# DBUS

Sources: Wikipedia, D-Bus Specification, Cardinal Peak

D-BUS is an Inter-Process Communication (IPC) specification. Communication is possible via multiple Transports (e.g., Unix sockets, systemd, TCP sockets).

Default D-Bus message buses:

- System-wide instance / message bus (see `/usr/share/dbus-1/system.conf` or `dbus-monitor --system`)
- User-session-specific instance / message bus (see `/usr/share/dbus-1/session.conf` or `dbus-monitor --session`)

Monitor DBUS signals: `dbus-monitor`

## DBUS Implementations

- GDBus (Gnome)
- QT/KDE D-Bus
- dbus-broker (default on Arch)

### dbus-broker

Source: dbus-broker package

- Each dbus-broker instance provides a message bus
- Each dbus-broker instance gets started by a parent process (*Controller*) (e.g., dbus-broker-launch)
- dbus-broker only implements message mediation (bus setup and external communication needs to be handled by the *Controller*.
- The *Controller* hands over a unix socket (as file descriptor *FD*) to the dbus-broker instance (`--controller=FD`)
- The unix socket is used by dbus-broker to accept commands from the *Controller*.
- Example commands issued by the Controller: Create listener sockets, specify bus policy, create activatable names, react to bus events.

dbus-broker-launch:

- dbus-broker-launch spawns and manages one message bus
- Configuration file is specified via `--config-file` or the default is taken from `/usr/share/dbus-1/` (`session.conf` for `--scope=user` and `system.conf` as fallback)
- `--scope` defines which systemd instance (user or system) is used to start the launcher

#### dbus-broker on Arch Linux

Source: Arch Wiki

During systemd installation, dbus-broker-units can be chosen as the *dbus-units* provider. This package provides the dbus-broker systemd service units.

Alternatively to dbus-broker, Arch also supports using the D-BUS reference implementation by installing the dbus-daemon-units instead of the dbus-broker service units.

## Desktop Notifications

- Send desktop notification: `notify-send test`
- Send desktop notification with icon: `notify-send -i face-glasses "test"` (icons are located in `/usr/share/icons/`)
- Monitor DBUS desktop notifications: `dbus-monitor "interface='org.freedesktop.Notifications'"`

### Desktop Notification Daemons

Primary source

Daemons:

- dunst: Lightweight, customizable, scriptable, minimalistic and provides features like rules, pausing, notification history
- NotifyOSD / notify-osd: Configurable, animations, sound, merging, overflow, flood prevention, do-not-disturb mode,

Depending on the Desktop Environment, notification daemons are built-in (Cinnamon, Gnome, KDE). Notification daemons can started standalone for other environments (e.g. i3wm). Standalone is possible via XDG autostart or as DBUS service.

Typically, DBUS service files are shipped with notification daemons, e.g., `/usr/share/dbus-1/services/org.freedesktop.Notifications.service`

### Troubleshooting

Tools:

- Bustle: Recording D-Bus communication, drawing sequence diagrams, inspecting communication
- D-Spy: Inspect buses/interfaces/properties/signals.methods, execute methods (e.g., send test desktop notifications)