

NAME

AnyEvent::Impl::Tk – AnyEvent adaptor for Tk

SYNOPSIS

```
use AnyEvent;
use Tk;

# this module gets loaded automatically as required
```

DESCRIPTION

This module provides transparent support for AnyEvent. You don't have to do anything to make Tk work with AnyEvent except by loading Tk before creating the first AnyEvent watcher.

Tk is buggy. Tk is extremely buggy. Tk is so unbelievably buggy that for each bug reported and fixed, you get one new bug followed by reintroduction of the old bug in a later revision. It is also basically unmaintained: the maintainers are not even interested in improving the situation – reporting bugs is considered rude, and fixing bugs is considered changing holy code, so it's apparently better to leave it broken.

I regularly run out of words to describe how bad it really is.

To work around some of the many, many bugs in Tk that don't get fixed, this adaptor **dup()**'s all filehandles that get passed into its I/O watchers, so if you register a read and a write watcher for one fh, AnyEvent will create two additional file descriptors (and handles).

This creates a high overhead and is slow, but seems to work around most known bugs in Tk::fileevent on 32 bit architectures (Tk seems to be terminally broken on 64 bit, do not expect more than 10 or so watchers to work on 64 bit machines).

Do not expect these workarounds to avoid segfaults and crashes inside Tk.

Note also that Tk event ids wrap around after 2**32 or so events, which on my machine can happen within less than 12 hours, after which Tk will stomp on random other events and kill them. So don't run Tk programs for more than an hour or so.

To be able to access the Tk event loop, this module creates a main window and withdraws it immediately. This might cause flickering on some platforms, but Tk perversely requires a window to be able to wait for file handle readiness notifications. This window is always created (in this version of AnyEvent) and can be accessed as `$AnyEvent::Impl::Tk::mw`.

SEE ALSO

AnyEvent, Tk.

AUTHOR

Marc Lehmann <schmorp@schmorp.de>
<http://anyevent.schmorp.de>