

Appsprites is a demonstration program that shows how to hook Tony Myles' SpriteWorld 1.0.b3 package into Bowers Development Corporation's AppMaker 1.5.2 under THINK C 5.0.4. The simple. $\pi$  SpriteWorld project was modified to provide basic functions for animation. Animation calls were then inserted into two critical AppMaker-generated routines (Dispatcher.c and MainWindow.c) and the AppMaker window record (in Globals.h) was extended to allow independent animation in each main window, although only one window is active at a time. A modal dialog was added that allows the user to set each sprite's move and frame time (the values must be the same in the demo) either from sprite 1 or independently for each sprite (Figure 1).

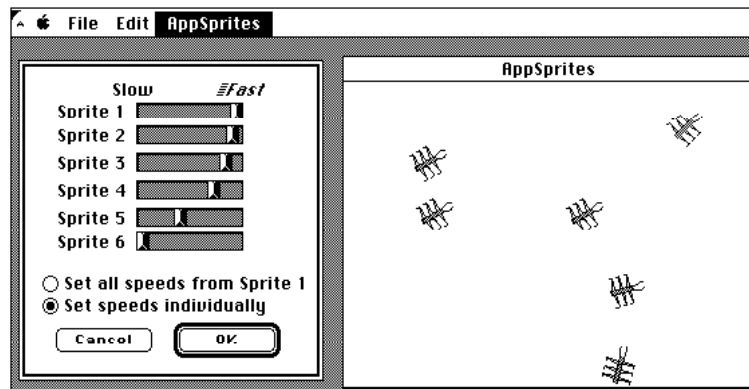


Figure 1. Appsprites screen showing dialog

AppSprites is only intended to be a starter program. Some things could be cleaned up in the package, such as the machine checking code (which could use the gestalt checking done by AppMaker). Appsprites is not multi-finder friendly, using a version of AppMaker's EventLoop.c that disables WaitNextEvent in favor of GetNextEvent to drive the animation with idle events (Figure 2).

```

/*-----*/
static Boolean GetEvent (void);
static Boolean GetEvent (void)
{
    Boolean    gotEvent;

    //    if (sysConfig.hasWNE) {
    //        gotEvent = WaitNextEvent (everyEvent, &curEvent,
    //                                GetSleep (), cursorRgn);
    //    } else {

```

```
SystemTask ();
gotEvent = GetNextEvent (everyEvent, &curEvent);
//    }
return (gotEvent);
} /*GetEvent*/
```

Replace the GetEvent() function in AppMaker's EventLoop.c with the one given here and recompile the AMLibraryC.π library before compiling Appsprites. ***If WaitNextEvent is used, the animation speed will be the rate of caret blinking — which is really slow!*** This can be changed by setting the sleep time to a smaller interval (replace the call to GetSleep() with a small number), or by implementing a function of your own.

Driving animation with idle events is slower than driving it with the tight loop in simple.π, but is still reasonably fast. How fast? Fast enough. As some of you may have observed, the sprites look remarkably like Stiquito robots. This is because AppSprites is the template for a Stiquito simulator based on SpriteWorld (you can find out more about Stiquito, and obtain the simulator and its source code when it is available, by anonymous ftp to cs.indiana.edu in /pub/stiquito).

Thanks *extremely* to Tony Myles for SpriteWorld, Bowers Development Company for AppMaker, and Symantec for THINK C 5.0.4. They work. 'Nuff said.

July 10, 1993

Jonathan W. Mills