

Dialog Utilities

```

/*
 * Dialog Utilities
 */

// Assumes inclusion of <MacHeaders>

/* Return topLeft point to center standard file dialogs. */
Point DU_StdPutWhere(void);
Point DU_StdGetWhere(void);

/*
 * Pre-load and center ALERT/DLOG template resources. Changes do not affect
 * the resource file, but if the ALERT/DLOG is created soon after the call
 * it will use the modified (centered) template in memory. Both routines
 * return the passed rsrc id to allow calls like:
 *     Alert(DU_CenterALRT(alertID),filter);
 *     GetNewDialog(DU_CenterDLOG(dlogID),dStorage,behindWindow);
 */
short DU_CenterALRT(short rsrcId);
short DU_CenterDLOG(short rsrcId);

static Rect DU_MouseGDRect(void) {
/* Returns the gdRect of the screen device the mouse is currently in. */
    Point      p;
    Rect       r = screenBits.bounds;
    GDHandle   curDevice = GetDeviceList();

    GetMouse(&p);

    while(curDevice){
    if (TestDeviceAttribute(curDevice,screenDevice) &&
        TestDeviceAttribute(curDevice,screenActive) &&
        PtInRect(p,&(*curDevice)->gdRect))
        r = (*curDevice)->gdRect;
        curDevice = GetNextDevice(curDevice);
    }
    return r;
}

static void DU_CenterRect(Rect* rect_p){
/*
 * Aligns *rect_p with screenBits.bounds - LR centered & with 1/3 of the
 * empty space above the rect.
 */
    Rect bRect = DU_MouseGDRect();

    bRect.top += GetMBarHeight();

    /* exactly centered */
    OffsetRect(rect_p, (bRect.right + bRect.left)/2 - (rect_p->right +
        rect_p->left)/2, (bRect.top + bRect.bottom)/2 -
        (rect_p->top + rect_p->bottom)/2);
}

```

```
/* 1/2 empty space above -> 1/3 empty space above */
OffsetRect(rect_p,0,-(rect_p->top - bRect.top)/3);
}

static Point DU_Where(short rsrcId){
/*
* Returns centering point for the topLeft corner of a dialog.
*/
    Handle h = GetResource('DLOG',rsrcId);
    Rect      r = {0,0,0,0};
    if(h){
        r = *((Rect*)(*h));
        DU_CenterRect(&r);
        ReleaseResource(h);
    }
    return topLeft(r);
}

Point DU_StdPutWhere(void){
    return DU_Where(putDlgID);
}

Point DU_StdGetWhere(void){
    return DU_Where(getDlgID);
}

static short DU_Center(ResType type, short rsrcId){
/*
* Reads in an 'ALRT' or 'DLOG' rsrc template & centers its display Rect.
*/
    Handle h = GetResource(type,rsrcId);

    if(h)
        DU_CenterRect((Rect*)(*h));
    return rsrcId;
}

short DU_CenterALRT(short rsrcId){
    return DU_Center('ALRT',rsrcId);
}

short DU_CenterDLOG(short rsrcId){
    return DU_Center('DLOG',rsrcId);
}
```