
jwm on Mint 18

jwm is a light-weight window manager for the X11 Window System. jwm is written in C and uses only Xlib at a minimum. Although jwm does not require them, it will take advantage of a number of libraries, for example libpng, libjpeg and libxpm to enhance its functionality. A full list is on <http://joewing.net/projects/jwm>.

In this example, we start with a working install of Linux Mint 18 MATE desktop. This will ensure that the basics of the X Window system (Xorg) are set up.

Initial Install and Setup.

First, we need to install jwm and some matching light weight applications. Use the Mint package manager to install the following ..

```
jwm      - the jwm window manager
xfe      - a file manager
rxvt     - a console terminal for X
qupzilla - a lightweight web browser
leafpad  - a basic text editor.
```

When these packages are installed, copy `/etc/jwm/system.jwmrc` to your home directory and rename it `.jwmrc`. As jwm will look in the users' home directory for its configuration file (`.jwmrc`) we can edit this file to add or change menu items and/or characteristics of the jwm desktop. So we will start out with the default configuration and modify to suit our requirements.

But first, a few words about the `.jwmrc` file format, which is XML.

XML stores data in a plain text format. This provides a software- and hardware-independent way of storing, transporting, and sharing data. An XML file contains tags, which are created by the author – there are no predefined tags in contrast to HTML. The data is enclosed in tags, and the structure of the tags and the data is determined by the author of the XML file.

To use the data in an XML file, there must be some software written to interpret and display it – in this case the software is the jwm window manager. As an example, consider the following XML snippet ..

```
<Nslig Meeting>
  <Date>18-July-2016</Date>
  <Start_Time>6.30 pm</Start_Time>
</Nslig Meeting>
```

This snippet has data which defines the date and start time of an Nslig meeting. Note that the tags are matched, for example we need both `<Date>` and `</Date>` to define the data.

Having installed jwm and some useful lightweight programs, we can now log out, select jwm in the login manager screen, and log in again. The jwm window manager will start with the default settings determined by the distros' maintainers. Click anywhere on the screen background or on the icon on the left hand side of the bottom panel. Now click on the "terminal" entry in the menu displayed.

The first thing we notice is that the terminal doesn't work "out of the box" because xterm is not in the default Xorg (X Windows) configuration for many distros). To fix this you can install xterm, or better, rxvt. You can do this by typing the following command ..

```
$ sudo apt-get install rxvt
```

jwm on Mint 18

You can now edit the .jwmrc file in your home directory and change

```
<Program icon="terminal.png" label="Terminal">xterm</Program>
```

to something like

```
<Program icon="terminal.png" label="Console">rxvt</Program>
```

You will most probably not be satisfied with the default rxvt/xterm settings, so it's time to consult the man pages for options. For rxvt, some sane(?) options are to have a scroll bar on the right hand side, the number of scroll lines equal to 1000, and a font of 9x15. You can get a list of X Window fonts by using the command

```
$ xlsfonts | more
```

If we decide to use the above options, the command to launch rxvt becomes ..

```
<Program icon="terminal.png" label="Console">rxvt -sr -sl 1000 -fn 9x15</Program>
```

In a similar way, we can add extra lines following the above <Program> line to set up menu entries for the xfe file manager, the qupzilla web browser and the leafpad text editor.

So now the Root Menu section of .jwmrc will be something like ...

```
<!-- The root menu. -->
<RootMenu onroot="12">
  <Program icon="terminal.png" label="Console">rxvt -fn 9x15 -sr -sl 1000</Program>
  <Program icon="xfe.xpm" label="File Manager">xfe</Program>
  <Program icon="qupzilla.png" label="Web Browser">qupzilla</Program>
  <Program icon="text-editor.png" label="Text Editor">leafpad</Program>
  <Separator/>
  <Program icon="lock.png" label="Lock">
    xlock -mode blank
  </Program>
  <Separator/>
  <Restart label="Restart" icon="restart.png"/>
  <Exit label="Exit" confirm="true" icon="quit.png"/>
</RootMenu>
```

If we want to add a submenu to the Root Menu, we use the <Menu> tag within the <RootMenu> tag. For example, if we want to add a couple of menu items in a "Utilities" submenu, we could arrange it like ...

```
<Separator/>
  <Menu icon="folder.png" label="Utilities">
    <Program icon="calculator.png" label="Calculator">calcoo</Program>
    <Program icon="viewer.png" label="Image Viewer">mirage</Program>
  </Menu>
```

which would update the Root Menu as shown below ...

jwm on Mint 18

```
<!-- The root menu. -->
<RootMenu onroot="1,2">
  <Program icon="terminal.png" label="Console">rxvt -fn 9x15 -sr -sl 1000</Program>
  <Program icon="xfe.xpm" label="File Manager">xfe</Program>
  <Program icon="qupzilla.png" label="Web Browser">qupzilla</Program>
  <Program icon="text-editor.png" label="Text Editor">leafpad</Program>
  <Separator/>
  <Menu icon="folder.png" label="Utilities">
    <Program icon="calculator.png" label="Calculator">calcoo</Program>
    <Program icon="viewer.png" label="Image Viewer">mirage</Program>
  </Menu>
  <Separator/>
  <Program icon="lock.png" label="Lock">
    xlock -mode blank
  </Program>
  <Separator/>
  <Restart label="Restart" icon="restart.png"/>
  <Exit label="Exit" confirm="true" icon="quit.png"/>
</RootMenu>
```

and would look like this ...



A few miscellaneous observations ..

Comments can be included in an XML file by using the character combinations `<!--` and `-->` at the beginning and the end of the comment respectively.

When making changes to the `.jwmrc` file, use the command `jwm -p` to check for errors – no news is good news. If something is wrong, error messages appear.

You can define the paths to be searched for icons using one or more instances of the `<IconPath>` tag. Key bindings can also be defined with the `<Key>` tag.

The option `onroot="1,2"` in the opening `<RootMenu>` tag defines the active mouse buttons to be the left button and the middle button (usually the scroll wheel) respectively. To set an option to active the right button simply add a 3. For example, to set right click only use `onroot="3"`

jwm on Mint 18

Some general tags ...

- <FocusModel> - can be sloppy or click - default is sloppy. You may prefer click.
- <StartupCommand> - used to automatically start a program when logging in, eg/. A system monitor.
- <Desktops> - can be used to define the number of virtual desktops, and their backgrounds. The background can be default for all virtual desktops or different for each one.

There are numerous other options – see the man page for jwm and examples on the jwm web site for more details.