

```

0001 *
0002 *****
0003 * Copyright (c) 2004 Zumasys, Inc. as an unpublished work. Permission is *
0004 * hereby granted, free of charge, to any person obtaining a copy of this *
0005 * software, to use the software without restriction, including without *
0006 * limitation the rights to use, copy, modify, merge, publish or *
0007 * distribute the software, and to permit persons to whom the software is *
0008 * furnished to do so, subject to the following conditions: This *
0009 * copyright notice and permission notice shall be included in all copies *
0010 * or substantial portions of the software. THE SOFTWARE IS PROVIDED "AS *
0011 * IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT *
0012 * NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A *
0013 * PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE *
0014 * COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, *
0015 * WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT *
0016 * OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN *
0017 * THE SOFTWARE. *
0018 *****
0019 *****
0020 *
0021 * USER INTERFACE EXAMPLE PROGRAM 6
0022 *
0023 *****
0024 *****
0025 *
0026 * This example illustrates a full Graphical User Interface (GUI) using
0027 * AccuTerm GUI. This program was initially created using the code
0028 * generator in the GUI Designer. Then application-specific initialization
0029 * code and event handling code was added.
0030 *
0031 *****
0032 *
0033 * EQUates for control characters and delimiters and other constants
0034 *
0035 EQU AM TO CHAR(254), VM TO CHAR(253), SVM TO CHAR(252)
0036 EQU FALSE TO 0
0037 EQU TRUE TO 1
0038 *
0039 *****
0040 *
0041 * The CUST.REC INCLUDE item declares the CUST.REC array, which is used to
0042 * store a working copy of the current data record. It also defines the
0043 * record layout by equating field names to array positions in the CUST.REC
0044 * array.
0045 *
0046 $INCLUDE UIEX.CUST.REC
0047 *
0048 *****
0049 *
0050 * This program displays pictures associated with records in the customer
0051 * file. This EQUate defines the path where the image files are located.
0052 $INCLUDE UIEX.PIX.PATH
0053 *
0054 *****
0055 *
0056 * Open our files...
0057 *
0058 OPEN 'CUST.SAMPLE' TO FN.CUST ELSE PRINT 'NO CUST.SAMPLE FILE'; STOP
0059 OPEN 'CUST.SAMPLE.CTRL' TO FN.CUST.CTRL ELSE PRINT 'NO CUST.SAMPLE.CTRL FILE'; STOP
0060 OPEN 'CUST.SAMPLE.XREF' TO FN.CUST.XREF ELSE PRINT 'NO CUST.SAMPLE.XREF FILE'; STOP
0061 OPEN 'UIEX' TO GUI.PROJECT.FILE ELSE PRINT 'NO UIEX FILE'; STOP

```

```

0062 *
0063 *****
0064 *
0065 * Build a list of the control IDs that are on the form. This list is used
0066 * when calling ATGUILOADVALUES to load the value of each control after
0067 * reading a record from the file.
0068
0069 ALL.CTLS = ''
0070 ALL.CTLS<-1> = 'txtCustID'
0071 ALL.CTLS<-1> = 'txtContact'
0072 ALL.CTLS<-1> = 'txtName'
0073 ALL.CTLS<-1> = 'txtAddress1'
0074 ALL.CTLS<-1> = 'txtAddress2'
0075 ALL.CTLS<-1> = 'txtCity'
0076 ALL.CTLS<-1> = 'txtSt'
0077 ALL.CTLS<-1> = 'txtZip'
0078 ALL.CTLS<-1> = 'txtCountry'
0079 ALL.CTLS<-1> = 'txtPhone'
0080 ALL.CTLS<-1> = 'txtFax'
0081 ALL.CTLS<-1> = 'txtNotes'
0082 ALL.CTLS<-1> = 'pix'
0083 *
0084 *****
0085 *
0086 * Read the GUI template from the project file into the TEMPLATE variable
0087 *
0088 READ TEMPLATE FROM GUI.PROJECT.FILE, 'UIEX6.GUI' ELSE STOP 202, 'UIEX6.GUI'
0089 *
0090 *
0091 *****
0092 *
0093 * All sections enclosed in *-->BEGIN ... <--* and *-->END ... <--* tags
0094 * are created and maintained by the GUI code generator. Except for the
0095 * event handler code, these sections should not be modified.
0096 *
0097 *****
0098 *
0099 *
0100 *****
0101 *
0102 *
0103 *-->BEGIN GUI HEADER<--*
0104 $INCLUDE GUIBP ATGUIEQUATES
0105 CALL ATGUIINIT(GUIERRORS, GUISTATE)
0106 IF GUIERRORS<1> >= 3 THEN GOTO GUI.ERROR
0107 CALL ATGUIRUNMACRO(TEMPLATE, '', GUIERRORS, GUISTATE)
0108 IF GUIERRORS<1> >= 3 THEN GOTO GUI.ERROR
0109 *-->END GUI HEADER<--*
0110 *
0111 *
0112 *****
0113 *
0114 *
0115 *-->BEGIN GUI STARTUP<--*
0116 CALL ATGUISHOW('UIEX6', 'FRMMAIN', '', '', GUIERRORS, GUISTATE)
0117 IF GUIERRORS<1> >= 2 THEN GOTO GUI.ERROR
0118 *-->END GUI STARTUP<--*
0119 *
0120 *
0121 *****
0122 *

```

```
0123 *
0124 *-->BEGIN EVENT LOOP<--*
0125 LOOP
0126 CALL ATGUIWAITEVENT(GUIAPP, GUIFRM, GUICTL, GUIEVT, GUIARGS, GUIERRORS, GUISTATE)
0127 IF GUIERRORS<1> >= 2 THEN GOTO GUI.ERROR
0128 UNTIL GUIEVT EQ GEQUIT DO
0129 GUIAPP=OCONV(GUIAPP, 'MCU')
0130 GUIFRM=OCONV(GUIFRM, 'MCU')
0131 GUICTL=OCONV(GUICTL, 'MCU')
0132 GOSUB GUI.DECODE.EVENT
0133 REPEAT
0134 *-->END EVENT LOOP<--*
0135 *
0136 *
0137 *****
0138 *
0139 *
0140 *-->BEGIN GUI TRAILER<--*
0141 CALL ATGUISHUTDOWN
0142 STOP
0143 *-->END GUI TRAILER<--*
0144 *
0145 *
0146 *****
0147 *
0148 *
0149 *-->BEGIN EVENT DECODER<--*
0150 GUI.DECODE.EVENT: *
0151 BEGIN CASE
0152 CASE GUIAPP='UIEX6'
0153 BEGIN CASE
0154 CASE GUIFRM='FRMMAIN'
0155 BEGIN CASE
0156 CASE GUICTL=''
0157 BEGIN CASE
0158 CASE GUIEVT=GECLOSE
0159 GOSUB GUI.UIEX6.FRMMAIN.CLOSE
0160 END CASE
0161 CASE GUICTL='CMDCANCEL'
0162 BEGIN CASE
0163 CASE GUIEVT=GECLICK
0164 GOSUB GUI.UIEX6.FRMMAIN.CMDCANCEL.CLICK
0165 END CASE
0166 CASE GUICTL='CMDDELETE'
0167 BEGIN CASE
0168 CASE GUIEVT=GECLICK
0169 GOSUB GUI.UIEX6.FRMMAIN.CMDDELETE.CLICK
0170 END CASE
0171 CASE GUICTL='CMDEXIT'
0172 BEGIN CASE
0173 CASE GUIEVT=GECLICK
0174 GOSUB GUI.UIEX6.FRMMAIN.CMDEXIT.CLICK
0175 END CASE
0176 CASE GUICTL='CMDNEW'
0177 BEGIN CASE
0178 CASE GUIEVT=GECLICK
0179 GOSUB GUI.UIEX6.FRMMAIN.CMDNEW.CLICK
0180 END CASE
0181 CASE GUICTL='CMDSAVE'
0182 BEGIN CASE
0183 CASE GUIEVT=GECLICK
```

```

0184     GOSUB GUI.UIEX6.FRMMAIN.CMDSAVE.CLICK
0185     END CASE
0186     CASE GUICTL='CMDSEARCH'
0187     BEGIN CASE
0188         CASE GUI EVT=GECLICK
0189         GOSUB GUI.UIEX6.FRMMAIN.CMDSEARCH.CLICK
0190         END CASE
0191     CASE GUICTL='MENU'
0192     GUIMNU=OCONV(GUIARGS<1,1>,'MCU')
0193     BEGIN CASE
0194         CASE GUIMNU='MNUCANCEL'
0195         GOSUB GUI.UIEX6.FRMMAIN.MENU.MNUCANCEL
0196         CASE GUIMNU='MNUDELETE'
0197         GOSUB GUI.UIEX6.FRMMAIN.MENU.MNUDELETE
0198         CASE GUIMNU='MNUEXIT'
0199         GOSUB GUI.UIEX6.FRMMAIN.MENU.MNUEXIT
0200         CASE GUIMNU='MNUNEW'
0201         GOSUB GUI.UIEX6.FRMMAIN.MENU.MNUNEW
0202         CASE GUIMNU='MNUSAVE'
0203         GOSUB GUI.UIEX6.FRMMAIN.MENU.MNUSAVE
0204         END CASE
0205     CASE GUICTL='TXTCUSTID'
0206     BEGIN CASE
0207         CASE GUI EVT=GEVALIDATE
0208         GOSUB GUI.UIEX6.FRMMAIN.TXTCUSTID.VALIDATE
0209         END CASE
0210     END CASE
0211     CASE GUIFRM='FRMXREF'
0212     BEGIN CASE
0213         CASE GUICTL=''
0214         BEGIN CASE
0215             CASE GUI EVT=GECLOSE
0216             GOSUB GUI.UIEX6.FRMXREF.CLOSE
0217             END CASE
0218         CASE GUICTL='CMDCANCEL'
0219         BEGIN CASE
0220             CASE GUI EVT=GECLICK
0221             GOSUB GUI.UIEX6.FRMXREF.CMDCANCEL.CLICK
0222             END CASE
0223         CASE GUICTL='CMDOK'
0224         BEGIN CASE
0225             CASE GUI EVT=GECLICK
0226             GOSUB GUI.UIEX6.FRMXREF.CMDOK.CLICK
0227             END CASE
0228         CASE GUICTL='LSTXREF'
0229         BEGIN CASE
0230             CASE GUI EVT=GEDBLCLICK
0231             GOSUB GUI.UIEX6.FRMXREF.LSTXREF.DBLCLICK
0232             END CASE
0233         END CASE
0234     END CASE
0235     END CASE
0236     RETURN
0237     *-->END EVENT DECODER<--*
0238     *
0239     *
0240     *****
0241     *
0242     *
0243     *-->BEGIN CLOSE EVENT HANDLER<--*
0244     GUI.UIEX6.FRMMAIN.CLOSE: *

```

```
0245 *
0246 * This is the 'Form Close' event handler. Here we check if
0247 * any changes have been made, and allow the user to save or
0248 * abandon changes, or cancel the action. If no changes (or
0249 * if changes were saved), the CLOSEIT subroutine is called,
0250 * closing the form, and as a result, the program will end.
0251 *
0252 GOSUB ASKSAVECHANGES
0253 IF OK THEN GOSUB CLOSEIT
0254 RETURN
0255 *-->END CLOSE EVENT HANDLER<--*
0256 *
0257 *
0258 *****
0259 *
0260 *
0261 *-->BEGIN EVENT HANDLER<--*
0262 GUI.UIEX6.FRMMAIN.COMDCANCEL.CLICK: *
0263 *
0264 * This is the 'cmdCancel Click' event handler. Here we simply
0265 * call the local CANCEL.RECORD subroutine.
0266 *
0267 GOSUB CANCEL.RECORD
0268 RETURN
0269 *-->END EVENT HANDLER<--*
0270 *
0271 *
0272 *****
0273 *
0274 *
0275 *-->BEGIN EVENT HANDLER<--*
0276 GUI.UIEX6.FRMMAIN.COMDELETE.CLICK: *
0277 *
0278 * This is the 'cmdDelete Click' event handler. Here we simply
0279 * call the local DELETE.RECORD subroutine to delete the item
0280 * from the data and index files.
0281 *
0282 GOSUB DELETE.RECORD
0283 RETURN
0284 *-->END EVENT HANDLER<--*
0285 *
0286 *
0287 *****
0288 *
0289 *
0290 *-->BEGIN EVENT HANDLER<--*
0291 GUI.UIEX6.FRMMAIN.COMDEXIT.CLICK: *
0292 *
0293 * This is the 'cmdExit Click' event handler. Here we check if
0294 * any changes have been made, and allow the user to save or
0295 * abandon changes, or cancel the action. If no changes (or
0296 * if changes were saved), the CLOSEIT subroutine is called,
0297 * closing the form, and as a result, the program will end.
0298 *
0299 GOSUB ASKSAVECHANGES
0300 IF OK THEN GOSUB CLOSEIT
0301 RETURN
0302 *-->END EVENT HANDLER<--*
0303 *
0304 *
0305 *****
```

```
0306 *
0307 *
0308 *-->BEGIN EVENT HANDLER<--*
0309 GUI.UIEX6.FRMMAIN.CMDNEW.CLICK: *
0310 *
0311 * This is the 'cmdNew Click' event handler. Here we simply
0312 * call the local NEW.RECORD subroutine to start a new record.
0313 *
0314 GOSUB NEW.RECORD
0315 RETURN
0316 *-->END EVENT HANDLER<--*
0317 *
0318 *
0319 *****
0320 *
0321 *
0322 *-->BEGIN EVENT HANDLER<--*
0323 GUI.UIEX6.FRMMAIN.CMDSAVE.CLICK: *
0324 *
0325 * This is the 'cmdSave Click' event handler. Here we simply
0326 * call the local UPDATE.RECORD subroutine to update the data
0327 * and index files with data from the GUI form.
0328 *
0329 GOSUB UPDATE.RECORD
0330 RETURN
0331 *-->END EVENT HANDLER<--*
0332 *
0333 *
0334 *****
0335 *
0336 *
0337 *-->BEGIN EVENT HANDLER<--*
0338 GUI.UIEX6.FRMMAIN.CMDSEARCH.CLICK: *
0339 *
0340 * This is the 'cmdSearch Click' event handler. Here we retrieve
0341 * the value of txtCustID from the GUI form and pass it to the
0342 * local SEARCH subroutine.
0343 *
0344 * Get the search string
0345 CALL ATGUIGETPROP('UIEX6','frmMAIN','txtCustID',GPVALUE,0,0,SRCH,GUIERRORS,GUISTATE)
0346 IF GUIERRORS<1> THEN RETURN ;* Ignore event if error
0347 GOSUB SEARCH
0348 RETURN
0349 *-->END EVENT HANDLER<--*
0350 *
0351 *
0352 *****
0353 *
0354 *
0355 *-->BEGIN EVENT HANDLER<--*
0356 GUI.UIEX6.FRMMAIN.MENU.MNUCANCEL: *
0357 *
0358 * This is the 'Menu mnuCancel' event handler.
0359 * Forward this to the cmdCancel Click handler.
0360 *
0361 GOSUB GUI.UIEX6.FRMMAIN.CMDCANCEL.CLICK
0362 RETURN
0363 *-->END EVENT HANDLER<--*
0364 *
0365 *
0366 *****
```

```

0367 *
0368 *
0369 *-->BEGIN EVENT HANDLER<--*
0370 GUI.UIEX6.FRMMAIN.MENU.MNUDELETE: *
0371 *
0372 * This is the 'Menu mnuDelete' event handler.
0373 * Forward this to the cmdDelete Click handler.
0374 *
0375 GOSUB GUI.UIEX6.FRMMAIN.CMDDELETE.CLICK
0376 RETURN
0377 *-->END EVENT HANDLER<--*
0378 *
0379 *
0380 *****
0381 *
0382 *
0383 *-->BEGIN EVENT HANDLER<--*
0384 GUI.UIEX6.FRMMAIN.MENU.MNUEXIT: *
0385 *
0386 * This is the 'Menu mnuExit' event handler.
0387 * Forward this to the cmdExit Click handler.
0388 *
0389 GOSUB GUI.UIEX6.FRMMAIN.CMDEXIT.CLICK
0390 RETURN
0391 *-->END EVENT HANDLER<--*
0392 *
0393 *
0394 *****
0395 *
0396 *
0397 *-->BEGIN EVENT HANDLER<--*
0398 GUI.UIEX6.FRMMAIN.MENU.MNUNEW: *
0399 *
0400 * This is the 'Menu mnuNew' event handler.
0401 * Forward this to the cmdNew Click handler.
0402 *
0403 GOSUB GUI.UIEX6.FRMMAIN.CMDNEW.CLICK
0404 RETURN
0405 *-->END EVENT HANDLER<--*
0406 *
0407 *
0408 *****
0409 *
0410 *
0411 *-->BEGIN EVENT HANDLER<--*
0412 GUI.UIEX6.FRMMAIN.MENU.MNUSAVE: *
0413 *
0414 * This is the 'Menu mnuSave' event handler.
0415 * Forward this to the cmdSave Click handler.
0416 *
0417 GOSUB GUI.UIEX6.FRMMAIN.CMDSAVE.CLICK
0418 RETURN
0419 *-->END EVENT HANDLER<--*
0420 *
0421 *
0422 *****
0423 *
0424 *
0425 *-->BEGIN EVENT HANDLER<--*
0426 GUI.UIEX6.FRMMAIN.TXTCUSTID.VALIDATE: *
0427 *

```

```

0428 * This is the 'txtCustID Validate' event handler. Check which
0429 * object triggered the event (some command buttons should not
0430 * cause a new record to be read). Extract the new ID from
0431 * GUIARGS<2>, then pass the ID to the local CHECK.ID subroutine.
0432 *
0433 WHICH = OCONV(FIELD(GUIARGS<1,1>,'*',3),'MCU')
0434 IF WHICH='CMDCANCEL' OR WHICH='CMDEXIT' THEN
0435   RETURN; * Ignore validate if caused by special command buttons
0436 END
0437 ID = GUIARGS<2> ;* New value of ID object
0438 GOSUB CHECK.ID
0439 RETURN
0440 *-->END EVENT HANDLER<--*
0441 *
0442 *
0443 *****
0444 *
0445 *
0446 *-->BEGIN CLOSE EVENT HANDLER<--*
0447 GUI.UIEX6.FRMXREF.CLOSE: *
0448 *
0449 * This is the search dialog 'form Close' event handler.
0450 * Forward this to the cmdCancel Click handler.
0451 *
0452 GOSUB GUI.UIEX6.FRMXREF.CMDCANCEL.CLICK
0453 RETURN
0454 *-->END CLOSE EVENT HANDLER<--*
0455 *
0456 *
0457 *****
0458 *
0459 *
0460 *-->BEGIN EVENT HANDLER<--*
0461 GUI.UIEX6.FRMXREF.CMDCANCEL.CLICK: *
0462 *
0463 * This is the search dialog 'cmdCancel Click' event handler.
0464 * Hide the search dialog box & activate txtCustID on the
0465 * main form.
0466 *
0467 CALL ATGUIHIDE(GUIAPP,GUIFRM,'',' ',GUIERRORS,GUISTATE)
0468 IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0469 CALL ATGUIACTIVATE('UIEX6','frmMAIN','txtCustID',GUIERRORS,GUISTATE)
0470 IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0471 RETURN
0472 *-->END EVENT HANDLER<--*
0473 *
0474 *
0475 *****
0476 *
0477 *
0478 *-->BEGIN EVENT HANDLER<--*
0479 GUI.UIEX6.FRMXREF.CMDOK.CLICK: *
0480 *
0481 * This is the search dialog 'cmdOK Click' event handler.
0482 * Forward this to the lstXref DblClick handler.
0483 *
0484 GOSUB GUI.UIEX6.FRMXREF.LSTXREF.DBLCLICK
0485 RETURN
0486 *-->END EVENT HANDLER<--*
0487 *
0488 *

```



```

0489 *****
0490 *
0491 *
0492 *-->BEGIN EVENT HANDLER<--*
0493 GUI.UIEX6.FRMXREF.LSTXREF.DBLCLICK: *
0494 *
0495 * This is the search dialog 'lstXref DblClick' event handler.
0496 * Retrieve the index of the selected item in the cross-reference
0497 * list, then extract the ID from the original list and call the
0498 * local READ.RECORD subroutine to read selected customer record
0499 * and load data to main form.
0500 *
0501 OK = FALSE
0502 CALL ATGUIGETPROP('UIEX6','frmXREF','lstXref',GPVALUE,0,0,IDX,GUIERRORS,GUISTATE)
0503 IF IDX > 0 THEN
0504     ID = XREF.LIST<1,IDX,1>
0505     GOSUB READ.RECORD
0506 END
0507 CALL ATGUIHIDE(GUIAPP,GUIFRM,'',' ',GUIERRORS,GUISTATE)
0508 IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0509 IF OK THEN
0510     CALL ATGUIACTIVATE('UIEX6','frmMAIN','txtContact',GUIERRORS,GUISTATE)
0511     IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0512 END ELSE
0513     CALL ATGUIACTIVATE('UIEX6','frmMAIN','txtCustID',GUIERRORS,GUISTATE)
0514     IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0515 END
0516 RETURN
0517 *-->END EVENT HANDLER<--*
0518 *
0519 *
0520 *****
0521 *
0522 *
0523 *-->BEGIN ERROR HANDLER<--*
0524 GUI.ERROR: *
0525 CALL ATGUISHUTDOWN
0526 PRINT 'The following errors have been reported by the GUI system:'
0527 NUMERRS=DCOUNT(GUIERRORS,AM)
0528 FOR EACHERR=2 TO NUMERRS
0529     PRINT GUIERRORS<EACHERR,6>
0530 NEXT EACHERR
0531 STOP
0532 *-->END ERROR HANDLER<--*
0533 *
0534 *
0535 *****
0536 *
0537 *
0538 *****
0539 *****
0540 * LOCAL SUBROUTINES (APPLICATION-SPECIFIC)
0541 *****
0542 *****
0543 *
0544 *
0545 *****
0546 *
0547 * The CHECK.ID subroutine validates a newly entered customer ID. The
0548 * READ.RECORD subroutine is called to read the specified record. If the
0549 * record does not exist, the ID is passed as a search string to the

```

```

0550 * SEARCH subroutine to display a list of matching records to choose from.
0551 *
0552 CHECK.ID: *
0553 *
0554 * Try to read a record using the entered ID
0555 GOSUB READ.RECORD
0556 IF NOT(OK) THEN
0557   * Record does not exist - use the ID as a search string
0558   SRCH = ID
0559   GOSUB SEARCH
0560 END
0561 RETURN
0562 *
0563 *
0564 *****
0565 *
0566 * The READ.RECORD subroutine reads a customer record, and if successful,
0567 * extracts record data and populates the main form with the data. The OK
0568 * variable is set to TRUE to indicate success. Otherwise, OK is set to
0569 * FALSE and the routine returns.
0570 *
0571 READ.RECORD: *
0572 *
0573 OK = FALSE
0574 *
0575 * Try to read a record using the entered ID
0576 MATREAD CUST.REC FROM FN.CUST, ID ELSE RETURN
0577 *
0578 * Read was successful - extract data
0579 CUST.ID = ID
0580 VALS = ''
0581 N = DCOUNT(ALL.CTLS,AM)
0582 FOR I = 1 TO N
0583   CTL = ALL.CTLS<I>
0584   BEGIN CASE
0585     CASE CTL EQ 'txtCustID'; VALS<I> = CUST.ID
0586     CASE CTL EQ 'txtContact'; VALS<I> = CUST.CONTACT
0587     CASE CTL EQ 'txtName'; VALS<I> = CUST.NAME
0588     CASE CTL EQ 'txtAddress1'; VALS<I> = CUST.ADDRESS1
0589     CASE CTL EQ 'txtAddress2'; VALS<I> = CUST.ADDRESS2
0590     CASE CTL EQ 'txtCity'; VALS<I> = CUST.CITY
0591     CASE CTL EQ 'txtST'; VALS<I> = CUST.ST
0592     CASE CTL EQ 'txtZip'; VALS<I> = CUST.ZIP
0593     CASE CTL EQ 'txtCountry'; VALS<I> = CUST.COUNTRY
0594     CASE CTL EQ 'txtPhone'; VALS<I> = CUST.PHONE
0595     CASE CTL EQ 'txtFax'; VALS<I> = CUST.FAX
0596     CASE CTL EQ 'txtNotes'; CONVERT SVM TO VM IN CUST.HISTORY; VALS<I> = CUST.HISTORY
0597     CASE CTL EQ 'pix'; VALS<I> = PIX.PATH:'CUST/':CUST.ID:'.JPG'
0598   END CASE
0599 NEXT I
0600 *
0601 * Populate main form with data from customer record
0602 CALL ATGUISETPROP('UIEX6', 'frmMAIN', 'pix', GPVALUE, 0, 0, '', GUIERRORS, GUISTATE) ;* In
case picture is missing
0603 CALL ATGUILOADVALUES('UIEX6', 'frmMAIN', ALL.CTLS, VALS, GUIERRORS, GUISTATE)
0604 IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0605 *
0606 * Enable the Delete button and menu item, since we have a record
0607 CALL ATGUIENABLE('UIEX6', 'frmMAIN', 'cmdDelete', '', GUIERRORS, GUISTATE)
0608 IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0609 CALL ATGUIENABLE('UIEX6', 'frmMAIN', 'Menu', 'mnuDelete', GUIERRORS, GUISTATE)

```

```

0610 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0611 CALL ATGUIENABLE('UIEX6','frmMAIN','cmdSave','',GUIERRORS,GUISTATE)
0612 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0613 CALL ATGUIENABLE('UIEX6','frmMAIN','Menu','mnuSave',GUIERRORS,GUISTATE)
0614 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0615 CALL ATGUISETPROP('UIEX6','frmMAIN','',GPCHANGED,0,0,0,GUIERRORS,GUISTATE)
0616 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0617 *
0618 OK = TRUE
0619 *
0620 RETURN
0621 *
0622 *
0623 *****
0624 *
0625 * The NEW.RECORD subroutine checks if any changes have been made, and
0626 * allows the user to save or abandon changes. If no changes (or if
0627 * changes were saved), the next record ID is read from the
0628 * CUST.SAMPLE.CTRL file and the form data is reset. The value of txtCustID
0629 * is set to the new ID and the txtContact field is activated, ready to
0630 * enter new customer information.
0631 *
0632 NEW.RECORD: *
0633 *
0634 GOSUB ASKSAVECHANGES
0635 IF OK THEN
0636   READVU CUST.ID FROM FN.CUST.CTRL,'NEXT',2 THEN
0637     WRITEV CUST.ID + 1 ON FN.CUST.CTRL,'NEXT',2
0638   END ELSE
0639     CALL ATGUIMSGBOX("Next item counter record not
found!", "Error", 3, 4, 0, RSP, GUIERRORS, GUISTATE)
0640     RETURN
0641   END
0642   MAT CUST.REC = ''
0643   CALL ATGUIRESET('UIEX6','frmMAIN',GUIERRORS,GUISTATE)
0644   IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0645   CALL
ATGUISETPROP('UIEX6','frmMAIN','txtCustID',GPVALUE,0,0,CUST.ID,GUIERRORS,GUISTATE)
0646 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0647 CALL ATGUIDISABLE('UIEX6','frmMAIN','cmdDelete','',GUIERRORS,GUISTATE)
0648 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0649 CALL ATGUIDISABLE('UIEX6','frmMAIN','Menu','mnuDelete',GUIERRORS,GUISTATE)
0650 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0651 CALL ATGUIENABLE('UIEX6','frmMAIN','cmdSave','',GUIERRORS,GUISTATE)
0652 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0653 CALL ATGUIENABLE('UIEX6','frmMAIN','Menu','mnuSave',GUIERRORS,GUISTATE)
0654 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0655 CALL ATGUIACTIVATE('UIEX6','frmMAIN','txtContact',GUIERRORS,GUISTATE)
0656 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0657 END
0658 RETURN
0659 *
0660 *
0661 *****
0662 *
0663 * The UPDATE.RECORD subroutine retrieves all updated data from the GUI
0664 * form and updates the CUST.REC array. It then calls the UIEX.CUST.UPDATE
0665 * subroutine to update the data and index files. After the update, the
0666 * Changed property of the main form is cleared, indicating that no data
0667 * (on the GUI form) has been updated.
0668 *

```

```

0669 UPDATE.RECORD: *
0670 *
0671 CALL ATGUIGETUPDATES('UIEX6','frmMAIN',CTLS,VALS,GUIERRORS,GUISTATE)
0672 IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0673 N = DCOUNT(CTLS,AM)
0674 FOR I = 1 TO N
0675   CTL = CTLS<I>
0676   BEGIN CASE
0677     CASE CTL EQ 'txtCustID'; CUST.ID = VALS<I>
0678     CASE CTL EQ 'txtContact'; CUST.CONTACT = VALS<I>
0679     CASE CTL EQ 'txtName'; CUST.NAME = VALS<I>
0680     CASE CTL EQ 'txtAddress1'; CUST.ADDRESS1 = VALS<I>
0681     CASE CTL EQ 'txtAddress2'; CUST.ADDRESS2 = VALS<I>
0682     CASE CTL EQ 'txtCity'; CUST.CITY = VALS<I>
0683     CASE CTL EQ 'txtST'; CUST.ST = VALS<I>
0684     CASE CTL EQ 'txtZip'; CUST.ZIP = VALS<I>
0685     CASE CTL EQ 'txtCountry'; CUST.COUNTRY = VALS<I>
0686     CASE CTL EQ 'txtPhone'; CUST.PHONE = VALS<I>
0687     CASE CTL EQ 'txtFax'; CUST.FAX = VALS<I>
0688     CASE CTL EQ 'txtNotes'; CUST.HISTORY = VALS<I>; CONVERT VM TO SVM IN CUST.HISTORY
0689   END CASE
0690 NEXT I
0691 IF CUST.ID EQ '' THEN
0692   CALL ATGUIMSGBOX('Missing customer ID - update
cancelled!', 'Warning', 2, 0, 0, RSP, GUIERRORS, GUISTATE)
0693 END ELSE
0694   * Update the file using the application-specific UIEX.CUST.UPDATE subroutine
0695   CALL UIEX.CUST.UPDATE(CUST.ID, MAT CUST.REC, FN.CUST, FN.CUST.XREF)
0696   CALL ATGUISETPROP('UIEX6','frmMAIN','', GPCHANGED, 0, 0, 0, GUIERRORS, GUISTATE)
0697   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0698 END
0699 RETURN
0700 *
0701 *
0702 *****
0703 *
0704 * The CANCEL.RECORD subroutine checks if any changes have been made,
0705 * and allows the user to save or abandon changes. If no changes (or
0706 * if changes were saved), the form data is reset and the CUST.REC array
0707 * is cleared.
0708 *
0709 CANCEL.RECORD: *
0710 *
0711 GOSUB ASKSAVECHANGES
0712 IF OK THEN
0713   CUST.ID = ''
0714   MAT CUST.REC = ''
0715   CALL ATGUIRESET('UIEX6','frmMAIN',GUIERRORS,GUISTATE)
0716   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0717   CALL ATGUIDISABLE('UIEX6','frmMAIN','cmdDelete','',GUIERRORS,GUISTATE)
0718   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0719   CALL ATGUIDISABLE('UIEX6','frmMAIN','Menu','mnuDelete',GUIERRORS,GUISTATE)
0720   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0721   CALL ATGUIDISABLE('UIEX6','frmMAIN','cmdSave','',GUIERRORS,GUISTATE)
0722   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0723   CALL ATGUIDISABLE('UIEX6','frmMAIN','Menu','mnuSave',GUIERRORS,GUISTATE)
0724   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0725   CALL ATGUIACTIVATE('UIEX6','frmMAIN','txtCustID',GUIERRORS,GUISTATE)
0726   IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0727 END
0728 RETURN

```

```
0729 *
0730 *
0731 *****
0732 *
0733 * The DELETE.RECORD subroutine confirms that the user intends to delete the
0734 * record, then calls the UIEX.CUST.DELETE subroutine to delete the record
0735 * from the data and index files.
0736 *
0737 DELETE.RECORD: *
0738 *
0739 IF CUST.ID EQ '' THEN
0740     CALL ATGUIMSGBOX('Missing customer ID - deletion
cancelled!', 'Warning', 2, 0, 0, RSP, GUIERRORS, GUISTATE)
0741 END ELSE
0742     CALL ATGUIMSGBOX("Are you sure you want to delete this record?", "Verify
Action", 3, 4, 0, RSP, GUIERRORS, GUISTATE)
0743     IF GUIERRORS<1> > 1 THEN CALL ATGUISHUTDOWN; STOP
0744     IF RSP = 6 THEN
0745         CALL UIEX.CUST.DELETE(CUST.ID, FN.CUST, FN.CUST.XREF)
0746         CUST.ID = ''
0747         MAT CUST.REC = ''
0748         CALL ATGUIRESET('UIEX6', 'frmMAIN', GUIERRORS, GUISTATE)
0749         IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0750         CALL ATGUIDISABLE('UIEX6', 'frmMAIN', 'cmdDelete', '', GUIERRORS, GUISTATE)
0751         IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0752         CALL ATGUIDISABLE('UIEX6', 'frmMAIN', 'Menu', 'mnuDelete', GUIERRORS, GUISTATE)
0753         IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0754         CALL ATGUIDISABLE('UIEX6', 'frmMAIN', 'cmdSave', '', GUIERRORS, GUISTATE)
0755         IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0756         CALL ATGUIDISABLE('UIEX6', 'frmMAIN', 'Menu', 'mnuSave', GUIERRORS, GUISTATE)
0757         IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0758         CALL ATGUIACTIVATE('UIEX6', 'frmMAIN', 'txtCustID', GUIERRORS, GUISTATE)
0759         IF GUIERRORS<1> > 2 THEN CALL ATGUISHUTDOWN; STOP
0760     END
0761 END
0762 RETURN
0763 *
0764 *
0765 *****
0766 *
0767 * The SEARCH subroutine calls the UIEX.GET.CUST.IDX.LIST subroutine to
0768 * build a list of matching records, loads the list into the search dialog
0769 * list box and displays the search dialog box.
0770 *
0771 SEARCH: *
0772 *
0773 * Get the cross-reference list
0774 XREF.LIST = ''
0775 CALL UIEX.GET.CUST.IDX.LIST(XREF.LIST, SRCH, FN.CUST, FN.CUST.XREF)
0776 *
0777 * Initialize the search form
0778 CALL ATGUIRESET('UIEX6', 'frmXREF', GUIERRORS, GUISTATE)
0779 CALL
ATGUISETPROP('UIEX6', 'frmXREF', 'lstXref', GPITEMS, 0, 0, XREF.LIST, GUIERRORS, GUISTATE)
0780 *
0781 * Display the search dialog box
0782 CALL ATGUISHOW('UIEX6', 'frmXREF', '', '', GUIERRORS, GUISTATE)
0783 *
0784 RETURN
0785 *
0786 *
```

```
0787 *****
0788 *
0789 * The ASKSAVECHANGES subroutine checks if any data on the GUI form has
0790 * been changed, and if so, asks the user if he wants to save changes. If
0791 * the response is negative, this routine returns with OK = FALSE.
0792 *
0793 ASKSAVECHANGES: *
0794 *
0795 OK = TRUE
0796 CALL ATGUIGETPROP('UIEX6','frmMAIN','',GPCHANGED,0,0,CHANGED,GUIERRORS,GUISTATE)
0797 IF GUIERRORS<1> > 2 THEN CALL ATGUIHUTDOWN; STOP
0798 IF CHANGED THEN
0799     CALL ATGUIMSGBOX('Do you want to save your changes?','Data
0800     Changed',3,3,0,RSP,GUIERRORS,GUISTATE)
0801     IF GUIERRORS<1> > 1 THEN CALL ATGUIHUTDOWN; STOP
0802     BEGIN CASE
0803     CASE RSP EQ 6
0804         * save changes
0805         GOSUB UPDATE.RECORD
0806     CASE RSP EQ 7
0807         * discard changes
0808     CASE 1
0809         * cancel action
0810         OK = FALSE
0811         RETURN
0812     END CASE
0813 END
0814 RETURN
0815 *
0816 *****
0817 *
0818 * The CLOSEIT subroutine hides the main form, then if the number of
0819 * visible forms is zero (it should be), the application object is deleted.
0820 * This causes the GUI runtime to return the Quit event, which terminates
0821 * the event loop, and the program ends.
0822 *
0823 CLOSEIT: *
0824 *
0825 CALL ATGUIHIDE('UIEX6','frmMAIN','',GUIERRORS,GUISTATE)
0826 IF GUIERRORS<1> >= 2 THEN GOTO GUI.ERROR
0827 CALL ATGUIGETPROP(GUIAPP','',GPSTATUS,0,0,NUM.FORMS,GUIERRORS,GUISTATE)
0828 IF GUIERRORS<1> >= 2 THEN GOTO GUI.ERROR
0829 IF NUM.FORMS = 0 THEN
0830     CALL ATGUIDELETE('UIEX6','',GUIERRORS,GUISTATE)
0831     IF GUIERRORS<1> >= 3 THEN GOTO GUI.ERROR
0832 END
0833 RETURN
0834 *
0835 END
0836
```