```
/****/
/**** Hard-set maximums */
/****/

#define         MaxCol                  255                     /* Maximum columns */
#define         MaxRow                  255                     /* Maximum rows */
#define         MaxHand                 16                      /* Maximum handlers */
#define         SizerSlush              3                       /* Slush area around sizers */

/****/
/**** Font specification */
/****/

typedef struct
{
     int                                font;                   /* Font */
     int                                size;                   /* Size */
     int                                face;                   /* Face */
} ListFontSpec;

/****/
/**** Small Font specification */
/****/

typedef struct
{
     unsigned char                      change;                 /* Change */
     unsigned char                      font;                   /* Font */
     unsigned char                      size;                   /* Size */
     unsigned char                      face;                   /* Face */
} SmallListFontSpec;

/****/
/**** List creation request structure */
/****/

typedef struct
{
     Rect                               visBox;                 /* Box */

     int                                xAddative;              /* Addition to X length */
     int                                yAddative;              /* Addition to Y length */
```

```
    int                         frame;                  /* Frame */
    int                         drawIt;                 /* Draw flag */
    ListFontSpec                font;                   /* Font specs */
    int                         boxFlags;               /* Boxflags */

    int                         columnWidth[MaxCol];    /* Width of column */
    int                         rowWidth[MaxRow];       /* Width of row */

    int                         XScroll;                /* X Scroll */
    int                         YScroll;                /* Y Scroll */

    int                         dataXMax;               /* X Amount of cells */
    int                         dataYMax;               /* Y Amount of cells */

    char                        selFlags;               /* Selection flags */
    char                        listFlags;              /* Scrolling flags */
} ListCreationReq;

/****/
/**** Handler definition */
/****/

typedef struct
{
    void                        (*handler)();           /* Handler routine */
} Handler;

/****/
/**** Cell structure */
/****/

typedef struct
{
    SmallListFontSpec           fontSpecs;              /* Font specifics */
    int                         boxFlags;               /* Boxing flags */
    unsigned char               handler;                /* Handler */
    unsigned char               selected;               /* Selected flag */
    unsigned char               dataLocal;              /* Local creation flag */
    Handle                      data;                   /* Data */
} NewCell,*NewCellPtr;
```

```
/****/
/**** List structure */
/****/

typedef struct
{
     Rect                visBox;                 /* Box */
     int                 visBoxXLength;          /* X Length of box */
     int                 visBoxYLength;          /* Y Length of box */
     int                 visXLength;             /* X Length in pixels */
     int                 visYLength;             /* Y Length in pixels */

     int                 xAddative;              /* Addition to X length */
     int                 yAddative;              /* Addition to Y length */

     int                 frameX;                 /* Frame width */
     int                 frameY;                 /* Frame height */

     int                 drawIt;                 /* Draw flag */

     ListFontSpec        fontSpecs;              /* Font specs */
     int                 boxFlags;               /* Boxflags */

     int                 columnWidth[MaxCol];    /* Width of column */
     int                 columnStart[MaxCol];    /* Start of the column */
     unsigned char       columnSel[MaxCol];      /* Column selected flag */
     int                 columnFil[MaxCol];      /* Column filter */

     int                 rowWidth[MaxRow];       /* Width of row */
     int                 rowStart[MaxRow];       /* Start of the column */
     unsigned char       rowSel[MaxCol];         /* Row selected flag */

     void                (*frameDraw)();         /* Frame drawing routine */

     ControlHandle       XScroll;                /* X Scroll */
     ControlHandle       YScroll;                /* Y Scroll */

     Boolean   active;                           /* Active */
```

```
    Handler    handler[MaxHand];       /* Handlers */

    Point                              firstCell;                  /* First cell visible */
    Point                              lastCell;                   /* Last cell visible */
    Rect                               visArea;                    /* Visible area */

    int                                dataXMax;                   /* X Amount of cells */
    int                                dataYMax;                   /* Y Amount of cells */
    Handle                             cells;                      /* Cell data */

    char                               selFlags;                   /* Selection flags */
    char                               listFlags;                  /* Scrolling flags */

    long                               lastClickTime;              /* Last click time */
    Point                              lastClickPos;               /* Last click position */

    char                               autoScrollMethod;           /* Auto scroll method */

    TEHandle cellEditText;             /* TextEdit handle to cell */
    Point                              cellEdit;                   /* Cell being edited */
} NewList,*NewListPtr,**NewListHandle;

/****/
/**** Routines */
/****/

    /**** Allocation */

NewListHandle ListNew(Rect *vBox,Rect *dBox,Point cellSize,
    int proc,WindowPtr window,int draw,int hasGrow,int horiz,int vert);
void ListDispose(NewListHandle nlh);

    /**** Action */

int ListPart(Point pt,Point *cell,NewListHandle nlh);
int ListCick(Point lpt,int mod,NewListHandle nlh);
int ListCickEnhanced(Point lpt,int mod,NewListHandle nlh,void (*clikLoop)());
pascal void ListTrackScrollVertical(ControlHandle cont,int part);
pascal void ListTrackScrollHorizontal(ControlHandle cont,int part);
void ListCellSizer(Rect dragArea,int part,Point cell,NewListHandle nlh);
void ListEmulatorDrag(int message,Point *oldCell,Point *newCell,
    NewListHandle nlh,int dataXMax,int dataYMax,char selFlags,int mod);
```

```
void ListSingleDrag(int message,Point *oldCell,Point *newCell,
      NewListHandle nlh,int dataXMax,int dataYMax,char selFlags,int mod);
void ListLineDrag(int message,Point *oldCell,Point *newCell,
      NewListHandle nlh,int dataXMax,int dataYMax,char selFlags,int mod);

      /**** Update */

void ListDraw(Point cell,NewListHandle nlh);
void ListInvert(Point cell,NewListHandle nlh);
void ListDrawCellReal(Rect *visBox,Point cell,NewCell *theCell,NewListPtr nlp,Rect *realVisBox,Boolean active);
void ListCalculate(NewListHandle nlh);
void ListUpdate(NewListHandle nlh,RgnHandle updRgn);
void ListUpdateWhole(NewListHandle nlh);
void ListScroll(int scrollX,int scrollY,NewListHandle nlh);
void ListInvertCell(Rect *visBox,Point cell,NewCell *theCell,NewListPtr nlp,Rect *realVisBox);

      /**** Poll */

void ListGetVisible(NewListHandle nlh,Rect *visArea);
int ListDrawStatus(NewListHandle nlh);
int ListGetCellAt(Point pt,NewListHandle nlh,Point *newCell);
int ListValidCell(Point cell,NewListHandle nlh);
int ListGetSelect(Boolean next,Point *cell,NewListHandle nlh);
void ListGetDataMax(int *xmax,int *ymax,NewListHandle nlh);
int ListCalTextWidth(Point cell,NewListHandle nlh);
Boolean ListCellEmpty(Point cell,NewListHandle nlh);
void ListFind(int *offset,int *len,Point cell,NewListHandle nlh);
void ListRect(Rect *cellRect,Point cell,NewListHandle nlh);
void ListClrCell(Point cell,NewListHandle nlh);
void ListGetCell(unsigned char *dataPtr,int *dataLen,Point cell,NewListHandle nlh);
Point ListLastClick(NewListHandle nlh);
void ListVisRect(Rect *visRect,Rect *destRect,Point cell,NewListHandle nlh);
Boolean ListIsActive(NewListHandle nlh);

      /**** Set */

void ListCleanList(NewListHandle nlh);
void ListSetBox(int boxFlags,Point cell,NewListHandle nlh);
void ListSetHandler(int handler,Point cell,NewListHandle nlh);
void ListSetRowWidth(int rowNum,int rowWidth,NewListHandle nlh);
void ListSetColumnWidth(int colNum,int colWidth,NewListHandle nlh);
void ListSetCell(unsigned char *dataPtr,int dataLen,Point cell,NewListHandle nlh);
void ListDoDraw(Boolean drawIt,NewListHandle nlh);
void ListCellFont(Point cell,int font,int size,int bold,NewListHandle nlh);
void ListSetGlobalFont(int font,int size,int face,NewListHandle nlh);
void ListSetGlobalBox(int boxFlags,NewListHandle nlh);
void ListSetAddative(int xAddative,int yAddative,NewListHandle nlh);
void ListSetSelect(Boolean status,Point cell,NewListHandle nlh);
```

```
void ListFlipSelect(Point cell,NewListHandle nlh);
void ListEstablishHandler(int handlerNum,void (*handler)(),NewListHandle nlh);
void ListSetData(Handle hand,Point cell,NewListHandle nlh);
```

```
void ListSetSelectionFlags(int selFlags,NewListHandle nlh);
void ListActivate(Boolean active,NewListHandle nlh);
void ListSetFrameWidth(int width,int height,NewListHandle nlh);
```

```
void ListSetFrameDrawer(void (*frameDraw)(),NewListHandle nlh);
void ListSetScrollMethod(int method,NewListHandle nlh);
void ListSetKeyFilter(int filter,int column,NewListHandle nlh);

	/**** Add/Remove Columns/Rows */

int ListAddColumn(int cols,int colNum,NewListHandle nlh);
void ListDelColumn(int cols,int colNum,NewListHandle nlh);
int ListAddRow(int rows,int rowStart,NewListHandle nlh);
void ListDelRow(int rows,int rowStart,NewListHandle nlh);

	/**** editText */

void ListEditStart(Point cell,NewListHandle nlh);
void ListEditStop(NewListHandle nlh);
void ListEditIdle(NewListHandle nlh);
void ListEditClick(Point lpt,Boolean extend,NewListHandle nlh);
void ListEditRect(NewListHandle nlh);
void ListEditKey(char ch,NewListHandle nlh);
void ListEditActivate(Boolean active,NewListHandle nlh);
TEHandle ListEditRec(NewListHandle nlh);

	/**** Miscellaneous */

void ListResetScrollBars(NewListHandle nlh);
void ListCopyRect(Rect *out,Rect *in);
void ListCopyBitString(unsigned char *inStr,unsigned char *outStr,long length);
void ListGetFont(ListFontSpec *spec);
void ListSetFont(ListFontSpec spec);
int ListAbs(int val);
void ListConstrain(Point *pt,Rect *constrain);
void ListCopyString(char *inStr,char *outStr);
void ListSetEditRect(Rect *vBox,Rect *dBox,TEHandle teRec);
Boolean ListCharacterFilter(char ch,int filter);
void ListSetDefaultType(TEHandle text,TextStyle style);

/****/
/**** Externals (how can I get around doing this?) */
/****/

#ifdef MAIN
	NewListHandle			outsideNLH;			/* Used during tracking */
#else
	extern NewListHandle		outsideNLH;			/* Used during tracking */
#endif

/****/
/**** Boxing flags */
```

```
/****/

#define     ListLeft            0x0001          /* Left line */
#define     ListRight           0x0002          /* Right line */
#define     ListTop             0x0004          /* Top line */
#define     ListBottom          0x0008          /* Bottom line */
#define     ListGray            0x0010          /* use gray pattern */
#define     ListDouble          0x0020          /* Double the pixel width */
```

```
/****/
/**** Handlers */
/****/

#define         DrawStringHandler       0                       /* use DrawString */
#define         TextBoxHandler          1                       /* use TextBox */
#define         TEUpdateHandler         2                       /* use TEUpdate */

/****/
/**** List Parts */
/****/

#define         inCell                  1                       /* In the cell area */
#define         inHorizScroll           2                       /* In the horizontal scroll */
#define         inVertScroll            3                       /* In the vertical scroll */
#define         inHorizSizer            4                       /* In the horizontal sizer */
#define         inVertSizer             5                       /* In the vertical sizer */
#define         inHorizFrame            6                       /* In the horizontal frame */
#define         inVertFrame             7                       /* In the vertical frame */
#define         inCornerFrame           8                       /* In the corner frame */
#define         inTextEditCell          9                       /* In the cell being edited */

/****/
/**** Messages to clikLoop userRoutines */
/****/

#define         startClikLoop           0                       /* Startup the clikLoop */
#define         newCellLocation         1                       /* New cell */

/****/
/**** Messages to frame drawer */
/****/

#define         HorizCell               0                       /* Horizontal cell */
#define         VertCell                1                       /* Vertical cell */

/****/
/**** Auto scrolling methods */
/****/

#define         cellMethod              0                       /* Scroll by cells */
#define         incMethod               1                       /* Scroll by pixels */

/****/
/**** Column key filters */
/****/

#define         noReturn                0x0001                  /* No return */
```

```
#define     noSpace         0x0002          /* No spaces */
#define     noCapital       0x0004          /* No capitals */
#define     noLowerCase     0x0008          /* No lower case */
#define     noPunc          0x0020          /* No punctuation */
```

```
#define     numberFilter        (noReturn|noSpace|noLowerCase|noCapital|noPunc)
#define     noLetter            (noLowerCase|noCapital)
```