



ibus with weston

Wayland IME Support

Presented by
Peng Wu
Software Engineer, Red Hat

Today's Topics

- ibus intro
- ibus on X Window
- ibus on Wayland
- New proposal for ibus on Wayland
- Pros and Cons
- Tasks
- Summary

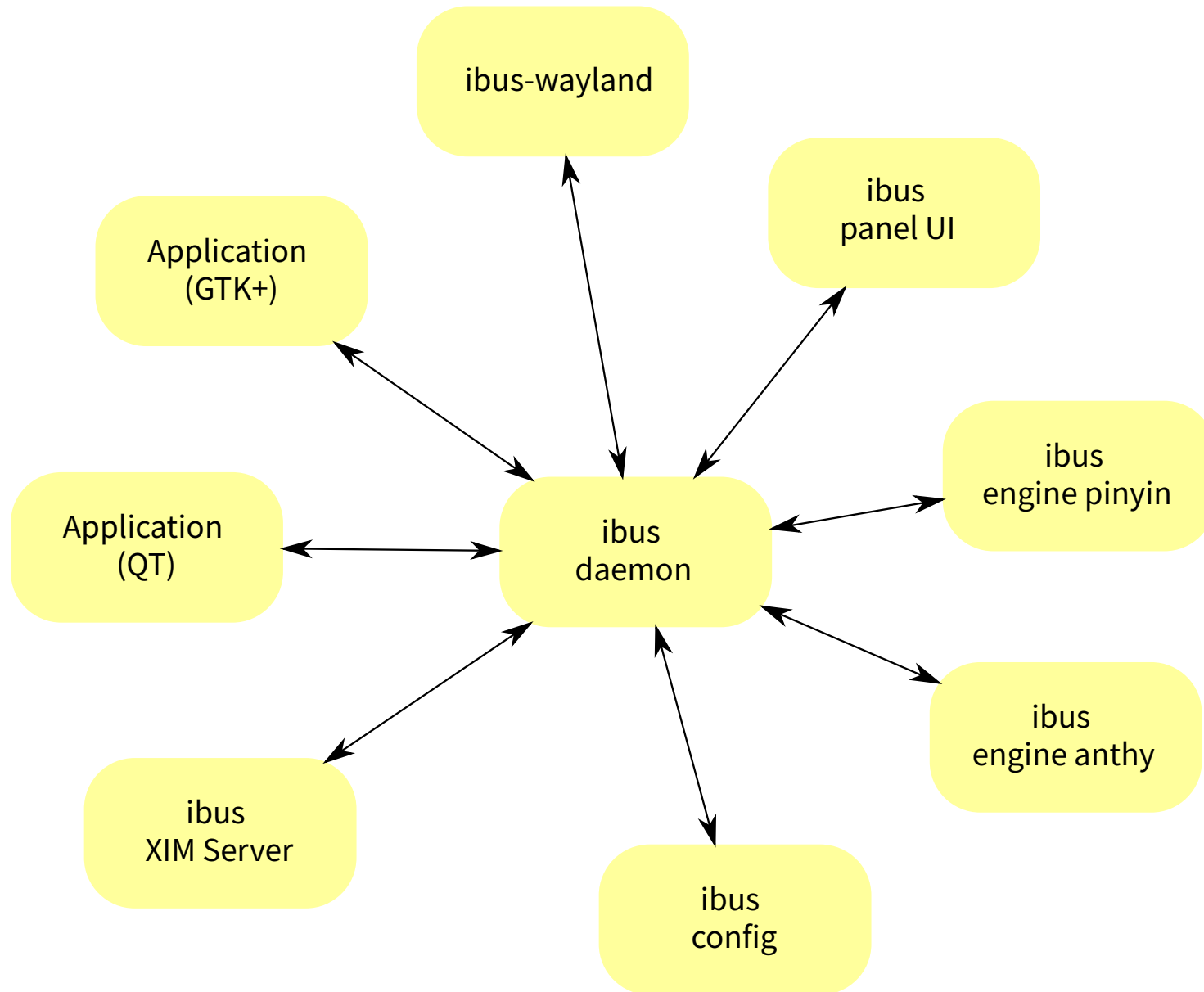
ibus intro



ibus intro

- Bus-Centric Architecture
 - It has an ibus-daemon, which manages all clients.
 - all engines, panel, config modules & clients are clients of ibus-daemon.
 - iBus is base on dbus IPC protocol.
- 

ibus architecture



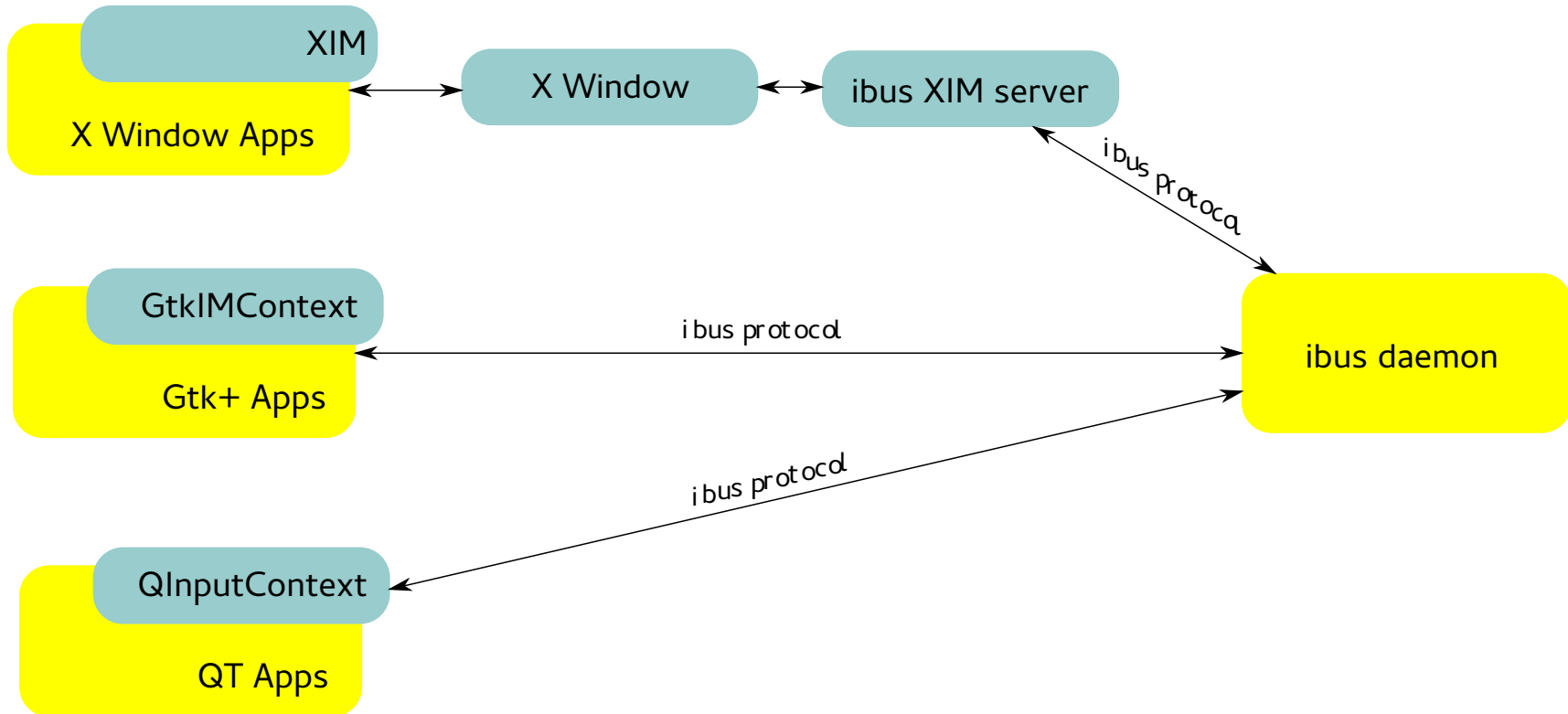
Advantage of ibus

- ibus uses dbus RPC for communication.
- The daemon, engines, clients & UI, all partitions are running in separated processes
 - One of them has problem, it will not impact all system
 - All partitions can be restarted on the fly.
 - Engine can be loaded & unloaded on demand.

ibus on X Window



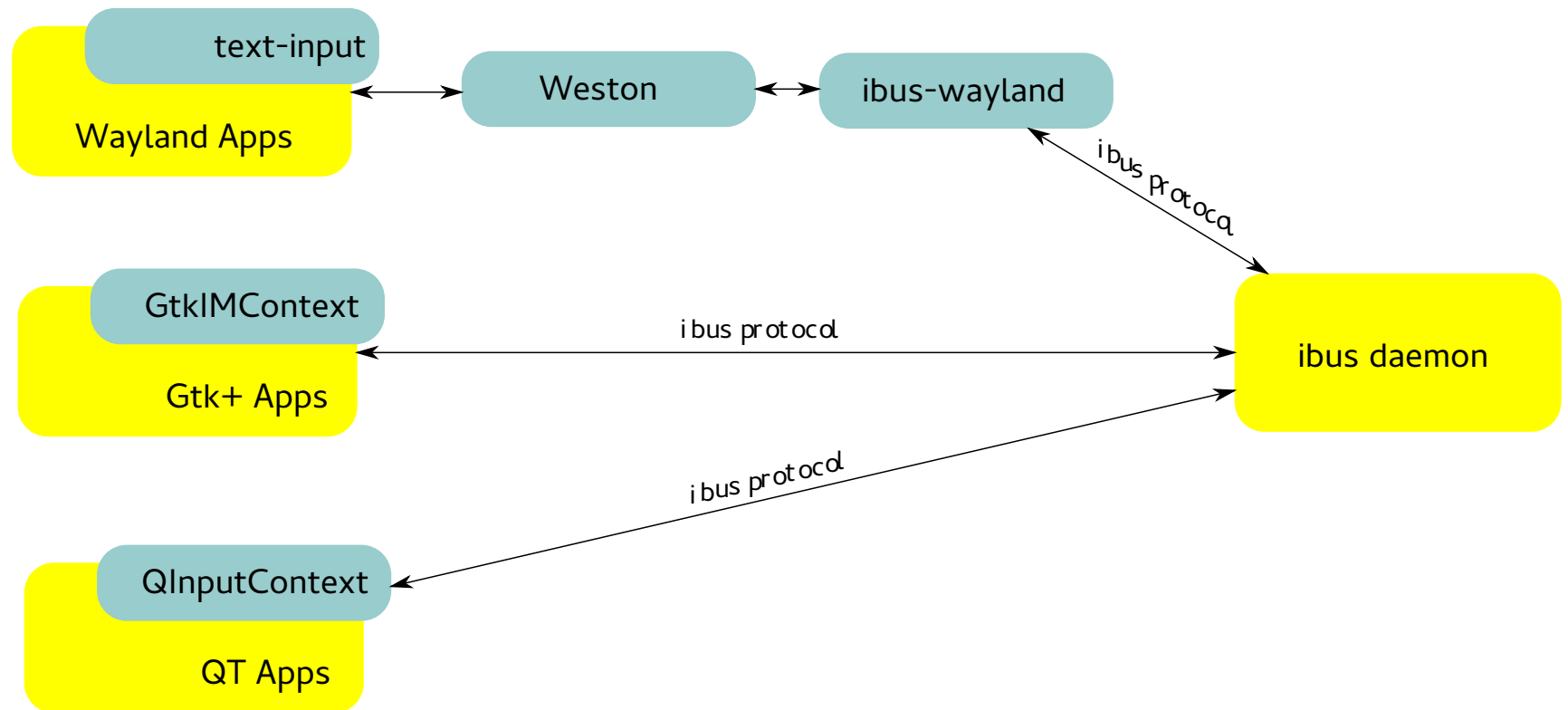
ibus on X Window



ibus on Wayland



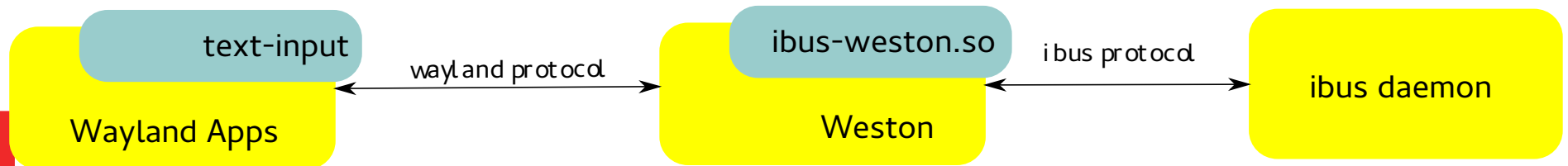
ibus on Wayland



New proposal for ibus on Wayland



New proposal



Pros and Cons



Pros

- more efficient
 - less processes involved



Cons

- ibus-weston plugin may crash weston
- Can glib main loop and wayland event loop work together?





Tasks

Tasks

- weston support to load input method plugin
 - GDBus support weston
 - GDBus tightly integrated with GMainContext
 - use `wl_event_loop_add_idle` to integrate with glib main loop
 - run glib main loop in `wl_event_loop_idle_func_t`
- 

Code Snippets

- ibus_main uses g_main_loop_run
- Maybe use g_main_context_iteration instead of g_main_loop_run

```
void wl_event_loop_idle_func_for_ibus(void *data)
{
    g_main_context_iteration (NULL, FALSE);
}
```

Summary

- weston display server acts like a event router for input method messages
 - Actually applications can communicate with ibus daemon directly
- may good to reduce the number of event proxy /routers for input method events

Questions?



License statement goes here. See https://fedoraproject.org/wiki/Licensing#Content_Licenses for acceptable licenses.