About me

Frederik Gladhorn

KDE

Digia

Accessible Applications with Qt

Qt Developer Days 2012

Frederik Gladhorn <frederik.gladhorn@digia.com>

Outline

Definition of Accessibility

Overview of APIs and Platforms

Making Applications Accessible

Tools

Summary

Accessibility

“degree to which a product, device, service, or environment is available to as many people as possible”

http://en.wikipedia.org/wiki/Accessibility

Enable people to
inform themselves
and participate in society

Legal

- Rehabilitation Act (1973) amended with section 504 and 508
- Americans with Disabilities Act (1990)
- Special Educational Needs and Disability Act (2001)
- EU Charter of Fundamental Rights

Assistive Technology

Virtual Keyboards

Screen Reader

Braille

Speech recognition

Special fonts

Example: OpenDyslexic

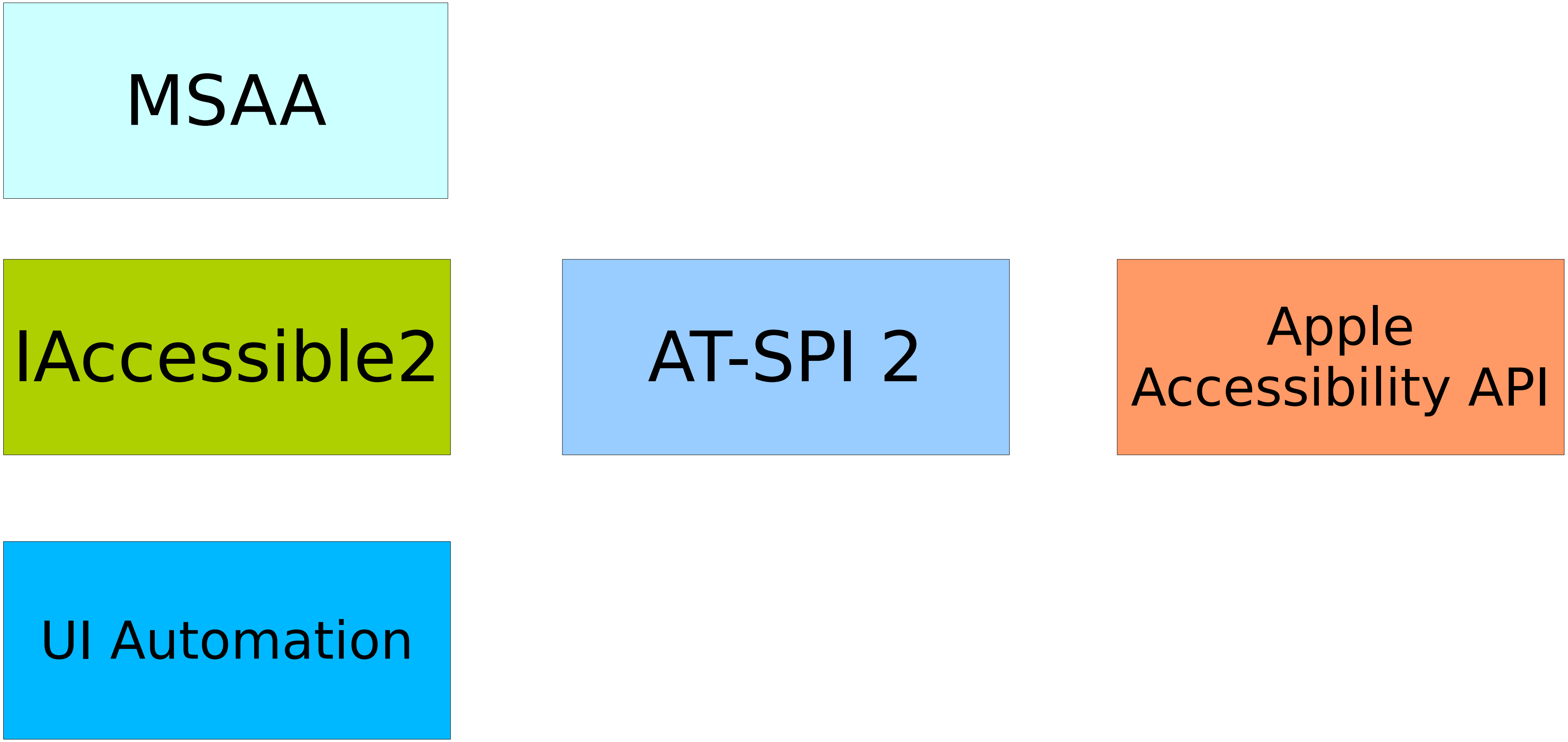
Allow platform or custom fonts!

http://dyslexicfonts.com/

APIs and Platforms

SAPPORO
13410 km
AVIEMOR
QUEENS
10689 km
ARE
10602 km
VERBIER
CERVINIA
8894 km
ZURS
8892 km
SKI SHOP
7 m
Craigh
RESERVATIONS
47 m
South Africa
SKI DECK
Craighall
MATT
8937 km

Accessibility APIs



MSAA

IAccessible2

AT-SPI 2

Apple Accessibility API

UI Automation

Qt

`QWidget::accessibleName` (and Description)

`QAccessibleInterface` (improved in Qt 5)

Qt Quick: Accessible (starting with Qt 5)

No external dependencies

Except Linux and Qt 4.8:
`qtatspi/qt-at-spi`

Making Applications Accessible

Colors

Fonts

Keyboard navigation

Accessible Name

Demo - Calculator

In Code

```
QWidget::setAccessibleName()
```

QAccessibleInterface

Important Properties

Name

Description

Role

State

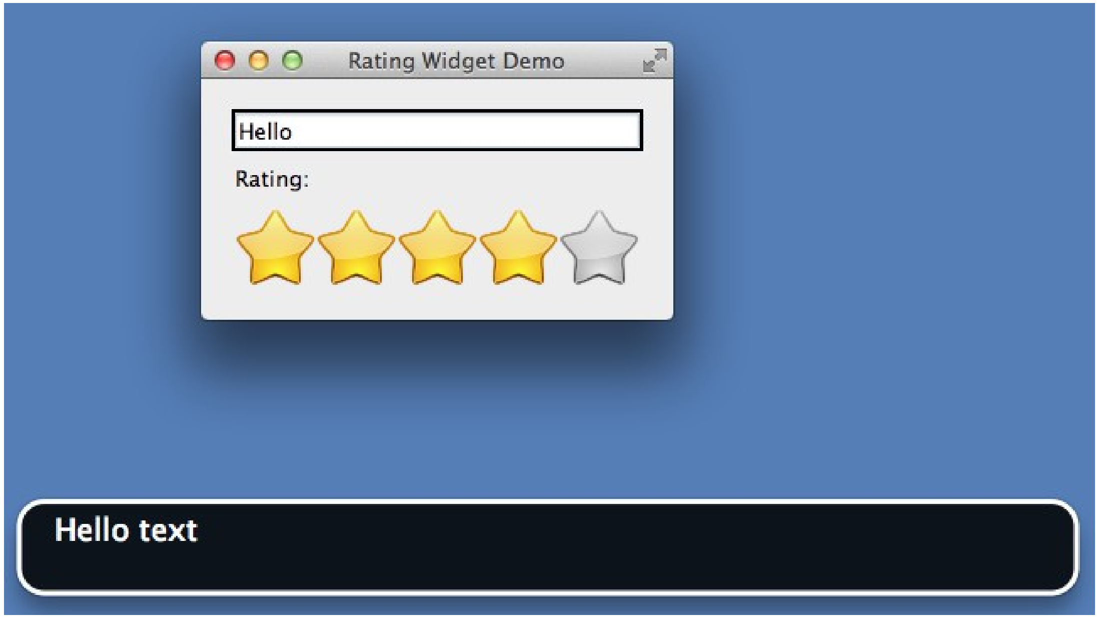
Demo – Rating Widget

Demo – Rating Widget in QML

Tools

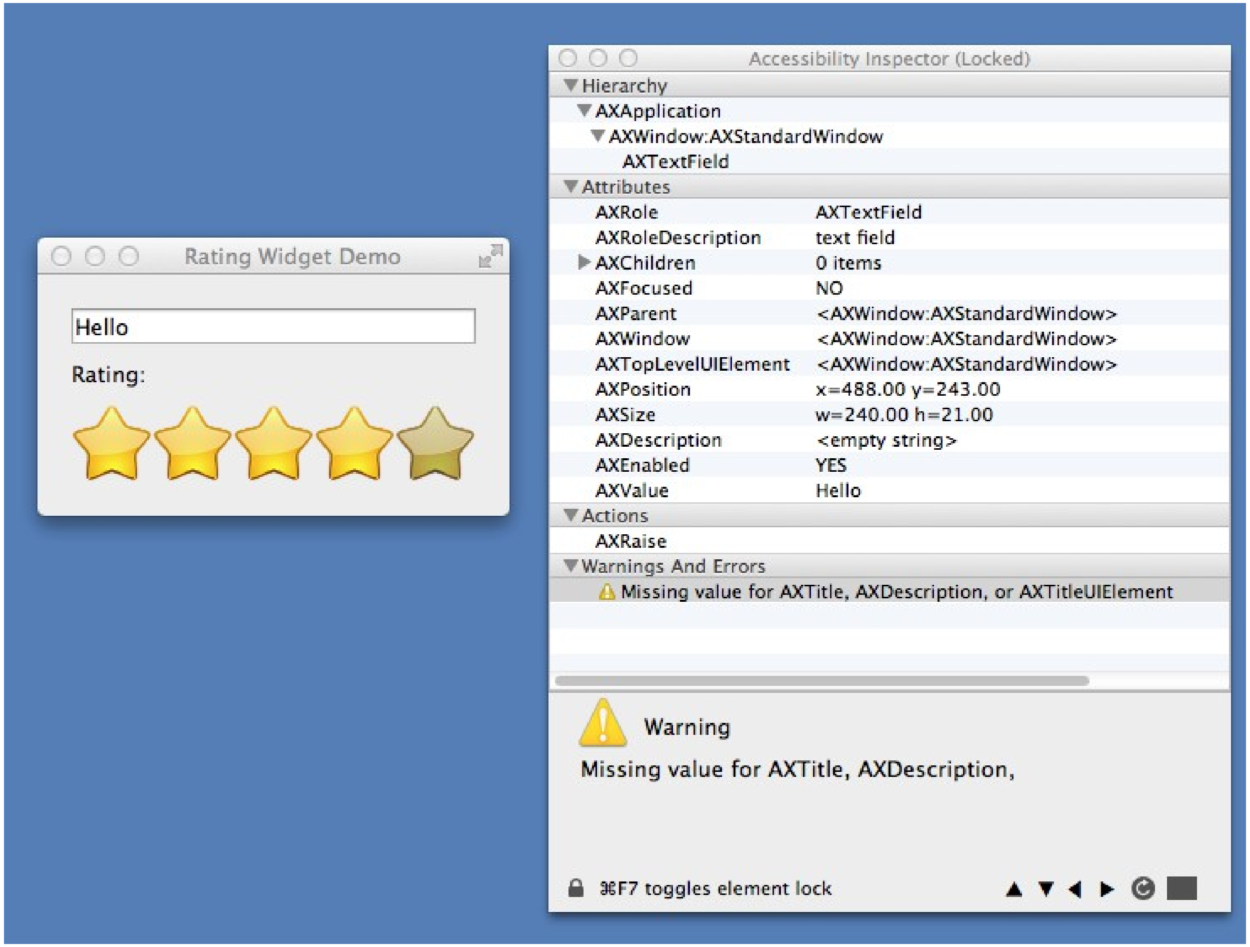
Mac Tools

Voice Over



Hello text

Accessibility Inspector



Hierarchy

- AXApplication
  - AXWindow:AXStandardWindow
    - AXTextField

Attributes

AXRole	AXTextField
AXRoleDescription	text field
AXChildren	0 items
AXFocused	NO
AXParent	<AXWindow:AXStandardWindow>
AXWindow	<AXWindow:AXStandardWindow>
AXTopLevelUIElement	<AXWindow:AXStandardWindow>
AXPosition	x=488.00 y=243.00
AXSize	w=240.00 h=21.00
AXDescription	<empty string>
AXEnabled	YES
AXValue	Hello

Actions

AXRaise

Warnings And Errors

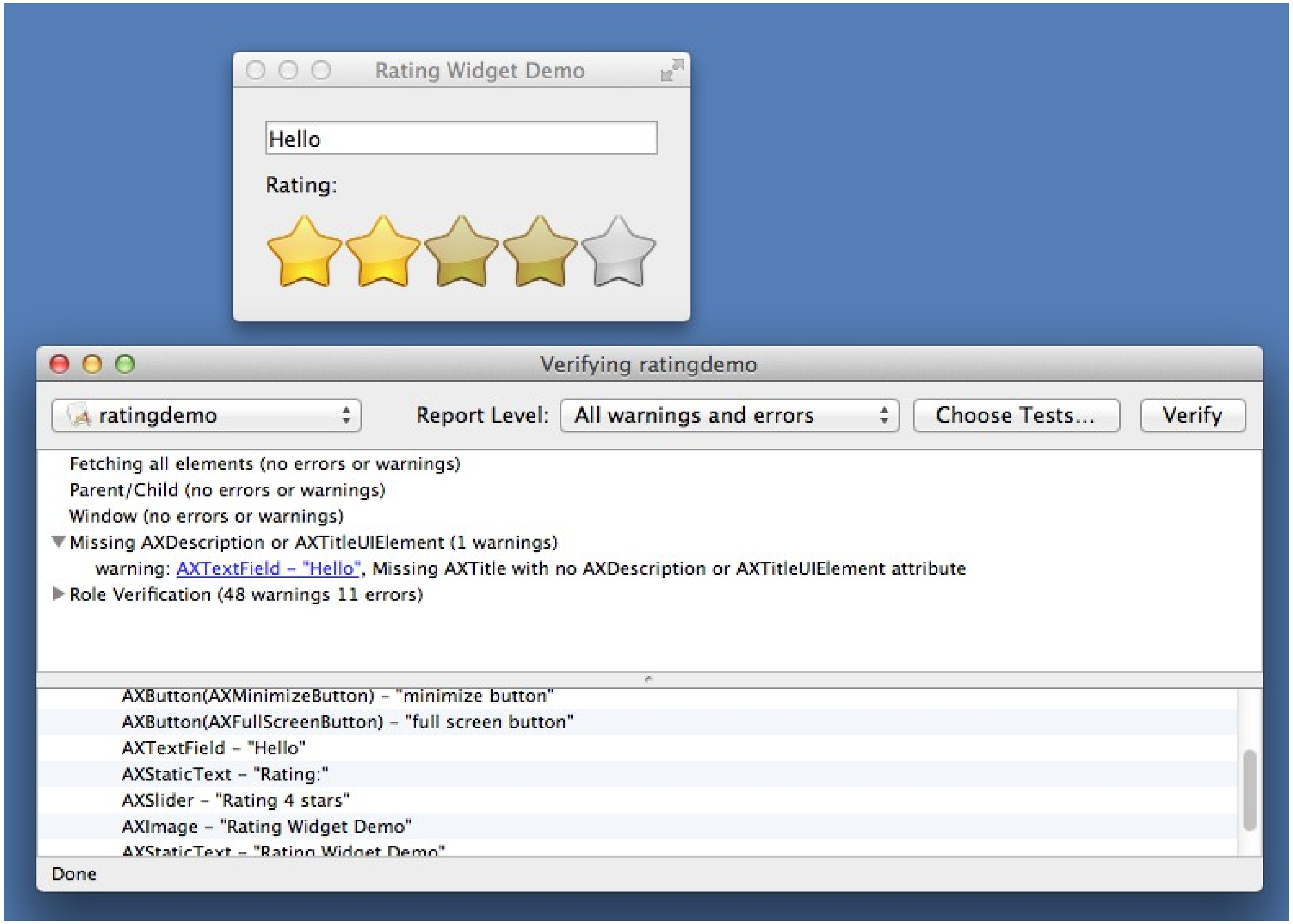
⚠ Missing value for AXTitle, AXDescription, or AXTitleUIElement

⚠ Warning

Missing value for AXTitle, AXDescription,

⌘F7 toggles element lock

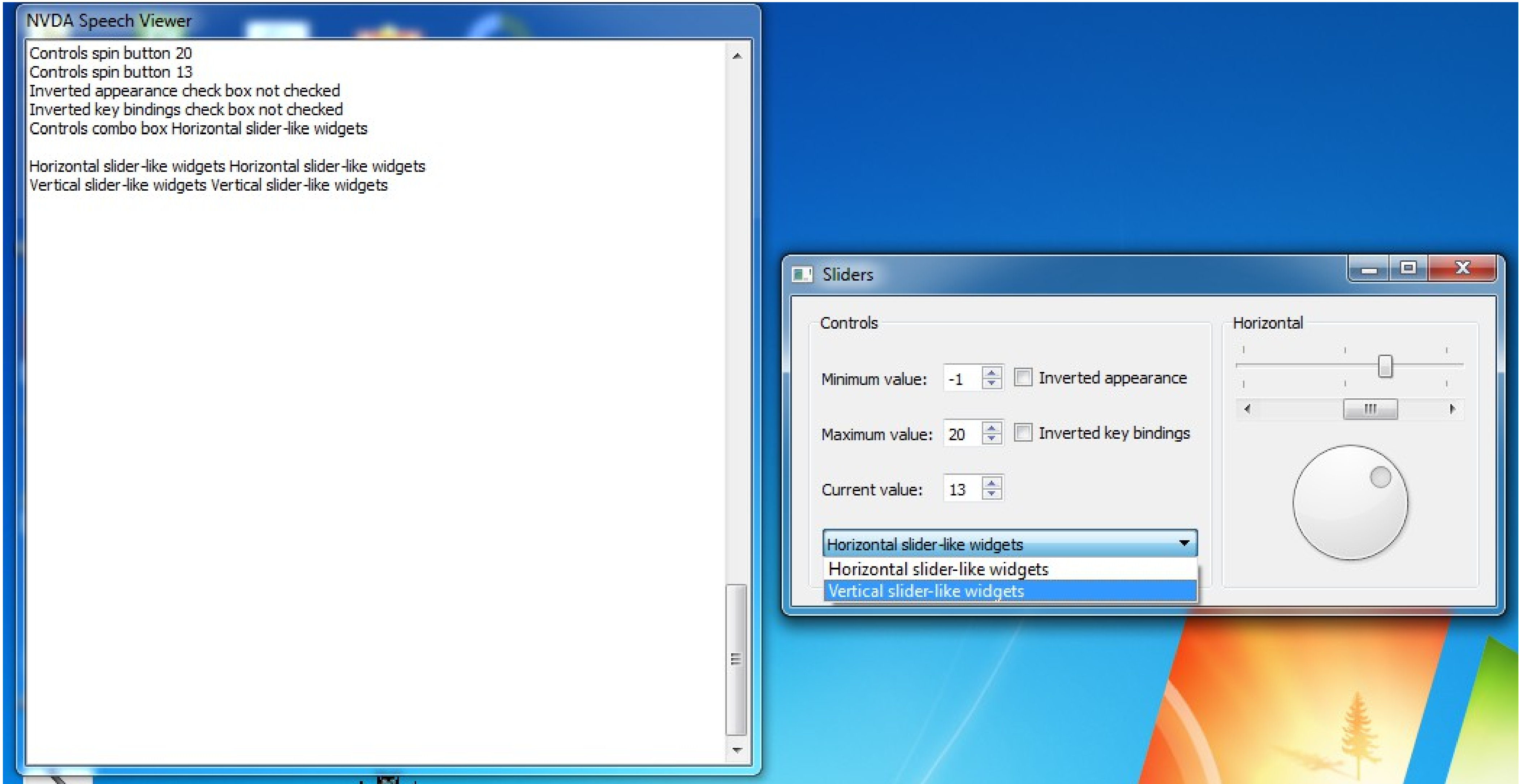
Accessibility Verifier



```
Fetching all elements (no errors or warnings)
Parent/Child (no errors or warnings)
Window (no errors or warnings)
▼Missing AXDescription or AXTitleUIElement (1 warnings)
    warning: AXTextField – "Hello", Missing AXTitle with no AXDescription or AXTitleUIElement attribute
▶Role Verification (48 warnings 11 errors)
```

```
        AXButton(AXMinimizeButton) – "minimize button"
        AXButton(AXFullScreenButton) – "full screen button"
        AXTextField – "Hello"
        AXStaticText – "Rating:"
        AXSlider – "Rating 4 stars"
        AXImage – "Rating Widget Demo"
        AXStaticText – "Rating Widget Demo"
```

NVDA



NVDA Speech Viewer

```
Controls spin button 20
Controls spin button 13
Inverted appearance check box not checked
Inverted key bindings check box not checked
Controls combo box Horizontal slider-like widgets

Horizontal slider-like widgets Horizontal slider-like widgets
Vertical slider-like widgets Vertical slider-like widgets
```

Sliders

Controls

Minimum value: -1 □ Inverted appearance

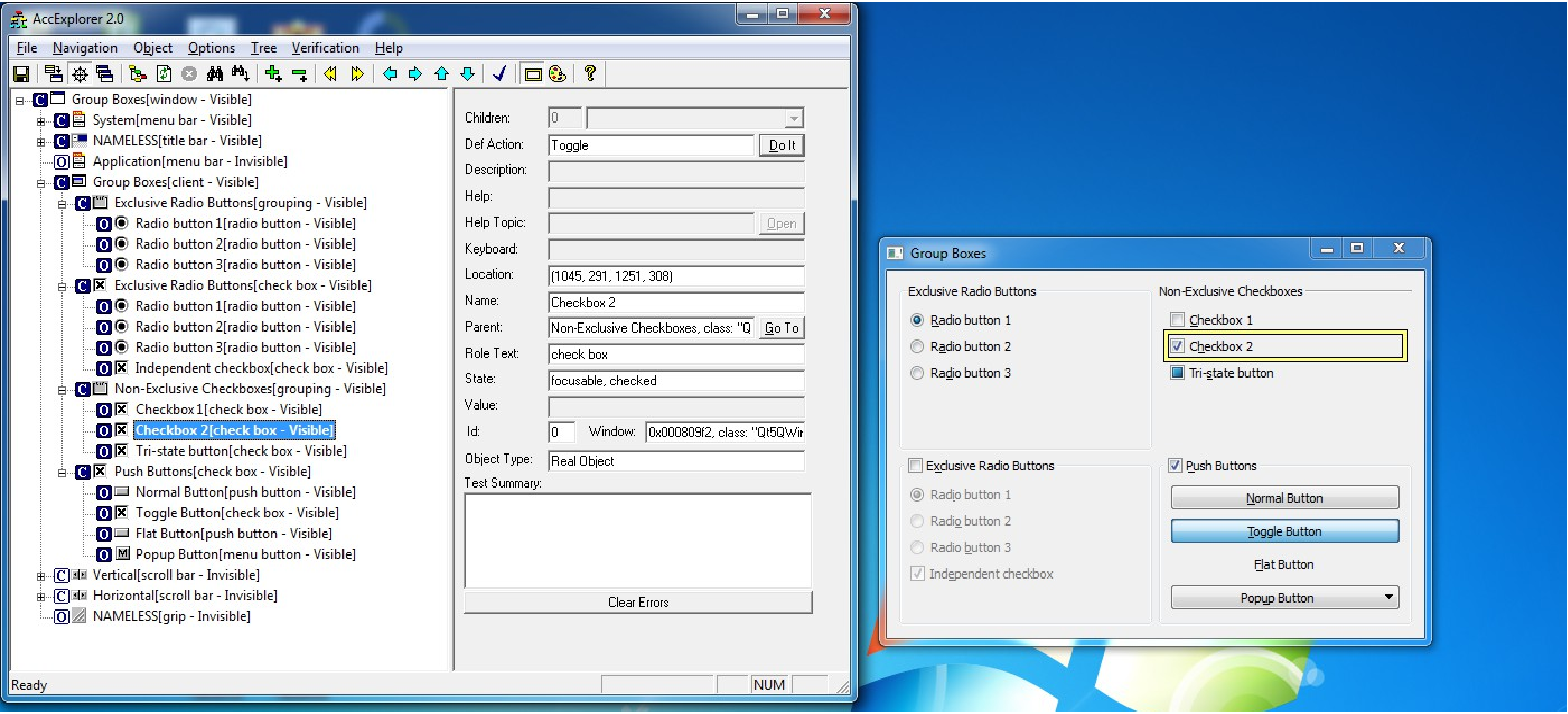
Maximum value: 20 □ Inverted key bindings

Current value: 13

Horizontal slider-like widgets

- Horizontal slider-like widgets
- Vertical slider-like widgets

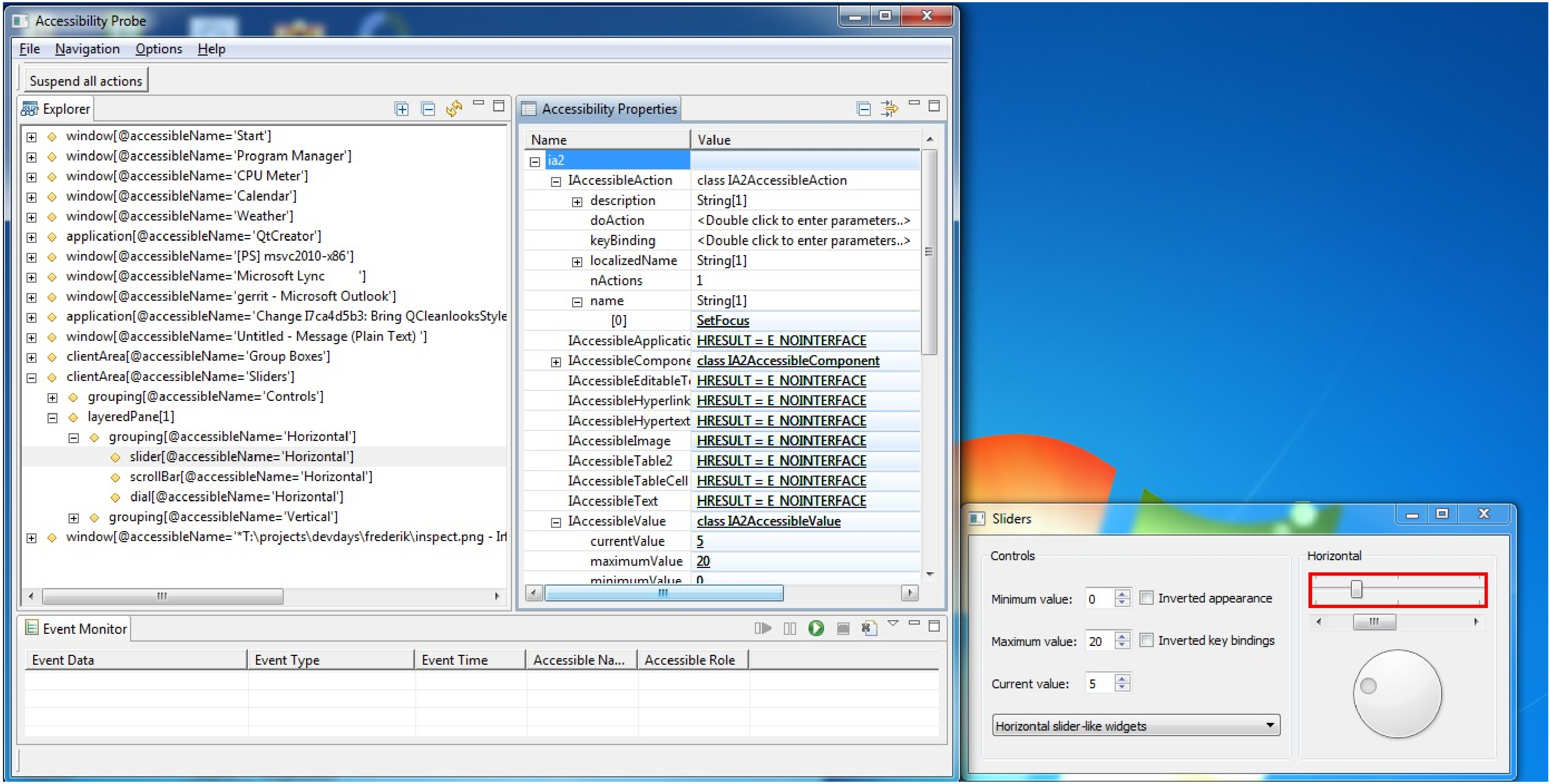
Acc Explorer



```
Group Boxes[window - Visible]
  System[menu bar - Visible]
  NAMELESS[title bar - Visible]
  Application[menu bar - Invisible]
  Group Boxes[client - Visible]
    Exclusive Radio Buttons[grouping - Visible]
      Radio button 1[radio button - Visible]
      Radio button 2[radio button - Visible]
      Radio button 3[radio button - Visible]
    Exclusive Radio Buttons[check box - Visible]
      Radio button 1[radio button - Visible]
      Radio button 2[radio button - Visible]
      Radio button 3[radio button - Visible]
      Independent checkbox[check box - Visible]
    Non-Exclusive Checkboxes[grouping - Visible]
      Checkbox 1[check box - Visible]
      Checkbox 2[check box - Visible]
      Tri-state button[check box - Visible]
    Push Buttons[check box - Visible]
      Normal Button[push button - Visible]
      Toggle Button[check box - Visible]
      Flat Button[push button - Visible]
      Popup Button[menu button - Visible]
  Vertical[scroll bar - Invisible]
  Horizontal[scroll bar - Invisible]
  NAMELESS[grip - Invisible]
```

Children:	0
Def Action:	Toggle
Description:	
Help:	
Help Topic:	
Keyboard:	
Location:	(1045, 291, 1251, 308)
Name:	Checkbox 2
Parent:	Non-Exclusive Checkboxes, class: "Q
Role Text:	check box
State:	focusable, checked
Value:	
Id:	0
Window:	0x000809f2, class: "Qt5QWi
Object Type:	Real Object
Test Summary:	

Accessibility Probe



```
⊞ window[@accessibleName='Start']
⊞ window[@accessibleName='Program Manager']
⊞ window[@accessibleName='CPU Meter']
⊞ window[@accessibleName='Calendar']
⊞ window[@accessibleName='Weather']
⊞ application[@accessibleName='QtCreator']
⊞ window[@accessibleName='[PS] msvc2010-x86']
⊞ window[@accessibleName='Microsoft Lync       ']
⊞ window[@accessibleName='gerrit - Microsoft Outlook']
⊞ application[@accessibleName='Change I7ca4d5b3: Bring QCleanlooksStyle
⊞ window[@accessibleName='Untitled - Message (Plain Text) ']
⊞ clientArea[@accessibleName='Group Boxes']
⊟ clientArea[@accessibleName='Sliders']
    ⊞ grouping[@accessibleName='Controls']
    ⊟ layeredPane[1]
        ⊟ grouping[@accessibleName='Horizontal']
              slider[@accessibleName='Horizontal']
              scrollBar[@accessibleName='Horizontal']
              dial[@accessibleName='Horizontal']
        ⊞ grouping[@accessibleName='Vertical']
⊞ window[@accessibleName='*T:\projects\devdays\frederik\inspect.png - Ir
```

Name	Value
⊟ ia2	
⊟ IAccessibleAction	class IA2AccessibleAction
⊞ description	String[1]
doAction	<Double click to enter parameters..>
keyBinding	<Double click to enter parameters..>
⊞ localizedName	String[1]
nActions	1
⊟ name	String[1]
[0]	SetFocus
IAccessibleApplicatic	HRESULT = E_NOINTERFACE
⊞ IAccessibleCompone	class IA2AccessibleComponent
IAccessibleEditableT	HRESULT = E_NOINTERFACE
IAccessibleHyperlink	HRESULT = E_NOINTERFACE
IAccessibleHypertext	HRESULT = E_NOINTERFACE
IAccessibleImage	HRESULT = E_NOINTERFACE
IAccessibleTable2	HRESULT = E_NOINTERFACE
IAccessibleTableCell	HRESULT = E_NOINTERFACE
IAccessibleText	HRESULT = E_NOINTERFACE
⊟ IAccessibleValue	class IA2AccessibleValue
currentValue	5
maximumValue	20
minimumValue	0

Event Data	Event Type	Event Time	Accessible Na...	Accessible Role

Controls

Minimum value: 0 ☐ Inverted appearance

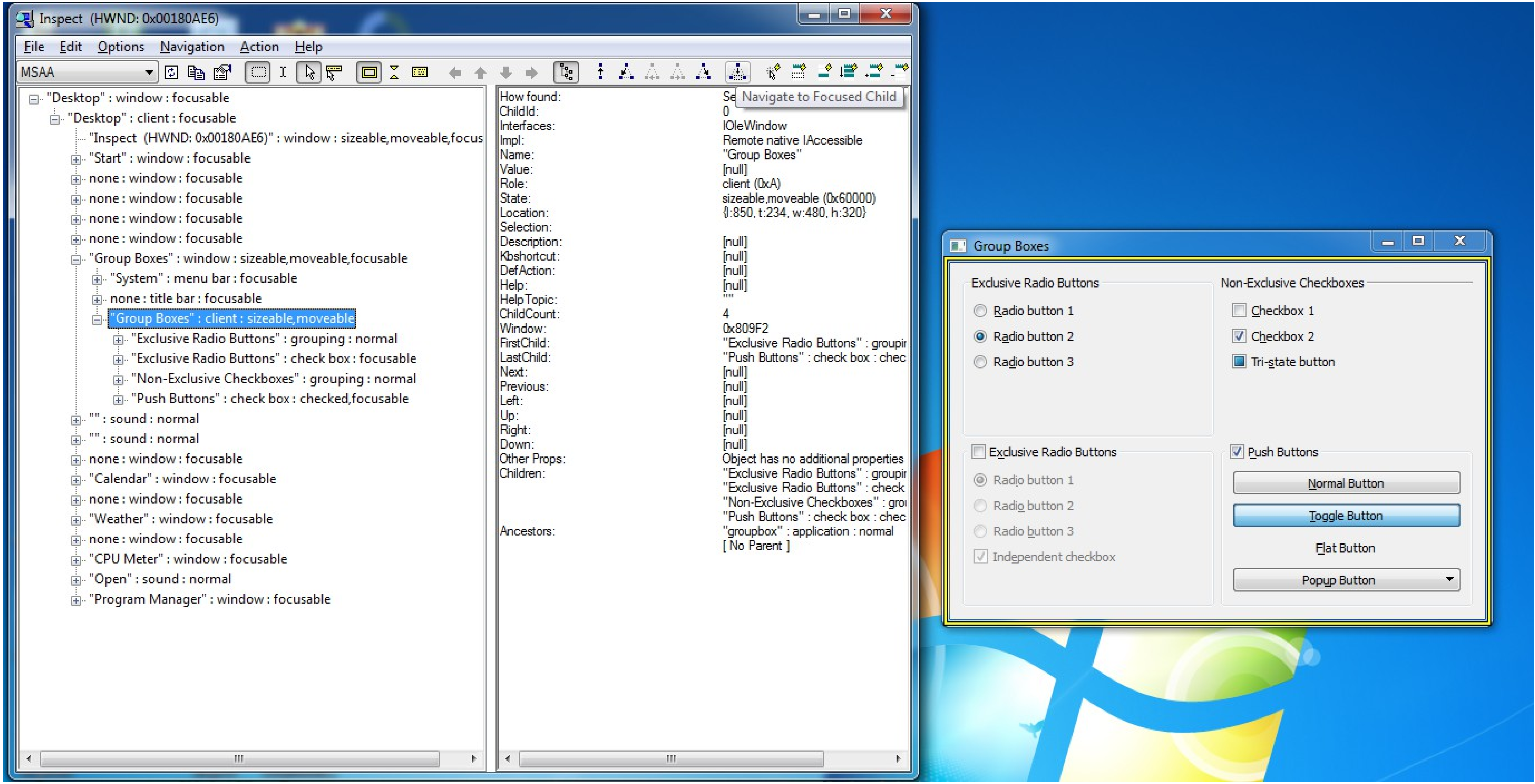
Maximum value: 20 ☐ Inverted key bindings

Current value: 5

Horizontal slider-like widgets

Horizontal

Microsoft Inspect

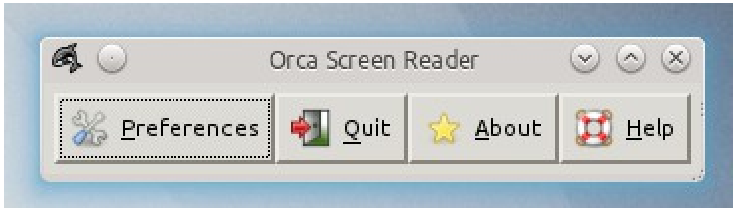


```
"Desktop" : window : focusable
  "Desktop" : client : focusable
    "Inspect  (HWND: 0x00180AE6)" : window : sizeable,moveable,focus
    "Start" : window : focusable
    none : window : focusable
    none : window : focusable
    none : window : focusable
    none : window : focusable
    "Group Boxes" : window : sizeable,moveable,focusable
      "System" : menu bar : focusable
      none : title bar : focusable
      "Group Boxes" : client : sizeable,moveable
        "Exclusive Radio Buttons" : grouping : normal
        "Exclusive Radio Buttons" : check box : focusable
        "Non-Exclusive Checkboxes" : grouping : normal
        "Push Buttons" : check box : checked,focusable
    "" : sound : normal
    "" : sound : normal
    none : window : focusable
    "Calendar" : window : focusable
    none : window : focusable
    "Weather" : window : focusable
    none : window : focusable
    "CPU Meter" : window : focusable
    "Open" : sound : normal
    "Program Manager" : window : focusable
```

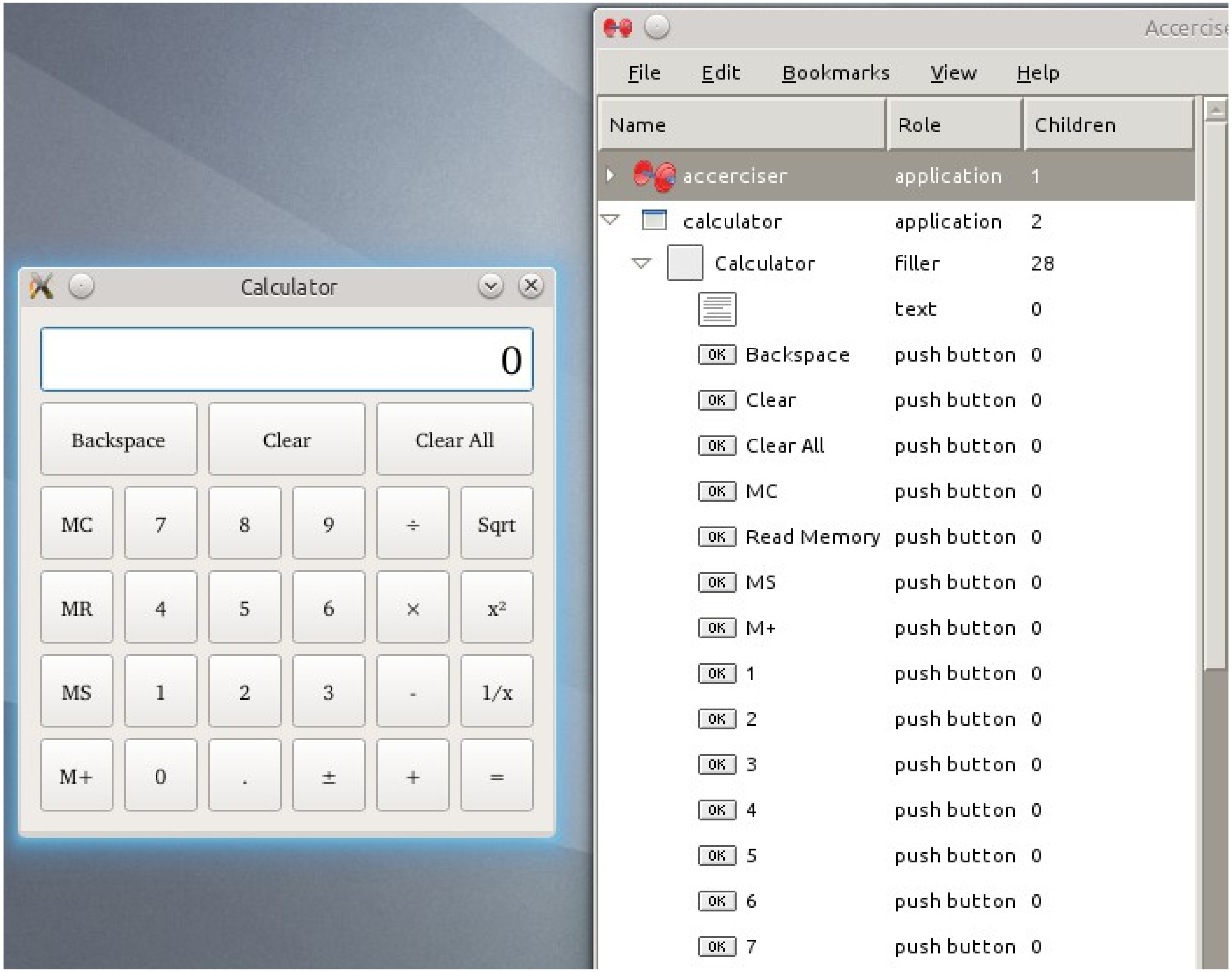
```
How found:     Se
ChildId:       0
Interfaces:    IOleWindow
Impl:          Remote native IAccessible
Name:          "Group Boxes"
Value:         [null]
Role:          client (0xA)
State:         sizeable,moveable (0x60000)
Location:      {l:850, t:234, w:480, h:320}
Selection:
Description:   [null]
Kbshortcut:    [null]
DefAction:     [null]
Help:          [null]
HelpTopic:     ""
ChildCount:    4
Window:        0x809F2
FirstChild:    "Exclusive Radio Buttons" : groupir
LastChild:     "Push Buttons" : check box : chec
Next:          [null]
Previous:      [null]
Left:          [null]
Up:            [null]
Right:         [null]
Down:          [null]
Other Props:   Object has no additional properties
Children:      "Exclusive Radio Buttons" : groupir
               "Exclusive Radio Buttons" : check
               "Non-Exclusive Checkboxes" : grou
               "Push Buttons" : check box : chec
Ancestors:     "groupbox" : application : normal
               [ No Parent ]
```

Linux Tools

Orca

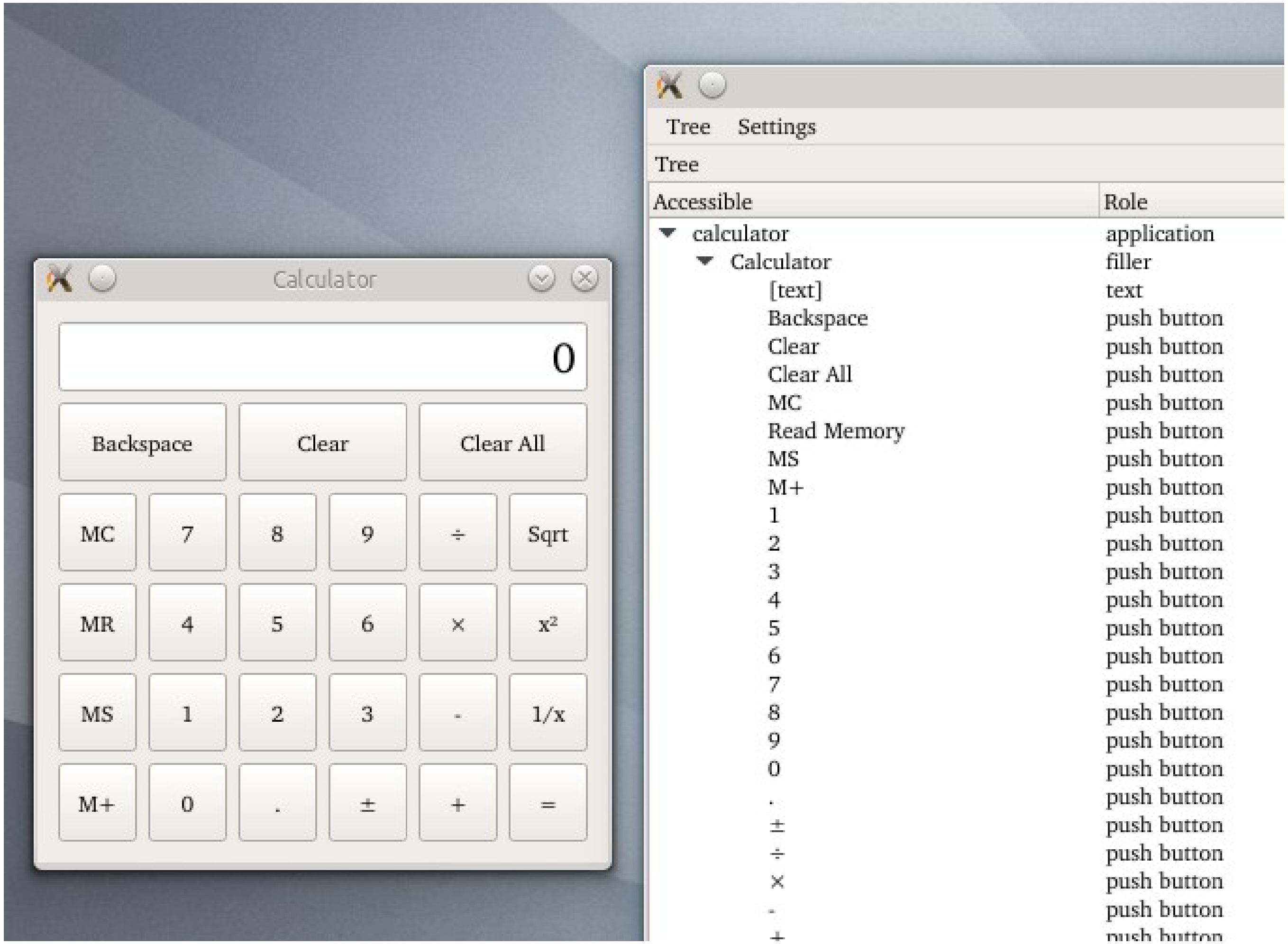


Accerciser



Name	Role	Children
accerciser	application	1
calculator	application	2
Calculator	filler	28
	text	0
Backspace	push button	0
Clear	push button	0
Clear All	push button	0
MC	push button	0
Read Memory	push button	0
MS	push button	0
M+	push button	0
1	push button	0
2	push button	0
3	push button	0
4	push button	0
5	push button	0
6	push button	0
7	push button	0

Randamizer



Accessible	Role
calculator	application
Calculator	filler
[text]	text
Backspace	push button
Clear	push button
Clear All	push button
MC	push button
Read Memory	push button
MS	push button
M+	push button
1	push button
2	push button
3	push button
4	push button
5	push button
6	push button
7	push button
8	push button
9	push button
0	push button
.	push button
±	push button
÷	push button
×	push button
-	push button
+	push button

Summary

Check Your Application

Colors, Fonts

Keyboard navigation

Screen Reader

Future Work

Qt Quick

Webkit

Questions?

Frederik Gladhorn <frederik.gladhorn@digia.com>

Differences Qt 4/5

- Improved notifications
  - Text
  - Tables
- Interface implementation much simpler
- IAccessible2 on Windows

About me

Frederik Gladhorn

KDE

Digia

German, from Bremerhaven, living in Oslo

University Stuttgart, Technical Cybernetics
Active in KDE since 2005
Joined Nokia in 2010 and Digia in 2012
Focus on Accessibility in the last year

I need assistive technology in the form of my glasses.

Enjoying climbing and skiing in Norway

Accessible Applications with Qt

Qt Developer Days 2012

Frederik Gladhorn <frederik.gladhorn@digia.com>

Outline

Definition of Accessibility

Overview of APIs and Platforms

Making Applications Accessible

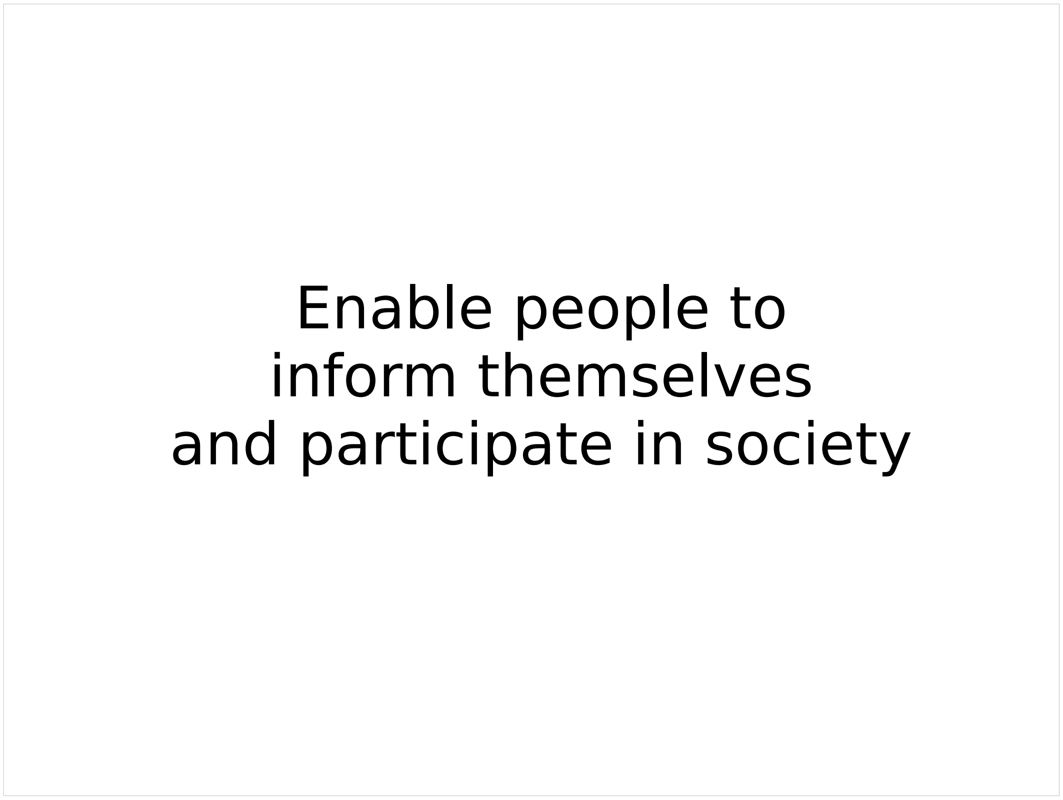
Tools

Summary

Accessibility

“degree to which a product, device, service, or environment is available to as many people as possible”

http://en.wikipedia.org/wiki/Accessibility



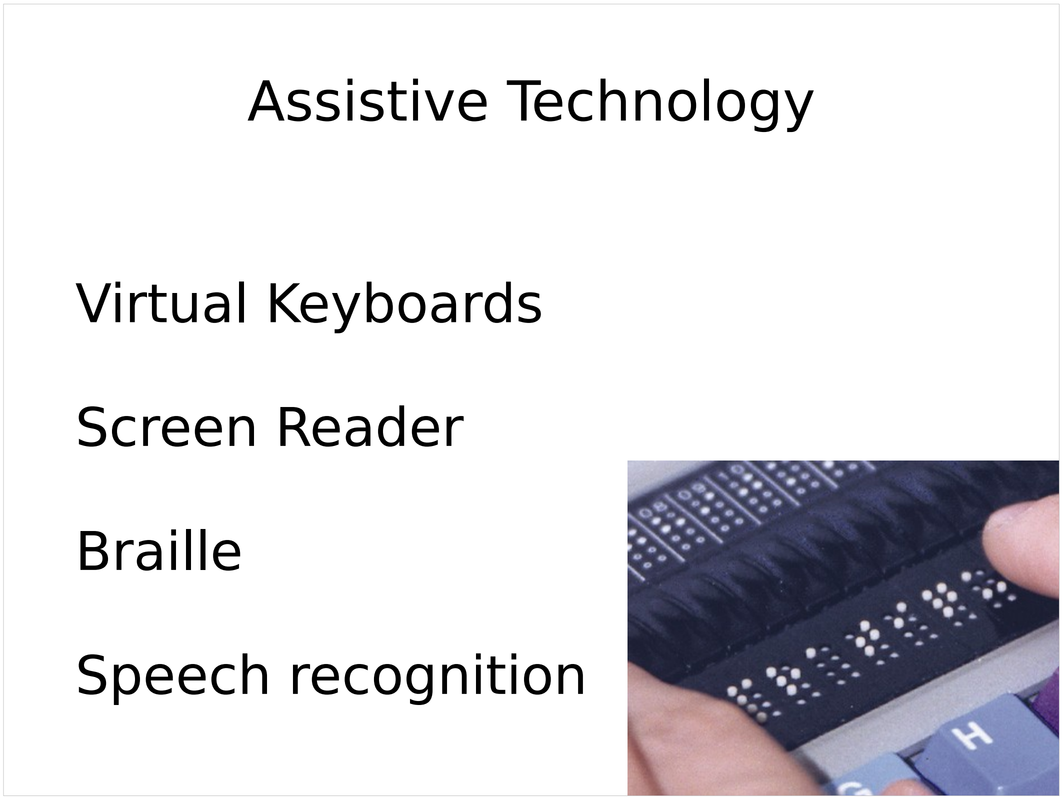
Enable people to
inform themselves
and participate in society

Study by forrester/microsoft 2003/2004:
57% of computer users between 18 and 64 could benefit from assistive technology

You should care because you make better apps and reach more people.

Legal

- Rehabilitation Act (1973) amended with section 504 and 508
- Americans with Disabilities Act (1990)
- Special Educational Needs and Disability Act (2001)
- EU Charter of Fundamental Rights



Assistive Technology

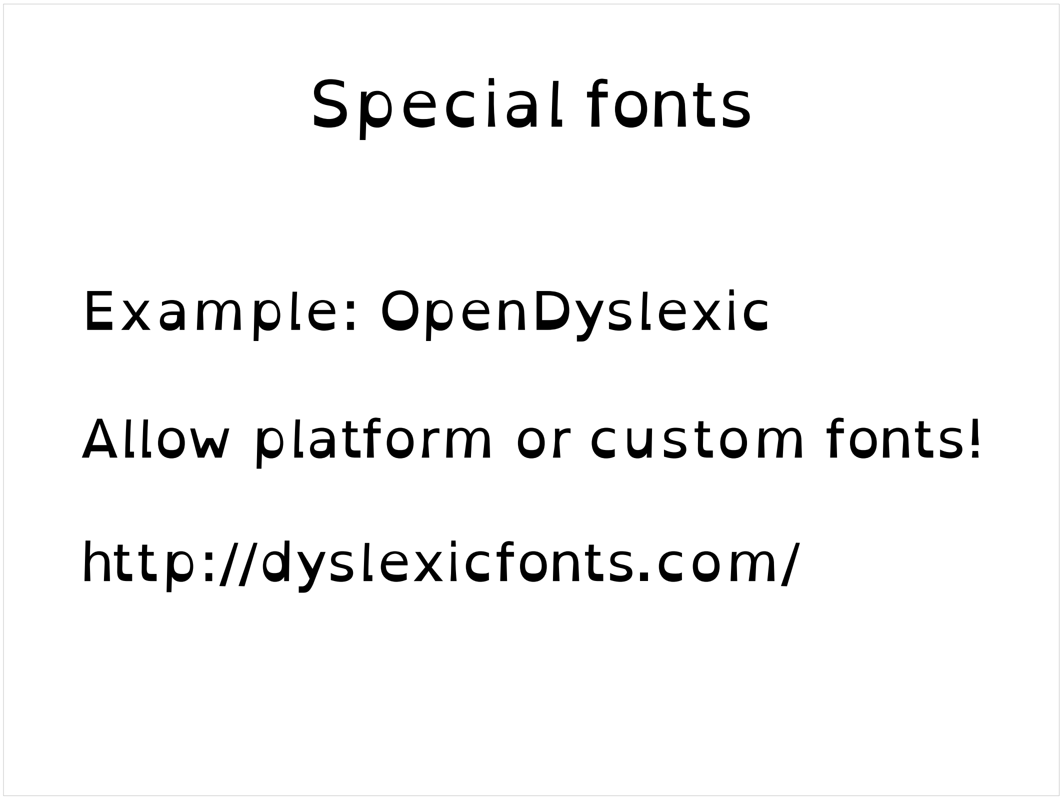
Virtual Keyboards

Screen Reader

Braille

Speech recognition

Deaf
Blind
Motorical
Dyslexia



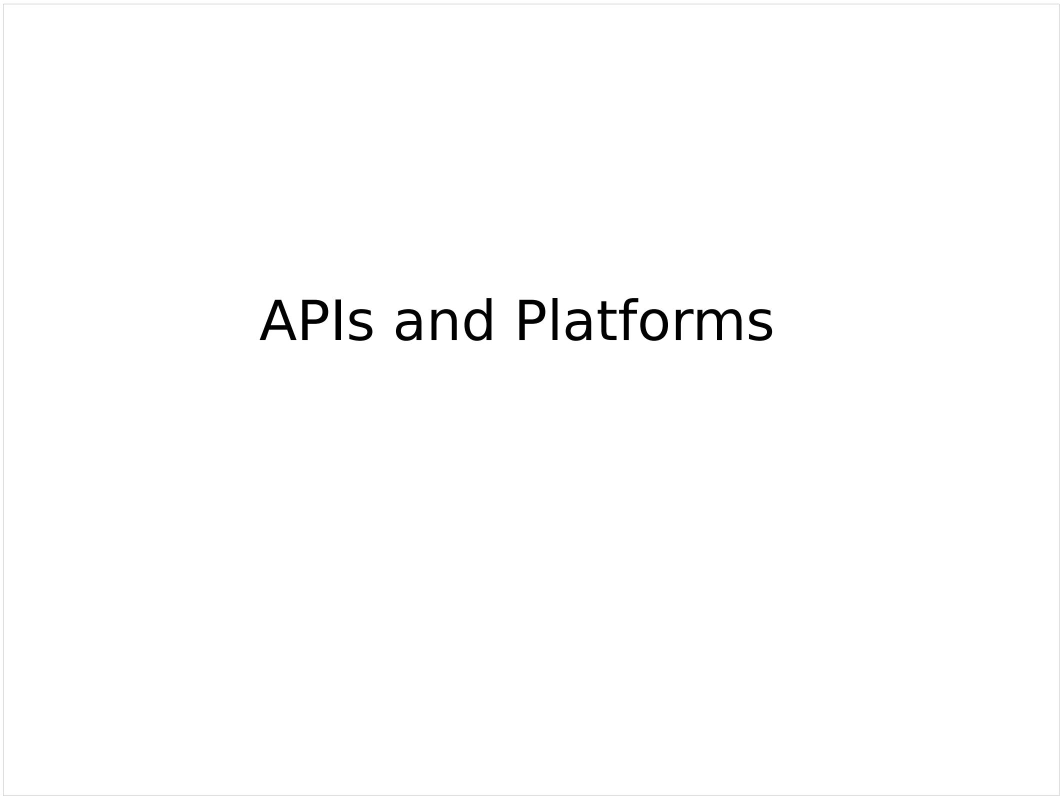
Special fonts

Example: OpenDyslexic

Allow platform or custom fonts!

http://dyslexicfonts.com/

Adds “gravity” to the letters so they don't turn around.

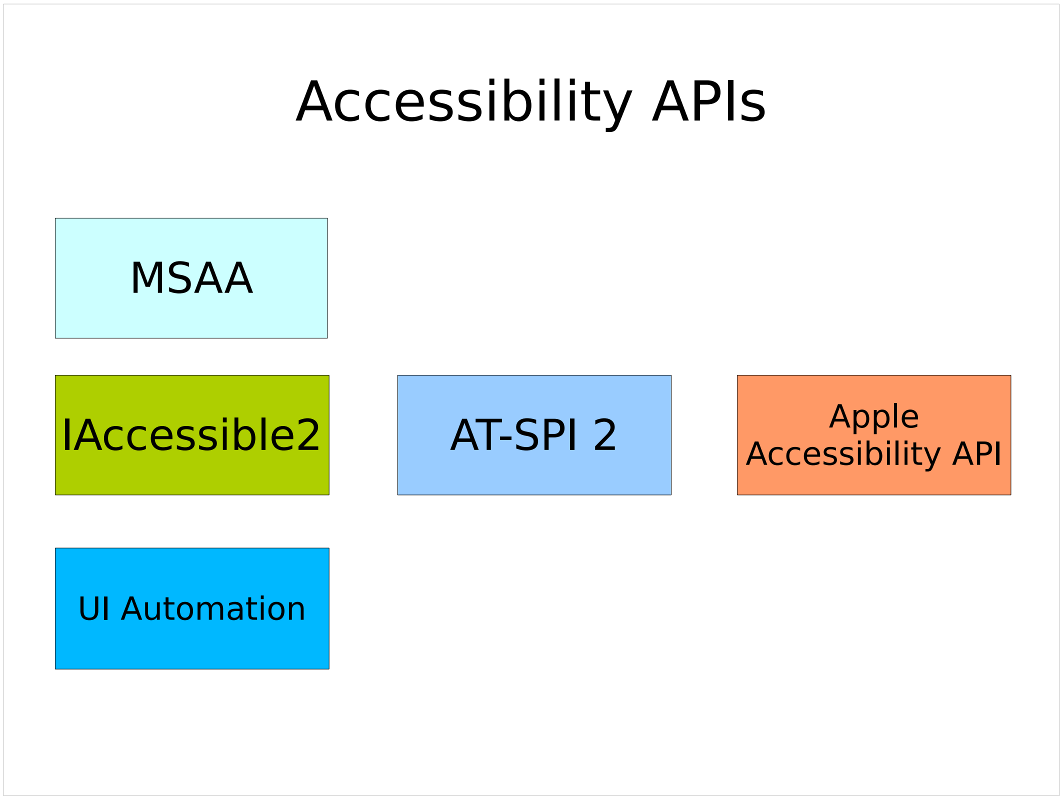


APIs and Platforms

In this section you will get a short overview of what Qt interfaces with for each plarform.

This is mostly interesting when debugging accessibility of your application.

Accessibility APIs – that's what they looked like to me in the beginning.



Accessibility APIs

MSAA

IAccessible2

AT-SPI 2

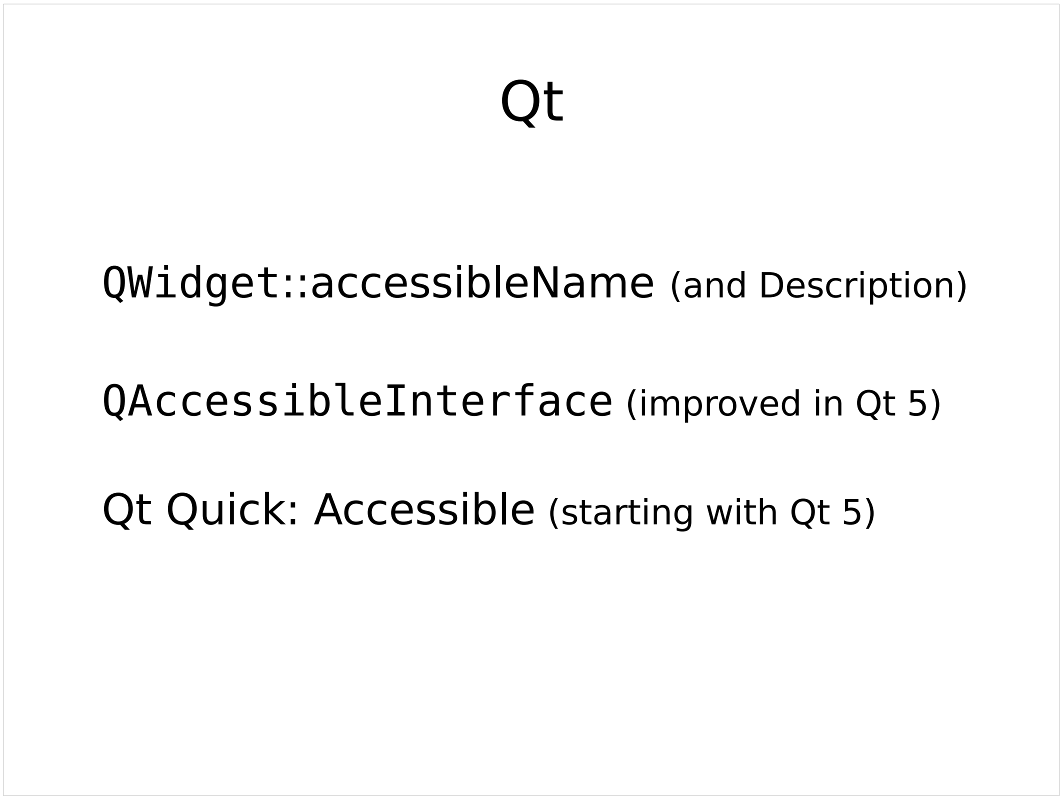
Apple Accessibility API

UI Automation

Not going to mention mobile (iOS/Android)

The left column is Windows, then Linux and Mac.

Luckily there is no need to care about these sine Qt abstracts the individual APIs away. It only matters for debugging with platform tools (see last section).



Qt

`QWidget::accessibleName` (and Description)

`QAccessibleInterface` (improved in Qt 5)

Qt Quick: Accessible (starting with Qt 5)

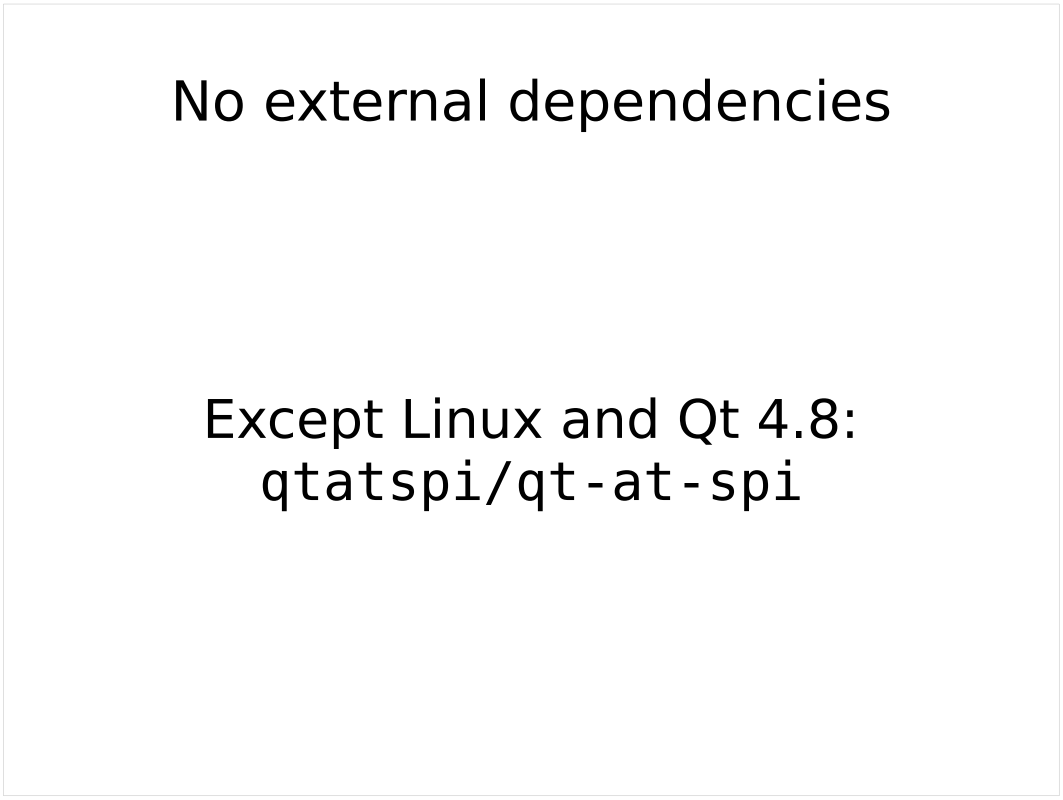
For QWidget the easy thing to improve accessibility is setting the accessibleName and accessibleDescription properties:

myWidget->setAccessibleName(“label”);

More complex solutions: subclass QAccessibleInterface

Qt Quick: attached property Accessible

Please refer to the documentation for each.



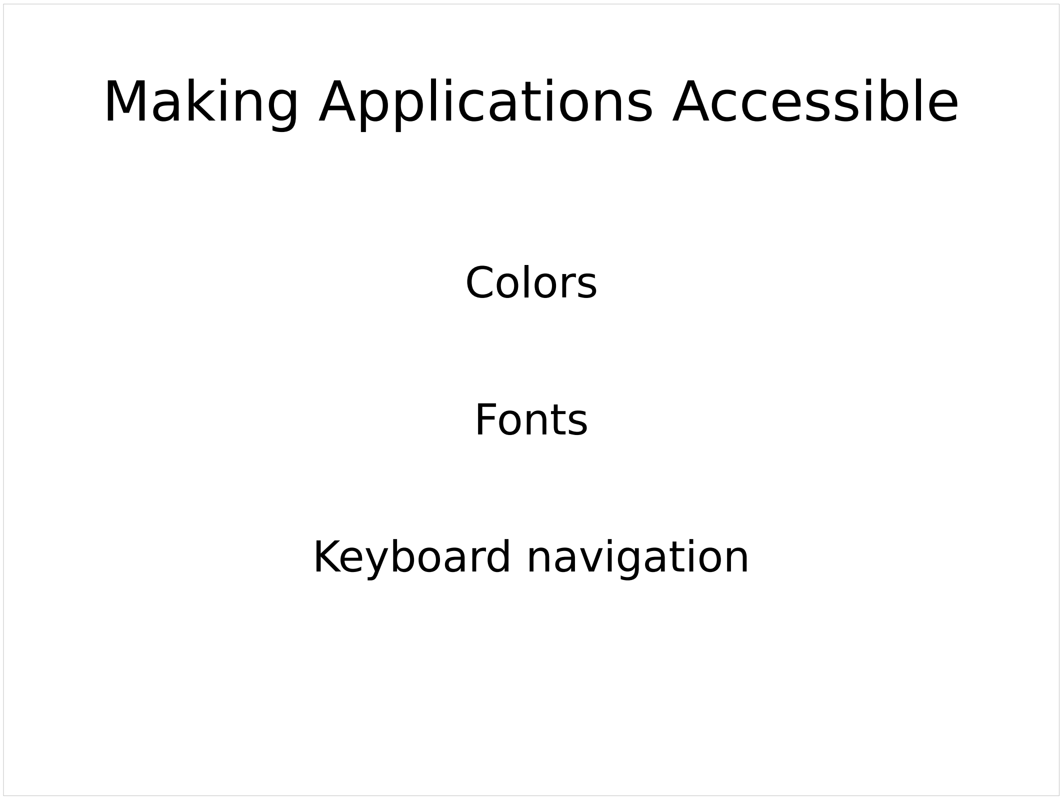
No external dependencies

Except Linux and Qt 4.8:
`qtatspi/qt-at-spi`

All accessibility features work with a Qt build since 4.8.

The one exception is Linux and Qt 4 where a plugin is needed. Almost all Linux distributions have packages for qt-at-spi. In Qt 5 the plugin has been merged and is part of qtbase.

(Much easier than a year ago where it took me two days to get it running at all.)



Making Applications Accessible

Colors

Fonts

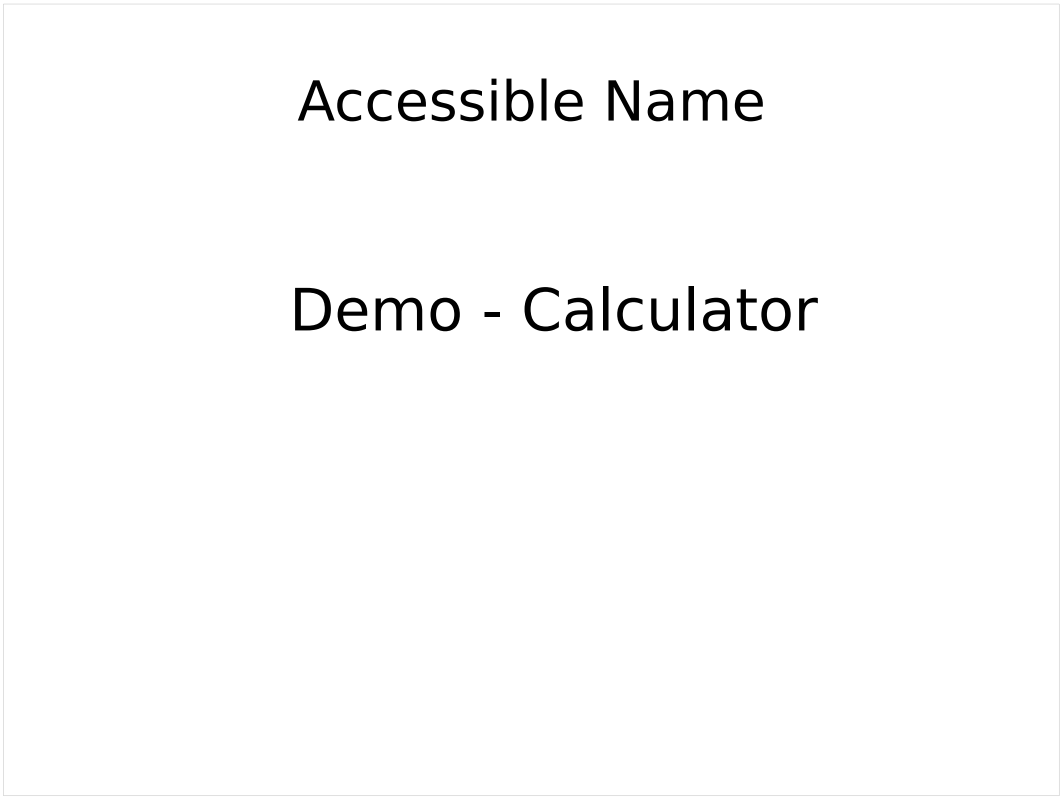
Keyboard navigation

The goal is clear: reach as many users as possible.

For that: follow platform color schemes if possible, otherwise make sure you have high enough contrasts. Everyone benefits from that.

For the fonts it's the same, go with the platform, allow big fonts and test with them.

Keyboard navigation is especially required by screen reader users. It is very important there since “pressing tab” is the primary means of discovering your application.

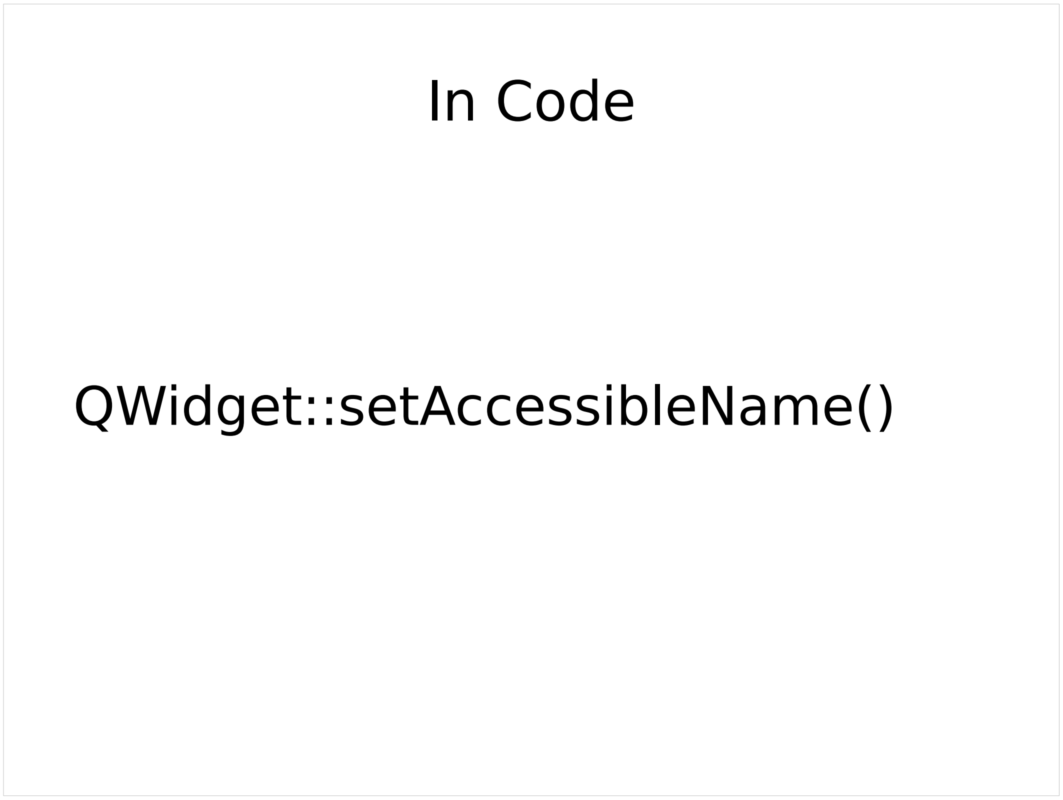


Accessible Name

Demo - Calculator

One of the examples shipped with Qt, the widget calculator one shows a funny behavior with the Orca screen reader.

The “MR” button is read as Mister. This demo shows how to fix it to read “read memory” instead by setting the AccessibleName.



In Code

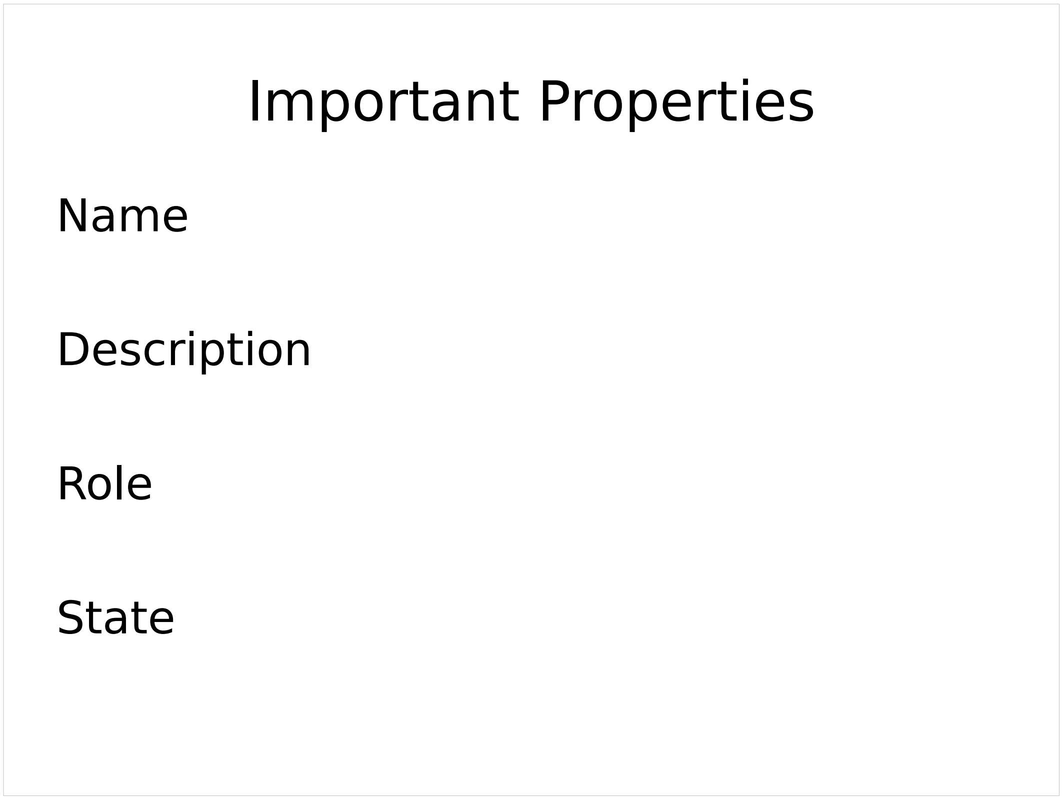
QWidget::setAccessibleName()

QAccessibleInterface

QAccessibleInterface can be subclassed for custom widgets. For convenience Qt offers QAccessibleObject/Widget.

This allows to get you started making custom widgets accessible easily.

Most important properties: Name, Description, State and Role.



Important Properties

Name

Description

Role

State

Things to keep in mind when subclassing QAccessibleInterface:

Name: short concise, should be translated and should not be the Role (“button” as name does not help, use “open” for an open button in a menu with no visible text).

Description: longer than name, gives more detailed information about an object

Role: the type of the object, see QAccessible::Role

State: the state, such as focused etc.

Demo – Rating Widget

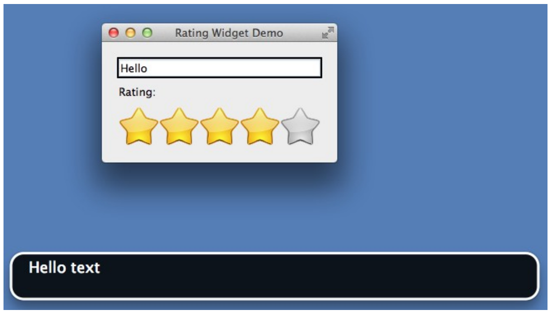
Demo – Rating Widget in QML

Tools

(Maybe blank screen)

Mac Tools

Voice Over



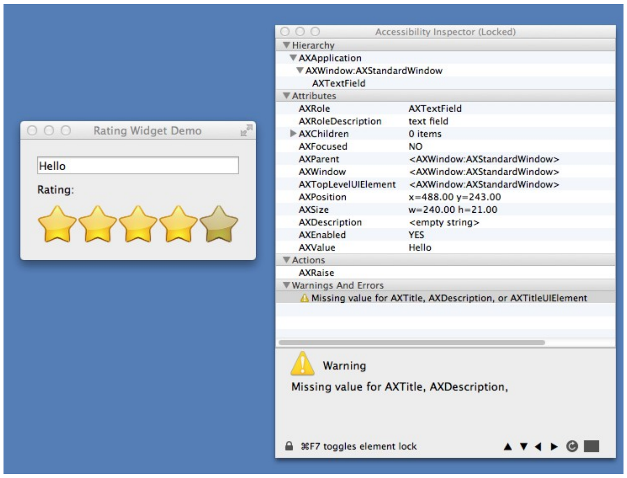
Hello

Hello text

Voice Over

Helpful for debugging: not reading but visual

Accessibility Inspector

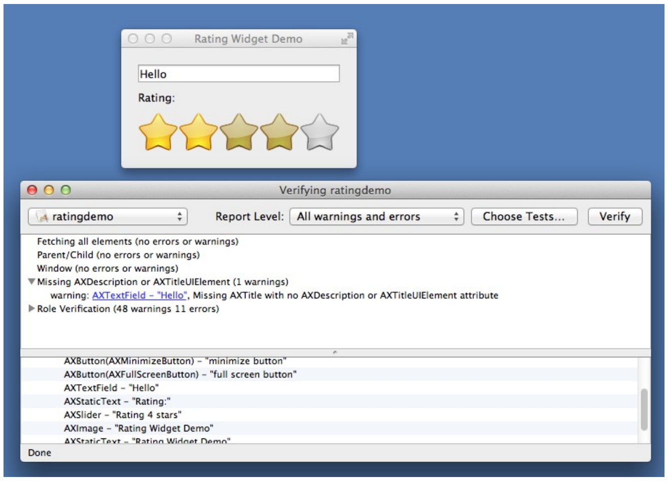


```
Accessibility Inspector (Locked)
▼Hierarchy
  ▼AXApplication
    ▼AXWindow:AXStandardWindow
       AXTextField
▼Attributes
   AXRole                AXTextField
   AXRoleDescription     text field
  ▶AXChildren            0 items
   AXFocused             NO
   AXParent              <AXWindow:AXStandardWindow>
   AXWindow              <AXWindow:AXStandardWindow>
   AXTopLevelUIElement   <AXWindow:AXStandardWindow>
   AXPosition            x=488.00 y=243.00
   AXSize                w=240.00 h=21.00
   AXDescription         <empty string>
   AXEnabled             YES
   AXValue               Hello
▼Actions
   AXRaise
▼Warnings And Errors
    ⚠ Missing value for AXTitle, AXDescription, or AXTitleUIElement

⚠ Warning
Missing value for AXTitle, AXDescription,

⌘F7 toggles element lock
```

Accessibility Verifier



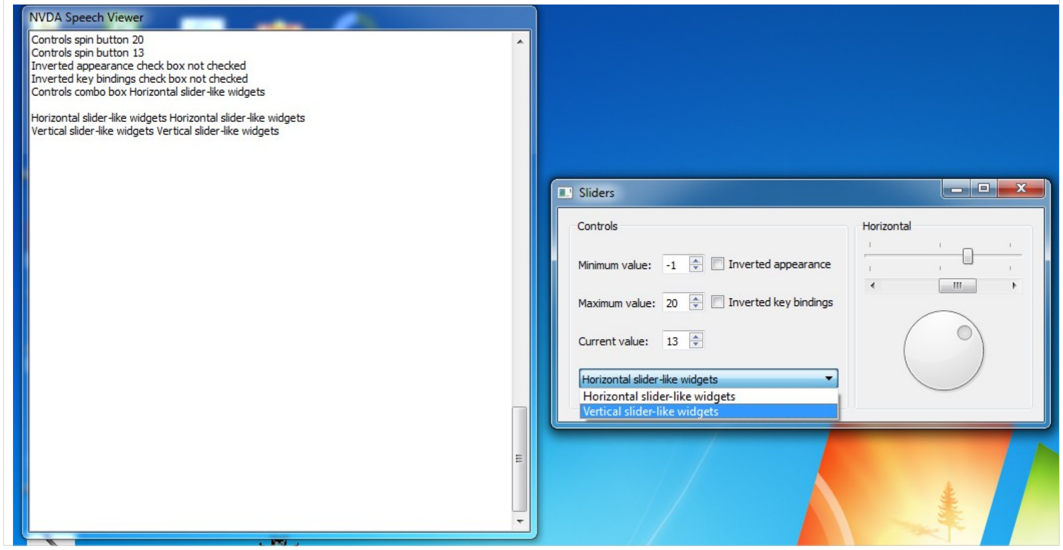
Report Level: All warnings and errors

```
  Fetching all elements (no errors or warnings)
  Parent/Child (no errors or warnings)
  Window (no errors or warnings)
▼ Missing AXDescription or AXTitleUIElement (1 warnings)
     warning: AXTextField – "Hello", Missing AXTitle with no AXDescription or AXTitleUIElement attribute
▶ Role Verification (48 warnings 11 errors)
```

```
        AXButton(AXMinimizeButton) – "minimize button"
        AXButton(AXFullScreenButton) – "full screen button"
        AXTextField – "Hello"
        AXStaticText – "Rating:"
        AXSlider – "Rating 4 stars"
        AXImage – "Rating Widget Demo"
        AXStaticText – "Rating Widget Demo"
```

Done

NVDA



NVDA Speech Viewer

```
Controls spin button 20
Controls spin button 13
Inverted appearance check box not checked
Inverted key bindings check box not checked
Controls combo box Horizontal slider-like widgets

Horizontal slider-like widgets Horizontal slider-like widgets
Vertical slider-like widgets Vertical slider-like widgets
```

Sliders

Controls

Minimum value: -1 □ Inverted appearance

Maximum value: 20 □ Inverted key bindings

Current value: 13

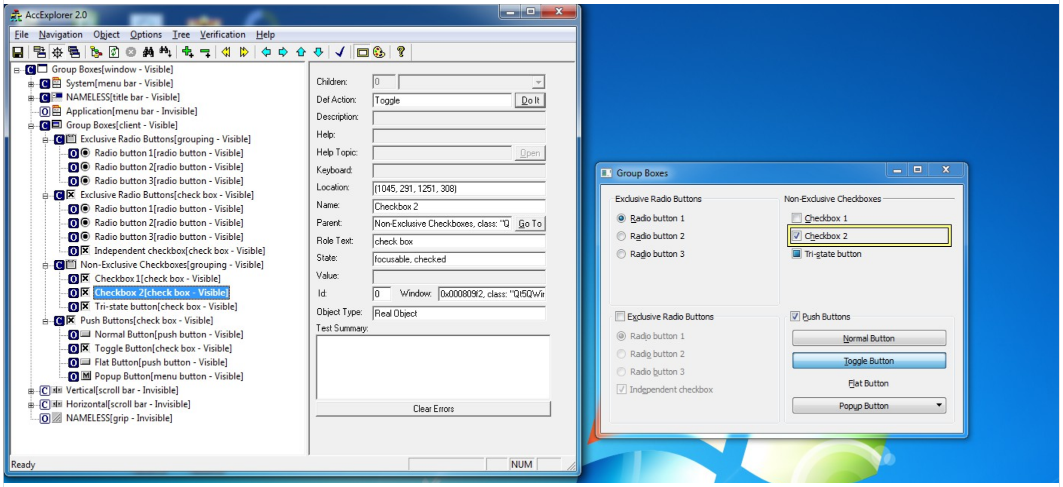
Horizontal slider-like widgets
- Horizontal slider-like widgets
- Vertical slider-like widgets

NVDA (NV Access)

Actual screen reader preferred by many blind people. It's free and Open source. It uses both MSAA and IA2.

Has a speech viewer that can be pleasant to use while developing.

Acc Explorer



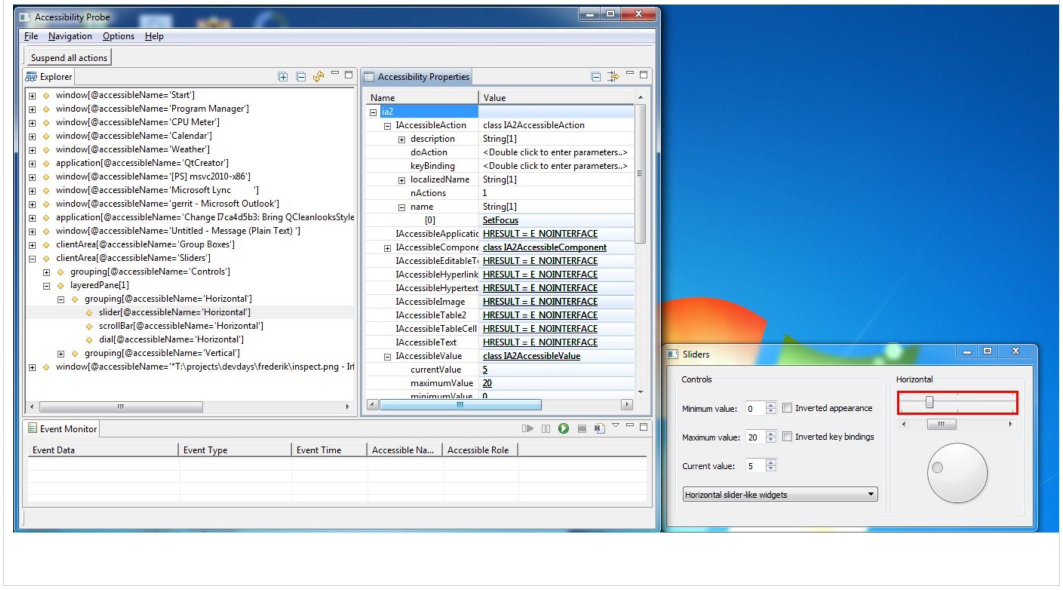
```
Group Boxes[window - Visible]
  System[menu bar - Visible]
  NAMELESS[title bar - Visible]
  Application[menu bar - Invisible]
  Group Boxes[client - Visible]
    Exclusive Radio Buttons[grouping - Visible]
      Radio button 1[radio button - Visible]
      Radio button 2[radio button - Visible]
      Radio button 3[radio button - Visible]
    Exclusive Radio Buttons[check box - Visible]
      Radio button 1[radio button - Visible]
      Radio button 2[radio button - Visible]
      Radio button 3[radio button - Visible]
      Independent checkbox[check box - Visible]
    Non-Exclusive Checkboxes[grouping - Visible]
      Checkbox 1[check box - Visible]
      Checkbox 2[check box - Visible]
      Tri-state button[check box - Visible]
    Push Buttons[check box - Visible]
      Normal Button[push button - Visible]
      Toggle Button[check box - Visible]
      Flat Button[push button - Visible]
      Popup Button[menu button - Visible]
  Vertical[scroll bar - Invisible]
  Horizontal[scroll bar - Invisible]
  NAMELESS[grip - Invisible]
```

Children: 0
Def Action: Toggle
Description:
Help:
Help Topic:
Keyboard:
Location: (1045, 291, 1251, 308)
Name: Checkbox 2
Parent: Non-Exclusive Checkboxes, class: "Q
Role Text: check box
State: focusable, checked
Value:
Id: 0 Window: 0x000809f2, class: "Qt5QWir
Object Type: Real Object
Test Summary:

Ready

Accessible Explorer 2.0 (Microsoft)
Allows you to see the hierarchy, to navigate it and interact with the actions and properties exposed.
For the hierarchy this is usually the preferred tool. However, it is obsolete, and might be hard to find.

Accessibility Probe



```
⊞ window[@accessibleName='Start']
⊞ window[@accessibleName='Program Manager']
⊞ window[@accessibleName='CPU Meter']
⊞ window[@accessibleName='Calendar']
⊞ window[@accessibleName='Weather']
⊞ application[@accessibleName='QtCreator']
⊞ window[@accessibleName='[PS] msvc2010-x86']
⊞ window[@accessibleName='Microsoft Lync        ']
⊞ window[@accessibleName='gerrit - Microsoft Outlook']
⊞ application[@accessibleName='Change I7ca4d5b3: Bring QCleanlooksStyle
⊞ window[@accessibleName='Untitled - Message (Plain Text) ']
⊞ clientArea[@accessibleName='Group Boxes']
⊟ clientArea[@accessibleName='Sliders']
    ⊞ grouping[@accessibleName='Controls']
    ⊟ layeredPane[1]
        ⊟ grouping[@accessibleName='Horizontal']
              slider[@accessibleName='Horizontal']
              scrollBar[@accessibleName='Horizontal']
              dial[@accessibleName='Horizontal']
        ⊞ grouping[@accessibleName='Vertical']
⊞ window[@accessibleName='*T:\projects\devdays\frederik\inspect.png - Ir
```

Name	Value
⊟ ia2	
⊟ IAccessibleAction	class IA2AccessibleAction
⊞ description	String[1]
doAction	<Double click to enter parameters..>
keyBinding	<Double click to enter parameters..>
⊞ localizedName	String[1]
nActions	1
⊟ name	String[1]
[0]	SetFocus
IAccessibleApplicatic	HRESULT = E_NOINTERFACE
⊞ IAccessibleCompone	class IA2AccessibleComponent
IAccessibleEditableT	HRESULT = E_NOINTERFACE
IAccessibleHyperlink	HRESULT = E_NOINTERFACE
IAccessibleHypertext	HRESULT = E_NOINTERFACE
IAccessibleImage	HRESULT = E_NOINTERFACE
IAccessibleTable2	HRESULT = E_NOINTERFACE
IAccessibleTableCell	HRESULT = E_NOINTERFACE
IAccessibleText	HRESULT = E_NOINTERFACE
⊟ IAccessibleValue	class IA2AccessibleValue
currentValue	5
maximumValue	20
minimumValue	0

Event Data	Event Type	Event Time	Accessible Na...	Accessible Role

Controls

Minimum value: 0 □ Inverted appearance

Maximum value: 20 □ Inverted key bindings

Current value: 5

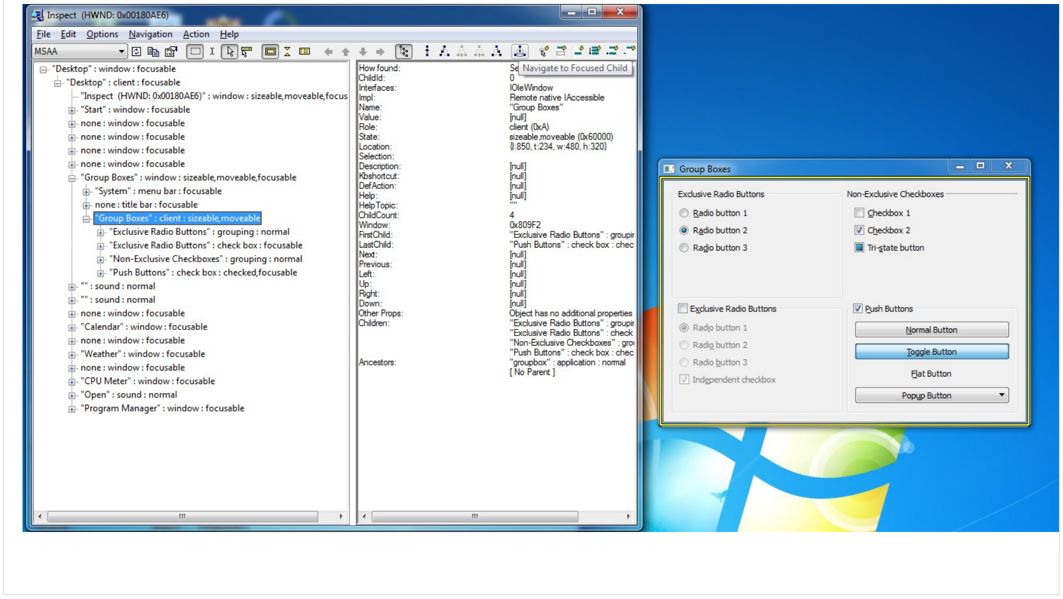
Horizontal slider-like widgets

Horizontal

Accessibility Probe (IBM)

Tool for inspecting both MSAA and IA2, with focus on IA2. Only usable "inspector" tool available for IA2 on windows.

Microsoft Inspect



```
⊟ "Desktop" : window : focusable
   ⊟ "Desktop" : client : focusable
      "Inspect  (HWND: 0x00180AE6)" : window : sizeable,moveable,focus
      ⊞ "Start" : window : focusable
      ⊞ none : window : focusable
      ⊞ none : window : focusable
      ⊞ none : window : focusable
      ⊞ none : window : focusable
      ⊟ "Group Boxes" : window : sizeable,moveable,focusable
         ⊞ "System" : menu bar : focusable
         ⊞ none : title bar : focusable
         ⊟ "Group Boxes" : client : sizeable,moveable
            ⊞ "Exclusive Radio Buttons" : grouping : normal
            ⊞ "Exclusive Radio Buttons" : check box : focusable
            ⊞ "Non-Exclusive Checkboxes" : grouping : normal
            ⊞ "Push Buttons" : check box : checked,focusable
      ⊞ "" : sound : normal
      ⊞ "" : sound : normal
      ⊞ none : window : focusable
      ⊞ "Calendar" : window : focusable
      ⊞ none : window : focusable
      ⊞ "Weather" : window : focusable
      ⊞ none : window : focusable
      ⊞ "CPU Meter" : window : focusable
      ⊞ "Open" : sound : normal
      ⊞ "Program Manager" : window : focusable
```

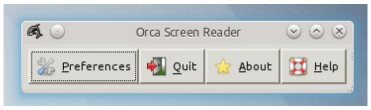
```
How found:      Se
ChildId:        0
Interfaces:     IOleWindow
Impl:           Remote native IAccessible
Name:           "Group Boxes"
Value:          [null]
Role:           client (0xA)
State:          sizeable,moveable (0x60000)
Location:       {l:850, t:234, w:480, h:320}
Selection:
Description:    [null]
Kbshortcut:     [null]
DefAction:      [null]
Help:           [null]
HelpTopic:      ""
ChildCount:     4
Window:         0x809F2
FirstChild:     "Exclusive Radio Buttons" : groupir
LastChild:      "Push Buttons" : check box : chec
Next:           [null]
Previous:       [null]
Left:           [null]
Up:             [null]
Right:          [null]
Down:           [null]
Other Props:    Object has no additional properties
Children:       "Exclusive Radio Buttons" : groupir
                "Exclusive Radio Buttons" : check
                "Non-Exclusive Checkboxes" : grou
                "Push Buttons" : check box : chec
Ancestors:      "groupbox" : application : normal
                [ No Parent ]
```

Inspect 7.1 (Microsoft)
Allows you to see the hierarchy, to navigate it and interact with the actions and properties exposed.

Allows more powerful focus testing. Supports MSAA and UI Automation but not IAccessible2.

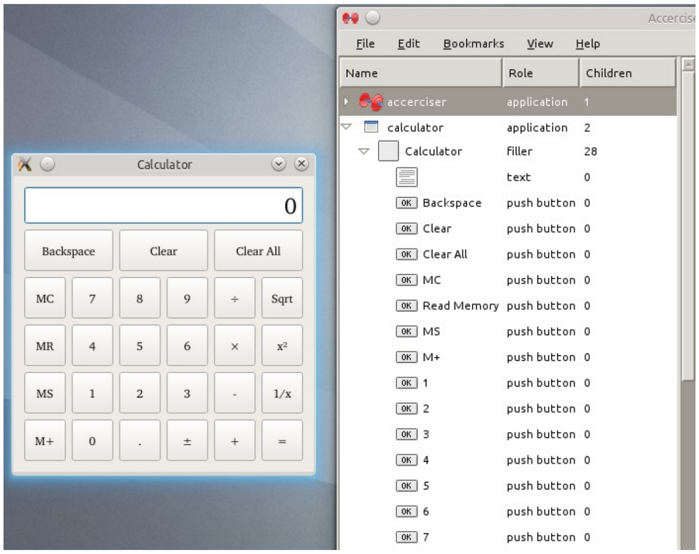
Linux Tools

Orca



Currently the only real screen reader option on Linux.

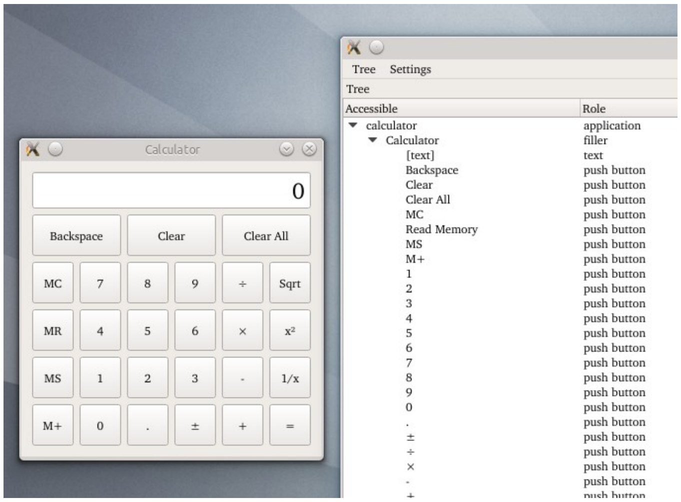
Accerciser



Name	Role	Children
accerciser	application	1
calculator	application	2
Calculator	filler	28
	text	0
Backspace	push button	0
Clear	push button	0
Clear All	push button	0
MC	push button	0
Read Memory	push button	0
MS	push button	0
M+	push button	0
1	push button	0
2	push button	0
3	push button	0
4	push button	0
5	push button	0
6	push button	0
7	push button	0

- Together with Orca one of the Gnome tools. Shows the hierarchy of accessible objects and has a scripting interface (Python).

Randamizer



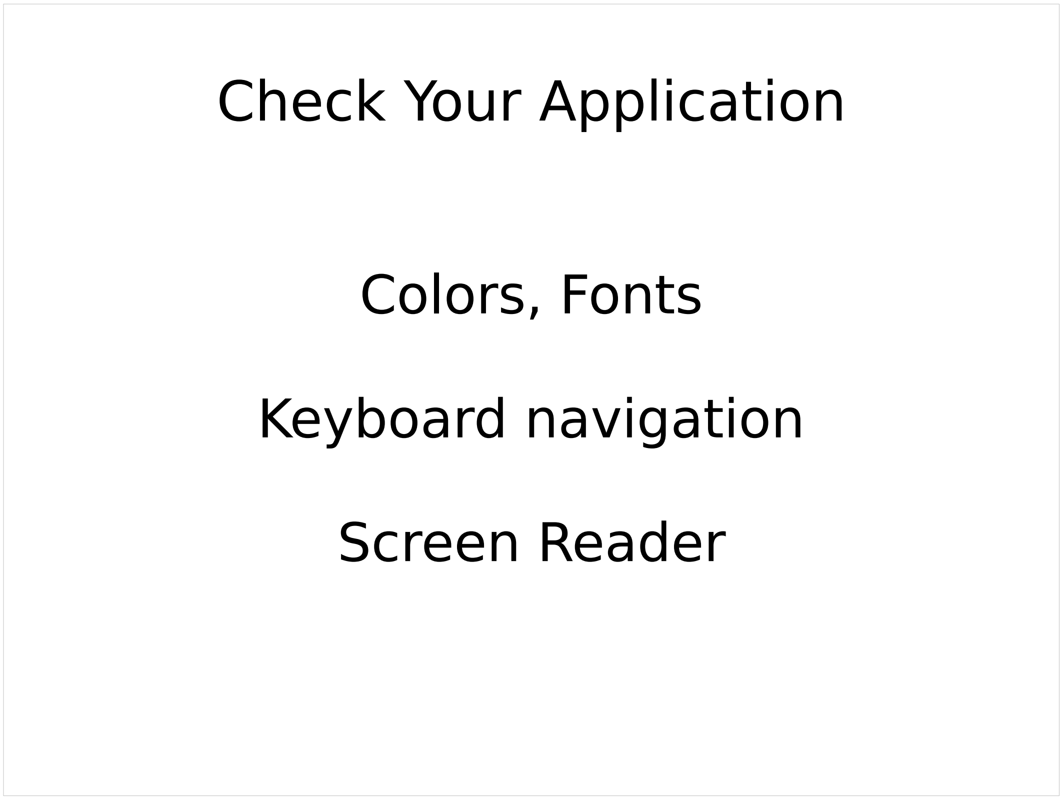
Tree

Accessible	Role
calculator	application
Calculator	filler
[text]	text
Backspace	push button
Clear	push button
Clear All	push button
MC	push button
Read Memory	push button
MS	push button
M+	push button
1	push button
2	push button
3	push button
4	push button
5	push button
6	push button
7	push button
8	push button
9	push button
0	push button
.	push button
±	push button
÷	push button
×	push button
-	push button
+	push button

Newly developed tool written in Qt, mostly to verify our APIs.

Becoming more and more useful for debugging.

Summary

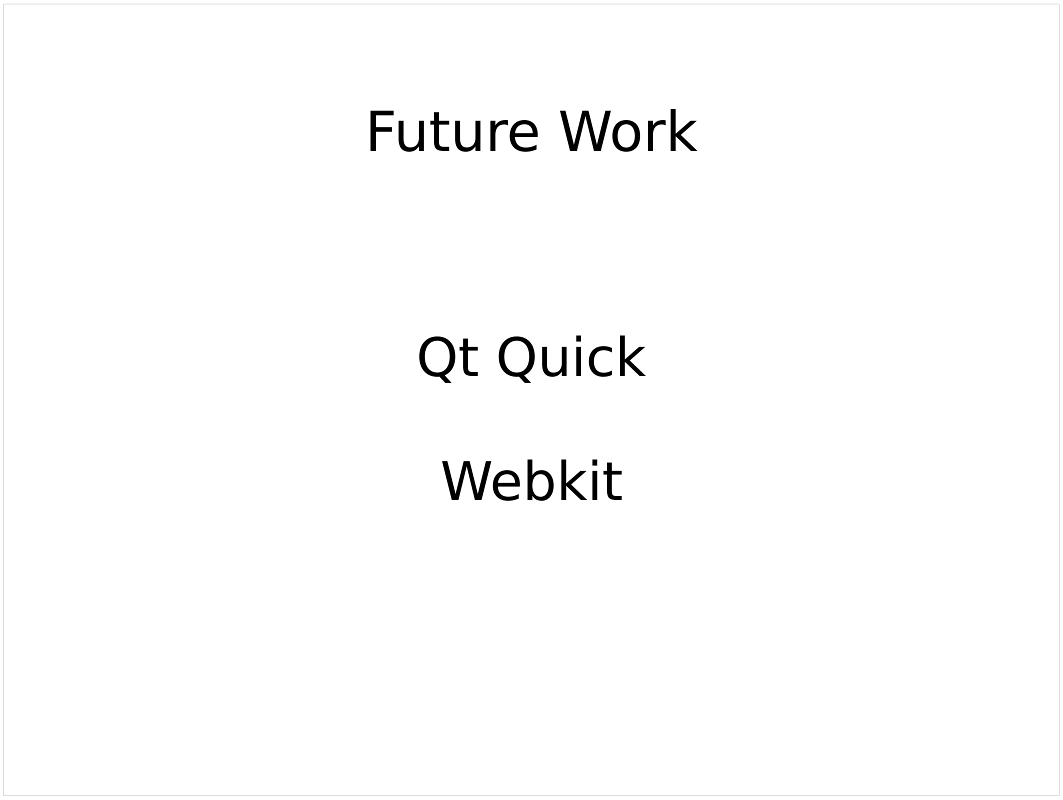


Check Your Application

Colors, Fonts

Keyboard navigation

Screen Reader



Future Work

Qt Quick

Webkit

Qt Quick works really nicely, but needs some improvements, for example when it comes to list views.

Qt WebKit is currently not accessible.

Questions?

Frederik Gladhorn <frederik.gladhorn@digia.com>

Differences Qt 4/5

- Improved notifications
    - Text
    - Tables
- Interface implementation much simpler
- IAccessible2 on Windows