

DBus

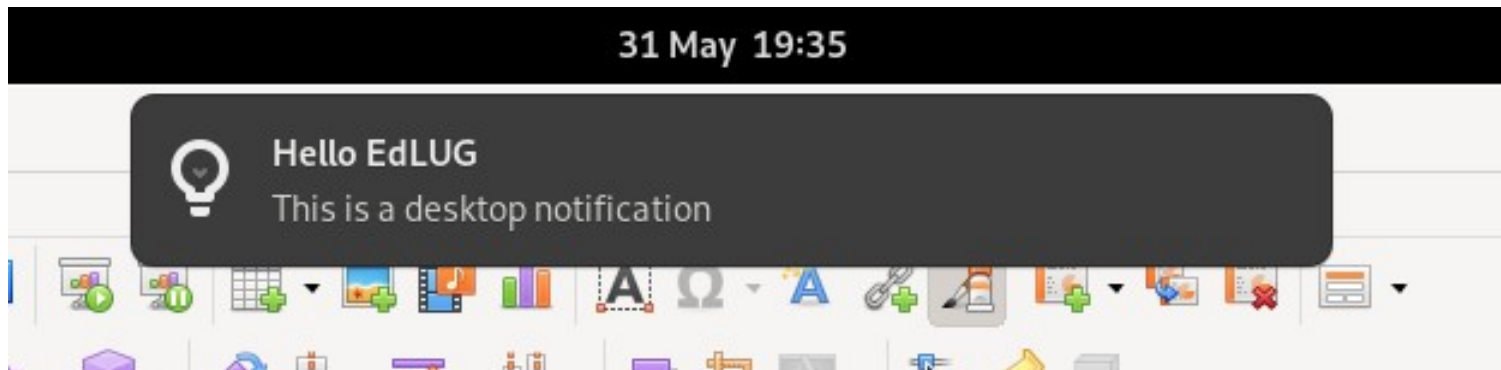
Communications for desktop apps

EdLUG, June 2021

Thomas Kluyver



Desktop notifications



`org.freedesktop.Notifications`

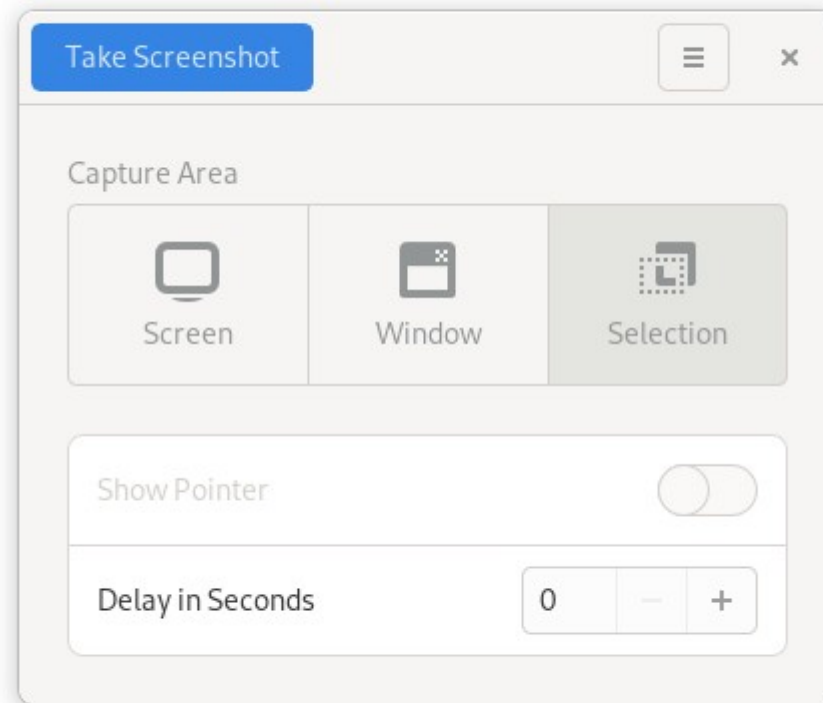
Screensaver

(or not)

org.freedesktop.ScreenSaver
org.gnome.ScreenSaver

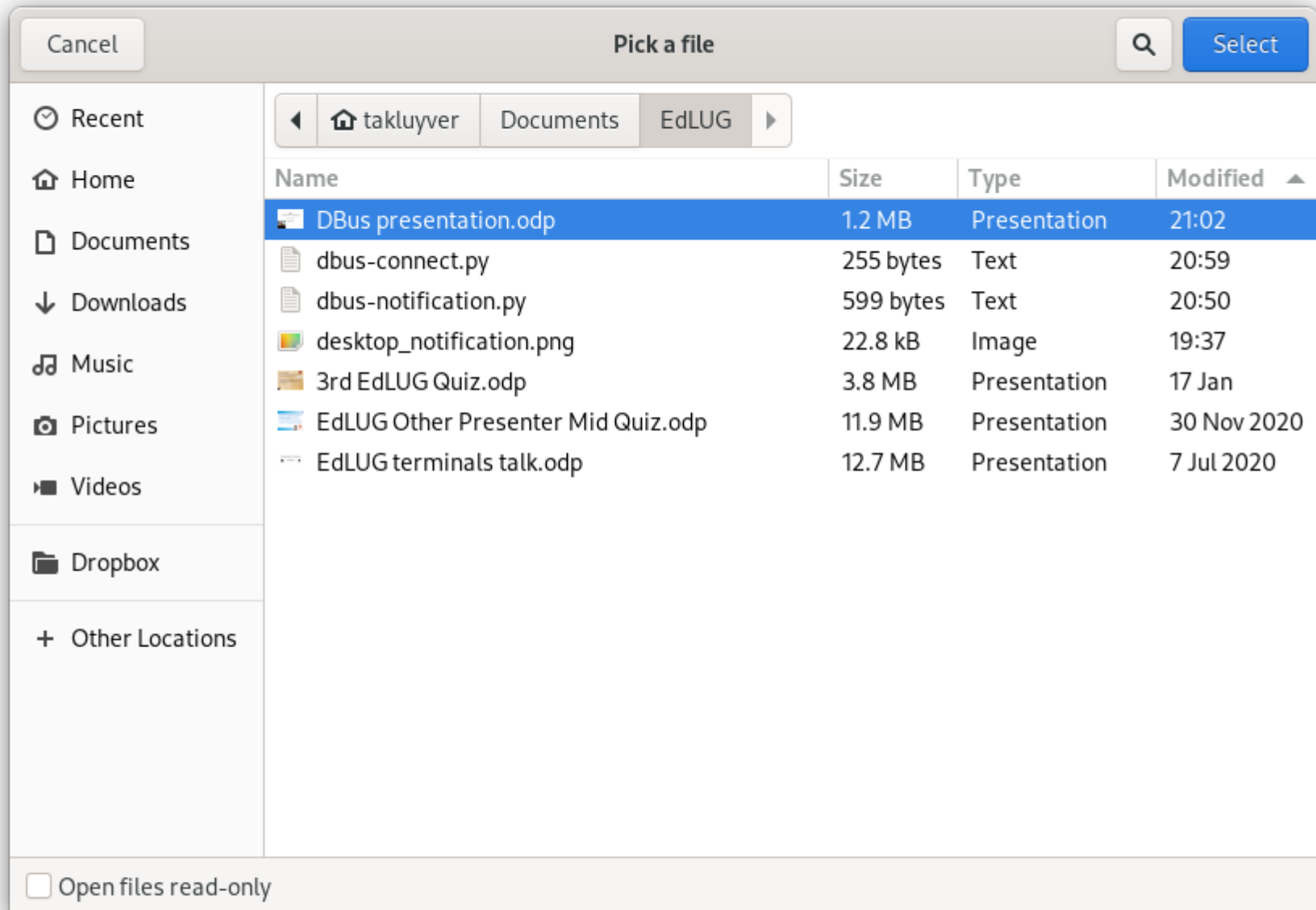
Screenshots

(on Wayland)



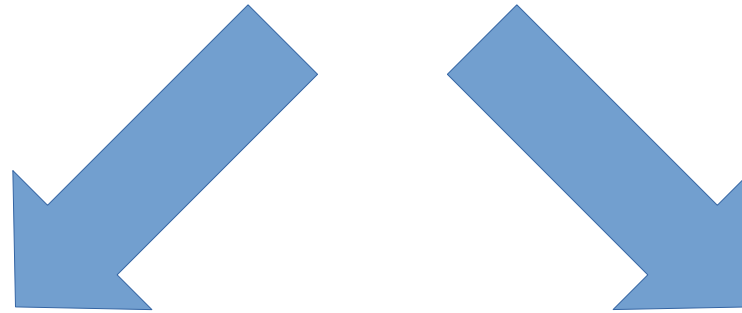
`org.gnome.Shell.Screenshot`

File picker



Is anyone there?

```
$ firefox https://edlug.gitlab.io/
```



Start a new browser

New tab in existing
window

*Send URL to browser
process*

```
org.mozilla.firefox.ZGVmYXVsdC1yZWxlYXNl
```

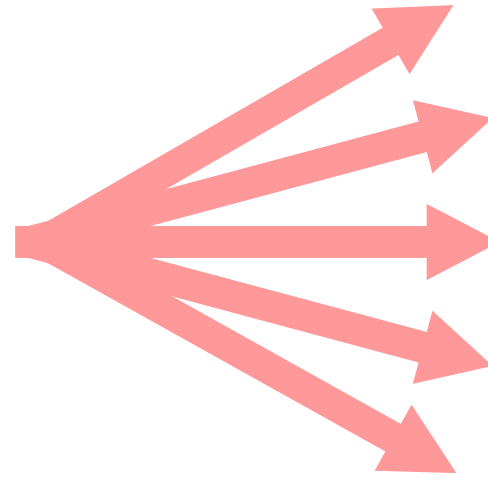
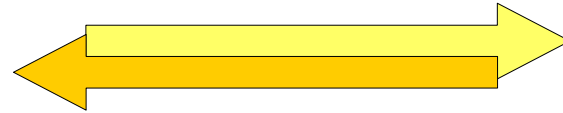
Profile name "default" in base64

How does this work?

- *Message bus*: central switchboard
 - Small program, always running
- Connect to the message bus
- Send messages addressed to a name
 - e.g. `org.freedesktop.Notifications`
- Claim your name (optional)
 - `RequestName` message to `org.freedesktop.DBus`
 - Name released if you disconnect

Communication patterns

- Method calls
 - One to one
 - Reply with result or error
- Signals
 - One to many
 - No reply
- Both can carry data



Naming stuff

- Well known names
 - `org.freedesktop.Notifications`
- Object paths
 - `/org/freedesktop/Notifications`
- Interfaces
 - `org.freedesktop.Notifications`
 - Group of methods and signals
 - Most things have a main interface + a few standard
 - e.g. `org.freedesktop.DBus.Introspectable`

Service activation

- Use a file to claim a name
- Start program when needed
- Less memory used!

e.g. `com.feralinteractive.GameMode`

`/usr/share/dbus-1/services/
com.feralinteractive.GameMode.service`

Talking to DBus

- Unix sockets
 - Like network sockets, but within computer
- On my system:
 - Session bus: `/run/user/1000/bus`
 - System bus: `/run/dbus/system_bus_socket`
- Authenticated with user ID
- Normally use `libdbus`

Demo

D-Feet

System Bus Session Bus

Search:

org.freedesktop.Notifications
activatable: no, pid: 1977

org.gnome.Disks.NotificationMonitor
activatable: no, pid: 2100

org.gnome.Evolution-alarm-notify
activatable: no, pid: 2105

org.gnome.SettingsDaemon.PrintNotificati
activatable: no, pid: 2007

org.gnome.Shell.Notifications
activatable: yes, pid: 1977

org.gtk.Notifications
activatable: no, pid: 1726

org.kde.StatusNotifierItem-2238-1
activatable: no, pid: 2238

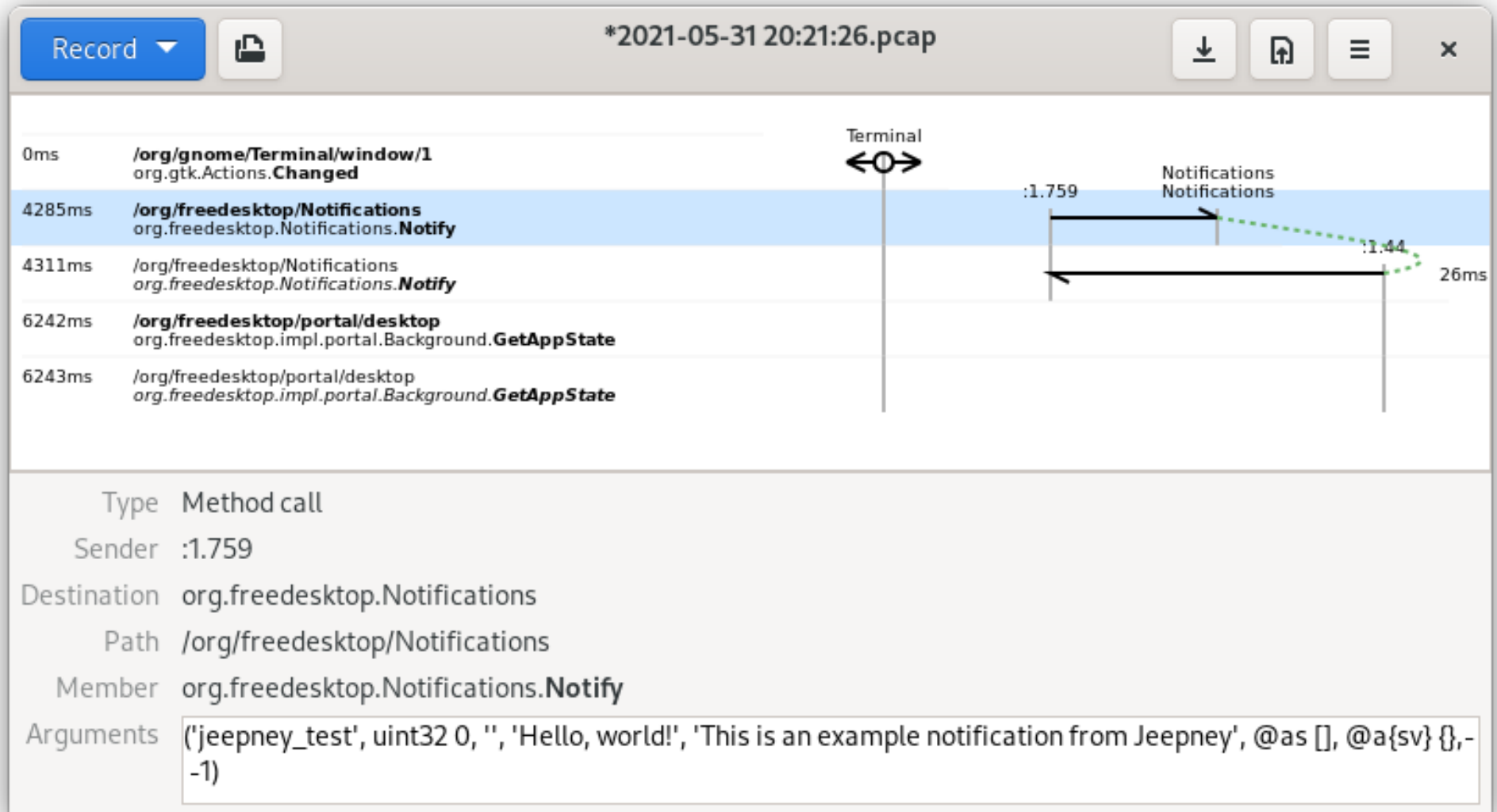
org.kde.StatusNotifierWatcher
activatable: no, pid: 1726

Address: unix:path=/run/user/1000/bus
Name: org.freedesktop.Notifications
Unique name: :1.44

Object path

- ▼ /org/freedesktop/Notifications
 - ▼ Interfaces
 - ▶ org.freedesktop.DBus.Introspectable
 - ▶ org.freedesktop.DBus.Peer
 - ▶ org.freedesktop.DBus.Properties
 - ▼ org.freedesktop.Notifications
 - ▼ Methods
 - CloseNotification (UInt32 arg_0) → ()
 - GetCapabilities () → (Array of [String] arg_0)
 - GetServerInformation () → (String arg_0, String arg_1, String arg_2)
 - Notify (String arg_0, UInt32 arg_1, String arg_2, String arg_3, String arg_4)
 - ▼ Signals
 - ActionInvoked (UInt32, String)
 - NotificationClosed (UInt32, UInt32)

Bustle



End

Questions?

D-Bus overview: <https://www.freedesktop.org/wiki/IntroductionToDBus/>
D-Bus specification: <https://dbus.freedesktop.org/doc/dbus-specification.html>

Is anyone there (other options)?

e.g. if Browser running, open a new tab

- Named pipe / Unix socket
 - Not cleaned up if process crashes
- Well known port
 - Only 65536 ports
- PID file
 - PIDs can be reused
 - Signals carry little information