

```
//#####
// FILENAME: WidePopupManager.h
// SUMMARY: Header for an AndrewStone™ PopUp Hack
// SUPERCLASS: Object:WidePopupManager
// PROTOCOLS: None
// INTERFACE: None
// AUTHOR: Andrew Stone & Charles Perkins
//#####
// Description
// > 2) How can I make the pop-up list resize to the longest string in the
// > list? I have tried various combinations of sizeToFit to both the
// > pop-up list and the button, but none give the desired results.
//
// Charles Perkins and I wrote a hack to do this [IE, when it's not popped,
// it's a certain fixed size in your UI, but when it's popped, it sizes to
// the widest entry] by creating a class which goes between the button and
// the pop up list as manager, and munges the frame of the Menu before it
// shows itself.
//
// Here's how you'd invoke it:
//
// - initializeFindButton:theButton
// {
//     NXAttachPopUpList(theButton,findPopUp);
//     [[WidePopupManager allocFromZone:[theButton
// zone]]initForButton:theButton andList:findPopUp];
//     .....
// }
//#####
// History
// 09/27/94: Snarfed from mlist.next-programmer archives.
//#####
// Imported Interfaces
//
// #import <appkit/appkit.h>
//
//*****
// Copyright (c) 1989,1990 Stone Design Corp. All rights reserved.
// programmer: Andrew C. Stone/ Charles L. Perkins
// Date: Feb 2 1992
// Purpose: To fool a popup into being wider than normal as needed
//*****/

@interface WidePopupManager:Object {
    id button;
    id popupList;
}
```



```
- window
{
    return [button window];
}

- display
{
    return [button display];
}

- (BOOL)needsDisplay
{
    return [button needsDisplay];
}

- (const char *)title
{
    return [button title];
}

- setTitle:(const char *)aString
{
    [button setTitle: aString];
    return self;
}
```

@end