igalia
Free Software Engineering

Debugging Accessibility Issues

Joanmarie Diggs | jdiggs@igalia.com | www.igalia.com

igalia
Free Software Engineering

Orca 101

Joanmarie Diggs | jdiggs@igalia.com | www.igalia.com

Screen Readers Present Screen Contents

- To users who are blind or visually impaired
- In synthesized speech and/or refreshable braille
- As they navigate using the application's commands:
  - Focus changes
  - Caret movement
  - Selection changes
- And when “events of interest” occur in non-focused widgets
  - Notifications
  - Incoming chat messages

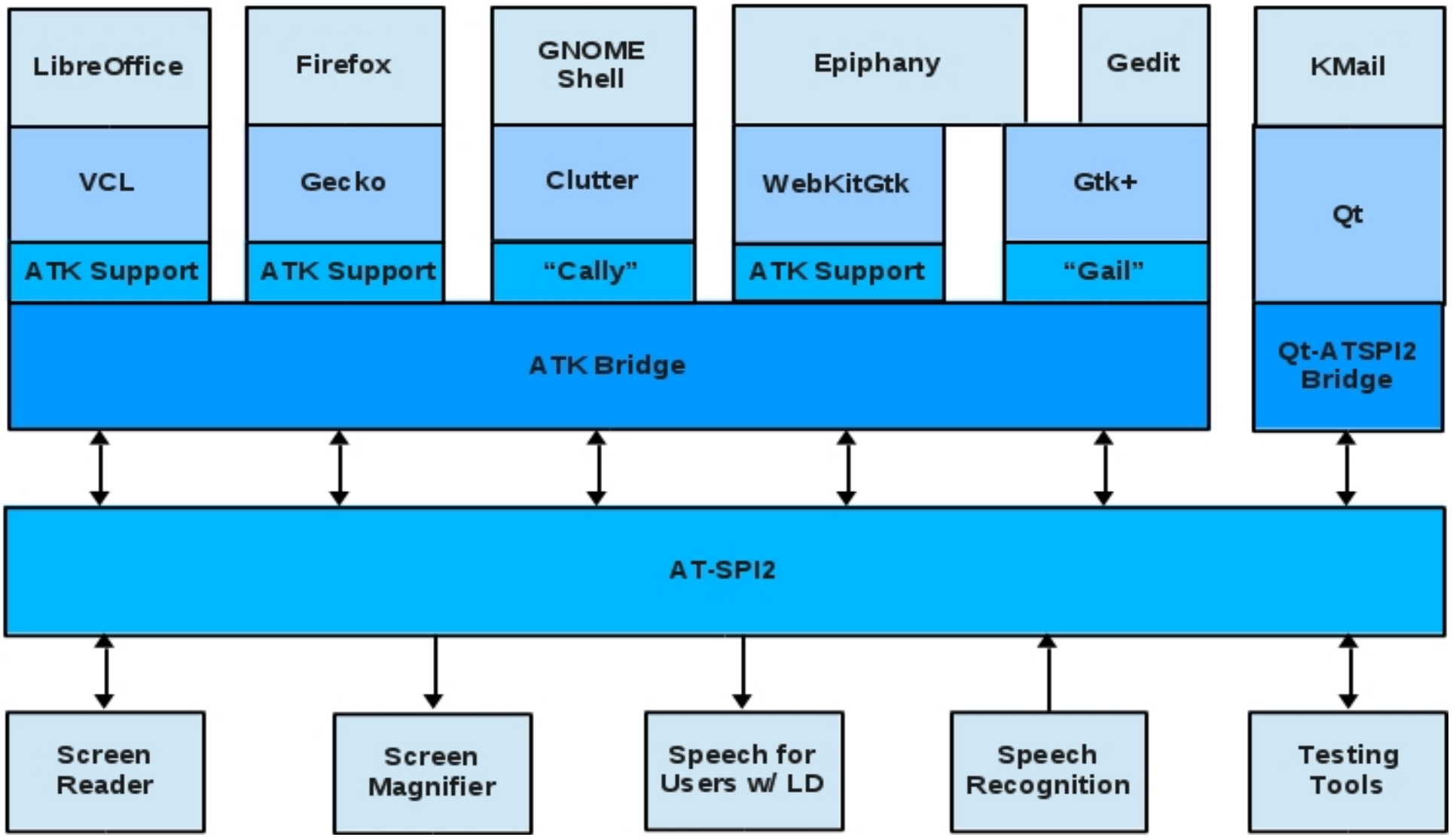
Screen Readers Provide Additional Commands For More Efficient Access

- Review chat messages and notifications
- Navigate by element type
- Obtain a summary of font/formatting information
- Search the screen for an object
- Etc.

Orca: A Screen Reader for the Free Desktop

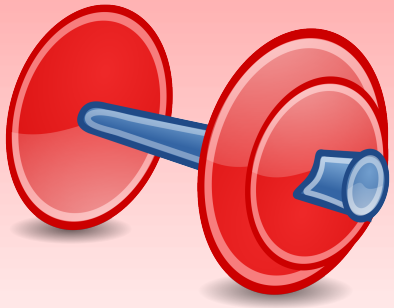
- Written in Python
- Access to graphical desktop environments
- Presentation:
  - Of objects, events, and other functionality via AT-SPI2
  - Speech synthesis via speech-dispatcher
  - Braille translation via liblouis
  - Braille display via brlapi

Where Is the Orca Bug?


LibreOffice
Firefox
GNOME Shell
Epiphany
Gedit
KMail
VCL
Gecko
Clutter
WebKitGtk
Gtk+
Qt
ATK Support
ATK Support
"Cally"
ATK Support
"Gail"
ATK Bridge
Qt-ATSPI2 Bridge
AT-SPI2
Screen Reader
Screen Magnifier
Speech for Users w/ LD
Speech Recognition
Testing Tools

igalia
Free Software Engineering

Accerciser 101

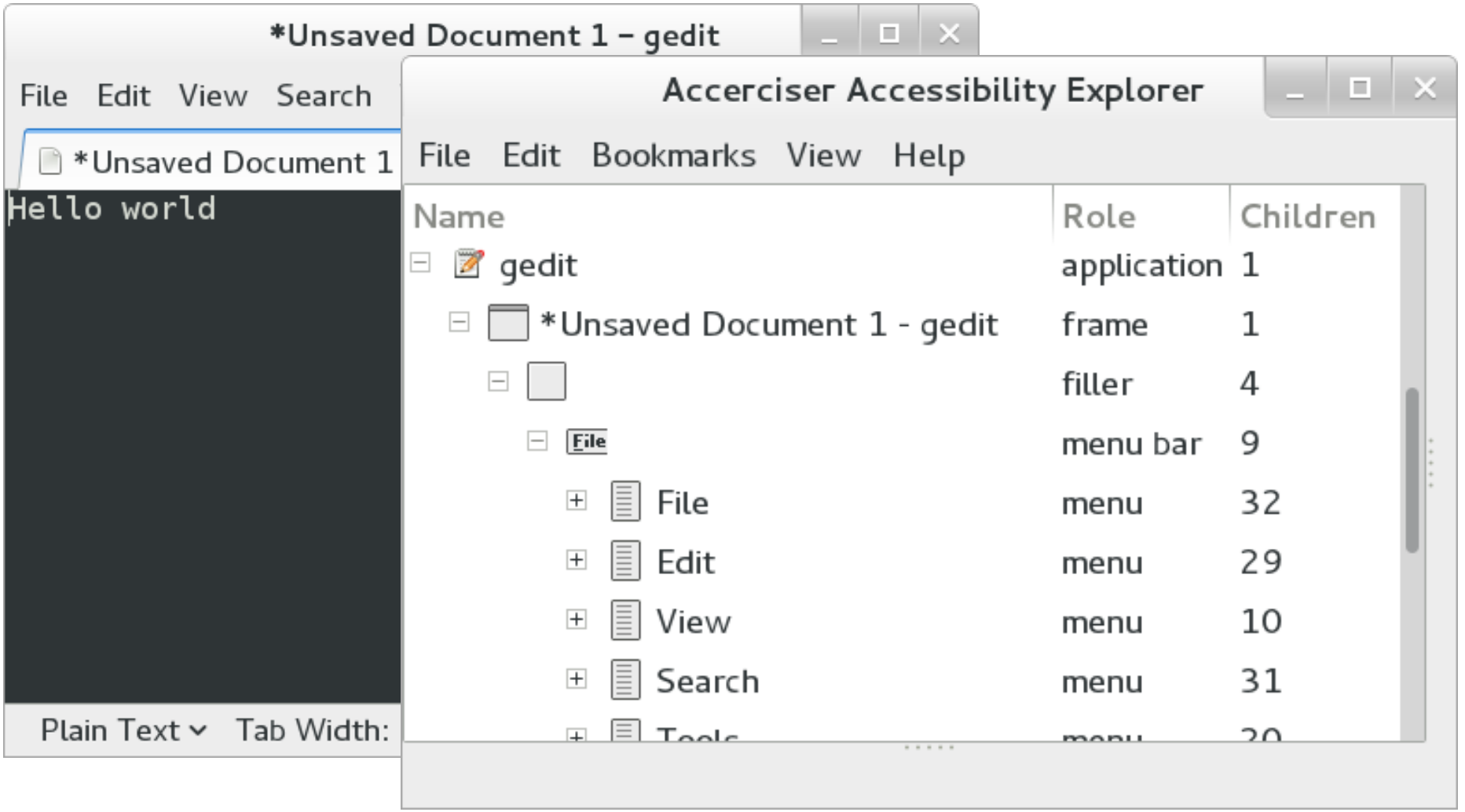


Joanmarie Diggs | jdiggs@igalia.com | www.igalia.com

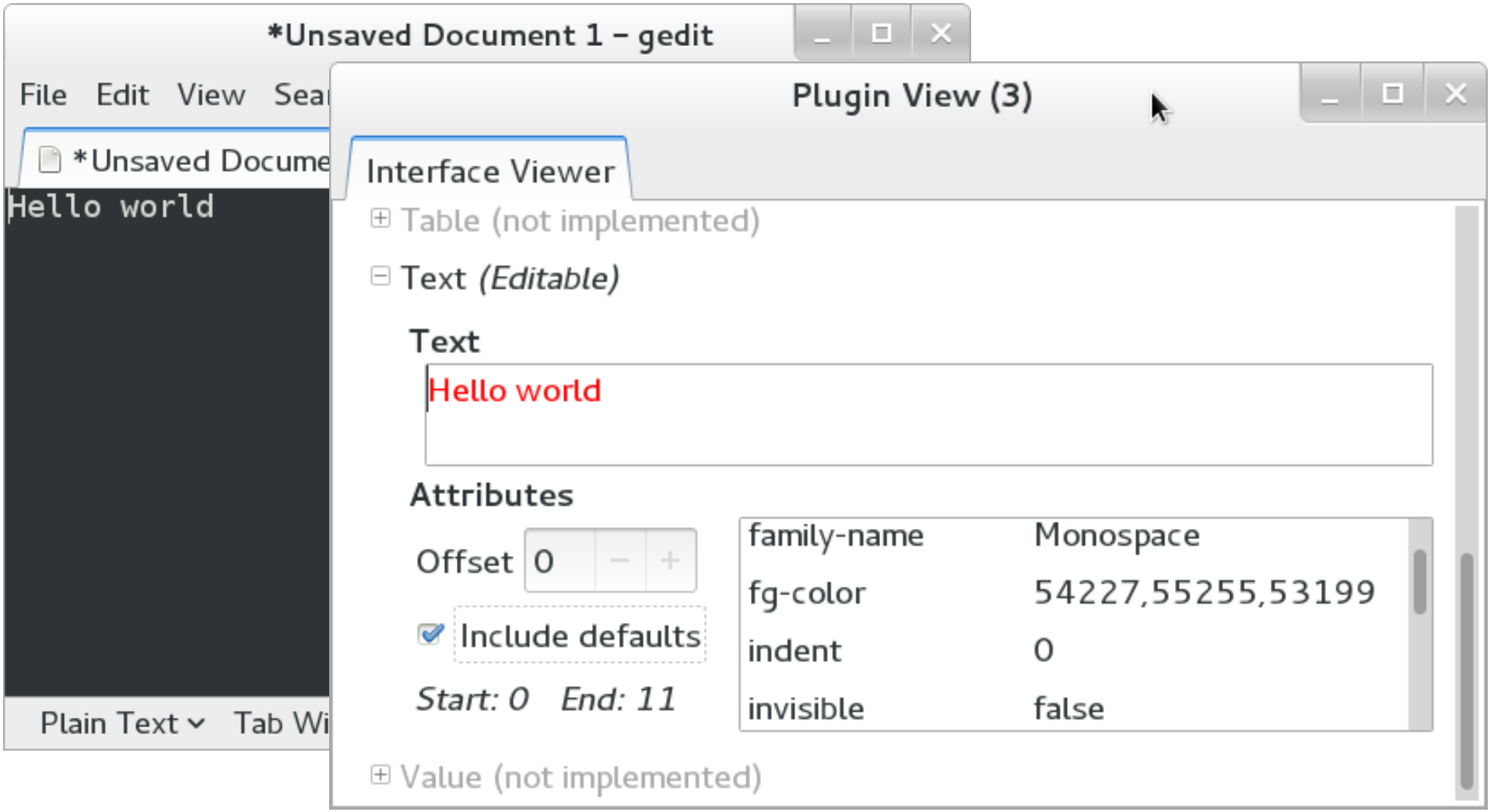
Accerciser: The Interactive “Accessibility Explorer”

- Written in Python
- View accessible objects
- Examine implemented accessibility interfaces
- Monitor accessibility events
- Interact with accessible objects through their interfaces

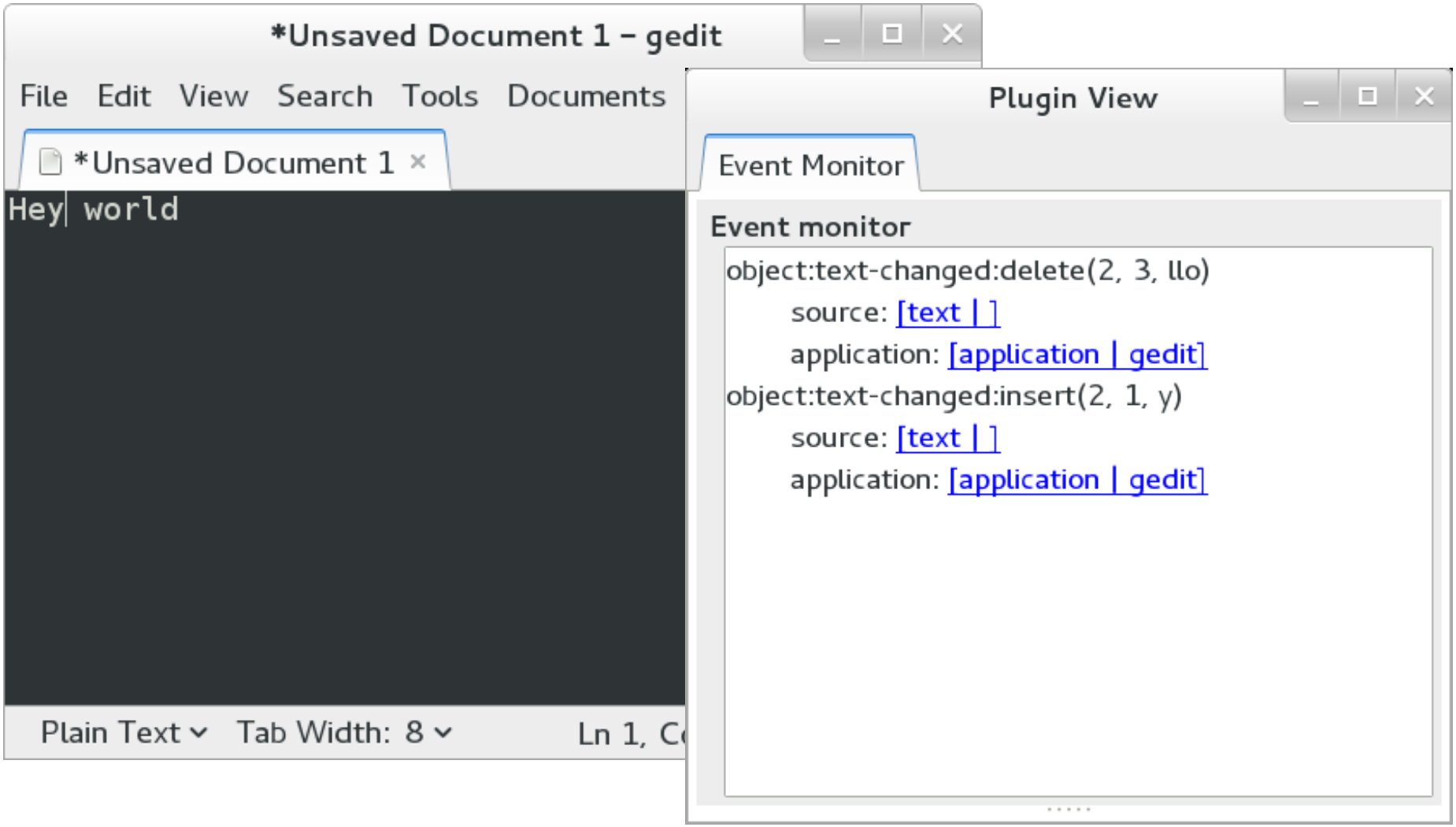
Accerciser: View Accessible Objects



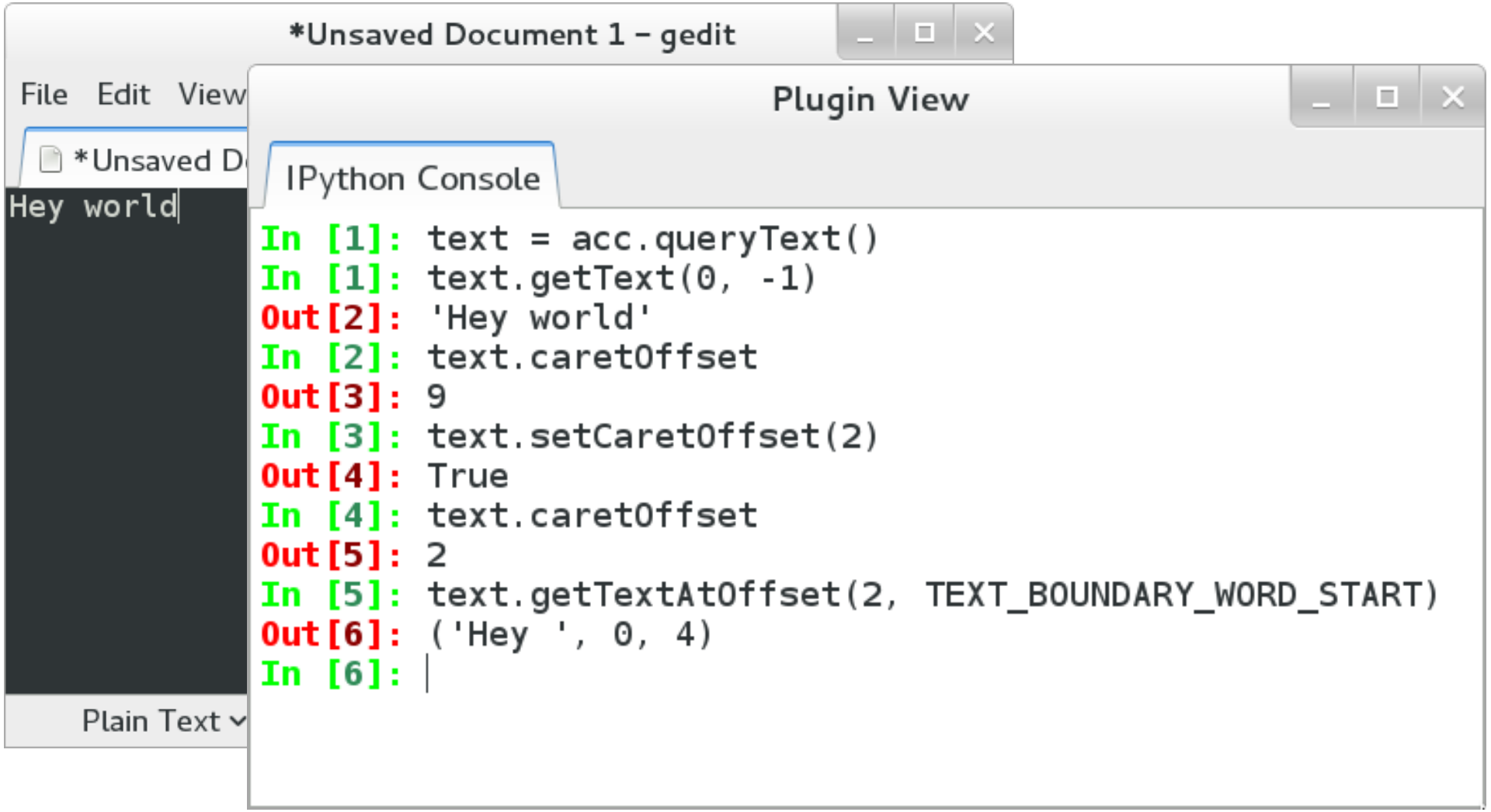
Accerciser: Examine Implemented Interfaces



Accerciser: Monitor Accessibility Events



Accerciser: Interact with Accessible Objects



Is the bug in the application or toolkit?

- Do you see the objects to be presented by Orca in Accerciser?
- Do the objects emit accessible events when you navigate?
- Do the objects have the needed accessible properties?
- Do the objects implement the needed accessible interfaces?

If you CANNOT answer “yes” to all of the above, the bug is likely in the application or toolkit

Rolling Your Own Listener

Joanmarie Diggs | jdiggs@igalia.com | www.igalia.com

Rolling Your Own Listener

- Pick the event type of interest
- Define the handler's behavior
- Register the listener
- Start the registry

Hello World Listener

helloworld.py – emacs@blockhead

File Edit Options Buffers Tools Python Help

```
#!/usr/bin/python
import pyatspi

def listener(e):
    print e

pyatspi.Registry.registerEventListener(listener, "object:state-changed")
pyatspi.Registry.start()
```

-:--- **helloworld.py**   All (9,0)   (Python)

Hello World Listener 2

helloworld2.py – emacs@blockhead

File Edit Options Buffers Tools Python Help

```
#!/usr/bin/python
import pyatspi

def listener(e):
    obj = e.source
    app = obj.getApplication()
    if app.name != 'gedit' or obj.getRole() != pyatspi.ROLE_MENU_ITEM:
        return

    print e

pyatspi.Registry.registerEventListener(
    listener, "object:state-changed:focused", "object:state-changed:selected")
pyatspi.Registry.start()
```

-:--- **helloworld2.py**   All (15,0)   (Python)

Hello World Listener 2 Results

```
object:state-changed:selected(1, 0, 0)
        source: [menu item | New]
        host_application: [application | gedit]
object:state-changed:selected(0, 0, 0)
        source: [menu item | New]
        host_application: [application | gedit]
object:state-changed:selected(1, 0, 0)
        source: [menu item | Open...]
        host_application: [application | gedit]
object:state-changed:selected(0, 0, 0)
        source: [menu item | Open...]
        host_application: [application | gedit]
object:state-changed:selected(1, 0, 0)
        source: [menu item | Save]
        host_application: [application | gedit]
object:state-changed:selected(0, 0, 0)
        source: [menu item | Save]
        host_application: [application | gedit]
```

Hello World Listener 2 Translation

```
object:state-changed:selected(1, 0, 0)
        source: [menu item | New]
        host_application: [application | gedit]
object:state-changed:selected(0, 0, 0)
        source: [menu item | New]
        host_application: [application | gedit]
object:state-changed:selected(1, 0, 0)
        source: [menu item | Open...]
        host_application: [application | gedit]
object:state-changed:selected(0, 0, 0)
        source: [menu item | Open...]
        host_application: [application | gedit]
object:state-changed:selected(1, 0, 0)
        source: [menu item | Save]
        host_application: [application | gedit]
object:state-changed:selected(0, 0, 0)
        source: [menu item | Save]
        host_application: [application | gedit]
```

To present the selected menu item as a user navigates:

- Listen for:
  - object:state-changed:selected
  - event.detail1 ==1
- Speak and braille:
  - event.source.name

TODO:

Investigate why we are not getting the expected focused events.