cmd-opt: 6.  
 bad eos: 6.  
 bad esc: 6.  
 bad filename: 6.  
 bad int-no: 6.  
 bad int-no range: 6.  
 bad univ-seq: 6.  
*c\_str*: 7, 11.  
*CAbs\_lr1\_sym*: 7.  
*clear*: 7.  
*close*: 7.  
 comment-overflow: 6.  
*cout*: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
 19, 20, 21, 22, 23, 24, 25.  
*dlno*: 7.  
*E\_sym*: 7.  
*endl*: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
 19, 20, 21, 22, 23, 24, 25.  
*eog*: 4.  
*error\_where*: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,  
 18, 19, 20, 21, 22, 23, 24, 25.  
*ext\_fle*: 7.  
*external\_file\_id\_*: 7.  
 file-inclusion: 6.  
*file\_name*: 11.  
 FILE\_TBL\_: 7.  
*find\_first\_of*: 7.  
*getline*: 7.  
*good*: 7.  
*id\_*: 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,  
 20, 21, 22, 23, 24, 25.  
*ifile*: 7.  
*ifstream*: 7.  
*line\_no\_*: 7.  
*line\_of\_data*: 7.  
*lno*: 7.  
*lrclg*: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
 19, 20, 21, 22, 23, 24, 25.  
 nested files exceeded: 6.  
*nested\_cnt*: 8.  
 no end-of-code: 6.  
 no filename: 6.  
 no int present: 6.  
*npos*: 7.  
*open*: 7.  
*o2\_err\_hdlr*: 2.  
*pos*: 7.  
*pos\_in\_line\_*: 7.  
*p1\_*: 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,  
 20, 21, 22, 23, 24, 25.  
*p2\_*: 21, 22, 23, 24.  
*rc\_pos\_*: 7.  
*replace*: 7.  
 Rerror: 5.  
*Rerror*: 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
 19, 20, 21, 22, 23, 24, 25.  
*Rerrors*: 5.  
 Rerrors: 4, 5.  
*Ro2\_err\_hdlr*: 4.  
*sf*: 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,  
 20, 21, 22, 23, 24, 25.  
*size\_type*: 7.  
*space*: 7.  
*std*: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
 19, 20, 21, 22, 23, 24, 25.  
*string*: 7.  
*tok\_co\_ords\_*: 7.  
 UINT: 7.  
*who\_file\_*: 7.  
*who\_line\_no\_*: 7.  
*yacco2*: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18,  
 19, 20, 21, 22, 23, 24, 25.

⟨ Error subrule 1 op directive 8 ⟩  
⟨ Error subrule 10 op directive 17 ⟩  
⟨ Error subrule 11 op directive 18 ⟩  
⟨ Error subrule 12 op directive 19 ⟩  
⟨ Error subrule 13 op directive 20 ⟩  
⟨ Error subrule 14 op directive 21 ⟩  
⟨ Error subrule 15 op directive 22 ⟩  
⟨ Error subrule 16 op directive 23 ⟩  
⟨ Error subrule 17 op directive 24 ⟩  
⟨ Error subrule 18 op directive 25 ⟩  
⟨ Error subrule 2 op directive 9 ⟩  
⟨ Error subrule 3 op directive 10 ⟩  
⟨ Error subrule 4 op directive 11 ⟩  
⟨ Error subrule 5 op directive 12 ⟩  
⟨ Error subrule 6 op directive 13 ⟩  
⟨ Error subrule 7 op directive 14 ⟩  
⟨ Error subrule 8 op directive 15 ⟩  
⟨ Error subrule 9 op directive 16 ⟩  
⟨ Error user-declaration directive 7 ⟩

# o2\_err\_hdlr Grammar

Date: January 2, 2015 at 15:37

File: o2\_err\_hdlr.lex

Ns: NS\_o2\_err\_hdlr

Version: 1.0

Debug: false

Grammar Comments:

Type: Monolithic

Logic sequencer: Print out errors from  $O_2$ .

	Section	Page
<b>Copyright</b> .....	<a href="#">1</a>	1
<i>o2_err_hdlr</i> <b>grammar</b> .....	<a href="#">2</a>	2
Fsm Co2_err_hdlr class .....	<a href="#">3</a>	2
<i>Ro2_err_hdlr</i> rule .....	<a href="#">4</a>	2
<i>Rerrors</i> rule .....	<a href="#">5</a>	2
<i>Error</i> rule .....	<a href="#">6</a>	3
Error user-declaration directive .....	<a href="#">7</a>	4
<i>Error</i> 's subrule 1 .....	<a href="#">8</a>	5
<i>Error</i> 's subrule 2 .....	<a href="#">9</a>	5
<i>Error</i> 's subrule 3 .....	<a href="#">10</a>	5
<i>Error</i> 's subrule 4 .....	<a href="#">11</a>	5
<i>Error</i> 's subrule 5 .....	<a href="#">12</a>	5
<i>Error</i> 's subrule 6 .....	<a href="#">13</a>	6
<i>Error</i> 's subrule 7 .....	<a href="#">14</a>	6
<i>Error</i> 's subrule 8 .....	<a href="#">15</a>	6
<i>Error</i> 's subrule 9 .....	<a href="#">16</a>	6
<i>Error</i> 's subrule 10 .....	<a href="#">17</a>	6
<i>Error</i> 's subrule 11 .....	<a href="#">18</a>	7
<i>Error</i> 's subrule 12 .....	<a href="#">19</a>	7
<i>Error</i> 's subrule 13 .....	<a href="#">20</a>	7
<i>Error</i> 's subrule 14 .....	<a href="#">21</a>	7
<i>Error</i> 's subrule 15 .....	<a href="#">22</a>	7
<i>Error</i> 's subrule 16 .....	<a href="#">23</a>	8
<i>Error</i> 's subrule 17 .....	<a href="#">24</a>	8
<i>Error</i> 's subrule 18 .....	<a href="#">25</a>	8
<b>First Set Language for <math>O_2^{linker}</math></b> .....	<a href="#">26</a>	9
<b>Lr1 State Network</b> .....	<a href="#">27</a>	10
<b>Index</b> .....	<a href="#">28</a>	13