NAME

urxvtd − urxvt terminal daemon

SYNOPSIS

urxvtd [−q|−−quiet] [−o|−−opendisplay] [−f|−−fork] [−m|−−mlock] [−e|−−eval *perlstring*]

urxvtd −q −o −f # for .xsession use

DESCRIPTION

This manpage describes the urxvtd daemon, which is the same vt102 terminal emulator as urxvt, but runs as a daemon that can open multiple terminal windows within the same process.

You can run it from your X startup scripts, for example, although it is not dependent on a working DISPLAY and, in fact, can open windows on multiple X displays on the same time.

Advantages of running a urxvt daemon include faster creation time for terminal windows and a lot of saved memory.

The disadvantage is a possible impact on stability − if the main program crashes, all processes in the terminal windows are terminated. For example, as there is no way to cleanly react to abnormal connection closes, `xkill` and server resets/restarts will kill the **urxvtd** instance including all windows it has opened.

OPTIONS

urxvtd currently understands a few options only. Bundling of options is not yet supported.

−q, **−−quiet**

Normally, **urxvtd** outputs the message `rxvt-unicode daemon listening on <path>` after binding to its control socket. This option will suppress this message (errors and warnings will still be logged).

−o, **−−opendisplay**

This forces **urxvtd** to open a connection to the current `$DISPLAY` and keep it open.

This is useful if you want to bind an instance of **urxvtd** to the lifetime of a specific display/server. If the server does a reset, **urxvtd** will be killed automatically.

−f, **−−fork**

This makes **urxvtd** fork after it has bound itself to its control socket.

−m, **−−mlock**

This makes **urxvtd** call *mlockall*(2) on itself. This locks **urxvtd** in RAM and prevents it from being swapped out to disk, at the cost of consuming a lot more memory on most operating systems.

Note: In order to use this feature, your system administrator must have set your user's RLIMIT_MEMLOCK to a size greater than or equal to the size of the **urxvtd** binary (or to unlimited). See */etc/security/limits.conf*.

Note 2: There is a known bug in glibc (possibly fixed in 2.8 and later versions) where calloc returns non-zeroed memory when mlockall is in effect. If you experience crashes or other odd behaviour while using −−mlock, try it without it.

−e, **−−eval** *perlstring*

Evaluate the given perl code after basic initialisation (requires perl support to be enabled when compiling **urxvtd**).

This can be used for example to configure the internal perl interpreter, which is shared between all terminal instances, or create additional listening sockets for additional protocols.

The code is currently executed *before* creating the normal listening sockets: this might change in future versions.

EXAMPLES

This is a useful invocation of **urxvtd** in a *.xsession*−style script:

```
urxvtd -q -f -o
```

This waits till the control socket is available, opens the current display and forks into the background. When you log-out, the server is reset and **urxvtd** is killed.

ENVIRONMENT

RXVT_SOCKET

Both **urxvtc** and **urxvtd** use the environment variable *RXVT_SOCKET* to create a listening socket and to contact the urxvtd, respectively. If the variable is missing then *$HOME/.urxvt/urxvtd−<nodename>* is used.

DISPLAY

Only used when the `--opendisplay` option is specified. Must contain a valid X display name.

SEE ALSO

urxvt(7), urxvtc(1)