Archaeology of Accessibility

A look at the past and future of the Accessibility stack
Emmanuele Bassi
Emmanuele Bassi
He/him
Emmanuele Bassi
He/him
GNOME Foundation
GTK
Let's talk about accessibility
Accessibility, n.: The design of products, devices, services, or environments so as to be usable by people with disabilities.
**Assistive technology**: Assistive, adaptive, and rehabilitative devices for people with disabilities or the elderly population.
Who benefits from accessibility?
Who benefits from accessibility?
Figure 1: © Frankie Fouganthin, CC SA-4.0
Figure 2: © Universal Studios
Figure 3:
Who benefits from accessibility?

We all do.
Some history

 GNOME Accessibility Kit (GTK) 1.3 → 2.0
Some history

- GTK 1.3 → 2.0
- ATK
Some history

- GTK 1.3 → 2.0
- ATK
- AT-SPI
Some history

- GTK 1.3 → 2.0
- ATK
- AT-SPI
- GAIL
Accessibility is **not** a checkbox
Accessibility is **not** a checkbox
The world has changed

- CORBA → DBus
The world has changed

- CORBA → DBus
- Out of tree modules → built in functionality
The world has changed

- CORBA → DBus
- Out of tree modules → built in functionality
- X11 → Wayland
The world has changed

- CORBA → DBus
- Out of tree modules → built in functionality
- X11 → Wayland
- Sandboxing
Accessibility in GNOME needs to change too
Accessibility in GNOME needs to change too
How do we change it?

🌈 Consolidate the effort
How do we change it?

- Consolidate the effort
- Simplify the toolkit
How do we change it?

- Consolidate the effort
- Simplify the toolkit
- Empower application developers
How do we change it?

- Consolidate the effort
- Simplify the toolkit
- Empower application developers
- Funding
New accessibility stack
WAI-ARIA

- W3C standard
WAI-ARIA

- W3C standard
- **element**: an accessible part of the UI
WAI-ARIA

- W3C standard
- **element**: an accessible part of the UI
- **role**: what the element *does*
WAI-ARIA

- W3C standard
- **element**: an accessible part of the UI
- **role**: what the element *does*
- **attributes**: what the element *has*
WAI-ARIA

- W3C standard
- **element**: an accessible part of the UI
- **role**: what the element *does*
- **attributes**: what the element *has*
  - property
WAI-ARIA

- W3C standard
- **element**: an accessible part of the UI
- **role**: what the element *does*
- **attributes**: what the element *has*
  - property
  - relation
WAI-ARIA

- W3C standard
- **element**: an accessible part of the UI
- **role**: what the element *does*
- **attributes**: what the element *has*
  - property
  - relation
  - state
Toolkit/1

c element: GtkAccessible
**Toolkit/1**

- **element**: GtkAccessible
- **role**: GtkAccessibleRole
element: GtkAccessible
role: GtkAccessibleRole
attributes
element: GtkAccessible
role: GtkAccessibleRole
attributes
  - GtkAccessibleProperty
element: GtkAccessible
role: GtkAccessibleRole
attributes
   GtkAccessibleProperty
   GtkAccessibleRelation
element: GtkAccessible
role: GtkAccessibleRole
attributes
- GtkAccessibleProperty
- GtkAccessibleRelation
- GtkAccessibleState
Toolkit/1

- **element**: GtkAccessible
- **role**: GtkAccessibleRole
- **attributes**
  - GtkAccessibleProperty
  - GtkAccessibleRelation
  - GtkAccessibleState
- **platform-specific backend**: GtkATContext
Toolkit/2

Set the role on your widget

```c
// GtkCheckBox: set role

gtk_widget_class_set_accessible_role (widget_class,
    GTK_ACCESSIBLE_ROLE_CHECKBOX);
```
Toolkit/3

Change states

// GtkCheckButton: state change

gtk_accessible_update_state (
    GTK_ACCESSIBLE (check_button),
    GTK_ACCESSIBLE_STATE_PRESSED, is_active, 
    -1
);

Change properties/1

```c
// GtkEntry → accessible label

gtk_accessible_update_property (GTK_ACCESSIBLE (username_entry),
    GTK_ACCESSIBLE_PROPERTY_LABEL, "User name",
    -1);
```
// GtkRange: change adjustment

gtk_accessible_update_property (
    GTK_ACCESSIBLE (scroll_bar),
    GTK_ACCESSIBLE_PROPERTY_VALUE_MIN, gtk_adjustment_get_lower (adj),
    GTK_ACCESSIBLE_PROPERTY_VALUE_MAX, gtk_adjustment_get_upper (adj),
    GTK_ACCESSIBLE_PROPERTY_VALUE_NOW, gtk_adjustment_get_value (adj),
    -1
);
Change relations

// GtkPasswordEntry → GtkLabel

gtk_accessible_update_relation (GTK_ACCESSIBLE (password_entry), GTK_ACCESSIBLE_RELATION_LABELLED_BY, password_label, -1);
Follow the rules
Five Rules of ARIA

1st: Reuse existing accessible UI elements
Five Rules of ARIA

1st: Reuse existing accessible UI elements
2nd: Do not change semantics
Five Rules of ARIA

1st: Reuse existing accessible UI elements
2nd: Do not change semantics
3rd: Keyboard accessibility
Five Rules of ARIA

1st: Reuse existing accessible UI elements
2nd: Do not change semantics
3rd: Keyboard accessibility
4th: Do not hide focusable elements
Five Rules of ARIA

1st: Reuse existing accessible UI elements
2nd: Do not change semantics
3rd: Keyboard accessibility
4th: Do not hide focusable elements
5th: Add an accessible label
Applications/2

GTK-BUILDER-TOOL
Applications/2

🎨 gtk-builder-tool
🎨 GtkTestATContext
Applications/2

- gtk-builder-tool
- GtkTestATContext
- gtk_test_accessible_assert_*
Current state

- Finalising the API
Current state

- Finalising the API
- Porting GTK widgets
Current state

- Finalising the API
- Porting GTK widgets
- Writing tests
Current state

- Finalising the API
- Porting GTK widgets
- Writing tests
- Documentation
Current state

- Finalising the API
- Porting GTK widgets
- Writing tests
- Documentation
- Implementing the AT-SPI backend
How can you help?

- 🌐 Help writing tests
How can you help?

- Help writing tests
- Documentation
How can you help?

- Help writing tests
- Documentation
- Port to Windows/macOS
How can you help?

- Help writing tests
- Documentation
- Port to Windows/macOS
- Improve access for sandboxed applications
How can you help?

- Help writing tests
- Documentation
- Port to Windows/macOS
- Improve access for sandboxed applications
- Funding
Shout out to

⁻ GNOME Accessibility team
Shout out to

- GNOME Accessibility team
- GNOME Foundation
Shout out to

👩‍💻 GNOME Accessibility team
👩‍💻 GNOME Foundation
👩‍💻 Hypra
Questions?
Thanks!