

BubbleMon and WMFishTime

Aquatic Utilities

Although most users appreciate some eye-candy on their desktops, they may feel a little guilty about wasting resources. Rather than being just colorful gimmicks BubbleMon and WMFishTime provide useful information. **BY ANDREA MÜLLER**

Although desktop environments provide all kinds of information systems, more simple window managers often leave their users to discover a suitable application themselves. As nearly everybody needs a desktop clock, and because a graphical system monitor is quite a useful gimmick too, you may find the sheer mass of applications of this type quite irritating. The DockApps [1], *WMFishTime* and *BubbleMon* [2], stand out from the crowd and fulfill this admittedly trivial task in an extremely creative way.

If your distribution does not include the packages, you will need to compile the sources before you start installing.

Installation

If you intend to use these tools, you will need a window manager that supports dock apps. Besides WindowMaker [3], managers such as *pwm* [4] or *waimea* [5] are potential candidates.

You can run the *make* and *make install* commands to compile and install the program. You will need root privileges to launch *make install*. As neither program provides a *configure* script, you

deskTOPia

Only you can decide how your desktop looks. With deskTOPia we regularly take you with us on a journey into the land of window managers and desktop environments, presenting the useful and the colorful viewers and pretty toys.



might prefer to edit the *Makefile* before you compile *WMFishTime* to install the program to `/usr/local/bin` instead of `/usr/X11R6/bin`. To do so, launch your favorite editor, open the *Makefile* and replace the following line:

```
PREFIX = $(DESTDIR)/usr/X11R6
```

with:

```
PREFIX = /usr/local
```

And Now for Something Completely Different...

Invoking *wmfishtime* & will place an analog clock on your desktop. But this clock is completely different from any you have seen before, in fact it is a virtual aquarium (Figure 1), although the maritime landscape does tend to distract you from the clock-hands in the foreground.

If you get bored of watching the seascape,

pointing at the fish with your mouse makes them scatter in all directions. As soon as you drag the pointer across the *WMFishTime* screen, the timid aquatic creatures swim for their lives and don't come back until the danger has passed. As this toy has little value apart from the time and date display at the bottom, the program's author also provides a message notification function that you can enable as follows:

```
wmfishtime -c &
```

This syntax tells *WMFishTime* to monitor the file that your *MAIL* environment variable points at. To find out what this is, type

GLOSSARY

System Load: This value is calculated by measuring the number of processes waiting for the CPU or hard disk within a specific interval. The system load on desktop systems is typically less than 2.



Figure 2:
WMFishTime
monitors
incoming mail



Figure 1:
WMFishTime
runtime

```
echo $MAIL
```

Most distributors set the variable to a meaningful value. When a new message arrives, *WMFishTime* waves the seaweed at the bottom of its window to let you know (Figure 2).

The waving seaweed disappears after opening the message with your preferred mail client to reveal the date display once more. However, using a mail program rather than *fetchmail* to collect your messages does cause a slight hitch, as *WMFishTime* will only notify you in case of incoming system messages.

Duck or Grouse?

Launching *BubbleMon* allows you to monitor your system's load in addition to displaying the time and indicating incoming messages. You might be wondering what on earth a duckling swimming back and forth on a pond has to do with the system load, when you launch the tool for the first time (Figure 3).

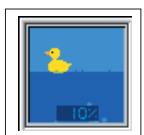


Figure 3: BubbleMon in Action

The answer is quite a lot, assuming you use your powers of observation in

Box 1: Aquariums and Duck Ponds For Ever

Of course, you will not want to launch these useful tools manually every time you reboot. An entry in the *.xinitrc* file or – depending on your configuration – in *.xsession* in your home directory will take care of that. SuSE 8.1 users can simply add both commands to *.xinitrc* below the line that says “# Add your own lines here...”:

```
bubblemon wakwak.sh &
wmfishtime -c &
```

It is even easier if you intend to run both applications on WindowMaker. Simply click on the window frame with the center mouse button and drag the window to the right to dock it. Now right-click on the window frame to open the drop-down menu and select *Settings*. Then use the configuration menu that then appears enter the required program syntax in the *Application path and arguments* fields, check the *Start when Window Maker is started* checkbox and then click on *OK* to confirm. WindowMaker will automatically launch the DockApps in future.

true field biologist style. The water level changes while you work, and the bubbles that stir up the surface also mean something. As you may have guessed the water level indicates the amount of memory in use on your system. If the water level keeps rising, you can assume that your machine is running a real memory hog. And the more bubbles displayed, the harder your CPU is working. Completely calm water indicates that your processor is more or less idle. If you start a task that requires a fair amount of processing power, the water will start to bubble to reflect the current CPU load. The percentage shown in the center of the window provides an additional CPU load display.

Beneath the Surface

Although this gives you a rough idea of how healthy your system is, there are times when you need to know more. Hiding behind the idyllic landscape there are two info-screens that overlay the foreground like holograms. When you drag the mouse pointer over the *BubbleMon* window, a monitor appears showing the **System Load** in numerical format at the top with a histogram below it (Figure 4).

Holding down the [Shift] key while dragging the mouse into the window displays a memory monitor with a histogram at the bottom and statistics for the working and swap memory in kilobytes and per cent at the top (Figure 5). To display the same statistics in megabytes, call *BubbleMon* with the *-k* flag instead.

The author's even provide for cases where you need to investigate the statistics more closely. Right-clicking the window will freeze the display. To unfreeze the display simply drag the mouse into and back out of the window. And if you do not require detailed information, simply invoke:

```
bubblemon -c -m &
```

to disable the display. The *-c* flag disables the

system load monitor, and *-m* the memory display. And if the duckling is too silly for your liking, you can always use the *-d* (for “duck”) flag to disable it.

Helper

In addition to the features already discussed, the program's author conveniently provides a function for launching an arbitrary application when you click on the *BubbleMon* with the center mouse key. To use this function, simply add the program you want to launch as the last parameter when invoking *BubbleMon*. An example should clarify this:

```
bubblemon -k sylpheed &
"xterm -e vim"
```

does the following. Left-clicking on the *BubbleMon* window, launches the *sylpheed* mail client, the center button calls an *xterm* window and launches the *vim* editor. As the second command contains space characters, it needs to be placed in quotes

What use is a duck that can't quack? Again the program's author has thought of that: if you take a look at the */misc* subdirectory in the *BubbleMon* source code, you will find a file called *wak.wav* and a shell script with the onomatopoeic name *wakwak.sh*. The latter calls the *play* command from the *sox* package that you may need to install. The following command:

```
chmod +x wakwak.sh
```

makes the script executable, and you can use your favorite editor to add the path to *wak.wav*. Now launch *BubbleMon* with the *wakwak.sh* parameter set to drive your neighbors mad by making your computer quack whenever you click *BubbleMon*.



Figure 4: BubbleMon's system load display



Figure 5: BubbleMon's memory monitor

INFO

- [1] Andreas Kneib: “Desktop Designer”, Linux Magazine, Issue 30, p 30
- [2] BubbleMon & WMFishTime: <http://www.ne.jp/asahi/linux/timecop/>
- [3] WindowMaker: <http://www.windowmaker.org>
- [4] PWM: <http://modeemi.cs.tut.fi/~tuomov/pwm/>
- [5] Waimea: <http://www.waimea.org/>