

new/usr/src/man/man1/Makefile

1

13186 Wed Jul 16 09:10:05 2014

new/usr/src/man/man1/Makefile

feedback from Hans

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet
9 # at http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011, Richard Lowe
14 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
15 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
16 #
```

```
18 include      $(SRC)/Makefile.master
```

```
20 MANSECT=     1
```

```
22 MANFILES=    acctcom.1      \
23               adb.1          \
24               addbib.1       \
25               alias.1        \
26               allocate.1     \
27               amt.1          \
28               appcert.1      \
29               apptrace.1     \
30               apropos.1      \
31               ar.1           \
32               arch.1         \
33               asa.1          \
34               at.1           \
35               atq.1          \
36               atrm.1         \
37               audioconvert.1 \
38               audiocntl.1    \
39               audioplay.1    \
40               audiorecord.1  \
41               audiotest.1    \
42               auths.1        \
43               awk.1          \
44               banner.1       \
45               basename.1     \
46               bc.1           \
47               bdiff.1        \
48               bfs.1          \
49               break.1        \
50               builtin.1      \
51               cal.1          \
52               calendar.1     \
53               cancel.1       \
54               cat.1          \
55               catman.1       \
56               cd.1           \
57               cdrw.1         \
58               checknr.1      \
59               chgrp.1        \
60               chkey.1        \
61               chmod.1        \
```

new/usr/src/man/man1/Makefile

2

```
61          chown.1            \
62          ckdate.1           \
63          ckgid.1            \
64          ckint.1            \
65          ckitem.1           \
66          ckkeywd.1          \
67          ckpath.1           \
68          ckrange.1          \
69          ckstr.1            \
70          cksum.1            \
71          cktime.1           \
72          ckuid.1            \
73          ckyorn.1           \
74          clear.1            \
75          cmp.1              \
76          col.1              \
77          comm.1             \
78          command.1          \
79          compress.1         \
80          cp.1               \
81          cpio.1             \
82          cputrack.1         \
83          crle.1             \
84          crontab.1          \
85          crypt.1            \
86          csh.1              \
87          csplit.1           \
88          ctags.1            \
89          ctrun.1            \
90          ctstat.1           \
91          ctwatch.1          \
92          cut.1              \
93          date.1             \
94          dc.1               \
95          deallocate.1       \
96          deroff.1           \
97          dhcpinfo.1         \
98          diff.1             \
99          diff3.1            \
100         diffmk.1            \
101         digest.1            \
102         dircmp.1            \
103         dis.1               \
104         disown.1            \
105         dispgid.1           \
106         dispuid.1           \
107         dos2unix.1          \
108         download.1          \
109         dpost.1             \
110         du.1                \
111         dump.1              \
112         dumpcs.1            \
113         echo.1              \
114         ed.1                \
115         egrep.1             \
116         eject.1             \
117         elfdump.1           \
118         elfedit.1           \
119         elfsign.1           \
120         elfwrap.1           \
121         enable.1            \
122         encrypt.1           \
123         enhance.1           \
124         env.1               \
125         eqn.1               \
126         exec.1              \
```

new/usr/src/man/man1/Makefile

```

127      exit.1          \
128      expand.1         \
129      expr.1           \
130      exstr.1          \
131      factor.1         \
132      fdformat.1       \
133      fgrep.1          \
134      file.1           \
135      filesync.1       \
136      find.1           \
137      finger.1         \
138      fmt.1            \
139      fmtmsg.1         \
140      fold.1           \
141      ftp.1            \
142      ftpcount.1       \
143      ftpwho.1         \
144      gcore.1          \
145      gencat.1         \
146      genmsg.1         \
147      getconf.1        \
148      getfacl.1        \
149      getlabel.1       \
150      getopt.1         \
151      getoptcvt.1      \
152      getopts.1        \
153      gettext.1        \
154      gettxt.1         \
155      getzonepath.1    \
156      glob.1           \
157      gprof.1          \
158      grep.1           \
159      groups.1         \
160      hash.1           \
161      head.1           \
162      history.1        \
163      hostid.1         \
164      hostname.1       \
165      iconv.1          \
166      indxbib.1        \
167      Intro.1          \
168      ipcrm.1          \
169      ipcs.1           \
170      isainfo.1        \
171      isalist.1        \
172      jobs.1           \
173      join.1           \
174      kbd.1            \
175      kdestroy.1       \
176      keylogin.1       \
177      keylogout.1      \
178      kill.1           \
179      kinit.1          \
180      klist.1          \
181      kmdb.1           \
182      kmfcfg.1         \
183      kpasswd.1        \
184      krb5-config.1    \
185      ksh93.1          \
186      ktutil.1         \
187      lari.1           \
188      last.1           \
189      lastcomm.1       \
190      ld.1             \
191      ldap.1           \
192      ldapdelete.1     \

```

3

new/usr/src/man/man1/Makefile

```

193      ldaplist.1       \
194      ldapmodify.1     \
195      ldapmodrdn.1     \
196      ldapsearch.1     \
197      ldd.1            \
198      ld.so.1.1        \
199      let.1            \
200      lex.1            \
201      lgrpinfo.1       \
202      limit.1          \
203      line.1           \
204      list_devices.1   \
205      listusers.1      \
206      ln.1             \
207      loadkeys.1       \
208      locale.1         \
209      localedef.1      \
210      logger.1         \
211      login.1          \
212      logname.1        \
213      logout.1         \
214      look.1           \
215      lookbib.1        \
216      lorder.1         \
217      lp.1             \
218      lpstat.1         \
219      ls.1             \
220      m4.1             \
221      mac.1            \
222      mach.1           \
223      machid.1         \
224      madv.so.1.1      \
225      mail.1           \
226      mailcompat.1     \
227      mailq.1          \
228      mailstats.1      \
229      mailx.1          \
230      makekey.1        \
231      man.1            \
232      mandoc.1         \
233      mconnect.1       \
234      mcs.1            \
235      mdb.1            \
236      msg.1            \
237      mkdir.1          \
238      mkmsgs.1         \
239      mktemp.1         \
240      moe.1            \
241      more.1           \
242      mpss.so.1.1      \
243      msgcc.1          \
244      msgcpp.1         \
245      msgcvt.1         \
246      msgfmt.1         \
247      msggen.1         \
248      msgget.1         \
249      mt.1             \
250      mv.1             \
251      nawk.1           \
252      nc.1             \
253      nca.1            \
254      ncab2clf.1       \
255      ncakmod.1        \
256      newform.1        \
257      newgrp.1         \
258      news.1           \

```

4

new/usr/src/man/man1/Makefile

```
259      newtask.1      \
260      nice.1         \
261      nl.1           \
262      nm.1           \
263      nohup.1        \
264      nroff.1         \
265      od.1           \
266      on.1           \
267      optisa.1        \
268      pack.1         \
269      pagesize.1      \
270      pargs.1         \
271      passwd.1        \
272      paste.1         \
273      pathchk.1       \
274      pax.1          \
275      pfexec.1        \
276      pg.1           \
277      pgrep.1         \
278      pkginfo.1       \
279      pkgmk.1         \
280      pkgparam.1      \
281      pkgproto.1      \
282      pkgtrans.1      \
283      pkttool.1       \
284      plabel.1        \
285      plgrp.1         \
286      plimit.1        \
287      pmadvise.1      \
288      pmap.1          \
289      postio.1        \
290      postprint.1     \
291      postreverse.1   \
292      ppgsz.1         \
293      ppriv.1         \
294      pr.1           \
295      praliases.1     \
296      prctl.1         \
297      preap.1         \
298      prex.1         \
299      print.1         \
300      printf.1        \
301      priocntl.1      \
302      proc.1          \
303      prof.1          \
304      profiles.1      \
305      projects.1      \
306      ps.1           \
307      ptree.1         \
308      pvs.1           \
309      pwd.1           \
310      ranlib.1        \
311      rcapstat.1      \
312      rcp.1           \
313      rdist.1         \
314      read.1          \
315      readonly.1      \
316      refer.1         \
317      regcmp.1        \
318      renice.1        \
319      rev.1           \
320      rlogin.1        \
321      rm.1            \
322      rmformat.1      \
323      rmmount.1       \
324      roffbib.1       \
```

5

new/usr/src/man/man1/Makefile

```
325      roles.1        \
326      rpcgen.1        \
327      rsh.1           \
328      runat.1         \
329      rup.1           \
330      ruptime.1       \
331      rusers.1        \
332      rwho.1          \
333      sar.1           \
334      scp.1           \
335      script.1        \
336      sdiff.1         \
337      sed.1           \
338      set.1           \
339      setfacl.1       \
340      setlabel.1      \
341      setpggrp.1      \
342      sftp.1          \
343      shcomp.1        \
344      shell_builtins.1 \
345      shift.1         \
346      size.1          \
347      sleep.1         \
348      smbutil.1       \
349      soelim.1        \
350      sort.1          \
351      sortbib.1       \
352      sotruss.1       \
353      spell.1         \
354      split.1         \
355      srchtxt.1       \
356      ssh.1           \
357      ssh-add.1        \
358      ssh-agent.1     \
359      ssh-http-proxy-connect.1 \
360      ssh-keygen.1     \
361      ssh-keyscan.1   \
362      ssh-socks5-proxy-connect.1 \
363      strchg.1        \
364      strings.1       \
365      strip.1         \
366      stty.1          \
367      sum.1           \
368      suspend.1       \
369      svcprop.1       \
370      svcs.1          \
371      symorder.1      \
372      sys-suspend.1   \
373      tabs.1          \
374      tail.1          \
375      talk.1          \
376      tar.1           \
377      tbl.1           \
378      tcopy.1         \
379      tee.1           \
380      telnet.1        \
381      test.1          \
382      tftp.1          \
383      time.1          \
384      times.1         \
385      timex.1         \
386      tip.1           \
387      tnfdump.1       \
388      tnfxtract.1     \
389      touch.1         \
390      tput.1          \
```

6

new/usr/src/man/man1/Makefile

7

```

391      tr.1          \
392      trap.1       \
393      troff.1      \
394      true.1       \
395      truss.1      \
396      tsort.1      \
397      tty.1        \
398      type.1       \
399      typeset.1    \
400      ul.1         \
401      umask.1      \
402      uname.1      \
403      unifdef.1    \
404      uniq.1       \
405      units.1      \
406      unix2dos.1   \
407      uptime.1     \
408      vacation.1   \
409      vgrind.1     \
410      volcheck.1   \
411      volrmmount.1 \
412      w.1          \
413      wait.1       \
414      wc.1         \
415      which.1      \
416      who.1        \
417      whocalls.1   \
418      whois.1      \
419      write.1      \
420      xargs.1       \
421      xgettext.1   \
422      xstr.1       \
423      yacc.1       \
424      yes.1        \
425      ypcat.1      \
426      ypmatch.1    \
427      yppasswd.1   \
428      ypwhich.1    \
429      zlogin.1     \
430      zonename.1   \
432  MANLINKS=      batch.1 \
433                  bg.1    \
434                  case.1  \
435                  chdir.1 \
436                  checkeq.1 \
437                  continue.1 \
438                  decrypt.1 \
439                  dirname.1 \
440                  dirs.1 \
441                  disable.1 \
442                  dumpkeys.1 \
443                  edit.1 \
444                  errrange.1 \
445                  errdate.1 \
446                  errgid.1 \
447                  errint.1 \
448                  erritem.1 \
449                  errpath.1 \
450                  errstr.1 \
451                  errtime.1 \
452                  erruid.1 \
453                  erryorn.1 \
454                  eval.1 \
455                  export.1 \
456                  false.1 \

```

new/usr/src/man/man1/Makefile

8

```

457      fc.1         \
458      fg.1         \
459      for.1        \
460      foreach.1     \
461      function.1    \
462      goto.1        \
463      hashcheck.1   \
464      hashmake.1    \
465      hashstat.1    \
466      helpdate.1    \
467      helpgid.1     \
468      helpint.1     \
469      helpitem.1    \
470      helppath.1    \
471      helprange.1   \
472      helpstr.1     \
473      helptime.1    \
474      helpuid.1     \
475      helpyorn.1    \
476      hist.1        \
477      i286.1        \
478      i386.1        \
479      i486.1        \
480      i860.1        \
481      iAPX286.1     \
482      if.1          \
483      intro.1       \
484      jsh.1         \
485      ksh.1         \
486      ldapadd.1     \
487      neqn.1        \
488      notify.1      \
489      onintr.1      \
490      page.1        \
491      pcat.1        \
492      pcred.1       \
493      pdp11.1       \
494      pfcsh.1       \
495      pfiles.1      \
496      pfksh.1       \
497      pflags.1      \
498      pfsh.1        \
499      pkill.1       \
500      pldd.1        \
501      popd.1        \
502      prun.1        \
503      psig.1        \
504      pstack.1      \
505      pstop.1       \
506      ptime.1       \
507      pushd.1       \
508      pwait.1       \
509      pwdx.1        \
510      red.1         \
511      rehash.1      \
512      remote_shell.1 \
513      remsh.1       \
514      repeat.1      \
515      return.1      \
516      rksh.1        \
517      rksh93.1      \
518      rmail.1       \
519      rmdir.1       \
520      rmumount.1    \
521      select.1      \
522      setenv.1      \

```

new/usr/src/man/man1/Makefile

```

523      settime.1      \
524      sh.1           \
525      snca.1         \
526      source.1       \
527      sparc.1        \
528      spellin.1      \
529      stop.1         \
530      strconf.1      \
531      sun.1          \
532      switch.1       \
533      u370.1         \
534      u3b.1          \
535      u3b15.1        \
536      u3b2.1         \
537      u3b5.1         \
538      ulimit.1       \
539      unalias.1      \
540      uncompress.1   \
541      unexpand.1     \
542      unhash.1       \
543      unlimit.1      \
544      unpack.1       \
545      unset.1        \
546      unsetenv.1     \
547      until.1        \
548      valdate.1      \
549      valgid.1       \
550      valint.1       \
551      valpath.1      \
552      valrange.1     \
553      valstr.1       \
554      valtime.1      \
555      valuid.1       \
556      valyorn.1      \
557      vax.1          \
558      vedit.1        \
559      whatis.1       \
560      whence.1       \
561      while.1        \
562      zcat.1         \

564 intro.1           := LINKSRC = Intro.1

566 whatis.1          := LINKSRC = apropos.1

568 unalias.1          := LINKSRC = alias.1

570 batch.1            := LINKSRC = at.1

572 dirname.1         := LINKSRC = basename.1

574 continue.1        := LINKSRC = break.1

576 chdir.1            := LINKSRC = cd.1
577 dirs.1             := LINKSRC = cd.1
578 popd.1             := LINKSRC = cd.1
579 pushd.1            := LINKSRC = cd.1

581 errdate.1          := LINKSRC = ckdate.1
582 helpdate.1         := LINKSRC = ckdate.1
583 valdate.1          := LINKSRC = ckdate.1

585 errgid.1           := LINKSRC = ckgid.1
586 helpgid.1          := LINKSRC = ckgid.1
587 valgid.1           := LINKSRC = ckgid.1

```

9

new/usr/src/man/man1/Makefile

```

589 errint.1          := LINKSRC = ckint.1
590 helpint.1          := LINKSRC = ckint.1
591 valint.1           := LINKSRC = ckint.1

593 erritem.1          := LINKSRC = ckitem.1
594 helpitem.1         := LINKSRC = ckitem.1

596 errpath.1          := LINKSRC = ckpath.1
597 helppath.1         := LINKSRC = ckpath.1
598 valpath.1          := LINKSRC = ckpath.1

600 errrange.1         := LINKSRC = ckrange.1
601 helprange.1        := LINKSRC = ckrange.1
602 valrange.1         := LINKSRC = ckrange.1

604 errstr.1           := LINKSRC = ckstr.1
605 helpstr.1          := LINKSRC = ckstr.1
606 valstr.1           := LINKSRC = ckstr.1

608 errtime.1          := LINKSRC = cktime.1
609 helptime.1         := LINKSRC = cktime.1
610 valtime.1          := LINKSRC = cktime.1

612 erruid.1           := LINKSRC = ckuid.1
613 helpuid.1          := LINKSRC = ckuid.1
614 valuid.1           := LINKSRC = ckuid.1

616 erryorn.1          := LINKSRC = kcyorn.1
617 helpyorn.1         := LINKSRC = kcyorn.1
618 valyorn.1          := LINKSRC = kcyorn.1

620 uncompress.1       := LINKSRC = compress.1
621 zcat.1             := LINKSRC = compress.1

623 red.1              := LINKSRC = ed.1

625 disable.1         := LINKSRC = enable.1

627 decrypt.1         := LINKSRC = encrypt.1

629 checkeq.1          := LINKSRC = eqn.1
630 neqn.1             := LINKSRC = eqn.1

632 eval.1             := LINKSRC = exec.1
633 source.1           := LINKSRC = exec.1

635 goto.1             := LINKSRC = exit.1
636 return.1           := LINKSRC = exit.1

638 unexpand.1         := LINKSRC = expand.1

640 hashstat.1         := LINKSRC = hash.1
641 rehash.1           := LINKSRC = hash.1
642 unhash.1           := LINKSRC = hash.1

644 fc.1               := LINKSRC = history.1
645 hist.1             := LINKSRC = history.1

647 bg.1               := LINKSRC = jobs.1
648 fg.1               := LINKSRC = jobs.1
649 notify.1           := LINKSRC = jobs.1
650 stop.1              := LINKSRC = jobs.1

652 jsh.1              := LINKSRC = ksh93.1
653 ksh.1              := LINKSRC = ksh93.1
654 rksh.1             := LINKSRC = ksh93.1

```

10

new/usr/src/man/man1/Makefile

11

```

655 rksh93.1      := LINKSRC = ksh93.1
656 sh.1          := LINKSRC = ksh93.1

658 ldapadd.1     := LINKSRC = ldapmodify.1

660 ulimit.1      := LINKSRC = limit.1
661 unlimit.1     := LINKSRC = limit.1

663 dumpkeys.1    := LINKSRC = loadkeys.1

665 i286.1        := LINKSRC = machid.1
666 i386.1        := LINKSRC = machid.1
667 i486.1        := LINKSRC = machid.1
668 i860.1        := LINKSRC = machid.1
669 iAPX286.1     := LINKSRC = machid.1
670 pdp11.1       := LINKSRC = machid.1
671 sparc.1       := LINKSRC = machid.1
672 sun.1         := LINKSRC = machid.1
673 u370.1        := LINKSRC = machid.1
674 u3b.1         := LINKSRC = machid.1
675 u3b15.1       := LINKSRC = machid.1
676 u3b2.1        := LINKSRC = machid.1
677 u3b5.1        := LINKSRC = machid.1
678 vax.1         := LINKSRC = machid.1

680 rmail.1       := LINKSRC = mail.1

682 page.1        := LINKSRC = more.1

684 snca.1        := LINKSRC = nca.1

686 pcat.1        := LINKSRC = pack.1
687 unpack.1      := LINKSRC = pack.1

689 pfcsh.1       := LINKSRC = pfexec.1
690 pfksh.1       := LINKSRC = pfexec.1
691 pfsh.1        := LINKSRC = pfexec.1

693 pkill.1       := LINKSRC = pgrep.1

695 pcred.1       := LINKSRC = proc.1
696 pfiles.1      := LINKSRC = proc.1
697 pflags.1      := LINKSRC = proc.1
698 pldd.1        := LINKSRC = proc.1
699 prun.1        := LINKSRC = proc.1
700 psig.1        := LINKSRC = proc.1
701 pstack.1      := LINKSRC = proc.1
702 pstop.1       := LINKSRC = proc.1
703 ptime.1       := LINKSRC = proc.1
704 pwait.1       := LINKSRC = proc.1
705 pwdx.1        := LINKSRC = proc.1

707 rmdir.1       := LINKSRC = rm.1

709 rmumount.1    := LINKSRC = rmmount.1

711 remote_shell.1 := LINKSRC = rsh.1
712 remsh.1       := LINKSRC = rsh.1

714 export.1      := LINKSRC = set.1
715 setenv.1      := LINKSRC = set.1
716 unset.1       := LINKSRC = set.1
717 unsetenv.1    := LINKSRC = set.1

719 case.1        := LINKSRC = shell_builtins.1
720 for.1          := LINKSRC = shell_builtins.1

```

new/usr/src/man/man1/Makefile

12

```

721 foreach.1     := LINKSRC = shell_builtins.1
722 function.1    := LINKSRC = shell_builtins.1
723 if.1          := LINKSRC = shell_builtins.1
724 repeat.1      := LINKSRC = shell_builtins.1
725 select.1      := LINKSRC = shell_builtins.1
726 switch.1      := LINKSRC = shell_builtins.1
727 until.1       := LINKSRC = shell_builtins.1
728 while.1       := LINKSRC = shell_builtins.1

730 hashcheck.1   := LINKSRC = spell.1
731 hashmake.1    := LINKSRC = spell.1
732 spellin.1     := LINKSRC = spell.1

734 strconf.1     := LINKSRC = strchg.1

736 settime.1     := LINKSRC = touch.1

738 onintr.1      := LINKSRC = trap.1

740 false.1       := LINKSRC = true.1

742 whence.1      := LINKSRC = typeset.1

744 # Links to usr/has/man

746 edit.1        := LINKSRC = ../../../has/man/man1has/edit.1has

748 vedit.1       := LINKSRC = ../../../has/man/man1has/vi.1has

750 .KEEP_STATE:

752 include       $(SRC)/man/Makefile.man

754 install:      $(ROOTMANFILES) $(ROOTMANLINKS)

```

new/usr/src/man/man1/apropos.1

1

1729 Wed Jul 16 09:10:05 2014

new/usr/src/man/man1/apropos.1

feedback from Hans

```
1 .\"
2 .\" This file and its contents are supplied under the terms of the
3 .\" Common Development and Distribution License (\"CDDL\"), version 1.0.
4 .\" You may only use this file in accordance with the terms of version
5 .\" 1.0 of the CDDL.
6 .\"
7 .\" A full copy of the text of the CDDL should have accompanied this
8 .\" source. A copy of the CDDL is also available via the Internet at
9 .\" http://www.illumos.org/license/CDDL.
10 .\"
11 .\"
12 .\" Copyright 2012 Nexenta Systems, Inc. All rights reserved.
13 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
14 .\"
15 .Dd Jul 13, 2014
16 .Dt APROPOS 1
17 .Os
18 .Sh NAME
19 .Nm apropos ,
20 .Nm whatis
21 .Nd keyword search in
22 .Nm whatis
23 database files
24 .Sh SYNOPSIS
25 .Nm
26 .Op Fl M Ar path
27 .Op Fl s Ar section
28 .Ar keyword ...
29 .Nm whatis
30 .Op Fl M Ar path
31 .Op Fl s Ar section
32 .Ar keyword ...
33 .Sh DESCRIPTION
34 The
35 .Nm
36 utility searches a set of
37 .Nm whatis
38 database files matching each
39 .Ar keyword .
40 The
41 .Nm whatis
42 utility does the same search but only on complete words. The
43 .Nm whatis
44 database files are created using
45 .Xr man 1
46 command.
47 .Bl -tag -width ".Fl d"
48 .It Fl M Ar path
49 Force a specific colon separated manual path instead of the default search path.
50 Overrides the
51 .Ev MANPATH
52 environment variable.
53 .It Fl s Ar section
54 Restrict search to specified
55 .Ar section .
56 .El
57 .Sh ENVIRONMENT
58 The following environment variables affect the execution of
59 .Nm :
60 .Bl -tag -width ".Ev MANPATH , PATH"
61 .It Ev MANPATH , PATH
```

new/usr/src/man/man1/apropos.1

2

```
62 Used to find the location of the
63 .Nm whatis
64 database files.
65 .El
66 .Sh DIAGNOSTICS
67 The
68 .Nm
69 utility exits 0 if a keyword matched and 1 if no keywords are matched or no
70 .Nm whatis
71 databases are found.
72 .Sh INTERFACE STABILITY
73 .Nm Committed .
73 Committed.
74 .Sh CODE SET INDEPENDENCE
75 Enabled.
76 .Sh SEE ALSO
77 .Xr man 1 ,
78 .Xr mandoc 1
```

new/usr/src/man/man1/man.1

1

3099 Wed Jul 16 09:10:05 2014

new/usr/src/man/man1/man.1

feedback from Hans

```
1 .\"
2 .\" This file and its contents are supplied under the terms of the
3 .\" Common Development and Distribution License (\"CDDL\"), version 1.0.
4 .\" You may only use this file in accordance with the terms of version
5 .\" 1.0 of the CDDL.
6 .\"
7 .\" A full copy of the text of the CDDL should have accompanied this
8 .\" source. A copy of the CDDL is also available via the Internet at
9 .\" http://www.illumos.org/license/CDDL.
10 .\"
11 .\"
12 .\" Copyright 2012 Nexenta Systems, Inc. All rights reserved.
13 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
14 .\"
15 .Dd October 18, 2012
16 .Dt MAN 1
17 .Os
18 .Sh NAME
19 .Nm man
20 .Nd find and display reference manual pages
21 .Sh SYNOPSIS
22 .Nm
23 .Op Fl alptw
24 .Op Fl M Ar manpath
25 .Op Fl s Ar mansect
26 .Ar page ...
27 .Nm
28 .Op Fl s Ar mansect
29 .Fl f
30 .Ar keyword ...
31 .Nm
32 .Op Fl s Ar mansect
33 .Fl k
34 .Ar keyword ...
35 .Sh DESCRIPTION
36 The
37 .Nm
38 utility finds and displays reference manual pages.
39 .Pp
40 Options that
41 .Nm
42 understands:
43 .Bl -tag -width indent
44 .It Fl M Ar manpath
45 Forces a specific colon separated manual path instead of the default
46 search path.
47 Overrides the
48 .Ev MANPATH
49 environment variable.
50 .It Fl a
51 Display all manual pages instead of just the first found for each
52 .Ar page
53 argument.
54 .It Fl f
55 Emulate
56 .Xr whatis 1 .
57 .It Fl k
58 Emulate
59 .Xr apropos 1 .
60 .It Fl l
61 Display the location of the manual page instead of the contents of
```

new/usr/src/man/man1/man.1

2

```
62 the manual page.
63 .It Fl p
64 Output current path used for searching.
65 .It Fl s Ar mansect
66 Restrict manual sections searched to the specified colon delimited list.
67 .It Fl t
68 Send the content formatted as PostScript to the default printer.
69 .It Fl w
70 Create
71 .Nm whatis
72 databases used by
73 .Xr apropos 1
74 and
75 .Xr whatis 1 .
76 .El
77 .Sh ENVIRONMENT
78 The following environment variables affect the execution of
79 .Nm :
80 .Bl -tag -width ".Ev MANPATH"
81 .It Ev LC_ALL , LC_CTYPE , LANG
82 Used to find locale-specific manual pages.
83 .It Ev MANPATH
84 Used to find the location of the manual files.
85 Corresponds to the
86 .Fl M
87 option.
88 .It Ev MANWIDTH
89 Defines the width of output. If set to
90 .Dq Li tty ,
91 and output is to a terminal, full width of terminal is used.
92 .It Ev PAGER
93 Program used to display files. If unset,
94 .Dq Li "less -ins"
95 is used.
96 .It Ev PATH
97 Used to find location of manual files if
98 .Ev MANPATH
99 and
100 .Fl M
101 are not specified.
102 .El
103 .Sh FILES
104 .Bl -tag -width indent -compact
105 .It Pa man.cf
106 Per-manpath configuration settings. The file is formatted as follows:
107 .Bd -literal -offset indent
108 MANSECT=\\fIsection\\fR[\\fIsection\\fR]...
109 .Ed
110 .Pp
111 Each section consists of a section in the reference manual. The file
112 may also contain comment blank lines or lines consisting of comments, where
113 the first character in the line is '#'. Both blank lines and comment lines are
114 ignored.
115 .El
116 .Sh CODE SET INDEPENDENCE
117 Enabled.
118 .Sh INTERFACE STABILITY
119 The
120 .Nm
121 utility is
122 .Nm Standard ,
123 as is the
124 utility is Standard, as is the
125 option. The other options are
126 .Nm Committed .
```



```
123 option. The other options are Committed.
127 .Sh SEE ALSO
128 .Xr apropos 1 ,
129 .Xr intro 1 ,
130 .Xr mandoc 1 ,
131 .Xr whatis 1 ,
132 .Xr man 5 ,
133 .Xr mdoc 5 ,
134 .Xr standards 5
135 .Sh NOTES
136 Some pages may contain information which cannot be properly displayed on
137 all terminals. In such cases, some information may be lost.
```

new/usr/src/man/man1/mandoc.1

1

14743 Wed Jul 16 09:10:05 2014

new/usr/src/man/man1/mandoc.1

feedback from Hans

```
1 .\"
2 .\" Permission to use, copy, modify, and distribute this software for any
3 .\" purpose with or without fee is hereby granted, provided that the above
4 .\" copyright notice and this permission notice appear in all copies.
5 .\"
6 .\" THE SOFTWARE IS PROVIDED \"AS IS\" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
7 .\" WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
8 .\" MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
9 .\" ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
10 .\" WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
11 .\" ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
12 .\" OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
13 .\"
14 .\"
15 .\" Copyright (c) 2009, 2010, 2011 Kristaps Dzonsons <kristaps@bsd.lv>
16 .\" Copyright 2012 Nexenta Systems, Inc. All rights reserved.
17 .\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
18 .\"
19 .Dd Jul 16, 2014
20 .Dt MANDOC 1
21 .Os
22 .Sh NAME
23 .Nm mandoc
24 .Nd format and display UNIX manuals
25 .Sh SYNOPSIS
26 .Nm mandoc
27 .Op Fl V
28 .Op Fl m Ns Ar format
29 .Op Fl O Ns Ar option
30 .Op Fl T Ns Ar output
31 .Op Fl W Ns Ar level
32 .Op Ar
33 .Sh DESCRIPTION
34 The
35 .Nm
36 utility formats
37 .Ux
38 manual pages for display.
39 .Pp
40 By default,
41 .Nm
42 reads
43 .Xr mdoc 5
44 or
45 .Xr man 5
46 text from stdin, implying
47 .Fl m Ns Cm andoc ,
48 and produces
49 .Fl T Ns Cm ascii
50 output.
51 .Pp
52 The arguments are as follows:
53 .Bl -tag -width Ds
54 .It Fl m Ns Ar format
55 Input format.
56 See
57 .Sx Input Formats
58 for available formats.
59 Defaults to
60 .Fl m Ns Cm andoc .
```

new/usr/src/man/man1/mandoc.1

2

```
61 .It Fl O Ns Ar option
62 Comma-separated output options.
63 .It Fl T Ns Ar output
64 Output format.
65 See
66 .Sx Output Formats
67 for available formats.
68 Defaults to
69 .Fl T Ns Cm ascii .
70 .It Fl V
71 Print version and exit.
72 .It Fl W Ns Ar level
73 Specify the minimum message
74 .Ar level
75 to be reported on the standard error output and to affect the exit status.
76 The
77 .Ar level
78 can be
79 .Cm warning ,
80 .Cm error ,
81 or
82 .Cm fatal .
83 The default is
84 .Fl W Ns Cm fatal ;
85 .Fl W Ns Cm all
86 is an alias for
87 .Fl W Ns Cm warning .
88 See
89 .Sx EXIT STATUS
90 and
91 .Sx DIAGNOSTICS
92 for details.
93 .Pp
94 The special option
95 .Fl W Ns Cm stop
96 tells
97 .Nm
98 to exit after parsing a file that causes warnings or errors of at least
99 the requested level.
100 No formatted output will be produced from that file.
101 If both a
102 .Ar level
103 and
104 .Cm stop
105 are requested, they can be joined with a comma, for example
106 .Fl W Ns Cm error , Ns Cm stop .
107 .It Ar file
108 Read input from zero or more files.
109 If unspecified, reads from stdin.
110 If multiple files are specified,
111 .Nm
112 will halt with the first failed parse.
113 .El
114 .Ss Input Formats
115 The
116 .Nm
117 utility accepts
118 .Xr mdoc 5
119 and
120 .Xr man 5
121 input with
122 .Fl m Ns Cm doc
123 and
124 .Fl m Ns Cm an ,
125 respectively.
126 The
```

```

127 .Xr mdoc 5
128 format is
129 .Em strongly
130 recommended;
131 .Xr man 5
132 should only be used for legacy manuals.
133 .Pp
134 A third option,
135 .Fl m Ns Cm andoc ,
136 which is also the default, determines encoding on-the-fly: if the first
137 non-comment macro is
138 .Sq \&Dd
139 or
140 .Sq \&Dt ,
141 the
142 .Xr mdoc 5
143 parser is used; otherwise, the
144 .Xr man 5
145 parser is used.
146 .Pp
147 If multiple
148 files are specified with
149 .Fl m Ns Cm andoc ,
150 each has its file-type determined this way.
151 If multiple files are
152 specified and
153 .Fl m Ns Cm doc
154 or
155 .Fl m Ns Cm an
156 is specified, then this format is used exclusively.
157 .Ss Output Formats
158 The
159 .Nm
160 utility accepts the following
161 .Fl T
162 arguments, which correspond to output modes:
163 .Bl -tag -width "Tlocale"
164 .It Fl T Ns Cm ascii
165 Produce 7-bit ASCII output.
166 This is the default.
167 See
168 .Sx ASCII Output .
169 .It Fl T Ns Cm html
170 Produce strict CSS1/HTML-4.01 output.
171 See
172 .Sx HTML Output .
173 .It Fl T Ns Cm lint
174 Parse only: produce no output.
175 Implies
176 .Fl W Ns Cm warning .
177 .It Fl T Ns Cm locale
178 Encode output using the current locale.
179 See
180 .Sx Locale Output .
181 .It Fl T Ns Cm man
182 Produce
183 .Xr man 5
184 format output.
185 See
186 .Sx Man Output .
187 .It Fl T Ns Cm pdf
188 Produce PDF output.
189 See
190 .Sx PDF Output .
191 .It Fl T Ns Cm ps
192 Produce PostScript output.

```

```

193 See
194 .Sx PostScript Output .
195 .It Fl T Ns Cm tree
196 Produce an indented parse tree.
197 .It Fl T Ns Cm utf8
198 Encode output in the UTF\ -8 multi-byte format.
199 See
200 .Sx UTF\ -8 Output .
201 .It Fl T Ns Cm xhtml
202 Produce strict CSS1/XHTML-1.0 output.
203 See
204 .Sx XHTML Output .
205 .El
206 .Pp
207 If multiple input files are specified, these will be processed by the
208 corresponding filter in-order.
209 .Ss ASCII Output
210 Output produced by
211 .Fl T Ns Cm ascii ,
212 which is the default, is rendered in standard 7-bit ASCII documented in
213 .Xr ascii 5 .
214 .Pp
215 Font styles are applied by using back-spaced encoding such that an
216 underlined character
217 .Sq c
218 is rendered as
219 .Sq _ Ns \e[bs] Ns c ,
220 where
221 .Sq \e[bs]
222 is the back-space character number 8.
223 Emboldened characters are rendered as
224 .Sq c Ns \e[bs] Ns c .
225 .Pp
226 The special characters documented in
227 .Xr mandoc_char 5
228 are rendered best-effort in an ASCII equivalent.
229 If no equivalent is found,
230 .Sq \&?
231 is used instead.
232 .Pp
233 Output width is limited to 78 visible columns unless literal input lines
234 exceed this limit.
235 .Pp
236 The following
237 .Fl O
238 arguments are accepted:
239 .Bl -tag -width Ds
240 .It Cm indent Ns = Ns Ar indent
241 The left margin for normal text is set to
242 .Ar indent
243 blank characters instead of the default of five for
244 .Xr mdoc 5
245 and seven for
246 .Xr man 5 .
247 Increasing this is not recommended; it may result in degraded formatting,
248 for example overfull lines or ugly line breaks.
249 .It Cm width Ns = Ns Ar width
250 The output width is set to
251 .Ar width ,
252 which will normalise to \(>=60.
253 .El
254 .Ss HTML Output
255 Output produced by
256 .Fl T Ns Cm html
257 conforms to HTML-4.01 strict.
258 .Pp

```

259 The
 260 .Pa example.style.css
 261 file documents style-sheet classes available for customising output.
 262 If a style-sheet is not specified with
 263 .Fl O Ns Ar style ,
 264 .Fl T Ns Cm html
 265 defaults to simple output readable in any graphical or text-based web
 266 browser.
 267 .Pp
 268 Special characters are rendered in decimal-encoded UTF\8.
 269 .Pp
 270 The following
 271 .Fl O
 272 arguments are accepted:
 273 .Bl -tag -width Ds
 274 .It Cm fragment
 275 Omit the
 276 .Aq !DOCTYPE
 277 declaration and the
 278 .Aq html ,
 279 .Aq head ,
 280 and
 281 .Aq body
 282 elements and only emit the subtree below the
 283 .Aq body
 284 element.
 285 The
 286 .Cm style
 287 argument will be ignored.
 288 This is useful when embedding manual content within existing documents.
 289 .It Cm includes Ns = Ns Ar fmt
 290 The string
 291 .Ar fmt ,
 292 for example,
 293 .Ar ../src/%I.html ,
 294 is used as a template for linked header files (usually via the
 295 .Sq \&In
 296 macro).
 297 Instances of
 298 .Sq \&%I
 299 are replaced with the include filename.
 300 The default is not to present a
 301 hyperlink.
 302 .It Cm man Ns = Ns Ar fmt
 303 The string
 304 .Ar fmt ,
 305 for example,
 306 .Ar ../html%S/%N.%S.html ,
 307 is used as a template for linked manuals (usually via the
 308 .Sq \&Xr
 309 macro).
 310 Instances of
 311 .Sq \&%N
 312 and
 313 .Sq %S
 314 are replaced with the linked manual's name and section, respectively.
 315 If no section is included, section 1 is assumed.
 316 The default is not to
 317 present a hyperlink.
 318 .It Cm style Ns = Ns Ar style.css
 319 The file
 320 .Ar style.css
 321 is used for an external style-sheet.
 322 This must be a valid absolute or
 323 relative URI.
 324 .El

325 .Ss Locale Output
 326 Locale-dependent output encoding is triggered with
 327 .Fl T Ns Cm locale .
 328 This option is not available on all systems: systems without locale
 329 support, or those whose internal representation is not natively UCS-4,
 330 will fall back to
 331 .Fl T Ns Cm ascii .
 332 See
 333 .Sx ASCII Output
 334 for font style specification and available command-line arguments.
 335 .Ss Man Output
 336 Translate input format into
 337 .Xr man 5
 338 output format.
 339 This is useful for distributing manual sources to legacy systems
 340 lacking
 341 .Xr mdoc 5
 342 formatters.
 343 .Pp
 344 If
 345 .Xr mdoc 5
 346 is passed as input, it is translated into
 347 .Xr man 5 .
 348 If the input format is
 349 .Xr man 5 ,
 350 the input is copied to the output, expanding any
 351 .Xr mandoc_roff 5
 352 .Sq so
 353 requests.
 354 The parser is also run, and as usual, the
 355 .Fl W
 356 level controls which
 357 .Sx DIAGNOSTICS
 358 are displayed before copying the input to the output.
 359 .Ss PDF Output
 360 PDF-1.1 output may be generated by
 361 .Fl T Ns Cm pdf .
 362 See
 363 .Sx PostScript Output
 364 for
 365 .Fl O
 366 arguments and defaults.
 367 .Ss PostScript Output
 368 PostScript
 369 .Qq Adobe-3.0
 370 Level-2 pages may be generated by
 371 .Fl T Ns Cm ps .
 372 Output pages default to letter sized and are rendered in the Times font
 373 family, 11-point.
 374 Margins are calculated as 1/9 the page length and width.
 375 Line-height is 1.4m.
 376 .Pp
 377 Special characters are rendered as in
 378 .Sx ASCII Output .
 379 .Pp
 380 The following
 381 .Fl O
 382 arguments are accepted:
 383 .Bl -tag -width Ds
 384 .It Cm paper Ns = Ns Ar name
 385 The paper size
 386 .Ar name
 387 may be one of
 388 .Ar a3 ,
 389 .Ar a4 ,
 390 .Ar a5 ,

```

391 .Ar legal ,
392 or
393 .Ar letter .
394 You may also manually specify dimensions as
395 .Ar NNxNN ,
396 width by height in millimetres.
397 If an unknown value is encountered,
398 .Ar letter
399 is used.
400 .El
401 .Ss UTF\~8 Output
402 Use
403 .Fl T Ns Cm utf8
404 to force a UTF\~8 locale.
405 See
406 .Sx Locale Output
407 for details and options.
408 .Ss XHTML Output
409 Output produced by
410 .Fl T Ns Cm xhtml
411 conforms to XHTML-1.0 strict.
412 .Pp
413 See
414 .Sx HTML Output
415 for details; beyond generating XHTML tags instead of HTML tags, these
416 output modes are identical.
417 .Sh EXIT STATUS
418 The
419 .Nm
420 utility exits with one of the following values, controlled by the message
421 .Ar level
422 associated with the
423 .Fl W
424 option:
425 .Pp
426 .Bl -tag -width Ds -compact
427 .It 0
428 No warnings or errors occurred, or those that did were ignored because
429 they were lower than the requested
430 .Ar level .
431 .It 2
432 At least one warning occurred, but no error, and
433 .Fl W Ns Cm warning
434 was specified.
435 .It 3
436 At least one parsing error occurred, but no fatal error, and
437 .Fl W Ns Cm error
438 or
439 .Fl W Ns Cm warning
440 was specified.
441 .It 4
442 A fatal parsing error occurred.
443 .It 5
444 Invalid command line arguments were specified.
445 No input files have been read.
446 .It 6
447 An operating system error occurred, for example memory exhaustion or an
448 error accessing input files.
449 Such errors cause
450 .Nm
451 to exit at once, possibly in the middle of parsing or formatting a file.
452 .El
453 .Pp
454 Note that selecting
455 .Fl T Ns Cm lint
456 output mode implies

```

```

457 .Fl W Ns Cm warning .
458 .Sh EXAMPLES
459 To page manuals to the terminal:
460 .Pp
461 .Dl $ mandoc \-Wall,stop mandoc.1 2\*(Gt&1 | less
462 .Dl $ mandoc mandoc.1 mdoc.5 | less
463 .Dl $ mandoc mandoc.1 mdoc.3 mdoc.7 | less
464 .Pp
464 To produce HTML manuals with
465 .Ar style.css
466 as the style-sheet:
467 .Pp
468 .Dl $ mandoc \-Thtml -Ostyle=style.css mdoc.5 \*(Gt mdoc.5.html
469 .Dl $ mandoc \-Thtml -Ostyle=style.css mdoc.7 \*(Gt mdoc.7.html
470 .Pp
470 To check over a large set of manuals:
471 .Pp
472 .Dl $ mandoc \-Tlint 'find /usr/src -name e*e.[1-9]'
473 .Pp
474 To produce a series of PostScript manuals for A4 paper:
475 .Pp
476 .Dl $ mandoc \-Tps \-Opaper=a4 mdoc.5 man.5 \*(Gt manuals.ps
477 .Dl $ mandoc \-Tps \-Opaper=a4 mdoc.7 man.7 \*(Gt manuals.ps
478 .Pp
478 Convert a modern
479 .Xr mdoc 5
480 manual to the older
481 .Xr man 5
482 format, for use on systems lacking an
483 .Xr mdoc 5
484 parser:
485 .Pp
486 .Dl $ mandoc \-Tman foo.mdoc \*(Gt foo.man
487 .Sh DIAGNOSTICS
488 Standard error messages reporting parsing errors are prefixed by
489 .Pp
490 .Sm off
491 .Dl Ar file : line : column : \ level :
492 .Sm on
493 .Pp
494 where the fields have the following meanings:
495 .Bl -tag -width "column"
496 .It Ar file
497 The name of the input file causing the message.
498 .It Ar line
499 The line number in that input file.
500 Line numbering starts at 1.
501 .It Ar column
502 The column number in that input file.
503 Column numbering starts at 1.
504 If the issue is caused by a word, the column number usually
505 points to the first character of the word.
506 .It Ar level
507 The message level, printed in capital letters.
508 .El
509 .Pp
510 Message levels have the following meanings:
511 .Bl -tag -width "warning"
512 .It Cm fatal
513 The parser is unable to parse a given input file at all.
514 No formatted output is produced from that input file.
515 .It Cm error
516 An input file contains syntax that cannot be safely interpreted,
517 either because it is invalid or because
518 .Nm
519 does not implement it yet.

```

520 By discarding part of the input or inserting missing tokens,
 521 the parser is able to continue, and the error does not prevent
 522 generation of formatted output, but typically, preparing that
 523 output involves information loss, broken document structure
 524 or unintended formatting.
 525 .It Cm warning
 526 An input file uses obsolete, discouraged or non-portable syntax.
 527 All the same, the meaning of the input is unambiguous and a correct
 528 rendering can be produced.
 529 Documents causing warnings may render poorly when using other
 530 formatting tools instead of
 531 .Nm .
 532 .El
 533 .Pp
 534 Messages of the
 535 .Cm warning
 536 and
 537 .Cm error
 538 levels are hidden unless their level, or a lower level, is requested using a
 539 .Fl W
 540 option or
 541 .Fl T Ns Cm lint
 542 output mode.
 543 .Pp
 544 The
 545 .Nm
 546 utility may also print messages related to invalid command line arguments
 547 or operating system errors, for example when memory is exhausted or
 548 input files cannot be read.
 549 Such messages do not carry the prefix described above.
 550 .Sh COMPATIBILITY
 551 This section summarises
 552 .Nm
 553 compatibility with GNU troff.
 554 Each input and output format is separately noted.
 555 .Ss ASCII Compatibility
 556 .Bl -bullet -compact
 557 .It
 558 Unrenderable unicode codepoints specified with
 559 .Sq \e[unNNN]
 560 escapes are printed as
 561 .Sq \&?
 562 in mandoc.
 563 In GNU troff, these raise an error.
 564 .It
 565 The
 566 .Sq \&Bd \-literal
 567 and
 568 .Sq \&Bd \-unfilled
 569 macros of
 570 .Xr mdoc 5
 571 in
 572 .Fl T Ns Cm ascii
 573 are synonyms, as are \-filled and \-ragged.
 574 .It
 575 In historic GNU troff, the
 576 .Sq \&Pa
 577 .Xr mdoc 5
 578 macro does not underline when scoped under an
 579 .Sq \&It
 580 in the FILES section.
 581 This behaves correctly in
 582 .Nm .
 583 .It
 584 A list or display following the
 585 .Sq \&Ss

586 .Xr mdoc 5
 587 macro in
 588 .Fl T Ns Cm ascii
 589 does not assert a prior vertical break, just as it doesn't with
 590 .Sq \&Sh .
 591 .It
 592 The
 593 .Sq \&na
 594 .Xr man 5
 595 macro in
 596 .Fl T Ns Cm ascii
 597 has no effect.
 598 .It
 599 Words aren't hyphenated.
 600 .El
 601 .Ss HTML/XHTML Compatibility
 602 .Bl -bullet -compact
 603 .It
 604 The
 605 .Sq \efP
 606 escape will revert the font to the previous
 607 .Sq \ef
 608 escape, not to the last rendered decoration, which is now dictated by
 609 CSS instead of hard-coded.
 610 It also will not span past the current scope,
 611 for the same reason.
 612 Note that in
 613 .Sx ASCII Output
 614 mode, this will work fine.
 615 .It
 616 The
 617 .Xr mdoc 5
 618 .Sq \&Bl \-hang
 619 and
 620 .Sq \&Bl \-tag
 621 list types render similarly (no break following overreached left-hand
 622 side) due to the expressive constraints of HTML.
 623 .It
 624 The
 625 .Xr man 5
 626 .Sq IP
 627 and
 628 .Sq TP
 629 lists render similarly.
 630 .El
 631 .Sh INTERFACE STABILITY
 632 The
 633 .Nm
 634 **utility is**
 635 **.Nm Committed ,**
 636 **but the details of specific output formats other than ASCII are**
 637 **.Nm Uncommitted .**
 638 *utility is Committed, but the details of specific output formats other than*
 639 *ASCII are Uncommitted.*
 638 .Sh SEE ALSO
 639 .Xr eqn 5 ,
 640 .Xr man 5 ,
 641 .Xr mandoc_char 5 ,
 642 .Xr mdoc 5 ,
 643 .Xr mandoc_roff 5 ,
 644 .Xr tbl 5
 645 .Sh AUTHORS
 646 The
 647 .Nm
 648 utility was written by
 649 .An Kristaps Dzonsons ,

```
650 .Mt kristaps@bsd.lv .
651 .Sh CAVEATS
652 In
653 .Fl T Ns Cm html
654 and
655 .Fl T Ns Cm xhtml ,
656 the maximum size of an element attribute is determined by
657 .Dv BUFSIZ ,
658 which is usually 1024 bytes.
659 Be aware of this when setting long link
660 formats such as
661 .Fl O Ns Cm style Ns = Ns Ar really/long/link .
662 .Pp
663 Nesting elements within next-line element scopes of
664 .Fl m Ns Cm an ,
665 such as
666 .Sq br
667 within an empty
668 .Sq B ,
669 will confuse
670 .Fl T Ns Cm html
671 and
672 .Fl T Ns Cm xhtml
673 and cause them to forget the formatting of the prior next-line scope.
674 .Pp
675 The
676 .Sq \{(aq
677 control character is an alias for the standard macro control character
678 and does not emit a line-break as stipulated in GNU troff.
```

new/usr/src/man/man1m/Makefile

1

14965 Wed Jul 16 09:10:05 2014

new/usr/src/man/man1m/Makefile

feedback from Hans

```
1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet
9 # at http://www.illumos.org/license/CDDL.
10 #
11 #
12 #
13 # Copyright 2011, Richard Lowe
14 # Copyright (c) 2012, Joyent, Inc. All rights reserved.
15 # Copyright 2013 Nexenta Systems, Inc. All rights reserved.
16 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
17 #
```

```
18 include      $(SRC)//Makefile.master
```

```
20 MANSECT=      1m
```

```
22 _MANFILES=    6to4relay.1m      \
23                Intro.1m         \
24                Uutry.1m          \
25                accept.1m         \
26                acct.1m           \
27                acctadm.1m        \
28                acctcms.1m        \
29                acctcon.1m        \
30                acctmerg.1m       \
31                acctprc.1m        \
32                acctsh.1m         \
33                adbggen.1m        \
34                add_allocatable.1m \
35                add_drv.1m        \
36                addbadsec.1m      \
37                arcstat.1m        \
38                arp.1m            \
39                atohexlabel.1m    \
40                audit.1m          \
41                audit_warn.1m     \
42                auditconfig.1m    \
43                auditd.1m         \
44                auditrecord.1m    \
45                auditreduce.1m    \
46                auditstat.1m      \
47                automount.1m      \
48                automountd.1m     \
49                autopush.1m       \
50                bart.1m           \
51                beadm.1m          \
52                boot.1m           \
53                bootadm.1m        \
54                bootconfchk.1m    \
55                busstat.1m        \
56                cachebsd.1m       \
57                cachebsdlog.1m    \
58                cachebspack.1m    \
59                cachebsdstat.1m   \
60                cachebsdswsize.1m \
```

new/usr/src/man/man1m/Makefile

2

```
61                captainfo.1m      \
62                catman.1m         \
63                cfgadm.1m         \
64                cfgadm_ac.1m      \
65                cfgadm_cardbus.1m \
66                cfgadm_fp.1m      \
67                cfgadm_ib.1m      \
68                cfgadm_pci.1m     \
69                cfgadm_sata.1m    \
70                cfgadm_sbd.1m     \
71                cfgadm_scsi.1m    \
72                cfgadm_sdcad.1m   \
73                cfgadm_sysctrl.1m \
74                cfgadm_usb.1m     \
75                cfsadmin.1m       \
76                chat.1m           \
77                check-hostname.1m \
78                check-permissions.1m \
79                chroot.1m         \
80                clear_locks.1m    \
81                clinfo.1m         \
82                clri.1m           \
83                consadm.1m        \
84                conv_lp.1m        \
85                conv_lpd.1m       \
86                coreadm.1m        \
87                cpustat.1m        \
88                cron.1m           \
89                cryptoadm.1m      \
90                datadm.1m         \
91                dd.1m             \
92                devattr.1m        \
93                devfree.1m        \
94                devfsadm.1m       \
95                device_remap.1m   \
96                devinfo.1m        \
97                devlinks.1m       \
98                devnm.1m          \
99                devprop.1m        \
100                devreserv.1m      \
101                df.1m             \
102                df_ufs.1m         \
103                dfmounts.1m       \
104                dfmounts_nfs.1m   \
105                dfshares.1m       \
106                dfshares_nfs.1m   \
107                dhcpagent.1m      \
108                dhcpconfig.1m     \
109                dhcpmgr.1m        \
110                dhtadm.1m         \
111                disks.1m          \
112                diskscan.1m       \
113                dispadmin.1m      \
114                dladm.1m          \
115                dlmgmtd.1m        \
116                dmesg.1m          \
117                dminfo.1m         \
118                dns-sd.1m         \
119                domainname.1m     \
120                drvconfig.1m      \
121                dsbitmap.1m       \
122                dscfg.1m          \
123                dscfgadm.1m       \
124                dscfglockd.1m     \
125                dsstat.1m         \
126                dsvclockd.1m      \
```


new/usr/src/man/man1m/Makefile

```
127      dtrace.1m      \
128      dumpadm.1m     \
129      editmap.1m     \
130      edquota.1m     \
131      eeprom.1m      \
132      embedded_su.1m \
133      etrn.1m         \
134      fcinfo.1m      \
135      fdetach.1m     \
136      fdisk.1m       \
137      ff.1m          \
138      ff_ufs.1m      \
139      fiocompress.1m \
140      flowadm.1m     \
141      fmadm.1m       \
142      fmd.1m         \
143      fmdump.1m      \
144      fmstat.1m      \
145      fmthard.1m     \
146      format.1m      \
147      fruadm.1m      \
148      fsck.1m        \
149      fsck_cacheofs.1m \
150      fsck_pcfs.1m   \
151      fsck_udfs.1m   \
152      fsck_ufs.1m    \
153      fsdb.1m        \
154      fsdb_udfs.1m   \
155      fsdb_ufs.1m    \
156      fsirand.1m     \
157      fssnap.1m      \
158      fssnap_ufs.1m  \
159      fsstat.1m      \
160      fstyp.1m       \
161      ftpaddhost.1m  \
162      ftpconfig.1m   \
163      ftprestart.1m  \
164      ftpshut.1m     \
165      fuser.1m       \
166      fwflash.1m     \
167      fwtmp.1m       \
168      getdev.1m      \
169      getdevpolicy.1m \
170      getdgrp.1m     \
171      getent.1m      \
172      gettable.1m    \
173      getty.1m       \
174      getvol.1m      \
175      groupadd.1m    \
176      groupdel.1m    \
177      groupmod.1m    \
178      growfs.1m      \
179      gsscred.1m     \
180      gssd.1m        \
181      hal-device.1m  \
182      hal-fdi-validate.1m \
183      hal-find.1m    \
184      hal-get-property.1m \
185      hald.1m        \
186      halt.1m        \
187      hextoalabel.1m \
188      hostconfig.1m  \
189      htable.1m      \
190      ickey.1m       \
191      id.1m          \
192      idmap.1m       \
```

3

new/usr/src/man/man1m/Makefile

```
193      idmapd.1m      \
194      idsconfig.1m   \
195      if_mpadm.1m    \
196      ifconfig.1m    \
197      ifparse.1m     \
198      iadm.1m        \
199      iicpbmp.1m     \
200      iicpsd.1m      \
201      ikeadm.1m      \
202      ikecert.1m     \
203      in.chargend.1m  \
204      in.comsat.1m   \
205      in.daytimed.1m \
206      in.dhcpd.1m    \
207      in.discardd.1m \
208      in.echod.1m    \
209      in.fingerd.1m  \
210      in.ftpd.1m     \
211      in.iked.1m     \
212      in.lpd.1m      \
213      in.mpathd.1m   \
214      in.ndpd.1m     \
215      in.rarpd.1m    \
216      in.rdisc.1m    \
217      in.rexecd.1m   \
218      in.ripngd.1m   \
219      in.rlogind.1m  \
220      in.routed.1m   \
221      in.rshd.1m     \
222      in.rwhod.1m    \
223      in.talkd.1m    \
224      in.telnetd.1m  \
225      in.tftpd.1m    \
226      in.timed.1m    \
227      in.uucpd.1m    \
228      inetadm.1m     \
229      inetconv.1m    \
230      inetd.1m       \
231      infocmp.1m     \
232      init.1m        \
233      inityp2l.1m    \
234      install.1m     \
235      installf.1m    \
236      installgrub.1m \
237      intrd.1m       \
238      intrstat.1m    \
239      iostat.1m      \
240      ipaddrsel.1m   \
241      ipadm.1m       \
242      ipdadm.1m      \
243      ipf.1m         \
244      ipfs.1m        \
245      ipfstat.1m     \
246      ipmon.1m       \
247      ipmpstat.1m    \
248      ipnat.1m       \
249      ippool.1m      \
250      ipqosconf.1m   \
251      ipsecalgs.1m   \
252      ipsecconf.1m   \
253      ipseckey.1m    \
254      iscsiadm.1m    \
255      isns.1m        \
256      isnsadm.1m     \
257      itadm.1m       \
258      k5srvutil.1m   \
```

4

new/usr/src/man/man1m/Makefile

```

259      kadb.1m      \
260      kadmin.1m    \
261      kadmind.1m   \
262      kclient.1m   \
263      kdb5_ldap_util.1m \
264      kdb5_util.1m \
265      kdcmgr.1m    \
266      kernel.1m    \
267      keyserv.1m   \
268      killall.1m   \
269      kprop.1m     \
270      kpropd.1m    \
271      kproplog.1m  \
272      krb5kdc.1m   \
273      ksslcfg.1m   \
274      kstat.1m     \
275      ktkk_warnd.1m \
276      labelit.1m   \
277      labelit_hsfs.1m \
278      labelit_udfs.1m \
279      labelit_ufs.1m \
280      latencytop.1m \
281      ldap_cachemgr.1m \
282      ldapaddent.1m \
283      ldapclient.1m \
284      link.1m      \
285      listdgrp.1m  \
286      listen.1m    \
287      locator.1m   \
288      lockfs.1m    \
289      lockstat.1m  \
290      lofiadm.1m   \
291      logadm.1m    \
292      logins.1m    \
293      lpadmin.1m   \
294      lpfilter.1m  \
295      lpforms.1m   \
296      lpget.1m     \
297      lpmove.1m    \
298      lpsched.1m   \
299      lpset.1m     \
300      lpshut.1m    \
301      lpsystem.1m  \
302      lpusers.1m   \
303      luxadm.1m    \
304      mail.local.1m \
305      makedbm.1m   \
306      makemap.1m   \
307      mdmonitor.1m \
308      medstat.1m   \
309      metaclear.1m \
310      metadb.1m    \
311      metadevadm.1m \
312      metahs.1m    \
313      metaimport.1m \
314      metainit.1m  \
315      metaoffline.1m \
316      metaparam.1m \
317      metarecover.1m \
318      metarename.1m \
319      metareplace.1m \
320      metaroot.1m  \
321      metaset.1m   \
322      metassist.1m \
323      metastat.1m  \
324      metasync.1m  \

```

5

new/usr/src/man/man1m/Makefile

```

325      metattach.1m \
326      mkdevalloc.1m \
327      mkdevmaps.1m \
328      mkfifo.1m    \
329      mkfile.1m    \
330      mkfs.1m      \
331      mkfs_pcfs.1m \
332      mkfs_udfs.1m \
333      mkfs_ufs.1m  \
334      mknod.1m     \
335      mkpwdict.1m  \
336      modinfo.1m   \
337      modload.1m   \
338      modunload.1m \
339      mount.1m     \
340      mount_cacheufs.1m \
341      mount_hsfs.1m \
342      mount_nfs.1m \
343      mount_pcfs.1m \
344      mount_smbfs.1m \
345      mount_tmpfs.1m \
346      mount_udfs.1m \
347      mount_ufs.1m \
348      mountall.1m  \
349      mountd.1m    \
350      mpathadm.1m  \
351      mpstat.1m    \
352      msgid.1m     \
353      mvdir.1m     \
354      ncaconfd.1m  \
355      ncheck.1m    \
356      ncheck_ufs.1m \
357      ndd.1m       \
358      ndmpadm.1m   \
359      ndmpd.1m     \
360      ndmpstat.1m  \
361      netstat.1m   \
362      netstrategy.1m \
363      newaliases.1m \
364      newfs.1m     \
365      newkey.1m    \
366      nfs4cbd.1m   \
367      nfsd.1m      \
368      nfslogd.1m   \
369      nfsmapid.1m  \
370      nfsstat.1m   \
371      nlsadmin.1m  \
372      nscadm.1m    \
373      nscd.1m      \
374      nwamd.1m     \
375      passmgmt.1m  \
376      pbind.1m     \
377      picld.1m     \
378      ping.1m      \
379      pkgadd.1m    \
380      pkgadm.1m    \
381      pkgask.1m    \
382      pkgchk.1m    \
383      pkgrm.1m     \
384      plockstat.1m \
385      pmadm.1m     \
386      pmconfig.1m  \
387      pntadm.1m    \
388      polkit-is-privileged.1m \
389      pooladm.1m   \
390      poolbind.1m  \

```

6

new/usr/src/man/man1m/Makefile

7

```

391      poolcfg.1m      \
392      poold.1m        \
393      poolstat.1m     \
394      ports.1m        \
395      powerd.1m       \
396      powertop.1m     \
397      ppdmgr.1m       \
398      pppd.1m         \
399      pppdump.1m      \
400      pppoec.1m       \
401      pppoed.1m       \
402      pppstats.1m     \
403      praudit.1m      \
404      print-service.1m \
405      printmgr.1m     \
406      projadd.1m      \
407      projdel.1m      \
408      projmod.1m      \
409      prstat.1m       \
410      prtconf.1m      \
411      prtdiag.1m      \
412      prtfru.1m       \
413      prtpicl.1m      \
414      prvtoc.1m       \
415      psradm.1m       \
416      psrinfo.1m      \
417      psrset.1m       \
418      putdev.1m       \
419      putdgrp.1m      \
420      pwck.1m         \
421      pwconv.1m       \
422      quot.1m         \
423      quota.1m        \
424      quotacheck.1m   \
425      quotaon.1m      \
426      raidctl.1m      \
427      ramdiskadm.1m   \
428      rcapadm.1m      \
429      rcapd.1m        \
430      rctladm.1m      \
431      rdate.1m        \
432      reboot.1m       \
433      rem_drv.1m       \
434      remove_allocatable.1m \
435      removef.1m      \
436      repquota.1m     \
437      rmmount.1m      \
438      rmt.1m          \
439      rmvolmgr.1m     \
440      roleadd.1m      \
441      roledel.1m      \
442      rolemod.1m      \
443      root_archive.1m \
444      route.1m        \
445      routeadm.1m     \
446      rpc.bootparamd.1m \
447      rpc.mdcommd.1m  \
448      rpc.metad.1m     \
449      rpc.metamedd.1m  \
450      rpc.metamhd.1m   \
451      rpc.rexd.1m      \
452      rpc.rstatd.1m    \
453      rpc.rusersd.1m   \
454      rpc.rwalld.1m    \
455      rpc.smsserverd.1m \
456      rpc.sprayd.1m    \

```

new/usr/src/man/man1m/Makefile

8

```

457      rpc.yppasswdd.1m \
458      rpc.yupdated.1m  \
459      rpcbind.1m       \
460      rpcinfo.1m       \
461      rquotad.1m       \
462      rsh.1m           \
463      rtc.1m           \
464      rtquery.1m       \
465      runacct.1m       \
466      rwall.1m         \
467      sac.1m           \
468      sacadm.1m        \
469      saf.1m           \
470      sar.1m           \
471      sasinfo.1m       \
472      savecore.1m      \
473      sbdadm.1m        \
474      scadm.1m         \
475      scmadm.1m        \
476      sdpadm.1m        \
477      sendmail.1m      \
478      setuname.1m      \
479      sftp-server.1m   \
480      share.1m         \
481      share_nfs.1m     \
482      shareall.1m      \
483      sharectl.1m      \
484      sharemgr.1m      \
485      showmount.1m     \
486      shutdown.1m      \
487      slpd.1m          \
488      smbadm.1m        \
489      smbd.1m          \
490      smbiod.1m        \
491      smbios.1m        \
492      smbstat.1m       \
493      smrsh.1m         \
494      smtnrhdb.1m      \
495      smtnrhttp.1m     \
496      smtnzonecfg.1m   \
497      sndradm.1m       \
498      sndrd.1m         \
499      sndrsyncd.1m     \
500      snoop.1m         \
501      soconfig.1m      \
502      sppptun.1m       \
503      spray.1m         \
504      ssh-keysign.1m   \
505      sshd.1m          \
506      statd.1m         \
507      stmfadm.1m       \
508      stmsboot.1m      \
509      strace.1m         \
510      strclean.1m      \
511      strerr.1m        \
512      sttydefs.1m      \
513      su.1m            \
514      sulogin.1m       \
515      svadm.1m         \
516      svc.configd.1m   \
517      svc.ipfd.1m      \
518      svc.startd.1m    \
519      svcadm.1m        \
520      svccfg.1m        \
521      swap.1m          \
522      sync.1m          \

```

new/usr/src/man/man1m/Makefile

9

```

523      syncinit.1m      \
524      syncloop.1m     \
525      syncstat.1m     \
526      sysdef.1m       \
527      syseventadm.1m  \
528      syseventconfd.1m \
529      syseventd.1m    \
530      syslogd.1m      \
531      tapes.1m        \
532      tcpd.1m         \
533      tcpdchk.1m      \
534      tcpdmatch.1m    \
535      th_define.1m    \
536      th_manage.1m    \
537      tic.1m          \
538      tnchkdb.1m      \
539      tnctl.1m        \
540      tnd.1m          \
541      tninfo.1m       \
542      tpmadm.1m       \
543      traceroute.1m   \
544      trapstat.1m     \
545      ttyadm.1m       \
546      ttymon.1m       \
547      tune.1m         \
548      txzonemgr.1m    \
549      tzselect.1m     \
550      uadmin.1m       \
551      ucodeadm.1m     \
552      ufsdump.1m      \
553      ufsrestore.1m   \
554      unshare.1m      \
555      unshare_nfs.1m  \
556      update_drv.1m   \
557      updatehome.1m   \
558      useradd.1m      \
559      userdel.1m      \
560      usermod.1m      \
561      utmpd.1m        \
562      uucheck.1m      \
563      uucico.1m       \
564      uucleanup.1m    \
565      uusched.1m      \
566      uuxqt.1m        \
567      vmstat.1m       \
568      volcopy.1m      \
569      volcopy_ufs.1m  \
570      vscanadm.1m     \
571      vscand.1m       \
572      wall.1m         \
573      wanboot_keygen.1m \
574      wanboot_keymgmt.1m \
575      wanboot_p12split.1m \
576      wanbootutil.1m  \
577      whodo.1m        \
578      wificonfig.1m   \
579      wpad.1m         \
580      wracct.1m       \
581      wusbadm.1m      \
582      ypbind.1m       \
583      ypinit.1m       \
584      ypmake.1m       \
585      ypmmap2src.1m   \
586      yppoll.1m       \
587      yppush.1m       \
588      ypserv.1m       \

```

new/usr/src/man/man1m/Makefile

10

```

589      ypset.1m        \
590      ypstart.1m      \
591      ypxfr.1m        \
592      zdb.1m          \
593      zdump.1m        \
594      zfs.1m          \
595      zic.1m          \
596      zoneadm.1m      \
597      zoneadmd.1m     \
598      zonecfg.1m      \
599      zpool.1m        \
600      zstreamdump.1m  \
\
602  i386_MANFILES=      lms.1m      \
603                      parted.1m   \
604                      mkntfs.1m    \
605                      ntfsclat.1m  \
606                      ntfsclone.1m \
607                      ntfscluster.1m \
608                      ntfscomp.1m  \
609                      ntfsd.1m     \
610                      ntfsfix.1m   \
611                      ntfsinfo.1m  \
612                      ntfslabel.1m \
613                      ntfsls.1m    \
614                      ntfsprogs.1m \
615                      ntfsresize.1m \
616                      ntfsundelete.1m \
\
618  sparc_MANFILES=     cvcd.1m     \
619                      dcs.1m      \
620                      drd.1m      \
621                      efdaemon.1m \
622                      ldmd.1m     \
623                      monitor.1m  \
624                      obpsym.1m   \
625                      oplhpd.1m   \
626                      prtdsc.1m   \
627                      sckmd.1m    \
628                      sf880drd.1m  \
629                      vntsd.1m    \
\
631  MANLINKS=           acctcon1.1m \
632                      acctcon2.1m  \
633                      acctdisk.1m  \
634                      acctdusg.1m  \
635                      accton.1m    \
636                      acctprc1.1m  \
637                      acctprc2.1m  \
638                      acctwtmp.1m  \
639                      bootparamd.1m \
640                      chargefee.1m \
641                      ckpacct.1m   \
642                      closewtmp.1m \
643                      comsat.1m    \
644                      dcopy.1m     \
645                      devfsadmd.1m \
646                      dodisk.1m    \
647                      fcadm.1m     \
648                      fingerd.1m   \
649                      ftpd.1m      \
650                      grpck.1m     \
651                      hal-find-by-capability.1m \
652                      hal-find-by-property.1m \
653                      hal-set-property.1m \
654                      intro.1m     \

```

new/usr/src/man/man1m/Makefile

11

```

655      kadmin.local.1m      \
656      lastlogin.1m         \
657      metadetch.1m         \
658      metaonline.1m        \
659      monacct.1m           \
660      nulladm.1m           \
661      poweroff.1m          \
662      prctmp.1m            \
663      prdaily.1m           \
664      prtacct.1m           \
665      quotaoff.1m          \
666      rarpd.1m             \
667      rdisc.1m             \
668      reject.1m            \
669      restricted_shell.1m   \
670      rexd.1m              \
671      rexecd.1m            \
672      rlogind.1m           \
673      routed.1m            \
674      rshd.1m              \
675      rstatd.1m            \
676      rusersd.1m           \
677      rwall.1m             \
678      rwhod.1m             \
679      sa1.1m               \
680      sa2.1m               \
681      sadc.1m              \
682      shutacct.1m          \
683      sprayd.1m            \
684      startup.1m           \
685      talkd.1m             \
686      telinit.1m           \
687      telnetd.1m           \
688      tftpd.1m             \
689      turnacct.1m          \
690      umount.1m            \
691      umount_smbfs.1m      \
692      umountall.1m         \
693      unlink.1m            \
694      unshareall.1m        \
695      utmp2wtmp.1m         \
696      uucpd.1m             \
697      uutry.1m             \
698      wtmpfix.1m           \
699      yppasswdd.1m         \
700      ypstop.1m            \
701      ypupdated.1m         \
702      ypxfr_1perday.1m     \
703      ypxfr_1perhour.1m    \
704      ypxfr_2perday.1m     \
705      ypxfrd.1m            \

707 MANFILES=      $_MANFILES) $($(MACH)_MANFILES)

709 intro.1m        := LINKSRC = Intro.1m

711 uutry.1m        := LINKSRC = Uutry.1m

713 reject.1m       := LINKSRC = accept.1m

715 acctdisk.1m     := LINKSRC = acct.1m
716 acctdusg.1m     := LINKSRC = acct.1m
717 accton.1m       := LINKSRC = acct.1m
718 acctwtmp.1m     := LINKSRC = acct.1m
719 closewtmp.1m    := LINKSRC = acct.1m
720 utmp2wtmp.1m    := LINKSRC = acct.1m

```

new/usr/src/man/man1m/Makefile

12

```

722 acctcon1.1m     := LINKSRC = acctcon.1m
723 acctcon2.1m     := LINKSRC = acctcon.1m

725 acctprc1.1m    := LINKSRC = acctprc.1m
726 acctprc2.1m    := LINKSRC = acctprc.1m

728 chargefee.1m   := LINKSRC = acctsh.1m
729 ckpacct.1m     := LINKSRC = acctsh.1m
730 dodisk.1m      := LINKSRC = acctsh.1m
731 lastlogin.1m   := LINKSRC = acctsh.1m
732 monacct.1m     := LINKSRC = acctsh.1m
733 nulladm.1m     := LINKSRC = acctsh.1m
734 prctmp.1m      := LINKSRC = acctsh.1m
735 prdaily.1m     := LINKSRC = acctsh.1m
736 prtacct.1m     := LINKSRC = acctsh.1m
737 shutacct.1m    := LINKSRC = acctsh.1m
738 startup.1m     := LINKSRC = acctsh.1m
739 turnacct.1m    := LINKSRC = acctsh.1m

741 dcopy.1m       := LINKSRC = clri.1m

743 devfsadmd.1m  := LINKSRC = devfsadm.1m

745 fcdm.1m       := LINKSRC = fcinfo.1m

747 wtmpfix.1m    := LINKSRC = fwtmp.1m

749 hal-find-by-capability.1m := LINKSRC = hal-find.1m
750 hal-find-by-property.1m  := LINKSRC = hal-find.1m

752 hal-set-property.1m     := LINKSRC = hal-get-property.1m

754 poweroff.1m            := LINKSRC = halt.1m

756 comsat.1m              := LINKSRC = in.comsat.1m
757 fingerd.1m             := LINKSRC = in.fingerd.1m
758 ftpd.1m                := LINKSRC = in.ftpd.1m
759 rarpd.1m                := LINKSRC = in.rarpd.1m
760 rdisc.1m                := LINKSRC = in.rdisc.1m
761 rexecd.1m               := LINKSRC = in.rexecd.1m
762 rlogind.1m              := LINKSRC = in.rlogind.1m
763 routed.1m               := LINKSRC = in.routed.1m
764 rshd.1m                 := LINKSRC = in.rshd.1m
765 rwhod.1m                := LINKSRC = in.rwhod.1m
766 talkd.1m                := LINKSRC = in.talkd.1m
767 telnetd.1m              := LINKSRC = in.telnetd.1m
768 tftpd.1m                := LINKSRC = in.tftpd.1m
769 uucpd.1m                := LINKSRC = in.uucpd.1m

771 telinit.1m             := LINKSRC = init.1m

773 kadmin.local.1m       := LINKSRC = kadmin.1m

775 unlink.1m             := LINKSRC = link.1m

777 metaonline.1m         := LINKSRC = metaoffline.1m

779 metadetch.1m         := LINKSRC = metattach.1m

781 umount.1m             := LINKSRC = mount.1m

783 umount_smbfs.1m       := LINKSRC = mount_smbfs.1m

785 umountall.1m          := LINKSRC = mountall.1m

```

```
787 grpck.1m           := LINKSRC = pwck.1m
789 quotaoff.1m        := LINKSRC = quotaon.1m

791 bootparamd.1m       := LINKSRC = rpc.bootparamd.1m
792 rexd.1m             := LINKSRC = rpc.rexd.1m
793 rstatd.1m           := LINKSRC = rpc.rstatd.1m
794 rusersd.1m          := LINKSRC = rpc.rusersd.1m
795 rwalld.1m           := LINKSRC = rpc.rwalld.1m
796 sprayd.1m           := LINKSRC = rpc.sprayd.1m
797 yppasswdd.1m        := LINKSRC = rpc.yppasswdd.1m
798 yppupdated.1m       := LINKSRC = rpc.yppupdated.1m

800 restricted_shell.1m := LINKSRC = rsh.1m

802 sa1.1m              := LINKSRC = sar.1m
803 sa2.1m              := LINKSRC = sar.1m
804 sadc.1m              := LINKSRC = sar.1m

806 unshareall.1m      := LINKSRC = shareall.1m

808 ypxfrd.1m           := LINKSRC = ypserv.1m

810 ypstop.1m           := LINKSRC = ypstart.1m

812 ypxfr_1perday.1m    := LINKSRC = ypxfr.1m
813 ypxfr_1perhour.1m   := LINKSRC = ypxfr.1m
814 ypxfr_2perday.1m    := LINKSRC = ypxfr.1m

817 .KEEP_STATE:

819 include      $(SRC)/man/Makefile.man

821 install:     $(ROOTMANFILES) $(ROOTMANLINKS)
```

new/usr/src/man/man1m/catman.1m

1

1543 Wed Jul 16 09:10:05 2014

new/usr/src/man/man1m/catman.1m

feedback from Hans

```
1.\"
2.\" This file and its contents are supplied under the terms of the
3.\" Common Development and Distribution License (\"CDDL\"), version 1.0.
4.\" You may only use this file in accordance with the terms of version
5.\" 1.0 of the CDDL.
6.\"
7.\" A full copy of the text of the CDDL should have accompanied this
8.\" source. A copy of the CDDL is also available via the Internet at
9.\" http://www.illumos.org/license/CDDL.
10.\"
11.\"
12.\" Copyright 2014 Garrett D'Amore <garrett@damore.org>
13.\"
14.Dd Jul 16, 2014
15.Dt CATAMN 1M
14.Dd Jul 13, 2014
15.Dt CATAMN 1
16.Os
17.Sh NAME
18.Nm catman
19.Nd generate
20.Nm whatis
21.database files
22.Sh SYNOPSIS
23.Nm
24.Op Fl M Ar path
25.Sh DESCRIPTION
26.The
27.Nm
28.utility generates a set of
29.Nm whatis
30.database files suitable for use with
31.Xr apropos 1
32.and
33.Xr whatis 1 .
34.It is supplied for compatibility reasons. The same databases can
35.be generated using the
36.Fl w
37.option with
38.Xr man 1 ,
39.and that command should be used instead.
40.Bl -tag -width ".Fl d"
41.It Fl M Ar path
42.Generate the
43.Nm whatis
44.database files within the specified colon separated manual paths.
45.Overrides the
46.Ev MANPATH
47.environment variable.
48.El
49.Sh ENVIRONMENT
50.The following environment variables affect the execution of
51.Nm :
52.Bl -tag -width ".Ev MANPATH"
53.It Ev MANPATH
54.Used to specify a colon seperated list of manual paths within
55.which to generate
56.Nm whatis
57.database files.
58.El
59.Sh DIAGNOSTICS
```

new/usr/src/man/man1m/catman.1m

2

```
60.The
61.Nm
62.utility exits 0 on success, and non-zero otherwise.
63.Sh INTERFACE STABILITY
64.Nm "Obsolete Committed" .
64.Obsolete.
65.Sh CODE SET INDEPENDENCE
66.Enabled.
67.Sh SEE ALSO
68.Xr apropos 1 ,
69.Xr man 1 ,
70.Xr whatis 1
```

23340 Wed Jul 16 09:10:05 2014

new/usr/src/man/man5/man.5

feedback from Hans

```

1 .\"
2 .\" Permission to use, copy, modify, and distribute this software for any
3 .\" purpose with or without fee is hereby granted, provided that the above
4 .\" copyright notice and this permission notice appear in all copies.
5 .\"
6 .\" THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
7 .\" WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
8 .\" MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
9 .\" ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
10 .\" WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
11 .\" ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
12 .\" OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
13 .\"
14 .\"
15 .\" Copyright (c) 2009, 2010, 2011 Kristaps Dzonsons <kristaps@bsd.lv>
16 .\" Copyright (c) 2011 Ingo Schwarze <schwarze@openbsd.org>
17 .\" Copyright 2012 Nexenta Systems, Inc. All rights reserved.
18 .\"
19 .Dd Jan 3, 2012
20 .Dt MAN 5
21 .Os
22 .Sh NAME
23 .Nm man
24 .Nd legacy formatting language for manual pages
25 .Sh DESCRIPTION
26 Traditionally, the
27 .Nm man
28 language has been used to write
29 .Ux
30 manuals for the
31 .Xr man 1
32 utility.
33 It supports limited control of presentational details like fonts,
34 indentation and spacing.
35 This reference document describes the structure of manual pages
36 and the syntax and usage of the man language.
37 .Pp
38 .Bf -emphasis
39 Do not use
40 .Nm
41 to write your manuals:
42 .Ef
43 It lacks support for semantic markup.
44 Use the
45 .Xr mdoc 5
46 language, instead.
47 .Pp
48 In a
49 .Nm
50 document, lines beginning with the control character
51 .Sq \&
52 are called
53 .Dq macro lines .
54 The first word is the macro name.
55 It usually consists of two capital letters.
56 For a list of available macros, see
57 .Sx MACRO OVERVIEW .
58 The words following the macro name are arguments to the macro.
59 .Pp
60 Lines not beginning with the control character are called
61 .Dq text lines .

```

```

62 They provide free-form text to be printed; the formatting of the text
63 depends on the respective processing context:
64 .Bd -literal -offset indent
65 \&.SH Macro lines change control state.
66 Text lines are interpreted within the current state.
67 .Ed
68 .Pp
69 Many aspects of the basic syntax of the
70 .Nm
71 language are based on the
72 .Xr roff 5
73 language; see the
74 .Em LANGUAGE SYNTAX
75 and
76 .Em MACRO SYNTAX
77 sections in the
78 .Xr roff 5
79 manual for details, in particular regarding
80 comments, escape sequences, whitespace, and quoting.
81 .Sh MANUAL STRUCTURE
82 Each
83 .Nm
84 document must contain the
85 .Sx \&TH
86 macro describing the document's section and title.
87 It may occur anywhere in the document, although conventionally it
88 appears as the first macro.
89 .Pp
90 Beyond
91 .Sx \&TH ,
92 at least one macro or text line must appear in the document.
93 .Pp
94 The following is a well-formed skeleton
95 .Nm
96 file for a utility
97 .Qq progname :
98 .Bd -literal -offset indent
99 \&.TH PROGNAME 1 "Oct 10, 2009"
100 \&.SH NAME
101 \efBprogname\efR \e(en a description goes here
102 \&.\e\{dq .SH LIBRARY
103 \&.\e\{dq For sections 2 & 3 only.
104 \&.SH SYNOPSIS
105 \efBprogname\efR [\efB\e-options\efR] arguments...
106 \&.SH DESCRIPTION
107 The \efBfoo\efR utility processes files...
108 \&.\e\{dq .SH IMPLEMENTATION NOTES
109 \&.\e\{dq .SH RETURN VALUES
110 \&.\e\{dq For sections 2, 3, & 9 only.
111 \&.\e\{dq .SH ENVIRONMENT
112 \&.\e\{dq For sections 1, 1M, 5, & 6 only.
113 \&.\e\{dq .SH FILES
114 \&.\e\{dq .SH EXIT STATUS
115 \&.\e\{dq For sections 1, 1M, & 6 only.
116 \&.\e\{dq .SH EXAMPLES
117 \&.\e\{dq .SH DIAGNOSTICS
118 \&.\e\{dq For sections 1, 1M, 5, 6, & 7 only.
119 \&.\e\{dq .SH ERRORS
120 \&.\e\{dq For sections 2, 3, & 9 only.
121 \&.\e\{dq .SH SEE ALSO
122 \&.\e\{dq .BR foo ( 1 )
123 \&.\e\{dq .SH STANDARDS
124 \&.\e\{dq .SH HISTORY
125 \&.\e\{dq .SH AUTHORS
126 \&.\e\{dq .SH CAVEATS
127 \&.\e\{dq .SH BUGS

```



```

128 \&.\e\{dq .SH SECURITY CONSIDERATIONS
129 .Ed
130 .Pp
131 The sections in a
132 .Nm
133 document are conventionally ordered as they appear above.
134 Sections should be composed as follows:
135 .Bl -ohang -offset indent
136 .It Em NAME
137 The name(s) and a short description of the documented material.
138 The syntax for this is generally as follows:
139 .Pp
140 .Dl \efBname\efR \e(en description
141 .It Em LIBRARY
142 The name of the library containing the documented material, which is
143 assumed to be a function in a section 2 or 3 manual.
144 For functions in the C library, this may be as follows:
145 .Pp
146 .Dl Standard C Library (libc, -lc)
147 .It Em SYNOPSIS
148 Documents the utility invocation syntax, function call syntax, or device
149 configuration.
150 .Pp
151 For the first, utilities (sections 1, 1M, and 6), this is
152 generally structured as follows:
153 .Pp
154 .Dl \efBname\efR [-\efBab\efR] [-\efBc\efR\efIarg\efR] \efBpath\efR...
155 .Pp
156 For the second, function calls (sections 2, 3, 9):
157 .Pp
158 .Dl \&.B char *name(char *\efIarg\efR);
159 .Pp
160 And for the third, configurations (section 7):
161 .Pp
162 .Dl \&.B name* at cardbus ? function ?
163 .Pp
164 Manuals not in these sections generally don't need a
165 .Em SYNOPSIS .
166 .It Em DESCRIPTION
167 This expands upon the brief, one-line description in
168 .Em NAME .
169 It usually contains a break-down of the options (if documenting a
170 command).
171 .It Em IMPLEMENTATION NOTES
172 Implementation-specific notes should be kept here.
173 This is useful when implementing standard functions that may have side
174 effects or notable algorithmic implications.
175 .It Em RETURN VALUES
176 This section documents the return values of functions in sections 2, 3, and 9.
177 .It Em ENVIRONMENT
178 Documents any usages of environment variables, e.g.,
179 .Xr environ 5 .
180 .It Em FILES
181 Documents files used.
182 It's helpful to document both the file name and a short description of how
183 the file is used (created, modified, etc.).
184 .It Em EXIT STATUS
185 This section documents the command exit status for
186 section 1, 6, and 8 utilities.
187 Historically, this information was described in
188 .Em DIAGNOSTICS ,
189 a practise that is now discouraged.
190 .It Em EXAMPLES
191 Example usages.
192 This often contains snippets of well-formed,
193 well-tested invocations.

```

```

194 Make sure that examples work properly!
195 .It Em DIAGNOSTICS
196 Documents error conditions.
197 This is most useful in section 4 manuals.
198 Historically, this section was used in place of
199 .Em EXIT STATUS
200 for manuals in sections 1, 6, and 8; however, this practise is
201 discouraged.
202 .It Em ERRORS
203 Documents error handling in sections 2, 3, and 9.
204 .It Em SEE ALSO
205 References other manuals with related topics.
206 This section should exist for most manuals.
207 .Pp
208 .Dl \&.BR bar \&( 1 \&),
209 .Pp
210 Cross-references should conventionally be ordered
211 first by section, then alphabetically.
212 .It Em STANDARDS
213 References any standards implemented or used, such as
214 .Pp
215 .Dl IEEE Std 1003.2 (\e(lqPOSIX.2\e(rq)
216 .Pp
217 If not adhering to any standards, the
218 .Em HISTORY
219 section should be used.
220 .It Em HISTORY
221 A brief history of the subject, including where support first appeared.
222 .It Em AUTHORS
223 Credits to the person or persons who wrote the code and/or documentation.
224 Authors should generally be noted by both name and email address.
225 .It Em CAVEATS
226 Common misuses and misunderstandings should be explained
227 in this section.
228 .It Em BUGS
229 Known bugs, limitations, and work-arounds should be described
230 in this section.
231 .It Em SECURITY CONSIDERATIONS
232 Documents any security precautions that operators should consider.
233 .El
234 .Sh MACRO OVERVIEW
235 This overview is sorted such that macros of similar purpose are listed
236 together, to help find the best macro for any given purpose.
237 Deprecated macros are not included in the overview, but can be found
238 in the alphabetical reference below.
239 .Ss Page header and footer meta-data
240 .Bl -column "PP, LP, P" description
241 .It Sx TH Ta set the title: Ar title section date Op Ar source Op Ar volume
242 .It Sx AT Ta display AT&T UNIX version in the page footer (<= 1 argument)
243 .It Sx UC Ta display BSD version in the page footer (<= 1 argument)
244 .El
245 .Ss Sections and paragraphs
246 .Bl -column "PP, LP, P" description
247 .It Sx SH Ta section header (one line)
248 .It Sx SS Ta subsection header (one line)
249 .It Sx PP , LP , P Ta start an undecorated paragraph (no arguments)
250 .It Sx RS , RE Ta reset the left margin: Op Ar width
251 .It Sx IP Ta indented paragraph: Op Ar head Op Ar width
252 .It Sx TP Ta tagged paragraph: Op Ar width
253 .It Sx HP Ta hanged paragraph: Op Ar width
254 .It Sx \&br Ta force output line break in text mode (no arguments)
255 .It Sx \&sp Ta force vertical space: Op Ar height
256 .It Sx fi , nf Ta fill mode and no-fill mode (no arguments)
257 .It Sx in Ta additional indent: Op Ar width
258 .El
259 .Ss Physical markup

```

```

260 .Bl -column "PP, LP, P" description
261 .It Sx B Ta boldface font
262 .It Sx I Ta italic font
263 .It Sx R Ta roman (default) font
264 .It Sx SB Ta small boldface font
265 .It Sx SM Ta small roman font
266 .It Sx BI Ta alternate between boldface and italic fonts
267 .It Sx BR Ta alternate between boldface and roman fonts
268 .It Sx IB Ta alternate between italic and boldface fonts
269 .It Sx IR Ta alternate between italic and roman fonts
270 .It Sx RB Ta alternate between roman and boldface fonts
271 .It Sx RI Ta alternate between roman and italic fonts
272 .El
273 .Ss Semantic markup
274 .Bl -column "PP, LP, P" description
275 .It Sx OP Ta optional arguments
276 .El
277 .Sh MACRO REFERENCE
278 This section is a canonical reference to all macros, arranged
279 alphabetically.
280 For the scoping of individual macros, see
281 .Sx MACRO SYNTAX .
282 .Ss \&AT
283 Sets the volume for the footer for compatibility with man pages from
284 .Tn AT&T UNIX
285 releases.
286 The optional arguments specify which release it is from.
287 .Ss \&B
288 Text is rendered in bold face.
289 .Pp
290 See also
291 .Sx \&I
292 and
293 .Sx \&R .
294 .Ss \&BI
295 Text is rendered alternately in bold face and italic.
296 Thus,
297 .Sq .BI this word and that
298 causes
299 .Sq this
300 and
301 .Sq and
302 to render in bold face, while
303 .Sq word
304 and
305 .Sq that
306 render in italics.
307 Whitespace between arguments is omitted in output.
308 .Pp
309 Examples:
310 .Pp
311 .Dl \&.BI bold italic bold italic
312 .Pp
313 The output of this example will be emboldened
314 .Dq bold
315 and italicised
316 .Dq italic ,
317 with spaces stripped between arguments.
318 .Pp
319 See also
320 .Sx \&IB ,
321 .Sx \&BR ,
322 .Sx \&RB ,
323 .Sx \&RI ,
324 and
325 .Sx \&IR .

```

```

326 .Ss \&BR
327 Text is rendered alternately in bold face and roman (the default font).
328 Whitespace between arguments is omitted in output.
329 .Pp
330 See
331 .Sx \&BI
332 for an equivalent example.
333 .Pp
334 See also
335 .Sx \&BI ,
336 .Sx \&IB ,
337 .Sx \&RB ,
338 .Sx \&RI ,
339 and
340 .Sx \&IR .
341 .Ss \&DT
342 Has no effect.
343 Included for compatibility.
344 .Ss \&HP
345 Begin a paragraph whose initial output line is left-justified, but
346 subsequent output lines are indented, with the following syntax:
347 .Bd -filled -offset indent
348 .Pf \. Sx \&HP
349 .Op Cm width
350 .Ed
351 .Pp
352 The
353 .Cm width
354 argument must conform to
355 .Sx Scaling Widths .
356 If specified, it's saved for later paragraph left-margins; if unspecified, the
357 saved or default width is used.
358 .Pp
359 See also
360 .Sx \&IP ,
361 .Sx \&LP ,
362 .Sx \&P ,
363 .Sx \&PP ,
364 and
365 .Sx \&TP .
366 .Ss \&I
367 Text is rendered in italics.
368 .Pp
369 See also
370 .Sx \&B
371 and
372 .Sx \&R .
373 .Ss \&IB
374 Text is rendered alternately in italics and bold face.
375 Whitespace between arguments is omitted in output.
376 .Pp
377 See
378 .Sx \&BI
379 for an equivalent example.
380 .Pp
381 See also
382 .Sx \&BI ,
383 .Sx \&BR ,
384 .Sx \&RB ,
385 .Sx \&RI ,
386 and
387 .Sx \&IR .
388 .Ss \&IP
389 Begin an indented paragraph with the following syntax:
390 .Bd -filled -offset indent
391 .Pf \. Sx \&IP

```

```

392 .Op Cm head Op Cm width
393 .Ed
394 .Pp
395 The
396 .Cm width
397 argument defines the width of the left margin and is defined by
398 .Sx Scaling Widths .
399 It's saved for later paragraph left-margins; if unspecified, the saved or
400 default width is used.
401 .Pp
402 The
403 .Cm head
404 argument is used as a leading term, flushed to the left margin.
405 This is useful for bulleted paragraphs and so on.
406 .Pp
407 See also
408 .Sx \&HP ,
409 .Sx \&LP ,
410 .Sx \&P ,
411 .Sx \&PP ,
412 and
413 .Sx \&TP .
414 .Ss \&IR
415 Text is rendered alternately in italics and roman (the default font).
416 Whitespace between arguments is omitted in output.
417 .Pp
418 See
419 .Sx \&BI
420 for an equivalent example.
421 .Pp
422 See also
423 .Sx \&BI ,
424 .Sx \&IB ,
425 .Sx \&BR ,
426 .Sx \&RB ,
427 and
428 .Sx \&RI .
429 .Ss \&LP
430 Begin an undecorated paragraph.
431 The scope of a paragraph is closed by a subsequent paragraph,
432 sub-section, section, or end of file.
433 The saved paragraph left-margin width is reset to the default.
434 .Pp
435 See also
436 .Sx \&HP ,
437 .Sx \&IP ,
438 .Sx \&P ,
439 .Sx \&PP ,
440 and
441 .Sx \&TP .
442 .Ss \&OP
443 Optional command-line argument.
444 This has the following syntax:
445 .Bd -filled -offset indent
446 .Pf \. Sx \&OP
447 .Cm key Op Cm value
448 .Ed
449 .Pp
450 The
451 .Cm key
452 is usually a command-line flag and
453 .Cm value
454 its argument.
455 .Ss \&P
456 Synonym for
457 .Sx \&LP .

```

```

458 .Pp
459 See also
460 .Sx \&HP ,
461 .Sx \&IP ,
462 .Sx \&LP ,
463 .Sx \&PP ,
464 and
465 .Sx \&TP .
466 .Ss \&PP
467 Synonym for
468 .Sx \&LP .
469 .Pp
470 See also
471 .Sx \&HP ,
472 .Sx \&IP ,
473 .Sx \&LP ,
474 .Sx \&P ,
475 and
476 .Sx \&TP .
477 .Ss \&R
478 Text is rendered in roman (the default font).
479 .Pp
480 See also
481 .Sx \&I
482 and
483 .Sx \&B .
484 .Ss \&RB
485 Text is rendered alternately in roman (the default font) and bold face.
486 Whitespace between arguments is omitted in output.
487 .Pp
488 See
489 .Sx \&BI
490 for an equivalent example.
491 .Pp
492 See also
493 .Sx \&BI ,
494 .Sx \&IB ,
495 .Sx \&BR ,
496 .Sx \&RI ,
497 and
498 .Sx \&IR .
499 .Ss \&RE
500 Explicitly close out the scope of a prior
501 .Sx \&RS .
502 The default left margin is restored to the state of the original
503 .Sx \&RS
504 invocation.
505 .Ss \&RI
506 Text is rendered alternately in roman (the default font) and italics.
507 Whitespace between arguments is omitted in output.
508 .Pp
509 See
510 .Sx \&BI
511 for an equivalent example.
512 .Pp
513 See also
514 .Sx \&BI ,
515 .Sx \&IB ,
516 .Sx \&BR ,
517 .Sx \&RB ,
518 and
519 .Sx \&IR .
520 .Ss \&RS
521 Temporarily reset the default left margin.
522 This has the following syntax:
523 .Bd -filled -offset indent

```

```

524 .Pf \. Sx \&RS
525 .Op Cm width
526 .Ed
527 .Pp
528 The
529 .Cm width
530 argument must conform to
531 .Sx Scaling Widths .
532 If not specified, the saved or default width is used.
533 .Pp
534 See also
535 .Sx \&RE .
536 .Ss \&SB
537 Text is rendered in small size (one point smaller than the default font)
538 bold face.
539 .Ss \&SH
540 Begin a section.
541 The scope of a section is only closed by another section or the end of
542 file.
543 The paragraph left-margin width is reset to the default.
544 .Ss \&SM
545 Text is rendered in small size (one point smaller than the default
546 font).
547 .Ss \&SS
548 Begin a sub-section.
549 The scope of a sub-section is closed by a subsequent sub-section,
550 section, or end of file.
551 The paragraph left-margin width is reset to the default.
552 .Ss \&TH
553 Sets the title of the manual page with the following syntax:
554 .Bd -filled -offset indent
555 .Pf \. Sx \&TH
556 .Ar title section date
557 .Op Ar source Op Ar volume
558 .Ed
559 .Pp
560 Conventionally, the document
561 .Ar title
562 is given in all caps.
563 The recommended
564 .Ar date
565 format is
566 .Sy YYYY-MM-DD
567 as specified in the ISO-8601 standard;
568 if the argument does not conform, it is printed verbatim.
569 If the
570 .Ar date
571 is empty or not specified, the current date is used.
572 The optional
573 .Ar source
574 string specifies the organisation providing the utility.
575 The
576 .Ar volume
577 string replaces the default rendered volume, which is dictated by the
578 manual section.
579 .Pp
580 Examples:
581 .Pp
582 .Dl \&TH CVS 5 "1992-02-12" GNU
583 .Ss \&TP
584 Begin a paragraph where the head, if exceeding the indentation width, is
585 followed by a newline; if not, the body follows on the same line after a
586 buffer to the indentation width.
587 Subsequent output lines are indented.
588 The syntax is as follows:
589 .Bd -filled -offset indent

```

```

590 .Pf \. Sx \&TP
591 .Op Cm width
592 .Ed
593 .Pp
594 The
595 .Cm width
596 argument must conform to
597 .Sx Scaling Widths .
598 If specified, it's saved for later paragraph left-margins; if
599 unspecified, the saved or default width is used.
600 .Pp
601 See also
602 .Sx \&HP ,
603 .Sx \&IP ,
604 .Sx \&LP ,
605 .Sx \&P ,
606 and
607 .Sx \&PP .
608 .Ss \&UC
609 Sets the volume for the footer for compatibility with man pages from
610 BSD releases.
611 The optional first argument specifies which release it is from.
612 .Ss \&br
613 Breaks the current line.
614 Consecutive invocations have no further effect.
615 .Pp
616 See also
617 .Sx \&sp .
618 .Ss \&fi
619 End literal mode begun by
620 .Sx \&nf .
621 .Ss \&ft
622 Change the current font mode.
623 See
624 .Sx Text Decoration
625 for a listing of available font modes.
626 .Ss \&in
627 Indent relative to the current indentation:
628 .Pp
629 .D1 Pf \. Sx \&in Op Cm width
630 .Pp
631 If
632 .Cm width
633 is signed, the new offset is relative.
634 Otherwise, it is absolute.
635 This value is reset upon the next paragraph, section, or sub-section.
636 .Ss \&na
637 Don't align to the right margin.
638 .Ss \&nf
639 Begin literal mode: all subsequent free-form lines have their end of
640 line boundaries preserved.
641 May be ended by
642 .Sx \&fi .
643 Literal mode is implicitly ended by
644 .Sx \&SH
645 or
646 .Sx \&SS .
647 .Ss \&sp
648 Insert vertical spaces into output with the following syntax:
649 .Bd -filled -offset indent
650 .Pf \. Sx \&sp
651 .Op Cm height
652 .Ed
653 .Pp
654 Insert
655 .Cm height

```

```

656 spaces, which must conform to
657 .Sx Scaling Widths .
658 If 0, this is equivalent to the
659 .Sx \&br
660 macro.
661 Defaults to 1, if unspecified.
662 .Pp
663 See also
664 .Sx \&br .
665 .Sh MACRO SYNTAX
666 The
667 .Nm
668 macros are classified by scope: line scope or block scope.
669 Line macros are only scoped to the current line (and, in some
670 situations, the subsequent line).
671 Block macros are scoped to the current line and subsequent lines until
672 closed by another block macro.
673 .Ss Line Macros
674 Line macros are generally scoped to the current line, with the body
675 consisting of zero or more arguments.
676 If a macro is scoped to the next line and the line arguments are empty,
677 the next line, which must be text, is used instead.
678 Thus:
679 .Bd -literal -offset indent
680 \&.I
681 foo
682 .Ed
683 .Pp
684 is equivalent to
685 .Sq \&.I foo .
686 If next-line macros are invoked consecutively, only the last is used.
687 If a next-line macro is followed by a non-next-line macro, an error is
688 raised, except for
689 .Sx \&br ,
690 .Sx \&sp ,
691 and
692 .Sx \&na .
693 .Pp
694 The syntax is as follows:
695 .Bd -literal -offset indent
696 \&.Y0 \&(\lBbody...\&(rB
697 \&(\lBbody...\&(rB
698 .Ed
699 .Bl -column "MacroX" "ArgumentsX" "ScopeXXXXX" "CompatX" -offset indent
700 .It Em Macro Ta Em Arguments Ta Em Scope Ta Em Notes
701 .It Sx \&AT Ta <=1 Ta current Ta \&
702 .It Sx \&BI Ta n Ta next-line Ta \&
703 .It Sx \&BI Ta n Ta current Ta \&
704 .It Sx \&BR Ta n Ta current Ta \&
705 .It Sx \&DT Ta 0 Ta current Ta \&
706 .It Sx \&I Ta n Ta next-line Ta \&
707 .It Sx \&IB Ta n Ta current Ta \&
708 .It Sx \&IR Ta n Ta current Ta \&
709 .It Sx \&OP Ta 0, 1 Ta current Ta compat
710 .It Sx \&R Ta n Ta next-line Ta \&
711 .It Sx \&RB Ta n Ta current Ta \&
712 .It Sx \&RI Ta n Ta current Ta \&
713 .It Sx \&SB Ta n Ta next-line Ta \&
714 .It Sx \&SM Ta n Ta next-line Ta \&
715 .It Sx \&TH Ta >1, <6 Ta current Ta \&
716 .It Sx \&UC Ta <=1 Ta current Ta \&
717 .It Sx \&br Ta 0 Ta current Ta compat
718 .It Sx \&fi Ta 0 Ta current Ta compat
719 .It Sx \&ft Ta 1 Ta current Ta compat
720 .It Sx \&in Ta 1 Ta current Ta compat
721 .It Sx \&na Ta 0 Ta current Ta compat

```

```

722 .It Sx \&nf Ta 0 Ta current Ta compat
723 .It Sx \&sp Ta 1 Ta current Ta compat
724 .El
725 .Pp
726 Macros marked as
727 .Qq compat
728 are included for compatibility with the significant corpus of existing
729 manuals that mix dialects of roff.
730 These macros should not be used for portable
731 .Nm
732 manuals.
733 .Ss Block Macros
734 Block macros comprise a head and body.
735 As with in-line macros, the head is scoped to the current line and, in
736 one circumstance, the next line (the next-line stipulations as in
737 .Sx Line Macros
738 apply here as well).
739 .Pp
740 The syntax is as follows:
741 .Bd -literal -offset indent
742 \&.Y0 \&(\lBhead...\&(rB
743 \&(\lBhead...\&(rB
744 \&(\lBbody...\&(rB
745 .Ed
746 .Pp
747 The closure of body scope may be to the section, where a macro is closed
748 by
749 .Sx \&SH ;
750 sub-section, closed by a section or
751 .Sx \&SS ;
752 part, closed by a section, sub-section, or
753 .Sx \&RE ;
754 or paragraph, closed by a section, sub-section, part,
755 .Sx \&HP ,
756 .Sx \&IP ,
757 .Sx \&LP ,
758 .Sx \&P ,
759 .Sx \&PP ,
760 or
761 .Sx \&TP .
762 No closure refers to an explicit block closing macro.
763 .Pp
764 As a rule, block macros may not be nested; thus, calling a block macro
765 while another block macro scope is open, and the open scope is not
766 implicitly closed, is syntactically incorrect.
767 .Bl -column "MacroX" "ArgumentsX" "Head ScopeX" "sub-sectionX" "compatX" -offset
768 .It Em Macro Ta Em Arguments Ta Em Head Scope Ta Em Body Scope Ta Em Notes
769 .It Sx \&HP Ta <2 Ta current Ta paragraph Ta \&
770 .It Sx \&IP Ta <3 Ta current Ta paragraph Ta \&
771 .It Sx \&LP Ta 0 Ta current Ta paragraph Ta \&
772 .It Sx \&P Ta 0 Ta current Ta paragraph Ta \&
773 .It Sx \&PP Ta 0 Ta current Ta paragraph Ta \&
774 .It Sx \&RE Ta 0 Ta current Ta none Ta compat
775 .It Sx \&RS Ta 1 Ta current Ta part Ta compat
776 .It Sx \&SH Ta >0 Ta next-line Ta section Ta \&
777 .It Sx \&SS Ta >0 Ta next-line Ta sub-section Ta \&
778 .It Sx \&TP Ta n Ta next-line Ta paragraph Ta \&
779 .El
780 .Pp
781 Macros marked
782 .Qq compat
783 are as mentioned in
784 .Sx Line Macros .
785 .Pp
786 If a block macro is next-line scoped, it may only be followed by in-line
787 macros for decorating text.

```

```

788 .Ss Font handling
789 In
790 .Nm
791 documents, both
792 .Sx Physical markup
793 macros and
794 .Xr roff 5
795 .Ql \ef
796 font escape sequences can be used to choose fonts.
797 In text lines, the effect of manual font selection by escape sequences
798 only lasts until the next macro invocation; in macro lines, it only lasts
799 until the end of the macro scope.
800 Note that macros like
801 .Sx \&BR
802 open and close a font scope for each argument.
803 .Sh COMPATIBILITY
804 This section documents areas of questionable portability between
805 implementations of the
806 .Nm
807 language.
808 .Pp
809 .Bl -dash -compact
810 .It
811 Do not depend on
812 .Sx \&SH
813 or
814 .Sx \&SS
815 to close out a literal context opened with
816 .Sx \&nf .
817 This behaviour may not be portable.
818 .It
819 In quoted literals, GNU troff allowed pair-wise double-quotes to produce
820 a standalone double-quote in formatted output.
821 It is not known whether this behaviour is exhibited by other formatters.
822 .It
823 troff suppresses a newline before
824 .Sq \{aq
825 macro output; in mandoc, it is an alias for the standard
826 .Sq \&.
827 control character.
828 .It
829 The
830 .Sq \eh
831 .Pq horizontal position ,
832 .Sq \ev
833 .Pq vertical position ,
834 .Sq \em
835 .Pq text colour ,
836 .Sq \eM
837 .Pq text filling colour ,
838 .Sq \ez
839 .Pq zero-length character ,
840 .Sq \ew
841 .Pq string length ,
842 .Sq \ek
843 .Pq horizontal position marker ,
844 .Sq \eo
845 .Pq text overstrike ,
846 and
847 .Sq \es
848 .Pq text size
849 escape sequences are all discarded in mandoc.
850 .It
851 The
852 .Sq \ef
853 scaling unit is accepted by mandoc, but rendered as the default unit.

```

```

854 .It
855 The
856 .Sx \&sp
857 macro does not accept negative values in mandoc.
858 In GNU troff, this would result in strange behaviour.
859 .It
860 In page header lines, GNU troff versions up to and including 1.21
861 only print
862 .Ar volume
863 names explicitly specified in the
864 .Sx \&TH
865 macro; mandoc and newer groff print the default volume name
866 corresponding to the
867 .Ar section
868 number when no
869 .Ar volume
870 is given, like in
871 .Xr mdoc 5 .
872 .El
873 .Pp
874 The
875 .Sx OP
876 macro is part of the extended
877 .Nm
878 macro set, and may not be portable to non-GNU troff implementations.
879 .Sh INTERFACE STABILITY
880 .Nm "Obsolete Committed" .
881 .Sh SEE ALSO
882 .Xr man 1 ,
883 .Xr mandoc 1 ,
884 .Xr eqn 5 ,
885 .Xr mandoc_char 5 ,
886 .Xr mdoc 5 ,
887 .Xr roff 5 ,
888 .Xr tbl 5 ,
889 .Sh HISTORY
890 The
891 .Nm
892 language first appeared as a macro package for the roff typesetting
893 system in
894 .At v7 .
895 It was later rewritten by James Clark as a macro package for groff.
896 Eric S. Raymond wrote the extended
897 .Nm
898 macros for groff in 2007.
899 The stand-alone implementation that is part of the
900 .Xr mandoc 1
901 utility written by Kristaps Dzonsons appeared in
902 .Ox 4.6 .
903 .Sh AUTHORS
904 This
905 .Nm
906 reference was written by
907 .An Kristaps Dzonsons ,
908 .Mt kristaps@bsd.lv .
909 .Sh CAVEATS
910 Do not use this language.
911 Use
912 .Xr mdoc 5 ,
913 instead.

```

76141 Wed Jul 16 09:10:05 2014

new/usr/src/man/man5/mdoc.5

feedback from Hans

```

1 .\"
2 .\" Permission to use, copy, modify, and distribute this software for any
3 .\" purpose with or without fee is hereby granted, provided that the above
4 .\" copyright notice and this permission notice appear in all copies.
5 .\"
6 .\" THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
7 .\" WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
8 .\" MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
9 .\" ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
10 .\" WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
11 .\" ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
12 .\" OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
13 .\"
14 .\"
15 .\" Copyright (c) 2009, 2010, 2011 Kristaps Dzonsons <kristaps@bsd.lv>
16 .\" Copyright (c) 2010, 2011 Ingo Schwarze <schwarze@openbsd.org>
17 .\" Copyright 2012 Nexenta Systems, Inc. All rights reserved.
18 .\" Copyright 2014 Garrett D'Amore <garrett@dmaore.org>
19 .\"
20 .Dd Jul 16, 2014
21 .Dd Jan 3, 2012
22 .Dt MDOC 5
23 .Os
24 .Sh NAME
25 .Nm mdoc
26 .Nd semantic markup language for formatting manual pages
27 The
28 .Nm mdoc
29 language supports authoring of manual pages for the
30 .Xr man 1
31 utility by allowing semantic annotations of words, phrases,
32 page sections and complete manual pages.
33 Such annotations are used by formatting tools to achieve a uniform
34 presentation across all manuals written in
35 .Nm ,
36 and to support hyperlinking if supported by the output medium.
37 .Pp
38 This reference document describes the structure of manual pages
39 and the syntax and usage of the
40 .Nm
41 language.
42 The reference implementation of a parsing and formatting tool is
43 .Xr mandoc 1 ;
44 the
45 .Sx COMPATIBILITY
46 section describes compatibility with other implementations.
47 .Pp
48 In an
49 .Nm
50 document, lines beginning with the control character
51 .Sq \& .
52 are called
53 .Dq macro lines .
54 The first word is the macro name.
55 It consists of two or three letters.
56 Most macro names begin with a capital letter.
57 For a list of available macros, see
58 .Sx MACRO OVERVIEW .
59 The words following the macro name are arguments to the macro, optionally
60 including the names of other, callable macros; see

```

```

61 .Sx MACRO SYNTAX
62 for details.
63 .Pp
64 Lines not beginning with the control character are called
65 .Dq text lines .
66 They provide free-form text to be printed; the formatting of the text
67 depends on the respective processing context:
68 .Bd -literal -offset indent
69 \&.Sh Macro lines change control state.
70 Text lines are interpreted within the current state.
71 .Ed
72 .Pp
73 Many aspects of the basic syntax of the
74 .Nm
75 language are based on the
76 .Xr roff 5
77 language; see the
78 .Em LANGUAGE SYNTAX
79 and
80 .Em MACRO SYNTAX
81 sections in the
82 .Xr roff 5
83 manual for details, in particular regarding
84 comments, escape sequences, whitespace, and quoting.
85 However, using
86 .Xr roff 5
87 requests in
88 .Nm
89 documents is discouraged;
90 .Xr mandoc 1
91 supports some of them merely for backward compatibility.
92 .Sh MANUAL STRUCTURE
93 A well-formed
94 .Nm
95 document consists of a document prologue followed by one or more
96 sections.
97 .Pp
98 The prologue, which consists of the
99 .Sx \&Dd ,
100 .Sx \&Dt ,
101 and
102 .Sx \&Os
103 macros in that order, is required for every document.
104 .Pp
105 The first section (sections are denoted by
106 .Sx \&Sh )
107 must be the NAME section, consisting of at least one
108 .Sx \&Nm
109 followed by
110 .Sx \&Nd .
111 .Pp
112 Following that, convention dictates specifying at least the
113 .Em SYNOPSIS
114 and
115 .Em DESCRIPTION
116 sections, although this varies between manual sections.
117 .Pp
118 The following is a well-formed skeleton
119 .Nm
120 file for a utility
121 .Qq progname :
122 .Bd -literal -offset indent
123 \&.Dd Jan 1, 1970
124 \&.Dt PROGNAME section
125 \&.Os
126 \&.Sh NAME

```

```

127 \&.Nm progname
128 \&.Nd one line description
129 \&.\&.\&(dq .Sh LIBRARY
130 \&.\&.\&(dq For sections 2, 3, & 9 only.
131 \&.Sh SYNOPSIS
132 \&.Nm progname
133 \&.Op Fl options
134 \&.Ar
135 \&.Sh DESCRIPTION
136 The
137 \&.Nm
138 utility processes files ...
139 \&.\&.\&(dq .Sh IMPLEMENTATION NOTES
140 \&.\&.\&(dq .Sh RETURN VALUES
141 \&.\&.\&(dq For sections 2, 3, & 9 only.
142 \&.\&.\&(dq .Sh ENVIRONMENT
143 \&.\&.\&(dq For sections 1, 1M, 5, & 6 only.
144 \&.\&.\&(dq .Sh FILES
145 \&.\&.\&(dq .Sh EXIT STATUS
146 \&.\&.\&(dq For sections 1, 1M, & 6 only.
147 \&.\&.\&(dq .Sh EXAMPLES
148 \&.\&.\&(dq .Sh DIAGNOSTICS
149 \&.\&.\&(dq For sections 1, 1M, 5, 6, & 7 only.
150 \&.\&.\&(dq .Sh ERRORS
151 \&.\&.\&(dq For sections 2, 3, & 9 only.
152 \&.\&.\&(dq .Sh ARCHITECTURE
153 \&.\&.\&(dq .Sh CODE SET INDEPENDENCE
154 \&.\&.\&(dq For sections 1, 1M, & 3 only.
155 \&.\&.\&(dq .Sh INTERFACE STABILITY
156 \&.\&.\&(dq .Sh MT-LEVEL
157 \&.\&.\&(dq For sections 2 & 3 only.
158 \&.\&.\&(dq .Sh SEE ALSO
159 \&.\&.\&(dq .Xr foobar 1
160 \&.\&.\&(dq .Sh STANDARDS
161 \&.\&.\&(dq .Sh HISTORY
162 \&.\&.\&(dq .Sh AUTHORS
163 \&.\&.\&(dq .Sh CAVEATS
164 \&.\&.\&(dq .Sh BUGS
165 \&.\&.\&(dq .Sh SECURITY CONSIDERATIONS
166 \&.\&.\&(dq Not used in OpenBSD.
167 .Ed
168 .Pp
169 The sections in an
170 .Nm
171 document are conventionally ordered as they appear above.
172 Sections should be composed as follows:
173 .Bl -ohang -offset Ds
174 .It Em NAME
175 The name(s) and a one line description of the documented material.
176 The syntax for this as follows:
177 .Bd -literal -offset indent
178 \&.Nm name0 ,
179 \&.Nm name1 ,
180 \&.Nm name2
181 \&.Nd a one line description
182 .Ed
183 .Pp
184 Multiple
185 .Sq \&Nm
186 names should be separated by commas.
187 .Pp
188 The
189 .Sx \&Nm
190 macro(s) must precede the
191 .Sx \&Nd
192 macro.

```

```

193 .Pp
194 See
195 .Sx \&Nm
196 and
197 .Sx \&Nd .
198 .It Em LIBRARY
199 The name of the library containing the documented material, which is
200 assumed to be a function in a section 2, 3, or 9 manual.
201 The syntax for this is as follows:
202 .Bd -literal -offset indent
203 \&.Lb libarm
204 .Ed
205 .Pp
206 See
207 .Sx \&Lb .
208 .It Em SYNOPSIS
209 Documents the utility invocation syntax, function call syntax, or device
210 configuration.
211 .Pp
212 For the first, utilities (sections 1, 1M, and 6), this is
213 generally structured as follows:
214 .Bd -literal -offset indent
215 \&.Nm bar
216 \&.Op Fl v
217 \&.Op Fl o Ar file
218 \&.Op Ar
219 \&.Nm foo
220 \&.Op Fl v
221 \&.Op Fl o Ar file
222 \&.Op Ar
223 .Ed
224 .Pp
225 Commands should be ordered alphabetically.
226 .Pp
227 For the second, function calls (sections 2, 3, 9):
228 .Bd -literal -offset indent
229 \&.In header.h
230 \&.Vt extern const char *global;
231 \&.Ft "char *"
232 \&.Fn foo "const char *src"
233 \&.Ft "char *"
234 \&.Fn bar "const char *src"
235 .Ed
236 .Pp
237 Ordering of
238 .Sx \&In ,
239 .Sx \&Vt ,
240 .Sx \&Fn ,
241 and
242 .Sx \&Fo
243 macros should follow C header-file conventions.
244 .Pp
245 And for the third, configurations (section 7):
246 .Bd -literal -offset indent
247 \&.Cd \&(dqit* at isa? port 0x2e\&(dq
248 \&.Cd \&(dqit* at isa? port 0x4e\&(dq
249 .Ed
250 .Pp
251 Manuals not in these sections generally don't need a
252 .Em SYNOPSIS .
253 .Pp
254 Some macros are displayed differently in the
255 .Em SYNOPSIS
256 section, particularly
257 .Sx \&Nm ,
258 .Sx \&Cd ,

```



```

259 .Sx \&Fd ,
260 .Sx \&Fn ,
261 .Sx \&Fo ,
262 .Sx \&In ,
263 .Sx \&Vt ,
264 and
265 .Sx \&Ft .
266 All of these macros are output on their own line.
267 If two such dissimilar macros are pairwise invoked (except for
268 .Sx \&Ft
269 before
270 .Sx \&Fo
271 or
272 .Sx \&Fn ) ,
273 they are separated by a vertical space, unless in the case of
274 .Sx \&Fo ,
275 .Sx \&Fn ,
276 and
277 .Sx \&Ft ,
278 which are always separated by vertical space.
279 .Pp
280 When text and macros following an
281 .Sx \&Nm
282 macro starting an input line span multiple output lines,
283 all output lines but the first will be indented to align
284 with the text immediately following the
285 .Sx \&Nm
286 macro, up to the next
287 .Sx \&Nm ,
288 .Sx \&Sh ,
289 or
290 .Sx \&Ss
291 macro or the end of an enclosing block, whichever comes first.
292 .It Em DESCRIPTION
293 This begins with an expansion of the brief, one line description in
294 .Em NAME :
295 .Bd -literal -offset indent
296 The
297 \&.Nm
298 utility does this, that, and the other.
299 .Ed
300 .Pp
301 It usually follows with a breakdown of the options (if documenting a
302 command), such as:
303 .Bd -literal -offset indent
304 The arguments are as follows:
305 \&.Bl \-tag \-width Ds
306 \&.It Fl v
307 Print verbose information.
308 \&.El
309 .Ed
310 .Pp
311 Manuals not documenting a command won't include the above fragment.
312 .Pp
313 Since the
314 .Em DESCRIPTION
315 section usually contains most of the text of a manual, longer manuals
316 often use the
317 .Sx \&Ss
318 macro to form subsections.
319 In very long manuals, the
320 .Em DESCRIPTION
321 may be split into multiple sections, each started by an
322 .Sx \&Sh
323 macro followed by a non-standard section name, and each having
324 several subsections, like in the present

```

```

325 .Nm
326 manual.
327 .It Em IMPLEMENTATION NOTES
328 Implementation-specific notes should be kept here.
329 This is useful when implementing standard functions that may have side
330 effects or notable algorithmic implications.
331 .It Em RETURN VALUES
332 This section documents the
333 return values of functions in sections 2, 3, and 9.
334 .Pp
335 See
336 .Sx \&Rv .
337 .It Em ENVIRONMENT
338 Lists the environment variables used by the utility,
339 and explains the syntax and semantics of their values.
340 The
341 .Xr environ 5
342 manual provides examples of typical content and formatting.
343 .Pp
344 See
345 .Sx \&Ev .
346 .It Em FILES
347 Documents files used.
348 It's helpful to document both the file name and a short description of how
349 the file is used (created, modified, etc.).
350 .Pp
351 See
352 .Sx \&Pa .
353 .It Em EXIT STATUS
354 This section documents the
355 command exit status for section 1, 6, and 8 utilities.
356 Historically, this information was described in
357 .Em DIAGNOSTICS ,
358 a practise that is now discouraged.
359 .Pp
360 See
361 .Sx \&Ex .
362 .It Em EXAMPLES
363 Example usages.
364 This often contains snippets of well-formed, well-tested invocations.
365 Make sure that examples work properly!
366 .It Em DIAGNOSTICS
367 Documents error conditions.
368 This is most useful in section 4 manuals.
369 Historically, this section was used in place of
370 .Em EXIT STATUS
371 for manuals in sections 1, 6, and 8; however, this practise is
372 discouraged.
373 .Pp
374 See
375 .Sx \&Bl
376 .Fl diag .
377 .It Em ERRORS
378 Documents error handling in sections 2, 3, and 9.
379 .Pp
380 See
381 .Sx \&Er .
382 .It Em ARCHITECTURE
383 This section is usually absent, but will be present when the
384 interface is specific to one or more architectures.
385 .It Em CODE SET INDEPENDENCE
386 Indicates whether the interface operates correctly with various different
387 code sets. True independent code sets will support not only ASCII and
388 Extended UNIX Codesets (EUC), but also other multi-byte encodings such as
389 UTF-8 and GB2312.
390 .Pp

```

391 Generally there will be some limitations that are fairly standard. See
 392 .Xr standards 5 for more information about some of these. Most interfaces
 393 should support at least UTF-8 in addition to ASCII.
 394 .It Em INTERFACE STABILITY
 395 Indicates the level of commitment to the interface. Interfaces can be described
 396 with in the following ways:
 397 .Bl -tag
 398 .It Nm Standard
 399 Indicates that the interface is defined by one or more standards bodies.
 400 Generally, changes to the interface will be carefully managed to conform
 401 to the relevant standards. These interfaces are generally the most suitable
 402 for use in portable programs.
 403 .It Nm Committed
 404 Indicates that the interface is intended to be preserved for the long-haul, and
 405 will rarely, if ever change, and never without notification (barring
 406 extraordinary and extenuating circumstances). These interfaces are
 407 preferred over other interfaces with the exception of
 408 .Nm Standard
 409 interfaces.
 410 .It Nm Uncommitted
 411 Indicates that the interface may change. Generally, changes to these interfaces
 412 should be infrequent, and some effort will be made to address compatibility
 413 considerations when changing or removing such interfaces. However, there is
 414 no firm commitment to the preservation of the interface. Most often this
 415 is applied to interfaces where operational experience with the interface
 416 is still limited and some need to change may be anticipated.
 417 .Pp
 418 Consumers should expect to revalidate any
 419 .Nm Uncommitted
 420 interfaces when crossing release boundaries. Products intended for
 421 use on many releases or intended to support compatibility with future
 422 releases should avoid these interfaces.
 423 .It Nm Volatile
 424 The interface can change at any time for any reason. Often this relates to
 425 interfaces that are part of external software components that are still evolving
 426 rapidly. Consumers should not expect that the interface (either binary or
 427 source level) will be unchanged from one release to the next.
 428 .It Nm Not-an-Interface
 429 Describes something that is specifically not intended for programmatic
 430 consumption. For example, specific human-readable output, or the layout
 431 of graphical items on a user interface, may be described this way. Generally
 432 programmatic alternatives to these will be available, and should be used
 433 when programmatic consumption is needed.
 434 .It Nm Private
 435 This is an internal interface. Generally these interfaces should only be
 436 used within the project, and should not be used by other programs or modules.
 437 The interface can and will change without notice as the project needs, at
 438 any time.
 439 .Pp
 440 Most often, Private interfaces will lack any documentation whatsoever, and
 441 generally any undocumented interface can be assumed to be Private.
 442 .It Nm Obsolete
 443 The interface is not intended for use in new projects or programs, and may
 444 be removed at a future date. The
 445 .Nm Obsolete
 446 word is a modifier that can
 447 be applied to other commitment levels. For example an
 448 .Nm Obsolete Committed
 449 interface is unlikely to be removed or changed, but nonetheless new use
 450 is discouraged (perhaps a better newer alternative is present).
 451 .El
 452 .It Em MT-LEVEL
 453 This section describes considerations for the interface when used within
 454 programs that use multiple threads. More discussion of these considerations
 455 is made in the MT-Level section of
 456 .Xr attributes 5 .

457 The interface can be described in the following ways.
 458 .Bl -tag
 459 .It Nm Safe
 460 Indicates the interface is safe for use within multiple threads. There
 461 may be additional caveats that apply, in which case those will be
 462 described. Note that some interfaces have semantics which may affect
 463 other threads, but these should be an intrinsic part of the interface
 464 rather than an unexpected side effect. For example, closing a file in
 465 one thread will cause that file to be closed in all threads.
 466 .It Nm Unsafe
 467 Indicates the interface is unsuitable for concurrent use within multiple
 468 threads. A threaded application may still make use of the interface, but
 469 will be required to provide external synchronization means to ensure that
 470 only a single thread calls the interface at a time.
 471 .It Nm MT-Safe
 472 Indicates that the interface is not only safe for concurrent use, but is
 473 designed for such use. For example, a
 474 .Nm Safe
 475 interface may make use of a global lock to provide safety, but at reduced
 476 internal concurrency, whereas an
 477 .Nm MT-Safe
 478 interface will be designed to be efficient even when used concurrently.
 479 .It Nm Async-Signal-Safe
 480 Indicates that the library is safe for use within a signal handler. An
 481 .Nm MT-Safe
 482 interface can be made
 483 .Nm Async-Signal-Safe
 484 by ensuring that it blocks signals when acquiring locks.
 485 .It Nm Safe with Exections
 486 As for
 487 .Nm Safe
 488 but with specific exceptions noted.
 489 .It Nm MT-Safe with Exections
 490 As for
 491 .Nm MT-Safe
 492 but with specific exceptions noted.
 493 .El
 494 .It Em SEE ALSO
 495 References other manuals with related topics.
 496 This section should exist for most manuals.
 497 Cross-references should conventionally be ordered first by section, then
 498 alphabetically.
 499 .Pp
 500 References to other documentation concerning the topic of the manual page,
 501 for example authoritative books or journal articles, may also be
 502 provided in this section.
 503 .Pp
 504 See
 505 .Sx \&Rs
 506 and
 507 .Sx \&Xr .
 508 .It Em STANDARDS
 509 References any standards implemented or used.
 510 If not adhering to any standards, the
 511 .Em HISTORY
 512 section should be used instead.
 513 .Pp
 514 See
 515 .Sx \&St .
 516 .It Em HISTORY
 517 A brief history of the subject, including where it was first implemented,
 518 and when it was ported to or reimplemented for the operating system at hand.
 519 .It Em AUTHORS
 520 Credits to the person or persons who wrote the code and/or documentation.
 521 Authors should generally be noted by both name and email address.
 522 .Pp

```

523 See
524 .Sx \&An .
525 .It Em CAVEATS
526 Common misuses and misunderstandings should be explained
527 in this section.
528 .It Em BUGS
529 Known bugs, limitations, and work-arounds should be described
530 in this section.
531 .It Em SECURITY CONSIDERATIONS
532 Documents any security precautions that operators should consider.
533 .El
534 .Sh MACRO OVERVIEW
535 This overview is sorted such that macros of similar purpose are listed
536 together, to help find the best macro for any given purpose.
537 Deprecated macros are not included in the overview, but can be found below
538 in the alphabetical
539 .Sx MACRO REFERENCE .
540 .Ss Document preamble and NAME section macros
541 .Bl -column "Brq, Bro, Brc" description
542 .It Sx \&Dd Ta document date: Ar month day , year
543 .It Sx \&Dt Ta document title: Ar TITLE SECTION Op Ar volume | arch
544 .It Sx \&Os Ta operating system version: Op Ar system Op Ar version
545 .It Sx \&Nm Ta document name (one argument)
546 .It Sx \&Nd Ta document description (one line)
547 .El
548 .Ss Sections and cross references
549 .Bl -column "Brq, Bro, Brc" description
550 .It Sx \&Sh Ta section header (one line)
551 .It Sx \&Ss Ta subsection header (one line)
552 .It Sx \&Sx Ta internal cross reference to a section or subsection
553 .It Sx \&Xr Ta cross reference to another manual page: Ar name section
554 .It Sx \&Pp , \&Lp Ta start a text paragraph (no arguments)
555 .El
556 .Ss Displays and lists
557 .Bl -column "Brq, Bro, Brc" description
558 .It Sx \&Bd , \&Ed Ta display block:
559 .Fl Ar type
560 .Op Fl offset Ar width
561 .Op Fl compact
562 .It Sx \&Dl Ta indented display (one line)
563 .It Sx \&DL Ta indented literal display (one line)
564 .It Sx \&Bl , \&El Ta list block:
565 .Fl Ar type
566 .Op Fl width Ar val
567 .Op Fl offset Ar val
568 .Op Fl compact
569 .It Sx \&It Ta list item (syntax depends on Fl Ar type )
570 .It Sx \&Ta Ta table cell separator in Sx \&Bl Fl column No lists
571 .It Sx \&Rs , \&%* , \&Re Ta bibliographic block (references)
572 .El
573 .Ss Spacing control
574 .Bl -column "Brq, Bro, Brc" description
575 .It Sx \&Pf Ta prefix, no following horizontal space (one argument)
576 .It Sx \&Ns Ta roman font, no preceding horizontal space (no arguments)
577 .It Sx \&Ap Ta apostrophe without surrounding whitespace (no arguments)
578 .It Sx \&Sm Ta switch horizontal spacing mode: Cm on | off
579 .It Sx \&Bk , \&Ek Ta keep block: Fl words
580 .It Sx \&br Ta force output line break in text mode (no arguments)
581 .It Sx \&sp Ta force vertical space: Op Ar height
582 .El
583 .Ss Semantic markup for command line utilities:
584 .Bl -column "Brq, Bro, Brc" description
585 .It Sx \&Nm Ta start a SYNOPSIS block with the name of a utility
586 .It Sx \&Fl Ta command line options (flags) (>=0 arguments)
587 .It Sx \&Cm Ta command modifier (>0 arguments)
588 .It Sx \&Ar Ta command arguments (>=0 arguments)

```

```

589 .It Sx \&Op , \&Oo , \&Oc Ta optional syntax elements (enclosure)
590 .It Sx \&Ic Ta internal or interactive command (>0 arguments)
591 .It Sx \&Ev Ta environmental variable (>0 arguments)
592 .It Sx \&Pa Ta file system path (>=0 arguments)
593 .El
594 .Ss Semantic markup for function libraries:
595 .Bl -column "Brq, Bro, Brc" description
596 .It Sx \&Lb Ta function library (one argument)
597 .It Sx \&In Ta include file (one argument)
598 .It Sx \&Ft Ta function type (>0 arguments)
599 .It Sx \&Fo , \&Fc Ta function block: Ar funcname
600 .It Sx \&Fn Ta function name:
601 .Op Ar functype
602 .Ar funcname
603 .Oo
604 .Op Ar argtype
605 .Ar argname
606 .Oc
607 .It Sx \&Fa Ta function argument (>0 arguments)
608 .It Sx \&Vt Ta variable type (>0 arguments)
609 .It Sx \&Va Ta variable name (>0 arguments)
610 .It Sx \&Dv Ta defined variable or preprocessor constant (>0 arguments)
611 .It Sx \&Er Ta error constant (>0 arguments)
612 .It Sx \&Ev Ta environmental variable (>0 arguments)
613 .El
614 .Ss Various semantic markup:
615 .Bl -column "Brq, Bro, Brc" description
616 .It Sx \&An Ta author name (>0 arguments)
617 .It Sx \&Lk Ta hyperlink: Ar uri Op Ar name
618 .It Sx \&Mt Ta Do mailto Dc hyperlink: Ar address
619 .It Sx \&Cd Ta kernel configuration declaration (>0 arguments)
620 .It Sx \&Ad Ta memory address (>0 arguments)
621 .It Sx \&Ms Ta mathematical symbol (>0 arguments)
622 .It Sx \&Tn Ta tradename (>0 arguments)
623 .El
624 .Ss Physical markup
625 .Bl -column "Brq, Bro, Brc" description
626 .It Sx \&Em Ta italic font or underline (emphasis) (>0 arguments)
627 .It Sx \&Sy Ta boldface font (symbolic) (>0 arguments)
628 .It Sx \&Li Ta typewriter font (literal) (>0 arguments)
629 .It Sx \&No Ta return to roman font (normal) (no arguments)
630 .It Sx \&Bf , \&Ef Ta font block:
631 .Op Fl Ar type | Cm \&Em | \&Li | \&Sy
632 .El
633 .Ss Physical enclosures
634 .Bl -column "Brq, Bro, Brc" description
635 .It Sx \&Dq , \&Do , \&Dc Ta enclose in typographic double quotes: Dq text
636 .It Sx \&Qq , \&Qo , \&Qc Ta enclose in typewriter double quotes: Qq text
637 .It Sx \&Sq , \&So , \&Sc Ta enclose in single quotes: Sq text
638 .It Sx \&Ql Ta single-quoted literal text: Ql text
639 .It Sx \&Pq , \&Po , \&Pc Ta enclose in parentheses: Pq text
640 .It Sx \&Bq , \&Bo , \&Bc Ta enclose in square brackets: Bq text
641 .It Sx \&Brq , \&Bro , \&Brc Ta enclose in curly braces: Brq text
642 .It Sx \&Aq , \&Ao , \&Ac Ta enclose in angle brackets: Aq text
643 .It Sx \&Eo , \&Ec Ta generic enclosure
644 .El
645 .Ss Text production
646 .Bl -column "Brq, Bro, Brc" description
647 .It Sx \&Ex Fl std Ta standard command exit values: Op Ar utility ...
648 .It Sx \&Rv Fl std Ta standard function return values: Op Ar function ...
649 .It Sx \&St Ta reference to a standards document (one argument)
650 .It Sx \&Ux Ta Ux
651 .It Sx \&At Ta At
652 .It Sx \&Bx Ta Bx
653 .It Sx \&Bsx Ta Bsx
654 .It Sx \&Nx Ta Nx

```

```

655 .It Sx \&Fx Ta Fx
656 .It Sx \&Ox Ta Ox
657 .It Sx \&Dx Ta Dx
658 .El
659 .Sh MACRO REFERENCE
660 This section is a canonical reference of all macros, arranged
661 alphabetically.
662 For the scoping of individual macros, see
663 .Sx MACRO SYNTAX .
664 .Ss \&%A
665 Author name of an
666 .Sx \&Rs
667 block.
668 Multiple authors should each be accorded their own
669 .Sx \&%A
670 line.
671 Author names should be ordered with full or abbreviated forename(s)
672 first, then full surname.
673 .Ss \&%B
674 Book title of an
675 .Sx \&Rs
676 block.
677 This macro may also be used in a non-bibliographic context when
678 referring to book titles.
679 .Ss \&%C
680 Publication city or location of an
681 .Sx \&Rs
682 block.
683 .Ss \&%D
684 Publication date of an
685 .Sx \&Rs
686 block.
687 Recommended formats of arguments are
688 .Ar month day , year
689 or just
690 .Ar year .
691 .Ss \&%I
692 Publisher or issuer name of an
693 .Sx \&Rs
694 block.
695 .Ss \&%J
696 Journal name of an
697 .Sx \&Rs
698 block.
699 .Ss \&%N
700 Issue number (usually for journals) of an
701 .Sx \&Rs
702 block.
703 .Ss \&%O
704 Optional information of an
705 .Sx \&Rs
706 block.
707 .Ss \&%P
708 Book or journal page number of an
709 .Sx \&Rs
710 block.
711 .Ss \&%Q
712 Institutional author (school, government, etc.) of an
713 .Sx \&Rs
714 block.
715 Multiple institutional authors should each be accorded their own
716 .Sx \&%Q
717 line.
718 .Ss \&%R
719 Technical report name of an
720 .Sx \&Rs

```

```

721 block.
722 .Ss \&%T
723 Article title of an
724 .Sx \&Rs
725 block.
726 This macro may also be used in a non-bibliographical context when
727 referring to article titles.
728 .Ss \&%U
729 URI of reference document.
730 .Ss \&%V
731 Volume number of an
732 .Sx \&Rs
733 block.
734 .Ss \&%Ac
735 Close an
736 .Sx \&Ao
737 block.
738 Does not have any tail arguments.
739 .Ss \&Ad
740 Memory address.
741 Do not use this for postal addresses.
742 .Pp
743 Examples:
744 .Dl \&.Ad [0,$]
745 .Dl \&.Ad 0x00000000
746 .Ss \&An
747 Author name.
748 Can be used both for the authors of the program, function, or driver
749 documented in the manual, or for the authors of the manual itself.
750 Requires either the name of an author or one of the following arguments:
751 .Pp
752 .Bl -tag -width "nosplit" -offset indent -compact
753 .It Fl split
754 Start a new output line before each subsequent invocation of
755 .Sx \&An .
756 .It Fl nosplit
757 The opposite of
758 .Fl split .
759 .El
760 .Pp
761 The default is
762 .Fl nosplit .
763 The effect of selecting either of the
764 .Fl split
765 modes ends at the beginning of the
766 .Em AUTHORS
767 section.
768 In the
769 .Em AUTHORS
770 section, the default is
771 .Fl nosplit
772 for the first author listing and
773 .Fl split
774 for all other author listings.
775 .Pp
776 Examples:
777 .Dl \&.An -nosplit
778 .Dl \&.An Kristaps Dzonsons \&Aq kristaps@bsd.lv
779 .Ss \&Ao
780 Begin a block enclosed by angle brackets.
781 Does not have any head arguments.
782 .Pp
783 Examples:
784 .Dl \&.Fl -key= \&Ns \&Ao \&Ar val \&Ac
785 .Pp
786 See also

```

```

787 .Sx \&Aq .
788 .Ss \&Ap
789 Inserts an apostrophe without any surrounding whitespace.
790 This is generally used as a grammatical device when referring to the verb
791 form of a function.
792 .Pp
793 Examples:
794 .Dl \&.Fn execve \&Ap d
795 .Ss \&Aq
796 Encloses its arguments in angle brackets.
797 .Pp
798 Examples:
799 .Dl \&.Fl -key= \&Ns \&Aq \&Ar val
800 .Pp
801 .Em Remarks :
802 this macro is often abused for rendering URIs, which should instead use
803 .Sx \&Lk
804 or
805 .Sx \&Mt ,
806 or to note pre-processor
807 .Dq Li #include
808 statements, which should use
809 .Sx \&In .
810 .Pp
811 See also
812 .Sx \&Ao .
813 .Ss \&Ar
814 Command arguments.
815 If an argument is not provided, the string
816 .Dq file ...\&
817 is used as a default.
818 .Pp
819 Examples:
820 .Dl ".Fl o Ar file"
821 .Dl ".Ar"
822 .Dl ".Ar arg1 , arg2 ."
823 .Pp
824 The arguments to the
825 .Sx \&Ar
826 macro are names and placeholders for command arguments;
827 for fixed strings to be passed verbatim as arguments, use
828 .Sx \&Fl
829 or
830 .Sx \&Cm .
831 .Ss \&At
832 Formats an AT&T version.
833 Accepts one optional argument:
834 .Pp
835 .Bl -tag -width "v[1-7] | 32vX" -offset indent -compact
836 .It Cm v[1-7] | 32v
837 A version of
838 .At .
839 .It Cm III
840 .At III .
841 .It Cm V[.1-4]]?
842 A version of
843 .At V .
844 .El
845 .Pp
846 Note that these arguments do not begin with a hyphen.
847 .Pp
848 Examples:
849 .Dl \&.At
850 .Dl \&.At III
851 .Dl \&.At V.1
852 .Pp

```

```

853 See also
854 .Sx \&Bsx ,
855 .Sx \&Bx ,
856 .Sx \&Dx ,
857 .Sx \&Fx ,
858 .Sx \&Nx ,
859 .Sx \&Ox ,
860 and
861 .Sx \&Ux .
862 .Ss \&Bc
863 Close a
864 .Sx \&Bo
865 block.
866 Does not have any tail arguments.
867 .Ss \&Bd
868 Begin a display block.
869 Its syntax is as follows:
870 .Bd -ragged -offset indent
871 .Pf \. Sx \&Bd
872 .Fl Ns Ar type
873 .Op Fl offset Ar width
874 .Op Fl compact
875 .Ed
876 .Pp
877 Display blocks are used to select a different indentation and
878 justification than the one used by the surrounding text.
879 They may contain both macro lines and text lines.
880 By default, a display block is preceded by a vertical space.
881 .Pp
882 The
883 .Ar type
884 must be one of the following:
885 .Bl -tag -width 13n -offset indent
886 .It Fl centered
887 Produce one output line from each input line, and centre-justify each line.
888 Using this display type is not recommended; many
889 .Nm
890 implementations render it poorly.
891 .It Fl filled
892 Change the positions of line breaks to fill each line, and left- and
893 right-justify the resulting block.
894 .It Fl literal
895 Produce one output line from each input line,
896 and do not justify the block at all.
897 Preserve white space as it appears in the input.
898 Always use a constant-width font.
899 Use this for displaying source code.
900 .It Fl ragged
901 Change the positions of line breaks to fill each line, and left-justify
902 the resulting block.
903 .It Fl unfilled
904 The same as
905 .Fl literal ,
906 but using the same font as for normal text, which is a variable width font
907 if supported by the output device.
908 .El
909 .Pp
910 The
911 .Ar type
912 must be provided first.
913 Additional arguments may follow:
914 .Bl -tag -width 13n -offset indent
915 .It Fl offset Ar width
916 Indent the display by the
917 .Ar width ,
918 which may be one of the following:

```

```

919 .Bl -item
920 .It
921 One of the pre-defined strings
922 .Cm indent
923 the width of a standard indentation (six constant width characters);
924 .Cm indent-two ,
925 twice
926 .Cm indent ;
927 .Cm left ,
928 which has no effect;
929 .Cm right ,
930 which justifies to the right margin; or
931 .Cm center ,
932 which aligns around an imagined centre axis.
933 .It
934 A macro invocation, which selects a predefined width
935 associated with that macro.
936 The most popular is the imaginary macro
937 .Ar \&Ds ,
938 which resolves to
939 .Sy 6n .
940 .It
941 A width using the syntax described in
942 .Sx Scaling Widths .
943 .It
944 An arbitrary string, which indents by the length of this string.
945 .El
946 .Pp
947 When the argument is missing,
948 .Fl offset
949 is ignored.
950 .It Fl compact
951 Do not assert vertical space before the display.
952 .El
953 .Pp
954 Examples:
955 .Bd -literal -offset indent
956 \&.Bd \-literal \-offset indent \-compact
957 Hello world.
958 \&.Ed
959 .Ed
960 .Pp
961 See also
962 .Sx \&D1
963 and
964 .Sx \&Dl .
965 .Ss \&Bf
966 Change the font mode for a scoped block of text.
967 Its syntax is as follows:
968 .Bd -ragged -offset indent
969 .Pf \. Sx \&Bf
970 .Oo
971 .Fl emphasis | literal | symbolic |
972 .Cm \&Em | \&Li | \&Sy
973 .Oc
974 .Ed
975 .Pp
976 The
977 .Fl emphasis
978 and
979 .Cm \&Em
980 argument are equivalent, as are
981 .Fl symbolic
982 and
983 .Cm \&Sy ,
984 and

```

```

985 .Fl literal
986 and
987 .Cm \&Li .
988 Without an argument, this macro does nothing.
989 The font mode continues until broken by a new font mode in a nested
990 scope or
991 .Sx \&Ef
992 is encountered.
993 .Pp
994 See also
995 .Sx \&Li ,
996 .Sx \&Ef ,
997 .Sx \&Em ,
998 and
999 .Sx \&Sy .
1000 .Ss \&Bk
1001 For each macro, keep its output together on the same output line,
1002 until the end of the macro or the end of the input line is reached,
1003 whichever comes first.
1004 Line breaks in text lines are unaffected.
1005 The syntax is as follows:
1006 .Pp
1007 .Dl Pf \. Sx \&Bk Fl words
1008 .Pp
1009 The
1010 .Fl words
1011 argument is required; additional arguments are ignored.
1012 .Pp
1013 The following example will not break within each
1014 .Sx \&Op
1015 macro line:
1016 .Bd -literal -offset indent
1017 \&.Bk \-words
1018 \&.Op Fl f Ar flags
1019 \&.Op Fl o Ar output
1020 \&.Ek
1021 .Ed
1022 .Pp
1023 Be careful in using over-long lines within a keep block!
1024 Doing so will clobber the right margin.
1025 .Ss \&Bl
1026 Begin a list.
1027 Lists consist of items specified using the
1028 .Sx \&It
1029 macro, containing a head or a body or both.
1030 The list syntax is as follows:
1031 .Bd -ragged -offset indent
1032 .Pf \. Sx \&Bl
1033 .Fl Ns Ar type
1034 .Op Fl width Ar val
1035 .Op Fl offset Ar val
1036 .Op Fl compact
1037 .Op HEAD ...
1038 .Ed
1039 .Pp
1040 The list
1041 .Ar type
1042 is mandatory and must be specified first.
1043 The
1044 .Fl width
1045 and
1046 .Fl offset
1047 arguments accept
1048 .Sx Scaling Widths
1049 or use the length of the given string.
1050 The

```

1051 .Fl offset
 1052 is a global indentation for the whole list, affecting both item heads
 1053 and bodies.
 1054 For those list types supporting it, the
 1055 .Fl width
 1056 argument requests an additional indentation of item bodies,
 1057 to be added to the
 1058 .Fl offset .
 1059 Unless the
 1060 .Fl compact
 1061 argument is specified, list entries are separated by vertical space.
 1062 .Pp
 1063 A list must specify one of the following list types:
 1064 .Bl -tag -width 12n -offset indent
 1065 .It Fl bullet
 1066 No item heads can be specified, but a bullet will be printed at the head
 1067 of each item.
 1068 Item bodies start on the same output line as the bullet
 1069 and are indented according to the
 1070 .Fl width
 1071 argument.
 1072 .It Fl column
 1073 A columnated list.
 1074 The
 1075 .Fl width
 1076 argument has no effect; instead, each argument specifies the width
 1077 of one column, using either the
 1078 .Sx Scaling Widths
 1079 syntax or the string length of the argument.
 1080 If the first line of the body of a
 1081 .Fl column
 1082 list is not an
 1083 .Sx \&It
 1084 macro line,
 1085 .Sx \&It
 1086 contexts spanning one input line each are implied until an
 1087 .Sx \&It
 1088 macro line is encountered, at which point items start being interpreted as
 1089 described in the
 1090 .Sx \&It
 1091 documentation.
 1092 .It Fl dash
 1093 Like
 1094 .Fl bullet ,
 1095 except that dashes are used in place of bullets.
 1096 .It Fl diag
 1097 Like
 1098 .Fl inset ,
 1099 except that item heads are not parsed for macro invocations.
 1100 Most often used in the
 1101 .Em DIAGNOSTICS
 1102 section with error constants in the item heads.
 1103 .It Fl enum
 1104 A numbered list.
 1105 No item heads can be specified.
 1106 Formatted like
 1107 .Fl bullet ,
 1108 except that cardinal numbers are used in place of bullets,
 1109 starting at 1.
 1110 .It Fl hang
 1111 Like
 1112 .Fl tag ,
 1113 except that the first lines of item bodies are not indented, but follow
 1114 the item heads like in
 1115 .Fl inset
 1116 lists.

1117 .It Fl hyphen
 1118 Synonym for
 1119 .Fl dash .
 1120 .It Fl inset
 1121 Item bodies follow items heads on the same line, using normal inter-word
 1122 spacing.
 1123 Bodies are not indented, and the
 1124 .Fl width
 1125 argument is ignored.
 1126 .It Fl item
 1127 No item heads can be specified, and none are printed.
 1128 Bodies are not indented, and the
 1129 .Fl width
 1130 argument is ignored.
 1131 .It Fl ohang
 1132 Item bodies start on the line following item heads and are not indented.
 1133 The
 1134 .Fl width
 1135 argument is ignored.
 1136 .It Fl tag
 1137 Item bodies are indented according to the
 1138 .Fl width
 1139 argument.
 1140 When an item head fits inside the indentation, the item body follows
 1141 this head on the same output line.
 1142 Otherwise, the body starts on the output line following the head.
 1143 .El
 1144 .Pp
 1145 Lists may be nested within lists and displays.
 1146 Nesting of
 1147 .Fl column
 1148 and
 1149 .Fl enum
 1150 lists may not be portable.
 1151 .Pp
 1152 See also
 1153 .Sx \&El
 1154 and
 1155 .Sx \&It .
 1156 .Ss \&Bo
 1157 Begin a block enclosed by square brackets.
 1158 Does not have any head arguments.
 1159 .Pp
 1160 Examples:
 1161 .Bd -literal -offset indent -compact
 1162 \&.Bo 1 ,
 1163 \&.Dv BUFSIZ \&Bc
 1164 .Ed
 1165 .Pp
 1166 See also
 1167 .Sx \&Bq .
 1168 .Ss \&Bq
 1169 Encloses its arguments in square brackets.
 1170 .Pp
 1171 Examples:
 1172 .Dl \&.Bq 1 , \&Dv BUFSIZ
 1173 .Pp
 1174 .Em Remarks :
 1175 this macro is sometimes abused to emulate optional arguments for
 1176 commands; the correct macros to use for this purpose are
 1177 .Sx \&Op ,
 1178 .Sx \&Oo ,
 1179 and
 1180 .Sx \&Oc .
 1181 .Pp
 1182 See also

```

1183 .Sx \&Bo .
1184 .Ss \&Brc
1185 Close a
1186 .Sx \&Bro
1187 block.
1188 Does not have any tail arguments.
1189 .Ss \&Bro
1190 Begin a block enclosed by curly braces.
1191 Does not have any head arguments.
1192 .Pp
1193 Examples:
1194 .Bd -literal -offset indent -compact
1195 \&.Bro 1 , ... ,
1196 \&.Va n \&Brc
1197 .Ed
1198 .Pp
1199 See also
1200 .Sx \&Brq .
1201 .Ss \&Brq
1202 Encloses its arguments in curly braces.
1203 .Pp
1204 Examples:
1205 .Dl \&.Brq 1 , ... , \&.Va n
1206 .Pp
1207 See also
1208 .Sx \&Bro .
1209 .Ss \&Bsx
1210 Format the BSD/OS version provided as an argument, or a default value if
1211 no argument is provided.
1212 .Pp
1213 Examples:
1214 .Dl \&.Bsx 1.0
1215 .Dl \&.Bsx
1216 .Pp
1217 See also
1218 .Sx \&At ,
1219 .Sx \&Bx ,
1220 .Sx \&Dx ,
1221 .Sx \&Fx ,
1222 .Sx \&Nx ,
1223 .Sx \&Ox ,
1224 and
1225 .Sx \&Ux .
1226 .Ss \&Bt
1227 Prints
1228 .Dq is currently in beta test.
1229 .Ss \&Bx
1230 Format the BSD version provided as an argument, or a default value if no
1231 argument is provided.
1232 .Pp
1233 Examples:
1234 .Dl \&.Bx 4.3 Tahoe
1235 .Dl \&.Bx 4.4
1236 .Dl \&.Bx
1237 .Pp
1238 See also
1239 .Sx \&At ,
1240 .Sx \&Bsx ,
1241 .Sx \&Dx ,
1242 .Sx \&Fx ,
1243 .Sx \&Nx ,
1244 .Sx \&Ox ,
1245 and
1246 .Sx \&Ux .
1247 .Ss \&Cd
1248 Kernel configuration declaration.

```

```

1249 This denotes strings accepted by
1250 .Xr config 8 .
1251 It is most often used in section 4 manual pages.
1252 .Pp
1253 Examples:
1254 .Dl \&.Cd device le0 at scode?
1255 .Pp
1256 .Em Remarks :
1257 this macro is commonly abused by using quoted literals to retain
1258 whitespace and align consecutive
1259 .Sx \&Cd
1260 declarations.
1261 This practise is discouraged.
1262 .Ss \&Cm
1263 Command modifiers.
1264 Typically used for fixed strings passed as arguments, unless
1265 .Sx \&Fl
1266 is more appropriate.
1267 Also useful when specifying configuration options or keys.
1268 .Pp
1269 Examples:
1270 .Dl ".Nm mt Fl f Ar device Cm rewind"
1271 .Dl ".Nm ps Fl o Cm pid , Ns Cm command"
1272 .Dl ".Nm dd Cm if= Ns Ar file1 Cm of= Ns Ar file2"
1273 .Dl ".Cm IdentityFile Pa ~/.ssh/id_rsa"
1274 .Dl ".Cm LogLevel Dv DEBUG"
1275 .Ss \&D1
1276 One-line indented display.
1277 This is formatted by the default rules and is useful for simple indented
1278 statements.
1279 It is followed by a newline.
1280 .Pp
1281 Examples:
1282 .Dl \&.D1 \&Fl abcdefgh
1283 .Pp
1284 See also
1285 .Sx \&Bd
1286 and
1287 .Sx \&Dl .
1288 .Ss \&Db
1289 Switch debugging mode.
1290 Its syntax is as follows:
1291 .Pp
1292 .D1 Pf \. Sx \&Db Cm on | off
1293 .Pp
1294 This macro is ignored by
1295 .Xr mandoc 1 .
1296 .Ss \&Dc
1297 Close a
1298 .Sx \&Do
1299 block.
1300 Does not have any tail arguments.
1301 .Ss \&Dd
1302 Document date.
1303 This is the mandatory first macro of any
1304 .Nm
1305 manual.
1306 Its syntax is as follows:
1307 .Pp
1308 .D1 Pf \. Sx \&Dd Ar month day , year
1309 .Pp
1310 The
1311 .Ar month
1312 is the full English month name, the
1313 .Ar day
1314 is an optionally zero-padded numeral, and the

```



```

1315 .Ar year
1316 is the full four-digit year.
1317 .Pp
1318 Other arguments are not portable; the
1319 .Xr mandoc 1
1320 utility handles them as follows:
1321 .Bl -dash -offset 3n -compact
1322 .It
1323 To have the date automatically filled in by the
1324 .Ox
1325 version of
1326 .Xr cvs 1 ,
1327 the special string
1328 .Dq $&Mdocdate$
1329 can be given as an argument.
1330 .It
1331 A few alternative date formats are accepted as well
1332 and converted to the standard form.
1333 .It
1334 If a date string cannot be parsed, it is used verbatim.
1335 .It
1336 If no date string is given, the current date is used.
1337 .El
1338 .Pp
1339 Examples:
1340 .Dl \&.Dd $&Mdocdate$
1341 .Dl \&.Dd $&Mdocdate: July 21 2007$
1342 .Dl \&.Dd July 21, 2007
1343 .Pp
1344 See also
1345 .Sx \&Dt
1346 and
1347 .Sx \&Os .
1348 .Ss \&Dl
1349 One-line intended display.
1350 This is formatted as literal text and is useful for commands and
1351 invocations.
1352 It is followed by a newline.
1353 .Pp
1354 Examples:
1355 .Dl \&.Dl % mandoc mdoc.5 \e(ba less
1356 .Dl \&.Dl % mandoc mdoc.7 \e(ba less
1357 .Pp
1358 See also
1359 .Sx \&Bd
1360 and
1361 .Sx \&D1 .
1362 .Ss \&Do
1363 Begin a block enclosed by double quotes.
1364 Does not have any head arguments.
1365 .Pp
1366 Examples:
1367 .Bd -literal -offset indent -compact
1368 \&.Do
1369 April is the cruellest month
1370 \e(em T.S. Eliot
1371 .Ed
1372 .Pp
1373 See also
1374 .Sx \&Dq .
1375 .Ss \&Dq
1376 Encloses its arguments in
1377 .Dq typographic
1378 double-quotes.
1379 .Pp

```

```

1380 Examples:
1381 .Bd -literal -offset indent -compact
1382 \&.Dq April is the cruellest month
1383 \e(em T.S. Eliot
1384 .Ed
1385 .Pp
1386 See also
1387 .Sx \&Qq ,
1388 .Sx \&Sq ,
1389 and
1390 .Sx \&Do .
1391 .Ss \&Dt
1392 Document title.
1393 This is the mandatory second macro of any
1394 .Nm
1395 file.
1396 Its syntax is as follows:
1397 .Bd -ragged -offset indent
1398 .Pf \. Sx \&Dt
1399 .Oo
1400 .Ar title
1401 .Oo
1402 .Ar section
1403 .Op Ar volume
1404 .Op Ar arch
1405 .Oc
1406 .Oc
1407 .Ed
1408 .Pp
1409 Its arguments are as follows:
1410 .Bl -tag -width Ds -offset Ds
1411 .It Ar title
1412 The document's title (name), defaulting to
1413 .Dq UNKNOWN
1414 if unspecified.
1415 It should be capitalised.
1416 .It Ar section
1417 The manual section. It should correspond to the manual's filename suffix
1418 and defaults to
1419 .Dq 1
1420 if unspecified.
1421 .It Ar volume
1422 This overrides the volume inferred from
1423 .Ar section .
1424 This field is optional.
1425 .It Ar arch
1426 This specifies the machine architecture a manual page applies to,
1427 where relevant.
1428 .El
1429 .Ss \&Dv
1430 Defined variables such as preprocessor constants, constant symbols,
1431 enumeration values, and so on.
1432 .Pp
1433 Examples:
1434 .Dl \&.Dv NULL
1435 .Dl \&.Dv BUFSIZ
1436 .Dl \&.Dv STDOUT_FILENO
1437 .Pp
1438 See also
1439 .Sx \&Er
1440 and
1441 .Sx \&Ev
1442 for special-purpose constants and
1443 .Sx \&Va
1444 for variable symbols.
1445 .Ss \&Dx

```

```

1446 Format the DragonFly BSD version provided as an argument, or a default
1447 value if no argument is provided.
1448 .Pp
1449 Examples:
1450 .Dl \&.Dx 2.4.1
1451 .Dl \&.Dx
1452 .Pp
1453 See also
1454 .Sx \&At ,
1455 .Sx \&Bsx ,
1456 .Sx \&Bx ,
1457 .Sx \&Fx ,
1458 .Sx \&Nx ,
1459 .Sx \&Ox ,
1460 and
1461 .Sx \&Ux .
1462 .Ss \&Ec
1463 Close a scope started by
1464 .Sx \&Eo .
1465 Its syntax is as follows:
1466 .Pp
1467 .Dl Pf \. Sx \&Ec Op Ar TERM
1468 .Pp
1469 The
1470 .Ar TERM
1471 argument is used as the enclosure tail, for example, specifying \e(rq
1472 will emulate
1473 .Sx \&Dc .
1474 .Ss \&Ed
1475 End a display context started by
1476 .Sx \&Bd .
1477 .Ss \&Ef
1478 End a font mode context started by
1479 .Sx \&Bf .
1480 .Ss \&Ek
1481 End a keep context started by
1482 .Sx \&Bk .
1483 .Ss \&El
1484 End a list context started by
1485 .Sx \&Bl .
1486 .Pp
1487 See also
1488 .Sx \&Bl
1489 and
1490 .Sx \&It .
1491 .Ss \&Em
1492 Denotes text that should be
1493 .Em emphasised .
1494 Note that this is a presentation term and should not be used for
1495 stylistically decorating technical terms.
1496 Depending on the output device, this is usually represented
1497 using an italic font or underlined characters.
1498 .Pp
1499 Examples:
1500 .Dl \&.Em Warnings!
1501 .Dl \&.Em Remarks :
1502 .Pp
1503 See also
1504 .Sx \&Bf ,
1505 .Sx \&Li ,
1506 .Sx \&No ,
1507 and
1508 .Sx \&Sy .
1509 .Ss \&En
1510 This macro is obsolete and not implemented in
1511 .Xr mandoc 1 .

```

```

1512 .Ss \&Eo
1513 An arbitrary enclosure.
1514 Its syntax is as follows:
1515 .Pp
1516 .Dl Pf \. Sx \&Eo Op Ar TERM
1517 .Pp
1518 The
1519 .Ar TERM
1520 argument is used as the enclosure head, for example, specifying \e(lq
1521 will emulate
1522 .Sx \&Do .
1523 .Ss \&Er
1524 Error constants for definitions of the
1525 .Va errno
1526 libc global variable.
1527 This is most often used in section 2 and 3 manual pages.
1528 .Pp
1529 Examples:
1530 .Dl \&.Er EPERM
1531 .Dl \&.Er ENOENT
1532 .Pp
1533 See also
1534 .Sx \&Dv
1535 for general constants.
1536 .Ss \&Es
1537 This macro is obsolete and not implemented.
1538 .Ss \&Ev
1539 Environmental variables such as those specified in
1540 .Xr environ 5 .
1541 .Xr environ 7 .
1542 .Pp
1543 Examples:
1544 .Dl \&.Ev DISPLAY
1545 .Dl \&.Ev PATH
1546 .Pp
1547 See also
1548 .Sx \&Dv
1549 for general constants.
1550 .Ss \&Ex
1551 Insert a standard sentence regarding command exit values of 0 on success
1552 and >0 on failure.
1553 This is most often used in section 1, 6, and 8 manual pages.
1554 Its syntax is as follows:
1555 .Pp
1556 .Dl Pf \. Sx \&Ex Fl std Op Ar utility ...
1557 .Pp
1558 If
1559 .Ar utility
1560 is not specified, the document's name set by
1561 .Sx \&Nm
1562 is used.
1563 Multiple
1564 .Ar utility
1565 arguments are treated as separate utilities.
1566 .Pp
1567 See also
1568 .Sx \&RV .
1569 .Ss \&Fa
1570 Function argument.
1571 Its syntax is as follows:
1572 .Dl -ragged -offset indent
1573 .Pp Pf \. Sx \&Fa
1574 .Op Cm argtype
1575 .Cm argname
1576 .Ed
1577 .Pp

```

```

1577 This may be invoked for names with or without the corresponding type.
1578 It is also used to specify the field name of a structure.
1579 Most often, the
1580 .Sx \&Fa
1581 macro is used in the
1582 .Em SYNOPSIS
1583 within
1584 .Sx \&Fo
1585 section when documenting multi-line function prototypes.
1586 If invoked with multiple arguments, the arguments are separated by a
1587 comma.
1588 Furthermore, if the following macro is another
1589 .Sx \&Fa ,
1590 the last argument will also have a trailing comma.
1591 .Pp
1592 Examples:
1593 .Dl \&Fa \((dqconst char *p\((dq
1594 .Dl \&Fa \((dqint a\((dq \((dqint b\((dq \((dqint c\((dq
1595 .Dl \&Fa foo
1596 .Pp
1597 See also
1598 .Sx \&Fo .
1599 .Ss \&Fc
1600 End a function context started by
1601 .Sx \&Fo .
1602 .Ss \&Fd
1603 Historically used to document include files.
1604 This usage has been deprecated in favour of
1605 .Sx \&In .
1606 Do not use this macro.
1607 .Pp
1608 See also
1609 .Sx MANUAL STRUCTURE
1610 and
1611 .Sx \&In .
1612 .Ss \&Fl
1613 Command-line flag or option.
1614 Used when listing arguments to command-line utilities.
1615 Prints a fixed-width hyphen
1616 .Sq \-
1617 directly followed by each argument.
1618 If no arguments are provided, a hyphen is printed followed by a space.
1619 If the argument is a macro, a hyphen is prefixed to the subsequent macro
1620 output.
1621 .Pp
1622 Examples:
1623 .Dl ".Fl R Op Fl H | L | P"
1624 .Dl ".Op Fl 1AaCcdFfgHhikLlmnopqRrSsTtux"
1625 .Dl ".Fl type Cm d Fl name Pa CVS"
1626 .Dl ".Fl Ar signal_number"
1627 .Dl ".Fl o Fl"
1628 .Pp
1629 See also
1630 .Sx \&Cm .
1631 .Ss \&Fn
1632 A function name.
1633 Its syntax is as follows:
1634 .Bd -ragged -offset indent
1635 .Pf \. Ns Sx \&Fn
1636 .Op Ar functype
1637 .Ar funcname
1638 .Op Oo Ar argtype Oc Ar argname
1639 .Ed
1640 .Pp
1641 Function arguments are surrounded in parenthesis and
1642 are delimited by commas.

```

```

1643 If no arguments are specified, blank parenthesis are output.
1644 In the
1645 .Em SYNOPSIS
1646 section, this macro starts a new output line,
1647 and a blank line is automatically inserted between function definitions.
1648 .Pp
1649 Examples:
1650 .Dl \&Fn \((dqint funcname\((dq \((dqint arg0\((dq \((dqint arg1\((dq
1651 .Dl \&Fn funcname \((dqint arg0\((dq
1652 .Dl \&Fn funcname arg0
1653 .Pp
1654 .Bd -literal -offset indent -compact
1655 \&.Ft functype
1656 \&.Fn funcname
1657 .Ed
1658 .Pp
1659 When referring to a function documented in another manual page, use
1660 .Sx \&Xr
1661 instead.
1662 See also
1663 .Sx MANUAL STRUCTURE ,
1664 .Sx \&Fo ,
1665 and
1666 .Sx \&Ft .
1667 .Ss \&Fo
1668 Begin a function block.
1669 This is a multi-line version of
1670 .Sx \&Fn .
1671 Its syntax is as follows:
1672 .Pp
1673 .Dl Pf \. Sx \&Fo Ar funcname
1674 .Pp
1675 Invocations usually occur in the following context:
1676 .Bd -ragged -offset indent
1677 .Pf \. Sx \&Ft Ar functype
1678 .br
1679 .Pf \. Sx \&Fo Ar funcname
1680 .br
1681 .Pf \. Sx \&Fa Oo Ar argtype Oc Ar argname
1682 .br
1683 \&.\.\.
1684 .br
1685 .Pf \. Sx \&Fc
1686 .Ed
1687 .Pp
1688 A
1689 .Sx \&Fo
1690 scope is closed by
1691 .Sx \&Fc .
1692 .Pp
1693 See also
1694 .Sx MANUAL STRUCTURE ,
1695 .Sx \&Fa ,
1696 .Sx \&Fc ,
1697 and
1698 .Sx \&Ft .
1699 .Ss \&Fr
1700 This macro is obsolete and not implemented in
1701 .Xr mandoc 1 .
1702 .Pp
1703 It was used to show function return values.
1704 The syntax was:
1705 .Pp
1706 .Dl Pf . Sx \&Fr Ar value
1707 .Ss \&Ft
1708 A function type.

```

1709 Its syntax is as follows:
 1710 .Pp
 1711 .Dl Pf \. Sx \&Ft Ar functype
 1712 .Pp
 1713 In the
 1714 .Em SYNOPSIS
 1715 section, a new output line is started after this macro.
 1716 .Pp
 1717 Examples:
 1718 .Dl \&Ft int
 1719 .Bd -literal -offset indent -compact
 1720 \&Ft functype
 1721 \&Fn funcname
 1722 .Ed
 1723 .Pp
 1724 See also
 1725 .Sx MANUAL STRUCTURE ,
 1726 .Sx \&Fn ,
 1727 and
 1728 .Sx \&Fo .
 1729 .Ss \&Fx
 1730 Format the
 1731 .Fx
 1732 version provided as an argument, or a default value
 1733 if no argument is provided.
 1734 .Pp
 1735 Examples:
 1736 .Dl \&Fx 7.1
 1737 .Dl \&Fx
 1738 .Pp
 1739 See also
 1740 .Sx \&At ,
 1741 .Sx \&Bsx ,
 1742 .Sx \&Bx ,
 1743 .Sx \&Dx ,
 1744 .Sx \&Nx ,
 1745 .Sx \&Ox ,
 1746 and
 1747 .Sx \&Ux .
 1748 .Ss \&Hf
 1749 This macro is not implemented in
 1750 .Xr mandoc 1 .
 1751 .Pp
 1752 It was used to include the contents of a (header) file literally.
 1753 The syntax was:
 1754 .Pp
 1755 .Dl Pf . Sx \&Hf Ar filename
 1756 .Ss \&Ic
 1757 Designate an internal or interactive command.
 1758 This is similar to
 1759 .Sx \&Cm
 1760 but used for instructions rather than values.
 1761 .Pp
 1762 Examples:
 1763 .Dl \&Ic :wq
 1764 .Dl \&Ic hash
 1765 .Dl \&Ic alias
 1766 .Pp
 1767 Note that using
 1768 .Sx \&Bd Fl literal
 1769 or
 1770 .Sx \&Dl
 1771 is preferred for displaying code; the
 1772 .Sx \&Ic
 1773 macro is used when referring to specific instructions.
 1774 .Ss \&In

1775 An
 1776 .Dq include
 1777 file.
 1778 When invoked as the first macro on an input line in the
 1779 .Em SYNOPSIS
 1780 section, the argument is displayed in angle brackets
 1781 and preceded by
 1782 .Dq #include ,
 1783 and a blank line is inserted in front if there is a preceding
 1784 function declaration.
 1785 This is most often used in section 2, 3, and 9 manual pages.
 1786 .Pp
 1787 Examples:
 1788 .Dl \&In sys/types.h
 1789 .Pp
 1790 See also
 1791 .Sx MANUAL STRUCTURE .
 1792 .Ss \&It
 1793 A list item.
 1794 The syntax of this macro depends on the list type.
 1795 .Pp
 1796 Lists
 1797 of type
 1798 .Fl hang ,
 1799 .Fl ohang ,
 1800 .Fl inset ,
 1801 and
 1802 .Fl diag
 1803 have the following syntax:
 1804 .Pp
 1805 .Dl Pf \. Sx \&It Ar args
 1806 .Pp
 1807 Lists of type
 1808 .Fl bullet ,
 1809 .Fl dash ,
 1810 .Fl enum ,
 1811 .Fl hyphen
 1812 and
 1813 .Fl item
 1814 have the following syntax:
 1815 .Pp
 1816 .Dl Pf \. Sx \&It
 1817 .Pp
 1818 with subsequent lines interpreted within the scope of the
 1819 .Sx \&It
 1820 until either a closing
 1821 .Sx \&El
 1822 or another
 1823 .Sx \&It .
 1824 .Pp
 1825 The
 1826 .Fl tag
 1827 list has the following syntax:
 1828 .Pp
 1829 .Dl Pf \. Sx \&It Op Cm args
 1830 .Pp
 1831 Subsequent lines are interpreted as with
 1832 .Fl bullet
 1833 and family.
 1834 The line arguments correspond to the list's left-hand side; body
 1835 arguments correspond to the list's contents.
 1836 .Pp
 1837 The
 1838 .Fl column
 1839 list is the most complicated.
 1840 Its syntax is as follows:

1841 .Pp
 1842 .D1 Pf \. Sx \&It Ar cell Op <TAB> Ar cell ...
 1843 .D1 Pf \. Sx \&It Ar cell Op Sx \&Ta Ar cell ...
 1844 .Pp
 1845 The arguments consist of one or more lines of text and macros
 1846 representing a complete table line.
 1847 Cells within the line are delimited by tabs or by the special
 1848 .Sx \&Ta
 1849 block macro.
 1850 The tab cell delimiter may only be used within the
 1851 .Sx \&It
 1852 line itself; on following lines, only the
 1853 .Sx \&Ta
 1854 macro can be used to delimit cells, and
 1855 .Sx \&Ta
 1856 is only recognised as a macro when called by other macros,
 1857 not as the first macro on a line.
 1858 .Pp
 1859 Note that quoted strings may span tab-delimited cells on an
 1860 .Sx \&It
 1861 line.
 1862 For example,
 1863 .Pp
 1864 .Dl .It \{(dqcol1 ; <TAB> col2 ;\}(dq \&;
 1865 .Pp
 1866 will preserve the semicolon whitespace except for the last.
 1867 .Pp
 1868 See also
 1869 .Sx \&Bl .
 1870 .Ss \&Lb
 1871 Specify a library.
 1872 The syntax is as follows:
 1873 .Pp
 1874 .D1 Pf \. Sx \&Lb Ar library
 1875 .Pp
 1876 The
 1877 .Ar library
 1878 parameter may be a system library, such as
 1879 .Cm libz
 1880 or
 1881 .Cm libpam ,
 1882 in which case a small library description is printed next to the linker
 1883 invocation; or a custom library, in which case the library name is
 1884 printed in quotes.
 1885 This is most commonly used in the
 1886 .Em SYNOPSIS
 1887 section as described in
 1888 .Sx MANUAL STRUCTURE .
 1889 .Pp
 1890 Examples:
 1891 .Dl \&.Lb libz
 1892 .Dl \&.Lb mdoc
 1893 .Ss \&Li
 1894 Denotes text that should be in a
 1895 .Li literal
 1896 font mode.
 1897 Note that this is a presentation term and should not be used for
 1898 stylistically decorating technical terms.
 1899 .Pp
 1900 On terminal output devices, this is often indistinguishable from
 1901 normal text.
 1902 .Pp
 1903 See also
 1904 .Sx \&Bf ,
 1905 .Sx \&Em ,
 1906 .Sx \&No ,

1907 and
 1908 .Sx \&Sy .
 1909 .Ss \&Lk
 1910 Format a hyperlink.
 1911 Its syntax is as follows:
 1912 .Pp
 1913 .D1 Pf \. Sx \&Lk Ar uri Op Ar name
 1914 .Pp
 1915 Examples:
 1916 .Dl \&.Lk http://bsd.lv \{(dqThe BSD.lv Project\}(dq
 1917 .Dl \&.Lk http://bsd.lv
 1918 .Pp
 1919 See also
 1920 .Sx \&Mt .
 1921 .Ss \&Lp
 1922 Synonym for
 1923 .Sx \&Pp .
 1924 .Ss \&Ms
 1925 Display a mathematical symbol.
 1926 Its syntax is as follows:
 1927 .Pp
 1928 .D1 Pf \. Sx \&Ms Ar symbol
 1929 .Pp
 1930 Examples:
 1931 .Dl \&.Ms sigma
 1932 .Dl \&.Ms aleph
 1933 .Ss \&Mt
 1934 Format a
 1935 .Dq mailto:
 1936 hyperlink.
 1937 Its syntax is as follows:
 1938 .Pp
 1939 .D1 Pf \. Sx \&Mt Ar address
 1940 .Pp
 1941 Examples:
 1942 .Dl \&.Mt discuss@manpages.bsd.lv
 1943 .Ss \&Nd
 1944 A one line description of the manual's content.
 1945 This may only be invoked in the
 1946 .Em SYNOPSIS
 1947 section subsequent the
 1948 .Sx \&Nm
 1949 macro.
 1950 .Pp
 1951 Examples:
 1952 .Dl Pf . Sx \&Nd mdoc language reference
 1953 .Dl Pf . Sx \&Nd format and display UNIX manuals
 1954 .Pp
 1955 The
 1956 .Sx \&Nd
 1957 macro technically accepts child macros and terminates with a subsequent
 1958 .Sx \&Sh
 1959 invocation.
 1960 Do not assume this behaviour: some
 1961 .Xr whatis 1
 1962 database generators are not smart enough to parse more than the line
 1963 arguments and will display macros verbatim.
 1964 .Pp
 1965 See also
 1966 .Sx \&Nm .
 1967 .Ss \&Nm
 1968 The name of the manual page, or \{(em in particular in section 1, 6,
 1969 and 8 pages \{(em of an additional command or feature documented in
 1970 the manual page.
 1971 When first invoked, the
 1972 .Sx \&Nm

1973 macro expects a single argument, the name of the manual page.
 1974 Usually, the first invocation happens in the
 1975 .Em NAME
 1976 section of the page.
 1977 The specified name will be remembered and used whenever the macro is
 1978 called again without arguments later in the page.
 1979 The
 1980 .Sx \&Nm
 1981 macro uses
 1982 .Sx Block full-implicit
 1983 semantics when invoked as the first macro on an input line in the
 1984 .Em SYNOPSIS
 1985 section; otherwise, it uses ordinary
 1986 .Sx In-line
 1987 semantics.
 1988 .Pp
 1989 Examples:
 1990 .Bd -literal -offset indent
 1991 \&.Sh SYNOPSIS
 1992 \&.Nm cat
 1993 \&.Op Fl benstuv
 1994 \&.Op Ar
 1995 .Ed
 1996 .Pp
 1997 In the
 1998 .Em SYNOPSIS
 1999 of section 2, 3 and 9 manual pages, use the
 2000 .Sx \&Fn
 2001 macro rather than
 2002 .Sx \&Nm
 2003 to mark up the name of the manual page.
 2004 .Ss \&No
 2005 Normal text.
 2006 Closes the scope of any preceding in-line macro.
 2007 When used after physical formatting macros like
 2008 .Sx \&Em
 2009 or
 2010 .Sx \&Sy
 2011 switches back to the standard font face and weight.
 2012 Can also be used to embed plain text strings in macro lines
 2013 using semantic annotation macros.
 2014 .Pp
 2015 Examples:
 2016 .Dl ".Em italic , Sy bold , No and roman"
 2017 .Pp
 2018 .Bd -literal -offset indent -compact
 2019 \&.Sm off
 2020 \&.Cm :C No / Ar pattern No / Ar replacement No /
 2021 \&.Sm on
 2022 .Ed
 2023 .Pp
 2024 See also
 2025 .Sx \&Em ,
 2026 .Sx \&Li ,
 2027 and
 2028 .Sx \&Sy .
 2029 .Ss \&Ns
 2030 Suppress a space between the output of the preceding macro
 2031 and the following text or macro.
 2032 Following invocation, input is interpreted as normal text
 2033 just like after an
 2034 .Sx \&No
 2035 macro.
 2036 .Pp
 2037 This has no effect when invoked at the start of a macro line.
 2038 .Pp

2039 Examples:
 2040 .Dl ".Ar name Ns = Ns Ar value"
 2041 .Dl ".Cm :M Ns Ar pattern"
 2042 .Dl ".Fl o Ns Ar output"
 2043 .Pp
 2044 See also
 2045 .Sx \&No
 2046 and
 2047 .Sx \&Sm .
 2048 .Ss \&Nx
 2049 Format the
 2050 .Nx
 2051 version provided as an argument, or a default value if
 2052 no argument is provided.
 2053 .Pp
 2054 Examples:
 2055 .Dl \&.Nx 5.01
 2056 .Dl \&.Nx
 2057 .Pp
 2058 See also
 2059 .Sx \&At ,
 2060 .Sx \&Bsx ,
 2061 .Sx \&Bx ,
 2062 .Sx \&Dx ,
 2063 .Sx \&Fx ,
 2064 .Sx \&Ox ,
 2065 and
 2066 .Sx \&Ux .
 2067 .Ss \&Oc
 2068 Close multi-line
 2069 .Sx \&Oo
 2070 context.
 2071 .Ss \&Oo
 2072 Multi-line version of
 2073 .Sx \&Op .
 2074 .Pp
 2075 Examples:
 2076 .Bd -literal -offset indent -compact
 2077 \&.Oo
 2078 \&.Op Fl flag Ns Ar value
 2079 \&.Oc
 2080 .Ed
 2081 .Ss \&Op
 2082 Optional part of a command line.
 2083 Prints the argument(s) in brackets.
 2084 This is most often used in the
 2085 .Em SYNOPSIS
 2086 section of section 1 and 8 manual pages.
 2087 .Pp
 2088 Examples:
 2089 .Dl \&.Op \&Fl a \&Ar b
 2090 .Dl \&.Op \&Ar a | b
 2091 .Pp
 2092 See also
 2093 .Sx \&Oo .
 2094 .Ss \&Os
 2095 Document operating system version.
 2096 This is the mandatory third macro of
 2097 any
 2098 .Nm
 2099 file.
 2100 Its syntax is as follows:
 2101 .Pp
 2102 .D1 Pf \. Sx \&Os Op Ar system Op Ar version
 2103 .Pp
 2104 The optional

```

2105 .Ar system
2106 parameter specifies the relevant operating system or environment.
2107 Left unspecified, it defaults to the local operating system version.
2108 This is the suggested form.
2109 .Pp
2110 Examples:
2111 .Dl \&.Os
2112 .Dl \&.Os KTH/CSC/TCS
2113 .Dl \&.Os BSD 4.3
2114 .Pp
2115 See also
2116 .Sx \&Dd
2117 and
2118 .Sx \&Dt .
2119 .Ss \&Ot
2120 This macro is obsolete and not implemented in
2121 .Xr mandoc 1 .
2122 .Pp
2123 Historical
2124 .Xr mdoc 5
2125 packages described it as
2126 .Dq "old function type (FORTRAN)" .
2127 .Ss \&Ox
2128 Format the
2129 .Ox
2130 version provided as an argument, or a default value
2131 if no argument is provided.
2132 .Pp
2133 Examples:
2134 .Dl \&.Ox 4.5
2135 .Dl \&.Ox
2136 .Pp
2137 See also
2138 .Sx \&At ,
2139 .Sx \&Bsx ,
2140 .Sx \&Bx ,
2141 .Sx \&Dx ,
2142 .Sx \&Fx ,
2143 .Sx \&Nx ,
2144 and
2145 .Sx \&Ux .
2146 .Ss \&Pa
2147 An absolute or relative file system path, or a file or directory name.
2148 If an argument is not provided, the character
2149 .Sq \{(ti
2150 is used as a default.
2151 .Pp
2152 Examples:
2153 .Dl \&.Pa /usr/bin/mandoc
2154 .Dl \&.Pa /usr/share/man/man5/mdoc.5
2035 .Dl \&.Pa /usr/share/man/man7/mdoc.7
2155 .Pp
2156 See also
2157 .Sx \&Lk .
2158 .Ss \&Pc
2159 Close parenthesised context opened by
2160 .Sx \&Po .
2161 .Ss \&Pf
2162 Removes the space between its argument
2163 .Pq Dq prefix
2164 and the following macro.
2165 Its syntax is as follows:
2166 .Pp
2167 .Dl .Pf Ar prefix macro arguments ...
2168 .Pp
2169 This is equivalent to:

```

```

2170 .Pp
2171 .Dl .No Ar prefix No \&Ns Ar macro arguments ...
2172 .Pp
2173 Examples:
2174 .Dl ".Pf $ Ar variable_name"
2175 .Dl ".Pf 0x Ar hex_digits"
2176 .Pp
2177 See also
2178 .Sx \&Ns
2179 and
2180 .Sx \&Sm .
2181 .Ss \&Po
2182 Multi-line version of
2183 .Sx \&Pq .
2184 .Ss \&Pp
2185 Break a paragraph.
2186 This will assert vertical space between prior and subsequent macros
2187 and/or text.
2188 .Pp
2189 Paragraph breaks are not needed before or after
2190 .Sx \&Sh
2191 or
2192 .Sx \&Ss
2193 macros or before displays
2194 .Pq Sx \&Bd
2195 or lists
2196 .Pq Sx \&Bl
2197 unless the
2198 .Fl compact
2199 flag is given.
2200 .Ss \&Pq
2201 Parenthesised enclosure.
2202 .Pp
2203 See also
2204 .Sx \&Po .
2205 .Ss \&Qc
2206 Close quoted context opened by
2207 .Sx \&Qo .
2208 .Ss \&Ql
2209 Format a single-quoted literal.
2210 See also
2211 .Sx \&Qq
2212 and
2213 .Sx \&Sq .
2214 .Ss \&Qo
2215 Multi-line version of
2216 .Sx \&Qq .
2217 .Ss \&Qq
2218 Encloses its arguments in
2219 .Qq typewriter
2220 double-quotes.
2221 Consider using
2222 .Sx \&Dq .
2223 .Pp
2224 See also
2225 .Sx \&Dq ,
2226 .Sx \&Sq ,
2227 and
2228 .Sx \&Qo .
2229 .Ss \&Re
2230 Close an
2231 .Sx \&Rs
2232 block.
2233 Does not have any tail arguments.
2234 .Ss \&Rs
2235 Begin a bibliographic

```

```

2236 .Pq Dq reference
2237 block.
2238 Does not have any head arguments.
2239 The block macro may only contain
2240 .Sx \&%A ,
2241 .Sx \&%B ,
2242 .Sx \&%C ,
2243 .Sx \&%D ,
2244 .Sx \&%I ,
2245 .Sx \&%J ,
2246 .Sx \&%N ,
2247 .Sx \&%O ,
2248 .Sx \&%P ,
2249 .Sx \&%Q ,
2250 .Sx \&%R ,
2251 .Sx \&%T ,
2252 .Sx \&%U ,
2253 and
2254 .Sx \&%V
2255 child macros (at least one must be specified).
2256 .Pp
2257 Examples:
2258 .Bd -literal -offset indent -compact
2259 \&.Rs
2260 \&.%A J. E. Hopcroft
2261 \&.%A J. D. Ullman
2262 \&.%B Introduction to Automata Theory, Languages, and Computation
2263 \&.%I Addison-Wesley
2264 \&.%C Reading, Massachusettes
2265 \&.%D 1979
2266 \&.Re
2267 .Ed
2268 .Pp
2269 If an
2270 .Sx \&Rs
2271 block is used within a SEE ALSO section, a vertical space is asserted
2272 before the rendered output, else the block continues on the current
2273 line.
2274 .Ss \&Rv
2275 Insert a standard sentence regarding a function call's return value of 0
2276 on success and \-1 on error, with the
2277 .Va errno
2278 libc global variable set on error.
2279 Its syntax is as follows:
2280 .Pp
2281 .D1 Pf \. Sx \&Rv Fl std Op Ar function ...
2282 .Pp
2283 If
2284 .Ar function
2285 is not specified, the document's name set by
2286 .Sx \&Nm
2287 is used.
2288 Multiple
2289 .Ar function
2290 arguments are treated as separate functions.
2291 .Pp
2292 See also
2293 .Sx \&Ex .
2294 .Ss \&Sc
2295 Close single-quoted context opened by
2296 .Sx \&So .
2297 .Ss \&Sh
2298 Begin a new section.
2299 For a list of conventional manual sections, see
2300 .Sx MANUAL STRUCTURE .
2301 These sections should be used unless it's absolutely necessary that

```

```

2302 custom sections be used.
2303 .Pp
2304 Section names should be unique so that they may be keyed by
2305 .Sx \&Sx .
2306 Although this macro is parsed, it should not consist of child node or it
2307 may not be linked with
2308 .Sx \&Sx .
2309 .Pp
2310 See also
2311 .Sx \&Pp ,
2312 .Sx \&Ss ,
2313 and
2314 .Sx \&Sx .
2315 .Ss \&Sm
2316 Switches the spacing mode for output generated from macros.
2317 Its syntax is as follows:
2318 .Pp
2319 .D1 Pf \. Sx \&Sm Cm on | off
2320 .Pp
2321 By default, spacing is
2322 .Cm on .
2323 When switched
2324 .Cm off ,
2325 no white space is inserted between macro arguments and between the
2326 output generated from adjacent macros, but text lines
2327 still get normal spacing between words and sentences.
2328 .Ss \&So
2329 Multi-line version of
2330 .Sx \&Sq .
2331 .Ss \&Sq
2332 Encloses its arguments in
2333 .Sq typewriter
2334 single-quotes.
2335 .Pp
2336 See also
2337 .Sx \&Dq ,
2338 .Sx \&Qq ,
2339 and
2340 .Sx \&So .
2341 .Ss \&Ss
2342 Begin a new subsection.
2343 Unlike with
2344 .Sx \&Sh ,
2345 there is no convention for the naming of subsections.
2346 Except
2347 .Em DESCRIPTION ,
2348 the conventional sections described in
2349 .Sx MANUAL STRUCTURE
2350 rarely have subsections.
2351 .Pp
2352 Sub-section names should be unique so that they may be keyed by
2353 .Sx \&Sx .
2354 Although this macro is parsed, it should not consist of child node or it
2355 may not be linked with
2356 .Sx \&Sx .
2357 .Pp
2358 See also
2359 .Sx \&Pp ,
2360 .Sx \&Sh ,
2361 and
2362 .Sx \&Sx .
2363 .Ss \&St
2364 Replace an abbreviation for a standard with the full form.
2365 The following standards are recognised:
2366 .Pp
2367 .Bl -tag -width "-p1003.1g-2000x" -compact

```



```

2368 .It \-p1003.1-88
2369 .St -p1003.1-88
2370 .It \-p1003.1-90
2371 .St -p1003.1-90
2372 .It \-p1003.1-96
2373 .St -p1003.1-96
2374 .It \-p1003.1-2001
2375 .St -p1003.1-2001
2376 .It \-p1003.1-2004
2377 .St -p1003.1-2004
2378 .It \-p1003.1-2008
2379 .St -p1003.1-2008
2380 .It \-p1003.1
2381 .St -p1003.1
2382 .It \-p1003.1b
2383 .St -p1003.1b
2384 .It \-p1003.1b-93
2385 .St -p1003.1b-93
2386 .It \-p1003.1c-95
2387 .St -p1003.1c-95
2388 .It \-p1003.1g-2000
2389 .St -p1003.1g-2000
2390 .It \-p1003.1i-95
2391 .St -p1003.1i-95
2392 .It \-p1003.2-92
2393 .St -p1003.2-92
2394 .It \-p1003.2a-92
2395 .St -p1003.2a-92
2396 .It \-p1387.2-95
2397 .St -p1387.2-95
2398 .It \-p1003.2
2399 .St -p1003.2
2400 .It \-p1387.2
2401 .St -p1387.2
2402 .It \-isoC
2403 .St -isoC
2404 .It \-isoC-90
2405 .St -isoC-90
2406 .It \-isoC-amd1
2407 .St -isoC-amd1
2408 .It \-isoC-tcor1
2409 .St -isoC-tcor1
2410 .It \-isoC-tcor2
2411 .St -isoC-tcor2
2412 .It \-isoC-99
2413 .St -isoC-99
2414 .It \-isoC-2011
2415 .St -isoC-2011
2416 .It \-iso9945-1-90
2417 .St -iso9945-1-90
2418 .It \-iso9945-1-96
2419 .St -iso9945-1-96
2420 .It \-iso9945-2-93
2421 .St -iso9945-2-93
2422 .It \-ansiC
2423 .St -ansiC
2424 .It \-ansiC-89
2425 .St -ansiC-89
2426 .It \-ansiC-99
2427 .St -ansiC-99
2428 .It \-ieee754
2429 .St -ieee754
2430 .It \-iso8802-3
2431 .St -iso8802-3
2432 .It \-iso8601
2433 .St -iso8601

```

```

2434 .It \-ieee1275-94
2435 .St -ieee1275-94
2436 .It \-xpg3
2437 .St -xpg3
2438 .It \-xpg4
2439 .St -xpg4
2440 .It \-xpg4.2
2441 .St -xpg4.2
2442 .It \-xpg4.3
2443 .St -xpg4.3
2444 .It \-xbd5
2445 .St -xbd5
2446 .It \-xcu5
2447 .St -xcu5
2448 .It \-xsh5
2449 .St -xsh5
2450 .It \-xns5
2451 .St -xns5
2452 .It \-xns5.2
2453 .St -xns5.2
2454 .It \-xns5.2d2.0
2455 .St -xns5.2d2.0
2456 .It \-xcurses4.2
2457 .St -xcurses4.2
2458 .It \-susv2
2459 .St -susv2
2460 .It \-susv3
2461 .St -susv3
2462 .It \-svid4
2463 .St -svid4
2464 .El
2465 .Ss \&Sx
2466 Reference a section or subsection in the same manual page.
2467 The referenced section or subsection name must be identical to the
2468 enclosed argument, including whitespace.
2469 .Pp
2470 Examples:
2471 .Dl \&Sx MANUAL STRUCTURE
2472 .Pp
2473 See also
2474 .Sx \&Sh
2475 and
2476 .Sx \&Ss .
2477 .Ss \&Sy
2478 Format enclosed arguments in symbolic
2479 .Pq Dq boldface .
2480 Note that this is a presentation term and should not be used for
2481 stylistically decorating technical terms.
2482 .Pp
2483 See also
2484 .Sx \&Bf ,
2485 .Sx \&Em ,
2486 .Sx \&Li ,
2487 and
2488 .Sx \&No .
2489 .Ss \&Ta
2490 Table cell separator in
2491 .Sx \&Bl Fl column
2492 lists; can only be used below
2493 .Sx \&It .
2494 .Ss \&Tn
2495 Format a tradename.
2496 .Pp
2497 Since this macro is often implemented to use a small caps font,
2498 it has historically been used for acronyms (like ASCII) as well.
2499 Such usage is not recommended because it would use the same macro

```

2500 sometimes for semantical annotation, sometimes for physical formatting.
 2501 .Pp
 2502 Examples:
 2503 .Dl \&.Tn IBM
 2504 .Ss \&Ud
 2505 Prints out
 2506 .Dq currently under development.
 2507 .Ss \&Ux
 2508 Format the UNIX name.
 2509 Accepts no argument.
 2510 .Pp
 2511 Examples:
 2512 .Dl \&.Ux
 2513 .Pp
 2514 See also
 2515 .Sx \&At ,
 2516 .Sx \&Bsx ,
 2517 .Sx \&Bx ,
 2518 .Sx \&Dx ,
 2519 .Sx \&Fx ,
 2520 .Sx \&Nx ,
 2521 and
 2522 .Sx \&Ox .
 2523 .Ss \&Va
 2524 A variable name.
 2525 .Pp
 2526 Examples:
 2527 .Dl \&.Va foo
 2528 .Dl \&.Va const char *bar ;
 2529 .Ss \&Vt
 2530 A variable type.
 2531 This is also used for indicating global variables in the
 2532 .Em SYNOPSIS
 2533 section, in which case a variable name is also specified.
 2534 Note that it accepts
 2535 .Sx Block partial-implicit
 2536 syntax when invoked as the first macro on an input line in the
 2537 .Em SYNOPSIS
 2538 section, else it accepts ordinary
 2539 .Sx In-line
 2540 syntax.
 2541 In the former case, this macro starts a new output line,
 2542 and a blank line is inserted in front if there is a preceding
 2543 function definition or include directive.
 2544 .Pp
 2545 Note that this should not be confused with
 2546 .Sx \&Ft ,
 2547 which is used for function return types.
 2548 .Pp
 2549 Examples:
 2550 .Dl \&.Vt unsigned char
 2551 .Dl \&.Vt extern const char * const sys_signame[] \&;
 2552 .Pp
 2553 See also
 2554 .Sx MANUAL STRUCTURE
 2555 and
 2556 .Sx \&Va .
 2557 .Ss \&Xc
 2558 Close a scope opened by
 2559 .Sx \&Xo .
 2560 .Ss \&Xo
 2561 Extend the header of an
 2562 .Sx \&It
 2563 macro or the body of a partial-implicit block macro
 2564 beyond the end of the input line.
 2565 This macro originally existed to work around the 9-argument limit

2566 of historic
 2567 .Xr roff 5 .
 2568 .Ss \&Xr
 2569 Link to another manual
 2570 .Pq Qq cross-reference .
 2571 Its syntax is as follows:
 2572 .Pp
 2573 .D1 Pf \. Sx \&Xr Ar name section
 2574 .Pp
 2575 The
 2576 .Ar name
 2577 and
 2578 .Ar section
 2579 are the name and section of the linked manual.
 2580 If
 2581 .Ar section
 2582 is followed by non-punctuation, an
 2583 .Sx \&Ns
 2584 is inserted into the token stream.
 2585 This behaviour is for compatibility with
 2586 GNU troff.
 2587 .Pp
 2588 Examples:
 2589 .Dl \&.Xr mandoc 1
 2590 .Dl \&.Xr mandoc 1 \&;
 2591 .Dl \&.Xr mandoc 1 \&Ns s behaviour
 2592 .Ss \&br
 2593 Emits a line-break.
 2594 This macro should not be used; it is implemented for compatibility with
 2595 historical manuals.
 2596 .Pp
 2597 Consider using
 2598 .Sx \&Pp
 2599 in the event of natural paragraph breaks.
 2600 .Ss \&sp
 2601 Emits vertical space.
 2602 This macro should not be used; it is implemented for compatibility with
 2603 historical manuals.
 2604 Its syntax is as follows:
 2605 .Pp
 2606 .D1 Pf \. Sx \&sp Op Ar height
 2607 .Pp
 2608 The
 2609 .Ar height
 2610 argument must be formatted as described in
 2611 .Sx Scaling Widths .
 2612 If unspecified,
 2613 .Sx \&sp
 2614 asserts a single vertical space.
 2615 .Sh MACRO SYNTAX
 2616 The syntax of a macro depends on its classification.
 2617 In this section,
 2618 .Sq \-arg
 2619 refers to macro arguments, which may be followed by zero or more
 2620 .Sq parm
 2621 parameters;
 2622 .Sq \&Yo
 2623 opens the scope of a macro; and if specified,
 2624 .Sq \&Yc
 2625 closes it out.
 2626 .Pp
 2627 The
 2628 .Em Callable
 2629 column indicates that the macro may also be called by passing its name
 2630 as an argument to another macro.
 2631 For example,

```

2632 .Sq \&.Op \&Fl 0 \&Ar file
2633 produces
2634 .Sq Op Fl 0 Ar file .
2635 To prevent a macro call and render the macro name literally,
2636 escape it by prepending a zero-width space,
2637 .Sq \e& .
2638 For example,
2639 .Sq \&.Op \e&Fl 0
2640 produces
2641 .Sq Op \&Fl 0 .
2642 If a macro is not callable but its name appears as an argument
2643 to another macro, it is interpreted as opaque text.
2644 For example,
2645 .Sq \&.Fl \&Sh
2646 produces
2647 .Sq Fl \&Sh .
2648 .Pp
2649 The
2650 .Em Parsed
2651 column indicates whether the macro may call other macros by receiving
2652 their names as arguments.
2653 If a macro is not parsed but the name of another macro appears
2654 as an argument, it is interpreted as opaque text.
2655 .Pp
2656 The
2657 .Em Scope
2658 column, if applicable, describes closure rules.
2659 .Ss Block full-explicit
2660 Multi-line scope closed by an explicit closing macro.
2661 All macros contains bodies; only
2662 .Sx \&Bf
2663 and
2664 .Pq optionally
2665 .Sx \&Bl
2666 contain a head.
2667 .Bd -literal -offset indent
2668 \&.Yo \&(\&B\&-arg \&(\&Bparm...\&(\&B\&(\&Bhead...\&(\&B
2669 \&(\&Bbody...\&(\&B
2670 \&.Yc
2671 .Ed
2672 .Bl -column "MacroX" "CallableX" "ParsedX" "closed by XXX" -offset indent
2673 .It Em Macro Ta Em Callable Ta Em Parsed Ta Em Scope
2674 .It Sx \&Bd Ta \&No Ta \&No Ta closed by Sx \&Ed
2675 .It Sx \&Bf Ta \&No Ta \&No Ta closed by Sx \&Ef
2676 .It Sx \&Bk Ta \&No Ta \&No Ta closed by Sx \&Ek
2677 .It Sx \&Bl Ta \&No Ta \&No Ta closed by Sx \&El
2678 .It Sx \&Ed Ta \&No Ta \&No Ta opened by Sx \&Bd
2679 .It Sx \&Ef Ta \&No Ta \&No Ta opened by Sx \&Bf
2680 .It Sx \&Ek Ta \&No Ta \&No Ta opened by Sx \&Bk
2681 .It Sx \&El Ta \&No Ta \&No Ta opened by Sx \&Bl
2682 .El
2683 .Ss Block full-implicit
2684 Multi-line scope closed by end-of-file or implicitly by another macro.
2685 All macros have bodies; some
2686 .Po
2687 .Sx \&It Fl bullet ,
2688 .Fl hyphen ,
2689 .Fl dash ,
2690 .Fl enum ,
2691 .Fl item
2692 .Pc
2693 don't have heads; only one
2694 .Po
2695 .Sx \&It
2696 in
2697 .Sx \&Bl Fl column

```

```

2698 .Pc
2699 has multiple heads.
2700 .Bd -literal -offset indent
2701 \&.Yo \&(\&B\&-arg \&(\&Bparm...\&(\&B\&(\&Bhead...\&(\&B\&(\&B
2702 \&(\&Bbody...\&(\&B
2703 .Ed
2704 .Bl -column "MacroX" "CallableX" "ParsedX" "closed by XXXXXXXXXX" -offset inden
2705 .It Em Macro Ta Em Callable Ta Em Parsed Ta Em Scope
2706 .It Sx \&It Ta \&No Ta Yes Ta closed by Sx \&It , Sx \&El
2707 .It Sx \&Nd Ta \&No Ta \&No Ta closed by Sx \&Sh
2708 .It Sx \&Nm Ta \&No Ta Yes Ta closed by Sx \&Nm , Sx \&Sh , Sx \&Ss
2709 .It Sx \&Sh Ta \&No Ta Yes Ta closed by Sx \&Sh
2710 .It Sx \&Ss Ta \&No Ta Yes Ta closed by Sx \&Sh , Sx \&Ss
2711 .El
2712 .Pp
2713 Note that the
2714 .Sx \&Nm
2715 macro is a
2716 .Sx Block full-implicit
2717 macro only when invoked as the first macro
2718 in a
2719 .Em SYNOPSIS
2720 section line, else it is
2721 .Sx In-line .
2722 .Ss Block partial-explicit
2723 Like block full-explicit, but also with single-line scope.
2724 Each has at least a body and, in limited circumstances, a head
2725 .Po
2726 .Sx \&Fo ,
2727 .Sx \&Eo
2728 .Pc
2729 and/or tail
2730 .Pq Sx \&Ec .
2731 .Bd -literal -offset indent
2732 \&.Yo \&(\&B\&-arg \&(\&Bparm...\&(\&B\&(\&Bhead...\&(\&B
2733 \&(\&Bbody...\&(\&B
2734 \&.Yc \&(\&Btail...\&(\&B

2736 \&.Yo \&(\&B\&-arg \&(\&Bparm...\&(\&B\&(\&Bhead...\&(\&B \
2737 \&(\&Bbody...\&(\&B \&.Yc \&(\&Btail...\&(\&B
2738 .Ed
2739 .Bl -column "MacroX" "CallableX" "ParsedX" "closed by XXXX" -offset indent
2740 .It Em Macro Ta Em Callable Ta Em Parsed Ta Em Scope
2741 .It Sx \&Ac Ta Yes Ta Yes Ta opened by Sx \&Ao
2742 .It Sx \&Ao Ta Yes Ta Yes Ta closed by Sx \&Ac
2743 .It Sx \&Bc Ta Yes Ta Yes Ta closed by Sx \&Bo
2744 .It Sx \&Bo Ta Yes Ta Yes Ta opened by Sx \&Bc
2745 .It Sx \&Brc Ta Yes Ta Yes Ta opened by Sx \&Bro
2746 .It Sx \&Bro Ta Yes Ta Yes Ta closed by Sx \&Brc
2747 .It Sx \&Dc Ta Yes Ta Yes Ta opened by Sx \&Do
2748 .It Sx \&Do Ta Yes Ta Yes Ta closed by Sx \&Dc
2749 .It Sx \&Ec Ta Yes Ta Yes Ta opened by Sx \&Eo
2750 .It Sx \&Eo Ta Yes Ta Yes Ta closed by Sx \&Ec
2751 .It Sx \&Fc Ta Yes Ta Yes Ta opened by Sx \&Fo
2752 .It Sx \&Fo Ta \&No Ta \&No Ta closed by Sx \&Fc
2753 .It Sx \&Oc Ta Yes Ta Yes Ta closed by Sx \&Oo
2754 .It Sx \&Oo Ta Yes Ta Yes Ta opened by Sx \&Oc
2755 .It Sx \&Pc Ta Yes Ta Yes Ta closed by Sx \&Po
2756 .It Sx \&Po Ta Yes Ta Yes Ta opened by Sx \&Pc
2757 .It Sx \&Qc Ta Yes Ta Yes Ta opened by Sx \&Oo
2758 .It Sx \&Qo Ta Yes Ta Yes Ta closed by Sx \&Oc
2759 .It Sx \&Re Ta \&No Ta \&No Ta opened by Sx \&Rs
2760 .It Sx \&Rs Ta \&No Ta \&No Ta closed by Sx \&Re
2761 .It Sx \&Sc Ta Yes Ta Yes Ta opened by Sx \&So
2762 .It Sx \&So Ta Yes Ta Yes Ta closed by Sx \&Sc
2763 .It Sx \&Xc Ta Yes Ta Yes Ta opened by Sx \&Xo

```

```

2764 .It Sx \&Xo Ta Yes Ta Yes Ta closed by Sx \&Xc
2765 .El
2766 .Ss Block partial-implicit
2767 Like block full-implicit, but with single-line scope closed by the
2768 end of the line.
2769 .Bd -literal -offset indent
2770 \&.Yo \(lB\arg \(lBval...\(rB\(rB \(lBbody...\(rB \(lBres...\(rB
2771 .Ed
2772 .Bl -column "MacroX" "CallableX" "ParsedX" -offset indent
2773 .It Em Macro Ta Em Callable Ta Em Parsed
2774 .It Sx \&Aq Ta Yes Ta Yes
2775 .It Sx \&Bq Ta Yes Ta Yes
2776 .It Sx \&Brq Ta Yes Ta Yes
2777 .It Sx \&D1 Ta \&No Ta \&Yes
2778 .It Sx \&Dl Ta \&No Ta Yes
2779 .It Sx \&Dq Ta Yes Ta Yes
2780 .It Sx \&Op Ta Yes Ta Yes
2781 .It Sx \&Pq Ta Yes Ta Yes
2782 .It Sx \&Ql Ta Yes Ta Yes
2783 .It Sx \&Qq Ta Yes Ta Yes
2784 .It Sx \&Sq Ta Yes Ta Yes
2785 .It Sx \&Vt Ta Yes Ta Yes
2786 .El
2787 .Pp
2788 Note that the
2789 .Sx \&Vt
2790 macro is a
2791 .Sx Block partial-implicit
2792 only when invoked as the first macro
2793 in a
2794 .Em SYNOPSIS
2795 section line, else it is
2796 .Sx In-line .
2797 .Ss Special block macro
2798 The
2799 .Sx \&Ta
2800 macro can only be used below
2801 .Sx \&It
2802 in
2803 .Sx \&Bl Fl column
2804 lists.
2805 It delimits blocks representing table cells;
2806 these blocks have bodies, but no heads.
2807 .Bl -column "MacroX" "CallableX" "ParsedX" "closed by XXXX" -offset indent
2808 .It Em Macro Ta Em Callable Ta Em Parsed Ta Em Scope
2809 .It Sx \&Ta Ta Yes Ta Yes Ta closed by Sx \&Ta , Sx \&It
2810 .El
2811 .Ss In-line
2812 Closed by the end of the line, fixed argument lengths,
2813 and/or subsequent macros.
2814 In-line macros have only text children.
2815 If a number (or inequality) of arguments is
2816 .Pq n
2817 then the macro accepts an arbitrary number of arguments.
2818 .Bd -literal -offset indent
2819 \&.Yo \(lB\arg \(lBval...\(rB\(rB \(lBargs...\(rB \(lBres...\(rB

2821 \&.Yo \(lB\arg \(lBval...\(rB\(rB \(lBargs...\(rB Yc...

2823 \&.Yo \(lB\arg \(lBval...\(rB\(rB arg0 arg1 argN
2824 .Ed
2825 .Bl -column "MacroX" "CallableX" "ParsedX" "Arguments" -offset indent
2826 .It Em Macro Ta Em Callable Ta Em Parsed Ta Em Arguments
2827 .It Sx \&%A Ta \&No Ta \&No Ta >0
2828 .It Sx \&%B Ta \&No Ta \&No Ta >0
2829 .It Sx \&%C Ta \&No Ta \&No Ta >0

```

```

2830 .It Sx \&%D Ta \&No Ta \&No Ta >0
2831 .It Sx \&%I Ta \&No Ta \&No Ta >0
2832 .It Sx \&%J Ta \&No Ta \&No Ta >0
2833 .It Sx \&%N Ta \&No Ta \&No Ta >0
2834 .It Sx \&%O Ta \&No Ta \&No Ta >0
2835 .It Sx \&%P Ta \&No Ta \&No Ta >0
2836 .It Sx \&%Q Ta \&No Ta \&No Ta >0
2837 .It Sx \&%R Ta \&No Ta \&No Ta >0
2838 .It Sx \&%T Ta \&No Ta \&No Ta >0
2839 .It Sx \&%U Ta \&No Ta \&No Ta >0
2840 .It Sx \&%V Ta \&No Ta \&No Ta >0
2841 .It Sx \&Ad Ta Yes Ta Yes Ta >0
2842 .It Sx \&An Ta Yes Ta Yes Ta >0
2843 .It Sx \&Ap Ta Yes Ta Yes Ta 0
2844 .It Sx \&Ar Ta Yes Ta Yes Ta n
2845 .It Sx \&At Ta Yes Ta Yes Ta 1
2846 .It Sx \&Bsx Ta Yes Ta Yes Ta n
2847 .It Sx \&Bt Ta \&No Ta \&No Ta 0
2848 .It Sx \&Bx Ta Yes Ta Yes Ta n
2849 .It Sx \&Cd Ta Yes Ta Yes Ta >0
2850 .It Sx \&Cm Ta Yes Ta Yes Ta >0
2851 .It Sx \&Db Ta \&No Ta \&No Ta 1
2852 .It Sx \&Dd Ta \&No Ta \&No Ta n
2853 .It Sx \&Dt Ta \&No Ta \&No Ta n
2854 .It Sx \&Dv Ta Yes Ta Yes Ta >0
2855 .It Sx \&Dx Ta Yes Ta Yes Ta n
2856 .It Sx \&Em Ta Yes Ta Yes Ta >0
2857 .It Sx \&En Ta \&No Ta \&No Ta 0
2858 .It Sx \&Er Ta Yes Ta Yes Ta >0
2859 .It Sx \&Es Ta \&No Ta \&No Ta 0
2860 .It Sx \&Ev Ta Yes Ta Yes Ta >0
2861 .It Sx \&Ex Ta \&No Ta \&No Ta n
2862 .It Sx \&Fa Ta Yes Ta Yes Ta >0
2863 .It Sx \&Fd Ta \&No Ta \&No Ta >0
2864 .It Sx \&Fl Ta Yes Ta Yes Ta n
2865 .It Sx \&Fn Ta Yes Ta Yes Ta >0
2866 .It Sx \&Fr Ta \&No Ta \&No Ta n
2867 .It Sx \&Ft Ta Yes Ta Yes Ta >0
2868 .It Sx \&Fx Ta Yes Ta Yes Ta n
2869 .It Sx \&Hf Ta \&No Ta \&No Ta n
2870 .It Sx \&Ic Ta Yes Ta Yes Ta >0
2871 .It Sx \&In Ta \&No Ta \&No Ta 1
2872 .It Sx \&Lb Ta \&No Ta \&No Ta 1
2873 .It Sx \&Li Ta Yes Ta Yes Ta >0
2874 .It Sx \&Lk Ta Yes Ta Yes Ta >0
2875 .It Sx \&Lp Ta \&No Ta \&No Ta 0
2876 .It Sx \&Ms Ta Yes Ta Yes Ta >0
2877 .It Sx \&Mt Ta Yes Ta Yes Ta >0
2878 .It Sx \&Nm Ta Yes Ta Yes Ta n
2879 .It Sx \&No Ta Yes Ta Yes Ta 0
2880 .It Sx \&Ns Ta Yes Ta Yes Ta 0
2881 .It Sx \&Nx Ta Yes Ta Yes Ta n
2882 .It Sx \&Os Ta \&No Ta \&No Ta n
2883 .It Sx \&Ot Ta \&No Ta \&No Ta n
2884 .It Sx \&Ox Ta Yes Ta Yes Ta n
2885 .It Sx \&Pa Ta Yes Ta Yes Ta n
2886 .It Sx \&Pf Ta Yes Ta Yes Ta 1
2887 .It Sx \&Pp Ta \&No Ta \&No Ta 0
2888 .It Sx \&Rv Ta \&No Ta \&No Ta n
2889 .It Sx \&Sm Ta \&No Ta \&No Ta 1
2890 .It Sx \&St Ta \&No Ta Yes Ta 1
2891 .It Sx \&Sx Ta Yes Ta Yes Ta >0
2892 .It Sx \&Sy Ta Yes Ta Yes Ta >0
2893 .It Sx \&Tn Ta Yes Ta Yes Ta >0
2894 .It Sx \&Ud Ta \&No Ta \&No Ta 0
2895 .It Sx \&UX Ta Yes Ta Yes Ta n

```

```

2896 .It Sx \&Va Ta Yes Ta Yes Ta n
2897 .It Sx \&Vt Ta Yes Ta Yes Ta >0
2898 .It Sx \&Xr Ta Yes Ta Yes Ta >0
2899 .It Sx \&br Ta \&No Ta \&No Ta 0
2900 .It Sx \&sp Ta \&No Ta \&No Ta 1
2901 .El
2902 .Ss Delimiters
2903 When a macro argument consists of one single input character
2904 considered as a delimiter, the argument gets special handling.
2905 This does not apply when delimiters appear in arguments containing
2906 more than one character.
2907 Consequently, to prevent special handling and just handle it
2908 like any other argument, a delimiter can be escaped by prepending
2909 a zero-width space
2910 .Pq Sq \e& .
2911 In text lines, delimiters never need escaping, but may be used
2912 as normal punctuation.
2913 .Pp
2914 For many macros, when the leading arguments are opening delimiters,
2915 these delimiters are put before the macro scope,
2916 and when the trailing arguments are closing delimiters,
2917 these delimiters are put after the macro scope.
2918 For example,
2919 .Pp
2920 .D1 Pf \. \&Aq "( [ word ] ) ."
2921 .Pp
2922 renders as:
2923 .Pp
2924 .D1 Aq ( [ word ] ) .
2925 .Pp
2926 Opening delimiters are:
2927 .Pp
2928 .Bl -tag -width Ds -offset indent -compact
2929 .It \&(
2930 left parenthesis
2931 .It \&[
2932 left bracket
2933 .El
2934 .Pp
2935 Closing delimiters are:
2936 .Pp
2937 .Bl -tag -width Ds -offset indent -compact
2938 .It \&.
2939 period
2940 .It \&,
2941 comma
2942 .It \&:
2943 colon
2944 .It \&;
2945 semicolon
2946 .It \&)
2947 right parenthesis
2948 .It \&]
2949 right bracket
2950 .It \&?
2951 question mark
2952 .It \&!
2953 exclamation mark
2954 .El
2955 .Pp
2956 Note that even a period preceded by a backslash
2957 .Pq Sq \e.\&
2958 gets this special handling; use
2959 .Sq \e&.
2960 to prevent that.
2961 .Pp

```

```

2962 Many in-line macros interrupt their scope when they encounter
2963 delimiters, and resume their scope when more arguments follow that
2964 are not delimiters.
2965 For example,
2966 .Pp
2967 .D1 Pf \. \&Fl "a ( b | c \e*(Ba d ) e"
2968 .Pp
2969 renders as:
2970 .Pp
2971 .D1 Fl a ( b | c \e*(Ba d ) e
2972 .Pp
2973 This applies to both opening and closing delimiters,
2974 and also to the middle delimiter:
2975 .Pp
2976 .Bl -tag -width Ds -offset indent -compact
2977 .It \&|
2978 vertical bar
2979 .El
2980 .Pp
2981 As a special case, the predefined string \e*(Ba is handled and rendered
2982 in the same way as a plain
2983 .Sq \&|
2984 character.
2985 Using this predefined string is not recommended in new manuals.
2986 .Ss Font handling
2987 In
2988 .Nm
2989 documents, usage of semantic markup is recommended in order to have
2990 proper fonts automatically selected; only when no fitting semantic markup
2991 is available, consider falling back to
2992 .Sx Physical markup
2993 macros.
2994 Whenever any
2995 .Nm
2996 macro switches the
2997 .Xr roff 5
2998 font mode, it will automatically restore the previous font when exiting
2999 its scope.
3000 Manually switching the font using the
3001 .Xr roff 5
3002 .Ql \ef
3003 font escape sequences is never required.
3004 .Sh COMPATIBILITY
3005 This section documents compatibility between mandoc and other other
3006 troff implementations, at this time limited to GNU troff
3007 .Pq Qq groff .
3008 The term
3009 .Qq historic groff
3010 refers to groff versions before 1.17,
3011 which featured a significant update of the
3012 .Pa doc.tmac
3013 file.
3014 .Pp
3015 Heirloom troff, the other significant troff implementation accepting
3016 \-mdoc, is similar to historic groff.
3017 .Pp
3018 The following problematic behaviour is found in groff:
3019 .ds hist (Historic groff only.)
3020 .Pp
3021 .Bl -dash -compact
3022 .It
3023 Display macros
3024 .Po
3025 .Sx \&Bd ,
3026 .Sx \&Dl ,
3027 and

```

```

3028 .Sx \&D1
3029 .Pc
3030 may not be nested.
3031 \*[hist]
3032 .It
3033 .Sx \&At
3034 with unknown arguments produces no output at all.
3035 \*[hist]
3036 Newer groff and mandoc print
3037 .Qq AT&T UNIX
3038 and the arguments.
3039 .It
3040 .Sx \&Bl Fl column
3041 does not recognise trailing punctuation characters when they immediately
3042 precede tabulator characters, but treats them as normal text and
3043 outputs a space before them.
3044 .It
3045 .Sx \&Bd Fl ragged compact
3046 does not start a new line.
3047 \*[hist]
3048 .It
3049 .Sx \&Dd
3050 with non-standard arguments behaves very strangely.
3051 When there are three arguments, they are printed verbatim.
3052 Any other number of arguments is replaced by the current date,
3053 but without any arguments the string
3054 .Dq Epoch
3055 is printed.
3056 .It
3057 .Sx \&Fl
3058 does not print a dash for an empty argument.
3059 \*[hist]
3060 .It
3061 .Sx \&Fn
3062 does not start a new line unless invoked as the line macro in the
3063 .Em SYNOPSIS
3064 section.
3065 \*[hist]
3066 .It
3067 .Sx \&Fo
3068 with
3069 .Pf non- Sx \&Fa
3070 children causes inconsistent spacing between arguments.
3071 In mandoc, a single space is always inserted between arguments.
3072 .It
3073 .Sx \&Ft
3074 in the
3075 .Em SYNOPSIS
3076 causes inconsistent vertical spacing, depending on whether a prior
3077 .Sx \&Fn
3078 has been invoked.
3079 See
3080 .Sx \&Ft
3081 and
3082 .Sx \&Fn
3083 for the normalised behaviour in mandoc.
3084 .It
3085 .Sx \&In
3086 ignores additional arguments and is not treated specially in the
3087 .Em SYNOPSIS .
3088 \*[hist]
3089 .It
3090 .Sx \&It
3091 sometimes requires a
3092 .Fl nested
3093 flag.

```

```

3094 \*[hist]
3095 In new groff and mandoc, any list may be nested by default and
3096 .Fl enum
3097 lists will restart the sequence only for the sub-list.
3098 .It
3099 .Sx \&Li
3100 followed by a delimiter is incorrectly used in some manuals
3101 instead of properly quoting that character, which sometimes works with
3102 historic groff.
3103 .It
3104 .Sx \&Lk
3105 only accepts a single link-name argument; the remainder is misformatted.
3106 .It
3107 .Sx \&Pa
3108 does not format its arguments when used in the FILES section under
3109 certain list types.
3110 .It
3111 .Sx \&Ta
3112 can only be called by other macros, but not at the beginning of a line.
3113 .It
3114 .Sx \&%C
3115 is not implemented.
3116 .It
3117 Historic groff only allows up to eight or nine arguments per macro input
3118 line, depending on the exact situation.
3119 Providing more arguments causes garbled output.
3120 The number of arguments on one input line is not limited with mandoc.
3121 .It
3122 Historic groff has many un-callable macros.
3123 Most of these (excluding some block-level macros) are callable
3124 in new groff and mandoc.
3125 .It
3126 .Sq \ (ba
3127 (vertical bar) is not fully supported as a delimiter.
3128 \*[hist]
3129 .It
3130 .Sq \ef
3131 .Pq font face
3132 and
3133 .Sq \ef
3134 .Pq font family face
3135 .Sx Text Decoration
3136 escapes behave irregularly when specified within line-macro scopes.
3137 .It
3138 Negative scaling units return to prior lines.
3139 Instead, mandoc truncates them to zero.
3140 .El
3141 .Pp
3142 The following features are unimplemented in mandoc:
3143 .Pp
3144 .Bl -dash -compact
3145 .It
3146 .Sx \&Bd
3147 .Fl file Ar file .
3148 .It
3149 .Sx \&Bd
3150 .Fl offset Ar center
3151 and
3152 .Fl offset Ar right .
3153 Groff does not implement centred and flush-right rendering either,
3154 but produces large indentations.
3155 .It
3156 The
3157 .Sq \eh
3158 .Pq horizontal position ,
3159 .Sq \ev

```

3160 .Pq vertical position ,
3161 .Sq \em
3162 .Pq text colour ,
3163 .Sq \eM
3164 .Pq text filling colour ,
3165 .Sq \ez
3166 .Pq zero-length character ,
3167 .Sq \ew
3168 .Pq string length ,
3169 .Sq \ek
3170 .Pq horizontal position marker ,
3171 .Sq \eo
3172 .Pq text overstrike ,
3173 and
3174 .Sq \es
3175 .Pq text size
3176 escape sequences are all discarded in mandoc.
3177 .It
3178 The
3179 .Sq \ef
3180 scaling unit is accepted by mandoc, but rendered as the default unit.
3181 .It
3182 In quoted literals, groff allows pairwise double-quotes to produce a
3183 standalone double-quote in formatted output.
3184 This is not supported by mandoc.
3185 .El
3186 .Sh SEE ALSO
3187 .Xr man 1 ,
3188 .Xr mandoc 1 ,
3189 .Xr eqn 5 ,
3190 .Xr man 5 ,
3191 .Xr mandoc_char 5 ,
3192 .Xr roff 5 ,
3193 .Xr tbl 5
3194 .Sh HISTORY
3195 The
3196 .Nm
3197 language first appeared as a troff macro package in
3198 .Bx 4.4 .
3199 It was later significantly updated by Werner Lemberg and Ruslan Ermilov
3200 in groff-1.17.
3201 The standalone implementation that is part of the
3202 .Xr mandoc 1
3203 utility written by Kristaps Dzonsons appeared in
3204 .Ox 4.6 .
3205 **in July, 2014.**
3206 .Sh AUTHORS
3207 The
3208 .Nm
3209 reference was written by
3210 .An Kristaps Dzonsons ,
3211 .Mt kristaps@bsd.lv .

new/usr/src/pkg/manifests/system-man.mf

1

```
*****
2543 Wed Jul 16 09:10:06 2014
new/usr/src/pkg/manifests/system-man.mf
feedback from Hans
*****

1 #
2 # This file and its contents are supplied under the terms of the
3 # Common Development and Distribution License ("CDDL"), version 1.0.
4 # You may only use this file in accordance with the terms of version
5 # 1.0 of the CDDL.
6 #
7 # A full copy of the text of the CDDL should have accompanied this
8 # source. A copy of the CDDL is also available via the Internet at
9 # http://www.illumos.org/license/CDDL.
10 #

12 #
13 # Copyright 2012 Nexenta Systems, Inc. All rights reserved.
14 # Copyright 2014 Garrett D'Amore <garrett@damore.org>
15 #

17 set name=pkg.fmri value=pkg:/system/man@$(PKGVERS)
18 set name=pkg.description \
19   value="utilities for display and formatting of reference manual pages"
20 set name=pkg.summary value="Reference Manual Pages Tools"
21 set name=info.classification \
22   value="org.opensolaris.category.2008:System/Text Tools"
23 set name=variant.arch value=$(ARCH)
24 dir path=usr/bin
25 dir path=usr/lib
26 dir path=usr/share
27 dir path=usr/share/man
28 dir path=usr/share/man/man1
29 dir path=usr/share/man/man1m
30 dir path=usr/share/man/man5
31 file path=usr/bin/catman mode=0555
32 file path=usr/bin/man mode=0555
33 file path=usr/bin/mandoc mode=0555
34 file path=usr/lib/makewhatis mode=0555
35 file path=usr/lib/mandoc_preconv mode=0555
36 file path=usr/share/man/man1/apropos.1
37 file path=usr/share/man/man1/catman.1
38 file path=usr/share/man/man1/man.1
39 file path=usr/share/man/man1/mandoc.1
40 file path=usr/share/man/man1m/catman.1m
41 file path=usr/share/man/man5/eqn.5
42 file path=usr/share/man/man5/man.5
43 file path=usr/share/man/man5/mandoc_char.5
44 file path=usr/share/man/man5/mandoc_roff.5
45 file path=usr/share/man/man5/mdoc.5
46 file path=usr/share/man/man5/tbl.5
47 hardlink path=usr/bin/apropos target=../usr/bin/man
48 hardlink path=usr/bin/whatis target=../usr/bin/man
49 hardlink path=usr/lib/makewhatis target=../usr/bin/man
50 license lic_CDDL license=lic_CDDL
51 license usr/src/cmd/man/THIRDPARTYLICENSE \
52   license=usr/src/cmd/man/THIRDPARTYLICENSE
53 license usr/src/cmd/mandoc/THIRDPARTYLICENSE \
54   license=usr/src/cmd/mandoc/THIRDPARTYLICENSE
55 link path=usr/man target=../share/man
56 link path=usr/share/man/man1/whatis.1 target=apropos.1
57 # arguably we also need lp, for man -t support, but really we don't
58 # want to make this mandatory, so we don't express the dependency here.
59 # gzcat/bzcat are used for displaying compressed manpages. However,
60 # as we don't format such pages this way by default, let's leave the
61 # dependency out.
```

new/usr/src/pkg/manifests/system-man.mf

2

```
60 #depend fmri=compress/bzip2 type=require
61 #depend fmri=compress/gzip type=require
62 # less is the default (per user environment) pager. We really should just
63 # import this into illumos-gate.
64 depend fmri=text/less type=require
```