

Archaeology of Accessibility

A look at the past and future of the Accessibility stack



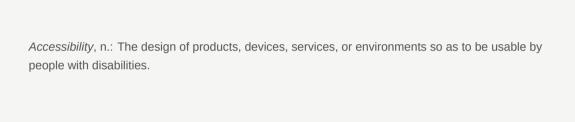
Emmanuele Bassi

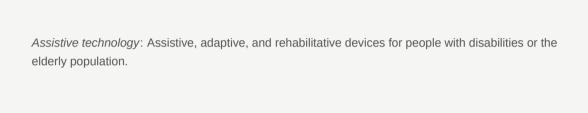
- Emmanuele Bassi

- **Emmanuele Bassi**
- **GNOME** Foundation

- **Emmanuele Bassi**
- **GNOME** Foundation
- **€** GTK

Let's talk about accessibility





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.

- **€** GTK 1.3 → 2.0
- ATK

- **€** GTK 1.3 → 2.0
- ATK
- **ℰ** AT-SPI

- **€** GTK 1.3 → 2.0
- ATK
- **ℰ** AT-SPI
- **GAIL**

Accessibility is **not** a checkbox

Accessibility is **not** a checkbox

€ CORBA → DBus

- **€** CORBA → DBus
- **₹** Out of tree modules → built in functionally

- **€** CORBA → DBus
- Out of tree modules → built in functionally

- **€** CORBA → DBus
- Out of tree modules → built in functionally
- Sandboxing

Accessibility in GNOME needs to change too

Accessibility in GNOME needs to change too

Consolidate the effort

- Consolidate the effort
- Simplify the toolkit

- Consolidate the effort
- Simplify the toolkit
- Empower application developers

- Consolidate the effort
- Simplify the toolkit
- Empower application developers
- Funding

New accessibility stack

WAI-ARIA

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

- **W3C** standard
- element: an accessible part of the UI
- vole: what the element does

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

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

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

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

ℰ element: GtkAccessible

₹ element: GtkAccessible

▼ role: GtkAccessibleRole

- **₹ element**: GtkAccessible
- **▼ role**: GtkAccessibleRole
- attributes

- **₹ element**: GtkAccessible
- **▼ role**: GtkAccessibleRole
- attributes
 - **ℰ** GtkAccessibleProperty

- **₹ element**: GtkAccessible
- **▼ role**: GtkAccessibleRole
- attributes
 - **ℰ** GtkAccessibleProperty
 - **ℰ** GtkAccessibleRelation

- **₹ element**: GtkAccessible
- **▼ role**: GtkAccessibleRole
- attributes
 - **ℰ** GtkAccessibleProperty
 - **ℰ** GtkAccessibleRelation
 - **₹** GtkAccessibleState

- **₹ element**: GtkAccessible
- **▼ role**: GtkAccessibleRole
- attributes
 - **ℰ** GtkAccessibleProperty
 - **ℰ** GtkAccessibleRelation
 - **ℰ** GtkAccessibleState
- platform-specific backend: GtkATContext

Set the role on your widget

```
// GtkCheckBox: set role

gtk_widget_class_set_accessible_role (
   widget_class,
   GTK_ACCESSIBLE_ROLE_CHECKBOX
);
```

Change states

```
// GtkCheckButton: state change

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

Change properties/1

```
// GtkEntry → accessible label
gtk_accessible_update_property (
   GTK_ACCESSIBLE (username_entry),
   GTK_ACCESSIBLE_PROPERTY_LABEL, "User name",
   -1
);
```

Change properties/2

```
// 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

1st: Reuse existing accessible UI elements

- 1st: Reuse existing accessible UI elements
- 2nd: Do not change semantics

- 1st: Reuse existing accessible UI elements
- 2nd: Do not change semantics
- 3rd: Keyboard accessibility

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

- 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

gtk-builder-tool

- gtk-builder-tool
- **ℰ** GtkTestATContext

- gtk-builder-tool
- **ℰ** GtkTestATContext
- gtk_test_accessible_assert_*

Finalising the API

- Finalising the API
- Porting GTK widgets

- Finalising the API
- Porting GTK widgets
- Writing tests

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

- Finalising the API
- Porting GTK widgets
- Writing tests
- Documentation
- Implementing the AT-SPI backend

Help writing tests

- Help writing tests
- Documentation

- Help writing tests
- Documentation
- Port to Windows/macOS

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

- 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!