****
**   17192 Tue Jan 15 10:36:52 2019**
new/usr/src/common/cmdparse/cmdparse.c
10097 indenting fixes in usr/src/{lib,common}
****
```
   1 /*
   2  * CDDL HEADER START
   3  *
   4  * The contents of this file are subject to the terms of the
   5  * Common Development and Distribution License (the "License").
   6  * You may not use this file except in compliance with the License.
   7  *
   8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
   9  * or http://www.opensolaris.org/os/licensing.
  10  * See the License for the specific language governing permissions
  11  * and limitations under the License.
  12  *
  13  * When distributing Covered Code, include this CDDL HEADER in each
  14  * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
  15  * If applicable, add the following below this CDDL HEADER, with the
  16  * fields enclosed by brackets "[]" replaced with your own identifying
  17  * information: Portions Copyright [yyyy] [name of copyright owner]
  18  *
  19  * CDDL HEADER END
  20  */
  21 /*
  22  * Copyright 2009 Sun Microsystems, Inc.  All rights reserved.
  23  * Use is subject to license terms.
  24  */
```
```
  26 /*
  27  * Copyright (c) 2018, Joyent, Inc.
  28  */
```
```
  30 #include <stdlib.h>
  31 #include <stdio.h>
  32 #include <sys/types.h>
  33 #include <unistd.h>
  34 #include <libintl.h>
  35 #include <errno.h>
  36 #include <string.h>
  37 #include <assert.h>
  38 #include <getopt.h>
  39 #include <cmdparse.h>
```
```
  42 /* Usage types */
  43 #define GENERAL_USAGE   1
  44 #define DETAIL_USAGE    2
```
```
  46 /* printable ascii character set len */
  47 #define MAXOPTIONS      (uint_t)('~' - '!' + 1)
```
```
  49 /*
  50  * MAXOPTIONSTRING is the max length of the options string used in getopt and
  51  * will be the printable character set + ':' for each character,
  52  * providing for options with arguments. e.g. "t:Cs:hglr:"
  53  */
  54 #define MAXOPTIONSTRING         MAXOPTIONS * 2
```
```
  56 /* standard command options table to support -?, -V */
  57 struct option standardCmdOptions[] = {
  58         {"help", no_argument, NULL, '?'},
  59         {"version", no_argument, NULL, 'V'},
  60         {NULL, 0, NULL, 0}
  61 };
```
_____unchanged_portion_omitted_

```
**********************************************************
   13286 Tue Jan 15 10:36:53 2019
new/usr/src/common/devid/devid.c
10097 indenting fixes in usr/src/{lib,common}
**********************************************************
    1 /*
    2  * CDDL HEADER START
    3  *
    4  * The contents of this file are subject to the terms of the
    5  * Common Development and Distribution License, Version 1.0 only
    6  * (the "License").  You may not use this file except in compliance
    7  * with the License.
    8  *
    9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
   10  * or http://www.opensolaris.org/os/licensing.
   11  * See the License for the specific language governing permissions
   12  * and limitations under the License.
   13  *
   14  * When distributing Covered Code, include this CDDL HEADER in each
   15  * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
   16  * If applicable, add the following below this CDDL HEADER, with the
   17  * fields enclosed by brackets "[]" replaced with your own identifying
   18  * information: Portions Copyright [yyyy] [name of copyright owner]
   19  *
   20  * CDDL HEADER END
   21  */
   22 /*
   23  * Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
   24  * Use is subject to license terms.
   25  */
   26 #pragma ident   "%Z%%M% %I%     %E% SMI"

   27 /*
   28  * Copyright (c) 2018, Joyent, Inc.
   29  */

   31 #include <sys/types.h>
   32 #include <sys/stropts.h>
   33 #include <sys/debug.h>
   34 #include <sys/isa_defs.h>
   35 #include <sys/dditypes.h>
   36 #include <sys/ddi_impldefs.h>
   37 #include "devid_impl.h"

   39 static int devid_str_decode_id(char *devidstr, ddi_devid_t *devidp,
   40     char **minor_namep, impl_devid_t *id);


   43 /*
   44  * Validate device id.
   45  */
   46 int
   47 #ifdef  _KERNEL
   48 ddi_devid_valid(ddi_devid_t devid)
   49 #else   /* !_KERNEL */
   50 devid_valid(ddi_devid_t devid)
   51 #endif  /* _KERNEL */
   52 {
   53         impl_devid_t    *id = (impl_devid_t *)devid;
   54         ushort_t        type;

   56         DEVID_ASSERT(devid != NULL);

   58         if (id->did_magic_hi != DEVID_MAGIC_MSB)
   59                 return (DEVID_RET_INVALID);
```

```
   61         if (id->did_magic_lo != DEVID_MAGIC_LSB)
   62                 return (DEVID_RET_INVALID);

   64         if (id->did_rev_hi != DEVID_REV_MSB)
   65                 return (DEVID_RET_INVALID);

   67         if (id->did_rev_lo != DEVID_REV_LSB)
   68                 return (DEVID_RET_INVALID);

   70         type = DEVID_GETTYPE(id);
   71         if ((type == DEVID_NONE) || (type > DEVID_MAXTYPE))
   72                 return (DEVID_RET_INVALID);

   74         return (DEVID_RET_VALID);
   75 }
_____unchanged_portion_omitted_
```

```
**********************************************************
   19132 Tue Jan 15 10:36:53 2019
new/usr/src/common/mc/mc-amd/mcamd_rowcol.c
10097 indenting fixes in usr/src/{lib,common}
**********************************************************
   1 /*
   2  * CDDL HEADER START
   3  *
   4  * The contents of this file are subject to the terms of the
   5  * Common Development and Distribution License (the "License").
   6  * You may not use this file except in compliance with the License.
   7  *
   8  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
   9  * or http://www.opensolaris.org/os/licensing.
  10  * See the License for the specific language governing permissions
  11  * and limitations under the License.
  12  *
  13  * When distributing Covered Code, include this CDDL HEADER in each
  14  * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
  15  * If applicable, add the following below this CDDL HEADER, with the
  16  * fields enclosed by brackets "[]" replaced with your own identifying
  17  * information: Portions Copyright [yyyy] [name of copyright owner]
  18  *
  19  * CDDL HEADER END
  20  *
  21  * Copyright 2007 Sun Microsystems, Inc.  All rights reserved.
  22  * Use is subject to license terms.
  23  */

  25 /*
  26  * Copyright (c) 2018, Joyent, Inc.
  27  */
  25 #pragma ident   "%Z%%M% %I%     %E% SMI"

  29 #include <mcamd_api.h>
  30 #include <mcamd_err.h>
  31 #include <mcamd_rowcol_impl.h>

  33 /*
  34  * Convenience structures to stash MC and CS properties in.
  35  */
  36 struct mcprops {
  37         mcamd_prop_t num;               /* corresponding chip number */
  38         mcamd_prop_t rev;               /* revision */
  39         mcamd_prop_t width;             /* access width */
  40         mcamd_prop_t base;              /* MC base address */
  41         mcamd_prop_t lim;               /* MC limit address */
  42         mcamd_prop_t csbnkmap_reg;      /* chip-select bank map */
  43         mcamd_prop_t intlven;           /* Node-intlv mask */
  44         mcamd_prop_t intlvsel;          /* Node-intlv selection for this node */
  45         mcamd_prop_t csintlvfctr;       /* cs intlv factor on this node */
  46         mcamd_prop_t bnkswzl;           /* bank-swizzle mode */
  47         mcamd_prop_t sparecs;           /* spare cs#, if any */
  48         mcamd_prop_t badcs;             /* substituted cs#, if any */
  49 };
_____unchanged_portion_omitted_

 467 /*
 468  * Given an MC, DIMM and offset (dimm rank, row, col, internal bank) we
 469  * find the corresponding chip-select for the rank and then reconstruct
 470  * a system address.  In the absence of serial number support it is possible
 471  * that we may be asked to perform this operation on a dimm which has been
 472  * swapped, perhaps even for a dimm of different size and number of ranks.
 473  * This may happen if fmadm repair has not been used.  There are some
 474  * unused bits in the offset and we could guard against this a little
 475  * by recording in those bit some of the physical characteristic of the
```

```
 476  * original DIMM such as size, number of ranks etc.
 477  */
 478 int
 479 mc_offset_to_pa(struct mcamd_hdl *hdl, mcamd_node_t *mc, mcamd_node_t *dimm,
 480     uint64_t offset, uint64_t *pap)
 481 {
 482         mcamd_node_t *cs;
 483         mcamd_dimm_offset_un_t off_un;
 484         uint32_t rank, rowaddr, bankaddr, coladdr;
 485         uint64_t iaddr = 0;
 486         const struct rct_bnkaddrmode *bamp;
 487         const struct rct_rcbmap *rcbmp;
 488         const struct rct_bnkswzlinfo *swzlp = NULL;
 489         struct rct_csintlv csi;
 490         struct mcprops mcp;
 491         struct csprops csp;
 492         uint64_t csmode;
 493         int maskhi_hi, maskhi_lo, masklo_hi, masklo_lo;

 495         off_un.do_offset = offset;
 496         rank = off_un.do_rank;
 497         bankaddr = off_un.do_bank;
 498         rowaddr = off_un.do_row;
 499         coladdr = off_un.do_col;

 501         mcamd_dprintf(hdl, MCAMD_DBG_FLOW, "mc_offset_to_pa: offset 0x%llx "
 502             "-> rank %d bank %d row 0x%x col 0x%x\n", offset,
 503             rank, bankaddr, rowaddr, coladdr);

 505         if (getmcprops(hdl, mc, "mc_offset_to_pa", &mcp) < 0)
 506                 return (-1);    /* errno already set */

 508         maskhi_hi = MC_CSMASKHI_HIBIT(mcp.rev);
 509         maskhi_lo = MC_CSMASKHI_LOBIT(mcp.rev);
 510         masklo_hi = MC_CSMASKLO_HIBIT(mcp.rev);
 511         masklo_lo = MC_CSMASKLO_LOBIT(mcp.rev);

 513         /*
 514          * Find the chip-select on this dimm using the given rank.
 515          */
 516         for (cs = mcamd_cs_next(hdl, dimm, NULL); cs != NULL;
 517             cs = mcamd_cs_next(hdl, dimm, cs)) {
 518                 if (getcsprops(hdl, cs, "mc_offset_to_pa", &csp) < 0)
 519                         return (-1);    /* errno already set */

 521                 if (csp.dimmrank == rank)
 522                         break;
 523         }

 525         if (cs == NULL) {
 526                 mcamd_dprintf(hdl, MCAMD_DBG_FLOW, "mc_offset_to_pa: Current "
 527                     "dimm in this slot does not have a cs using rank %d\n",
 528                     rank);
 529                 return (mcamd_set_errno(hdl, EMCAMD_NOADDR));
 530         }

 532         /*
 533          * If the cs# has been substituted by the online spare then the
 534          * given unum is not actually contributing to the system address
 535          * map since all accesses to it are redirected.
 536          *
 537          * If the cs# failed BIOS test it is not in the address map.
 538          *
 539          * If the cs# is the online spare cs# then it is contributing to
 540          * the system address map only if swapped in, and the csbase etc
 541          * parameters to use must be those of the bad cs#.
```

```
542                 */
543          if (mcp.badcs != MC_INVALNUM && csp.num == mcp.badcs) {
544                 return (mcamd_set_errno(hdl, EMCAMD_NOADDR));
545          } else if (csp.testfail) {
546                 return (mcamd_set_errno(hdl, EMCAMD_NOADDR));
547          } else if (mcp.sparecs != MC_INVALNUM && csp.num == mcp.sparecs &&
548             mcp.badcs != MC_INVALNUM) {
549                 /*
550                  * Iterate over all cs# of this memory controller to find
551                  * the bad one - the bad cs# need not be on the same dimm
552                  * as the spare.
553                  */
554                 for (cs = mcamd_cs_next(hdl, mc, NULL); cs != NULL;
555                     cs = mcamd_cs_next(hdl, mc, cs)) {
556                         mcamd_prop_t csnum;

558                         if (!mcamd_get_numprop(hdl, cs, MCAMD_PROP_NUM,
559                             &csnum)) {
560                                 mcamd_dprintf(hdl, MCAMD_DBG_ERR,
561                                     "mcamd_offset_to_pa: csnum lookup failed "
562                                     "while looking for bad cs#");
563                                 return (mcamd_set_errno(hdl,
564                                     EMCAMD_TREEINVALID));
565                         }
566                         if (csnum == mcp.badcs)
567                                 break;
568                 }

570                 if (cs == NULL) {
571                         mcamd_dprintf(hdl, MCAMD_DBG_ERR, "mcamd_offset_to_pa: "
572                             "failed to find cs for bad cs#%d\n", mcp.badcs);
573                         return (mcamd_set_errno(hdl, EMCAMD_TREEINVALID));
571                         return (mcamd_set_errno(hdl,
572                             EMCAMD_TREEINVALID));
574                 }

576                 /* found bad cs - reread properties from it instead of spare */
577                 if (getcsprops(hdl, cs, "mc_offset_to_pa", &csp) < 0)
578                         return (-1);    /* errno already set */
579          }

581          csmode = MC_CS_MODE(mcp.csbnkmap_reg, csp.num);

583          if (gettbls(hdl, csmode, &mcp, &bamp, &rcbmp,
584             mcp.bnkswzl ? &swzlp : NULL, &csi,
585             "mc_offset_to_pa") < 0)
586                 return (-1);    /* errno already set */

588          /*
589           * If there are umaskable DRAM InputAddr bits the add those bits
590           * to iaddr from the cs base address.
591           */
592          if (MC_CSMASK_UNMASKABLE(mcp.rev) != 0) {
593                 iaddr |= iaddr_add(hdl, iaddr,
594                     BITS(csp.base, maskhi_hi + MC_CSMASK_UNMASKABLE(mcp.rev),
595                     maskhi_hi + 1), "unmaskable cs basehi bits");
596          }

598          /*
599           * basehi bits not meing masked pass straight through to the
600           * iaddr.
601           */
602          iaddr |= iaddr_add(hdl, iaddr,
603             BITS(csp.base, maskhi_hi, maskhi_lo) &
604             ~BITS(csp.mask, maskhi_hi, maskhi_lo),
605             "cs basehi bits not being masked");
```

```
607          /*
608           * if cs interleaving is active then baselo address bit are being
609           * masked - pass the rest through.
610           */
611          if (mcp.csintlvfctr > 1) {
612                 iaddr |= iaddr_add(hdl, iaddr,
613                     BITS(csp.base, masklo_hi, masklo_lo) &
614                     ~BITS(csp.mask, masklo_hi, masklo_lo),
615                     "cs baselo bits not being masked");
616          }

618          /*
619           * Reconstruct iaddr bits from known row address
620           */
621          iaddr |= iaddr_add(hdl, iaddr,
622             row_to_iaddr(hdl, bamp, rcbmp, &csi, rowaddr),
623             "add iaddr bits from row");

625          /*
626           * Reconstruct iaddr bits from known column address
627           */
628          iaddr |= iaddr_add(hdl, iaddr,
629             col_to_iaddr(hdl, bamp, rcbmp, coladdr),
630             "add iaddr bits from col");

632          /*
633           * Reconstruct iaddr bits from known internal banksel address
634           */
635          iaddr |= iaddr_add(hdl, iaddr,
636             bank_to_iaddr(hdl, rcbmp, swzlp, iaddr, bankaddr),
637             "add iaddr bits from bank");

639          /*
640           * Move iaddr up into the range for this MC and insert any
641           * node interleave selection bits.
642           */
643          if (iaddr_unnormalize(hdl, &mcp, iaddr, pap) < 0)
644                 return (-1);    /* errno already set */

646          return (0);
647 }
_____unchanged_portion_omitted_
```

```
**********************************************************
    5296 Tue Jan 15 10:36:53 2019
new/usr/src/lib/libeti/form/common/regcmp.c
10097 indenting fixes in usr/src/{lib,common}
**********************************************************
   1 /*
   2  * CDDL HEADER START
   3  *
   4  * The contents of this file are subject to the terms of the
   5  * Common Development and Distribution License, Version 1.0 only
   6  * (the "License").  You may not use this file except in compliance
   7  * with the License.
   8  *
   9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
  10  * or http://www.opensolaris.org/os/licensing.
  11  * See the License for the specific language governing permissions
  12  * and limitations under the License.
  13  *
  14  * When distributing Covered Code, include this CDDL HEADER in each
  15  * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
  16  * If applicable, add the following below this CDDL HEADER, with the
  17  * fields enclosed by brackets "[]" replaced with your own identifying
  18  * information: Portions Copyright [yyyy] [name of copyright owner]
  19  *
  20  * CDDL HEADER END
  21  */
  22 /*      Copyright (c) 1988 AT&T */
  23 /*        All Rights Reserved   */


  26 /*
  27  * Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
  28  * Use is subject to license terms.
  29  */

  31 /*
  32  * Copyright (c) 2018, Joyent, Inc.
  33  */
  31 #pragma ident   "%Z%%M% %I%     %E% SMI"        /* SVr4.0 1.2   */

  35 /*LINTLIBRARY*/

  37 #include <sys/types.h>
  38 #include <stdlib.h>
  39 #include "utility.h"

  41 /* this code was taken from REGCMP(3X) */

  43 #define SSIZE   16
  44 #define TGRP    48
  45 #define A256    02
  46 #define ZERO    01
  47 #define NBRA    10
  48 #define CIRCFL  32;
  49 #define SLOP    5
  50 #define FEOF    0 /* This was originally EOF but it clashes with the header */
  51                        /* definition so it was changed to FEOF */

  53 #define CBRA    60
  54 #define GRP     40
  55 #define SGRP    56
  56 #define PGRP    68
  57 #define EGRP    44
  58 #define RNGE    03
  59 #define CCHR    20
  60 #define CDOT    64
```

```
  61 #define CCL     24
  62 #define NCCL    8
  63 #define CDOL    28
  64 #define FCEOF   52 /* This was originally CEOF but it clashes with the header */
  65                        /* definition so it was changed to FCEOF */
  66 #define CKET    12

  68 #define STAR    01
  69 #define PLUS    02
  70 #define MINUS   16

  72 intptr_t        *__sp_;
  73 intptr_t        *__stmax;
  74 int     __i_size;

  76 /*ARGSUSED2*/
  77 char *
  78 libform_regcmp(char *cs1, char *cs2)
  79 {
  80         char c;
  81         char *ep, *sp;
  82         int *adx;
  83         int i, cflg;
  84         char *lastep, *sep, *eptr;
  85         int nbra, ngrp;
  86         int cclcnt;
  87         intptr_t stack[SSIZE];

  89         __sp_ = stack;
  90         *__sp_ = -1;
  91         __stmax = &stack[SSIZE];

  93         adx = (int *)&cs1;
  94         i = nbra = ngrp = 0;
  95         while (*adx)
  96                 i += __size((char *)(intptr_t)*adx++);
  97         adx = (int *)&cs1;
  98         sp = (char *)(intptr_t)*adx++;
  99         if ((sep = ep = malloc((unsigned)(2 * i + SLOP))) == NULL)
 100                 return (NULL);
 101         if ((c = *sp++) == FEOF)
 102                 goto cerror;
 103         if (c == '^') {
 104                 c = *sp++;
 105                 *ep++ = CIRCFL;
 106         }
 107         if ((c == '*') || (c == '+') || (c == '{'))
 108                 goto cerror;
 109         sp--;
 110         for (;;) {
 111                 if ((c = *sp++) == FEOF) {
 112                         if (*adx) {
 113                                 sp = (char *)(intptr_t)*adx++;
 114                                 continue;
 115                         }
 116                         *ep++ = FCEOF;
 117                         if (--nbra > NBRA || *__sp_ != -1)
 118                                 goto cerror;
 119                         __i_size = (int) (ep - sep);
 120                         return (sep);
 121                 }
 122                 if ((c != '*') && (c != '{') && (c != '+'))
 123                         lastep = ep;
 124                 switch (c) {

 126                 case '(':
```

```
127                                if (!__rpush(ep)) goto cerror;
128                                *ep++ = CBRA;
129                                *ep++ = -1;
130                                continue;
131                        case ')':
132                                if (!(eptr = (char *)__rpop())) goto cerror;
133                                if ((c = *sp++) == '$') {
134                                        if ('0' > (c = *sp++) || c > '9')
135                                                goto cerror;
136                                        *ep++ = CKET;
137                                        *ep++ = *++eptr = nbra++;
138                                        *ep++ = (c-'0');
139                                        continue;
140                                }
141                                *ep++ = EGRP;
142                                *ep++ = ngrp++;
143                                sp--;
144                                switch (c) {
145                                case '+':
146                                        *eptr = PGRP;
147                                        break;
148                                case '*':
149                                        *eptr = SGRP;
150                                        break;
151                                case '{':
152                                        *eptr = TGRP;
153                                        break;
154                                default:
155                                        *eptr = GRP;
156                                        continue;
157                                }
158                                i = (int) (ep - eptr - 2);
159                                for (cclcnt = 0; i >= 256; cclcnt++)
160                                        i -= 256;
161                                if (cclcnt > 3) goto cerror;
162                                *eptr |= cclcnt;
163                                *++eptr = (char) i;
164                                continue;

166                        case '\\':
167                                *ep++ = CCHR;
168                                if ((c = *sp++) == FEOF)
169                                        goto cerror;
170                                *ep++ = c;
171                                continue;

173                        case '{':
174                                *lastep |= RNGE;
175                                cflg = 0;
176                        nlim:
177                                if ((c = *sp++) == '}') goto cerror;
178                                i = 0;
179                                do {
180                                        if ('0' <= c && c <= '9')
181                                                i = (i*10+(c-'0'));
182                                        else goto cerror;
183                                } while (((c = *sp++) != '}') && (c != ','));
184                                if (i > 255) goto cerror;
185                                *ep++ = (char) i;
186                                if (c == ',') {
187                                        if (cflg++) goto cerror;
188                                        if ((c = *sp++) == '}') {
189                                                *ep++ = -1;
190                                                continue;
191                                        } else {
192                                                sp--;
```

```
193                                                goto nlim;
194                                        }
195                                }
196                                if (!cflg)
197                                        *ep++ = (char) i;
198                                else if ((ep[-1]&0377) < (ep[-2]&0377))
199                                        goto cerror;
200                                continue;

202                        case '.':
203                                *ep++ = CDOT;
204                                continue;

206                        case '+':
207                                if (*lastep == CBRA || *lastep == CKET)
208                                        goto cerror;
209                                *lastep |= PLUS;
210                                continue;

212                        case '*':
213                                if (*lastep == CBRA || *lastep == CKET)
214                                        goto cerror;
215                                *lastep |= STAR;
216                                continue;

218                        case '$':
219                                if ((*sp != FEOF) || (*adx))
220                                        goto defchar;
221                                *ep++ = CDOL;
222                                continue;

224                        case '[':
225                                *ep++ = CCL;
226                                *ep++ = 0;
227                                cclcnt = 1;
228                                if ((c = *sp++) == '^') {
229                                        c = *sp++;
230                                        ep[-2] = NCCL;
231                                }
232                                do {
233                                        if (c == FEOF)
234                                                goto cerror;
235                                        if ((c == '-') && (cclcnt > 1) &&
236                                            (*sp != ']')) {
237                                                *ep = ep[-1];
238                                                ep++;
239                                                ep[-2] = MINUS;
240                                                cclcnt++;
241                                                continue;
242                                        }
243                                        *ep++ = c;
244                                        cclcnt++;
245                                } while ((c = *sp++) != ']');
246                                lastep[1] = (char) cclcnt;
247                                continue;

249                        defchar:
250                        default:
251                                *ep++ = CCHR;
252                                *ep++ = c;
253                        }
254                }
255        cerror:
256                free(sep);
257                return (0);
258        }
```
_____*unchanged_portion_omitted_*

```
**********************************************************
    7633 Tue Jan 15 10:36:53 2019
new/usr/src/lib/libeti/form/common/regex.c
10097 indenting fixes in usr/src/{lib,common}
**********************************************************
   1 /*
   2  * CDDL HEADER START
   3  *
   4  * The contents of this file are subject to the terms of the
   5  * Common Development and Distribution License, Version 1.0 only
   6  * (the "License").  You may not use this file except in compliance
   7  * with the License.
   8  *
   9  * You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
  10  * or http://www.opensolaris.org/os/licensing.
  11  * See the License for the specific language governing permissions
  12  * and limitations under the License.
  13  *
  14  * When distributing Covered Code, include this CDDL HEADER in each
  15  * file and include the License file at usr/src/OPENSOLARIS.LICENSE.
  16  * If applicable, add the following below this CDDL HEADER, with the
  17  * fields enclosed by brackets "[]" replaced with your own identifying
  18  * information: Portions Copyright [yyyy] [name of copyright owner]
  19  *
  20  * CDDL HEADER END
  21  */
  22 /*      Copyright (c) 1988 AT&T */
  23 /*        All Rights Reserved   */


  26 /*
  27  * Copyright 2004 Sun Microsystems, Inc.  All rights reserved.
  28  * Use is subject to license terms.
  29  */

  31 /*
  32  * Copyright (c) 2018, Joyent, Inc.
  33  */

  35 /*LINTLIBRARY*/

  37 #include <sys/types.h>
  38 #include <stdlib.h>
  39 #include <unistd.h>
  40 #include "utility.h"

  42 /*
  43  *      this code was taken from REGCMP(3X)
  44  */
  45 /*VARARGS*/
  46 /*ARGSUSED*/

  48 #define SSIZE    50
  49 #define TGRP     48
  50 #define A256     01
  51 #define A512     02
  52 #define A768     03
  53 #define NBRA     10
  54 #define CIRCFL   32

  56 #define CBRA     60
  57 #define GRP      40
  58 #define SGRP     56
  59 #define PGRP     68
  60 #define EGRP     44
  61 #define RNGE     03
```

```
  62 #define CCHR     20
  63 #define CDOT     64
  64 #define CCL      24
  65 #define NCCL     8
  66 #define CDOL     28
  67 #define FCEOF    52 /* This was originally CEOF but it clashes with the header */
  68                    /* definition so it was changed to FCEOF */
  69 #define CKET     12

  71 #define STAR     01
  72 #define PLUS     02
  73 #define MINUS    16

  75 char     *__braslist[NBRA];
  76 char     *__braelist[NBRA];
  77 char     *__loc1;
  78 intptr_t        __bravar[NBRA];
  79 intptr_t       *__st[SSIZE + 1];
  80 intptr_t       *__eptr_, *__lptr_;
  81 intptr_t        __cflg;

  83 char *
  84 libform_regex(char *addrc, char *addrl, char *a1)
  85 {
  86         intptr_t cur, in;
  87         intptr_t *adx;
  88         char *p1, *p2;

  90         for (in = 0; in < NBRA; in++) {
  91                 __braslist[in] = 0;
  92                 __bravar[in] = -1;
  93         }
  94         __cflg = 0;
  95         cur = __execute(addrc, addrl);
  96         adx = (intptr_t *)&a1;
  97         for (in = 0; in < NBRA; in++) {
  98                 if (((p1 = __braslist[in]) != 0) && (__bravar[in] >= 0)) {
  99                         p2 = (char *)adx[__bravar[in]];
 100                         while (p1 < __braelist[in]) *p2++ = *p1++;
 101                         *p2 = '\0';
 102                 }
 103         }
 104         if (!__cflg)
 105                 return ((addrl == (char *)cur) ? (char *)0 : (char *)cur);
 106         else
 107                 return ((char *)cur);
 108 }
_____unchanged_portion_omitted_
```

```
**********************************************************
    4563 Tue Jan 15 10:36:53 2019
new/usr/src/lib/libmp/common/mdiv.c
10097 indenting fixes in usr/src/{lib,common}
**********************************************************
   1 /*      Copyright (c) 1984, 1986, 1987, 1988, 1989 AT&T */
   2 /*        All Rights Reserved   */


   5 /*
   6  * Copyright (c) 1980 Regents of the University of California.
   7  * All rights reserved.  The Berkeley software License Agreement
   8  * specifies the terms and conditions for redistribution.
   9  */
  10 /*      Portions Copyright(c) 1988, Sun Microsystems Inc.       */
  11 /*        All Rights Reserved                                   */

  13 /*
  14  * Copyright (c) 1997, by Sun Microsystems, Inc.
  15  * All rights reserved.
  16  */

  18 /*
  19  * Copyright (c) 2018, Joyent, Inc.
  20  */
  18 #ident  "%Z%%M% %I%      %E% SMI"          /* SVr4.0 1.1   */

  22 /* LINTLIBRARY */

  24 #include <mp.h>
  25 #include <stdio.h>
  26 #include <stdlib.h>
  27 #include <sys/types.h>
  28 #include "libmp.h"

  30 static void m_div(MINT *, MINT *, MINT *, MINT *);

  32 void
  33 mp_mdiv(MINT *a, MINT *b, MINT *q, MINT *r)
  34 {
  35         MINT x, y;
  36         int sign;

  38         sign = 1;
  39         x.len = y.len = 0;
  40         _mp_move(a, &x);
  41         _mp_move(b, &y);
  42         if (x.len < 0) {
  43                 sign = -1;
  44                 x.len = -x.len;
  45         }
  46         if (y.len < 0) {
  47                 sign = -sign;
  48                 y.len = -y.len;
  49         }
  50         _mp_xfree(q);
  51         _mp_xfree(r);
  52         m_div(&x, &y, q, r);
  53         if (sign == -1) {
  54                 q->len = -q->len;
  55                 r->len = -r->len;
  56         }
  57         _mp_xfree(&x);
  58         _mp_xfree(&y);
  59 }
_____unchanged_portion_omitted_
```

****
**   3142 Tue Jan 15 10:36:53 2019**
new/usr/src/lib/libmp/common/mout.c
10097 indenting fixes in usr/src/{lib,common}
****
     1 /*   Copyright (c) 1984, 1986, 1987, 1988, 1989 AT&T */
     2 /*   All Rights Reserved   */

     5 /*
     6 * Copyright (c) 1980 Regents of the University of California.
     7 * All rights reserved. The Berkeley software License Agreement
     8 * specifies the terms and conditions for redistribution.
     9 */
    10 /*   Portions Copyright(c) 1988, Sun Microsystems Inc.   */
    11 /*   All Rights Reserved   */

    13 /*
    14 * Copyright (c) 1997, by Sun Microsystems, Inc.
    15 * All rights reserved.
    16 */

    18 **/***
    19 ** * Copyright (c) 2018, Joyent, Inc.**
    20 ** */**
    18 *#ident "%Z%%M% %I%   %E% SMI"   /* SVr4.0 1.1   */*

    22 /* LINTLIBRARY */

    24 #include <stdio.h>
    25 #include <mp.h>
    26 #include <sys/types.h>
    27 #include "libmp.h"
    28 #include <stdlib.h>

    30 static int
    31 m_in(MINT *a, short b, FILE *f)
    32 {
    33   MINT x, y, ten;
    34   int sign, c;
    35   short qten, qy;

    37   _mp_xfree(a);
    38   sign = 1;
    39   ten.len = 1;
    40   ten.val = &qten;
    41   qten = b;
    42   x.len = 0;
    43   y.len = 1;
    44   y.val = &qy;
    45   **while ((c = getc(f)) != EOF) {**
    43   *while ((c = getc(f)) != EOF)*
    46   switch (c) {

    48   case '\\':
    49   (void) getc(f);
    50   continue;
    51   case '\t':
    52   case '\n':
    53   a->len *= sign;
    54   _mp_xfree(&x);
    55   return (0);
    56   case ' ':
    57   continue;
    58   case '-':
    59   sign = -sign;

    60   continue;
    61   default:
    62   if (c >= '0' && c <= '9') {
    63   qy = c - '0';
    64   mp_mult(&x, &ten, a);
    65   mp_madd(a, &y, a);
    66   _mp_move(a, &x);
    67   continue;
    68   } else {
    69   (void) ungetc(c, stdin);
    70   a->len *= sign;
    71   return (0);
    72   }
    73   }
    74   **}**

    76   return (EOF);
    77 }
_____unchanged_portion_omitted_

****
   6831 Tue Jan 15 10:36:53 2019
new/usr/src/lib/pam_modules/authtok_check/packlib.c
10097 indenting fixes in usr/src/{lib,common}
****
    1 /*
    2   * Copyright 2009 Sun Microsystems, Inc. All rights reserved.
    3   * Use is subject to license terms.
    4   */

    6 /*
    7   *** Copyright (c) 2018, Joyent, Inc.**
    8   ***/**

   10 **/***
   11   * This program is copyright Alec Muffett 1993. The author disclaims all
   12   * responsibility or liability with respect to it's usage or its effect
   13   * upon hardware or computer systems, and maintains copyright as set out
   14   * in the "LICENCE" document which accompanies distributions of Crack v4.0
   15   * and upwards.
   16   */

   18 #include "packer.h"

   20 void
   21 PWRemove(char *path)
   22 {
   23   char fname[PATH_MAX];

   25   (void) snprintf(fname, sizeof (fname), "%s/%s", path,
   26   DICT_DATABASE_PWI);
   27   (void) unlink(fname);
   28   (void) snprintf(fname, sizeof (fname), "%s/%s", path,
   29   DICT_DATABASE_PWD);
   30   (void) unlink(fname);
   31   (void) snprintf(fname, sizeof (fname), "%s/%s", path,
   32   DICT_DATABASE_HWM);
   33   (void) unlink(fname);
   34 }
_____unchanged_portion_omitted_