

Debugging DBus

Ted Gould
Device Sprint
Oct 2014

What is DBus?

- Simple Message Bus
- Divided into System/Session instances
- De facto IPC for services on Ubuntu

DBus Message

| key | values | description |
|-------------|---|--|
| type | signal, method_call, method_return, error | How the message behaves |
| sender | Address in the form :0.1 | Who the message originated from |
| interface | String in the reverse domain form (com.canonical.foo) | Chooses which interface the client is expecting to use (<i>next slide</i>) |
| member | string | Select which part of the interface to use |
| path | String in the form /foo/bar | Provides a way for services to provide multiple end points to clients |
| destination | Address (<i>typically unset for signals</i>) | Provides routing suggestions to the DBus daemon |
| payload | Serialized data | Let's talk about this in a slide or two |

DBus Interfaces

```
<?xml version="1.0" encoding="UTF-8" ?>
<node name="/org/freedesktop/Notifications">
  <interface name="org.freedesktop.Notifications">

    <method name="Notify">
      <arg type="s" name="app_name" direction="in" />
      <arg type="u" name="id" direction="in" />
      <arg type="s" name="icon" direction="in" />
      <arg type="s" name="summary" direction="in" />
      <arg type="s" name="body" direction="in" />
      <arg type="as" name="actions" direction="in" />
      <arg type="a{sv}" name="hints" direction="in" />
      <arg type="i" name="timeout" direction="in" />
      <arg type="u" name="return_id" direction="out" />
    </method>

    <method name="CloseNotification">
      <arg type="u" name="id" direction="in" />
    </method>
  </interface>
</node>
```

DBus Types For Payloads

| Character | Type |
|-----------|--|
| y | 8-bit unsigned integer |
| b | boolean |
| n | 16-bit signed integer |
| q | 16-bit unsigned integer |
| i | 32-bit signed integer |
| u | 32-bit unsigned integer |
| x | 64-bit signed integer |
| t | 64-bit unsigned integer |
| d | double-precision floating point |
| s | UTF8 string (no embedded null) |
| o | DBus object path |
| a | Array |
| v | Variant |
| h | Unix file descriptor |
| () | Structure (i.e. a(ss), a(iisv), a(sa(ss)inqay) |

DBus Introspection

```
$ gdbus call --session --dest
org.freedesktop.Notifications --object-
path /org/freedesktop/Notifications --method
org.freedesktop.DBus.Introspectable.Introspe
ct
```

```
(<!DOCTYPE node PUBLIC "-//freedesktop//DTD D-BUS Object Introspection
1.0//EN"\n"http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd">\n<node>\n <interface
name="org.freedesktop.DBus.Introspectable">\n <method name="Introspect">\n <arg name="data" direction="out"
type="s"/>\n </method>\n </interface>\n <interface name="org.freedesktop.DBus.Properties">\n <method
name="Get">\n <arg name="interface" direction="in" type="s"/>\n <arg name="propname" direction="in"
type="s"/>\n <arg name="value" direction="out" type="v"/>\n </method>\n <method name="Set">\n <arg
name="interface" direction="in" type="s"/>\n <arg name="propname" direction="in" type="s"/>\n <arg
name="value" direction="in" type="v"/>\n </method>\n <method name="GetAll">\n <arg name="interface"
direction="in" type="s"/>\n <arg name="props" direction="out" type="a{sv}"/>\n </method>\n </interface>\n
<interface name="org.freedesktop.Notifications">\n <method name="GetServerInformation">\n <arg
name="return_name" type="s" direction="out"/>\n <arg name="return_vendor" type="s" direction="out"/>\n <arg
name="return_version" type="s" direction="out"/>\n <arg name="return_spec_version" type="s" direction="out"/>\n
</method>\n <method name="GetCapabilities">\n <arg name="return_caps" type="as" direction="out"/>\n
</method>\n <method name="CloseNotification">\n <arg name="id" type="u" direction="in"/>\n </method>\n
<method name="Notify">\n <arg name="app_name" type="s" direction="in"/>\n <arg name="id" type="u"
direction="in"/>\n <arg name="icon" type="s" direction="in"/>\n <arg name="summary" type="s"
direction="in"/>\n <arg name="body" type="s" direction="in"/>\n <arg name="actions" type="as"
direction="in"/>\n <arg name="hints" type="a{sv}" direction="in"/>\n <arg name="timeout" type="i"
direction="in"/>\n <arg name="return_id" type="u" direction="out"/>\n </method>\n </interface>\n</node>\n',)
```

DBus Monitor

```
$ dbus-monitor interface=org.freedesktop.Notifications
```

```
method call sender=:1.156 -> dest=:1.59 serial=6 path=/org/freedesktop/Notifications;  
interface=org.freedesktop.Notifications; member=GetServerInformation
```

```
method call sender=:1.156 -> dest=:1.59 serial=7 path=/org/freedesktop/Notifications;  
interface=org.freedesktop.Notifications; member=Notify
```

```
    string "notify-send"
```

```
    uint32 0
```

```
    string ""
```

```
    string "example"
```

```
    string ""
```

```
    array [
```

```
    ]
```

```
    array [
```

```
        dict entry(
```

```
            string "urgency"
```

```
            variant          byte 1
```

```
        )
```

```
    ]
```

```
int32 -1
```

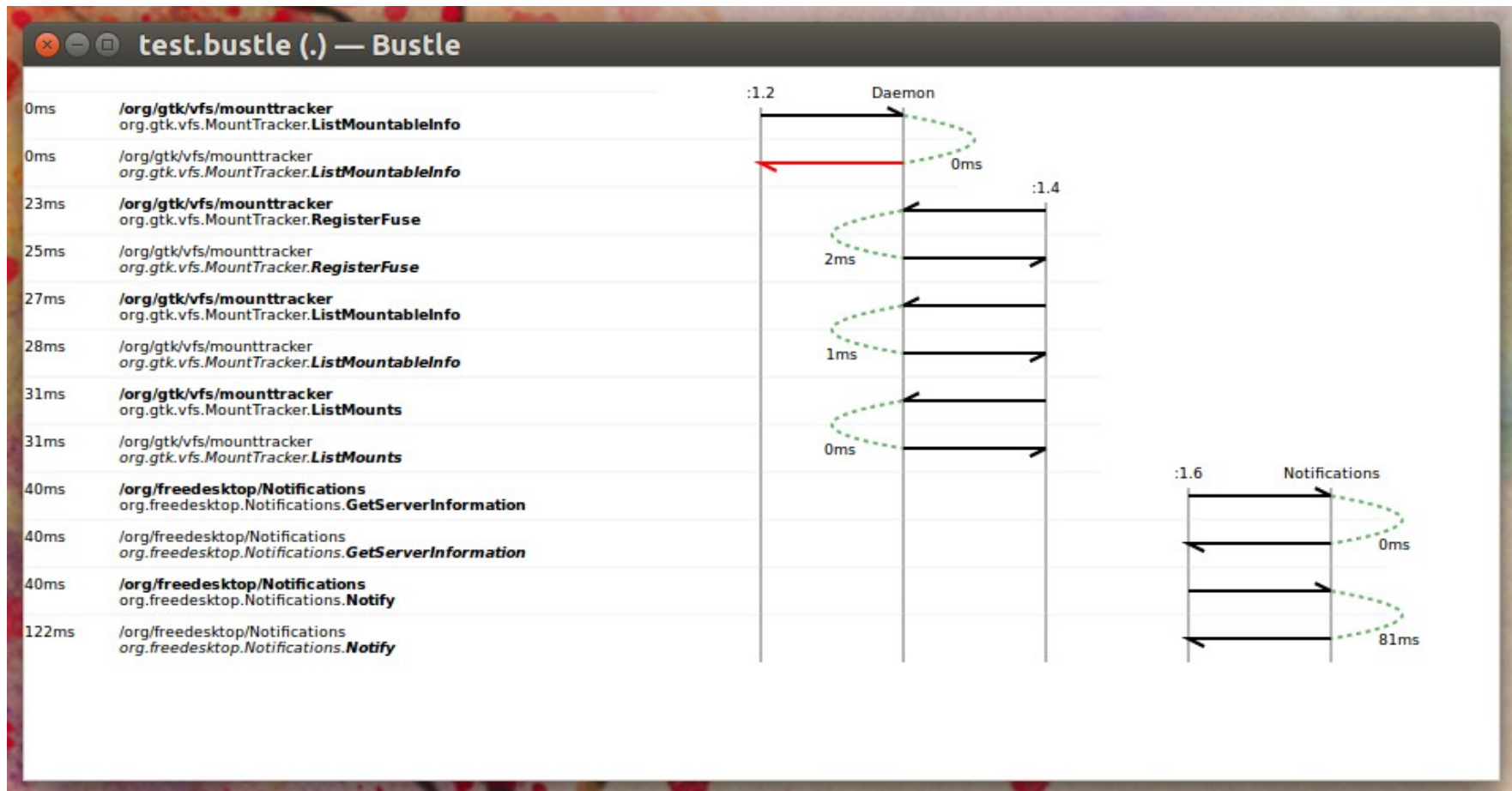
Bustle

```
$ bustle-pcap test.bustle
```

```
Logging D-Bus traffic to 'test.bustle'...
```

```
Hit Control-C to stop logging.
```

```
$ bustle test.bustle
```



D-Feet (Introspection)

The screenshot shows the d-feet application window. At the top, there are tabs for 'System Bus' and 'Session Bus'. Below the tabs is a 'Filter:' input field. A list of bus names is displayed, with 'org.freedesktop.Notifications' selected and highlighted in orange. To the right of the list, a panel shows the details for the selected object:

- Address:** unix:abstract=/tmp/dbus-eemligE2Tz
- Name:** org.freedesktop.Notifications
- Unique name:** :1.196
- Object path:** /org/freedesktop/Notifications

Below the object path, the application shows the introspected interfaces and methods:

- Interfaces:**
 - org.freedesktop.DBus.Introspectable
 - org.freedesktop.DBus.Properties
 - org.freedesktop.Notifications
- Methods:**
 - CloseNotification (UInt32 id) → ()
 - GetCapabilities () → (Array of [String] return_caps)
 - GetServerInformation () → (String return_name, String return_vendor, String return_version)
 - Notify (String app_name, UInt32 id, String icon, String summary, String body, Array of [String] actions)

At the bottom of the window, a summary of the selected object is shown:

- Name:** org.freedesktop.Notifications
- Process ID:** 3364
- Command line:** /usr/lib/x86_64-linux-gnu/notify-osd

D-Feet (Method Call)

Execute D-Bus Method

Method name: GetCapabilities () ↪ (Array of [String] return_caps)

Object Path: /org/freedesktop/Notifications

Interface: org.freedesktop.Notifications

Method input

Method output

['body', 'body-markup', 'icon-static', 'image/svg+xml', 'x-canonical-private-synchronous', 'x-canonical-append', 'x-canonical-private-icon-only', 'x-canonical-truncation']

Pretty print Source

Method execution 1 - +

Ø: 0.0033 Min: 0.0033 Max: 0.0033

Close Execute