

SELENIUM + AT-SPI

The most important project in KDE

Harald Sitter



Harald

sitter@kde.org

Doing KDE things since forever

Working for Blue Systems

From Austria living in Austria

Not KDE e.V. President (yet)



Oh, It's Me

Vision

A world in which every application has GUI tests and enjoys quality.



Easy

GUI testing is often a drag. It needs to be made easy to write and easy to run.



Reproducible

Must be somewhat reproducible locally so it is easy to debug and fix.



The Problem

Nothing Ever Happened
for our Software

Research

Pays off

During unrelated research of technology stacks to use for WS-Discovery in our KIO SMB worker I've stumbled upon WinAppDriver.

Surprisingly enough just what we need, albeit for the wrong platform. But, what if...

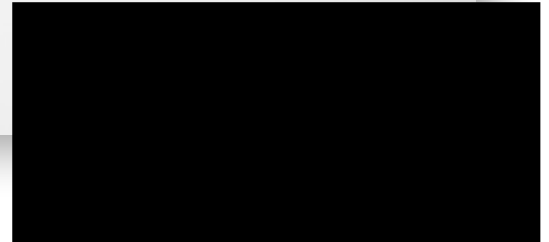


Selenium

Popular web-oriented UI automation & testing system.
Server-client architecture.
Client libraries for all the languages.

Appium

UI automation & testing for native/mobile applications.
Built on top of Selenium.

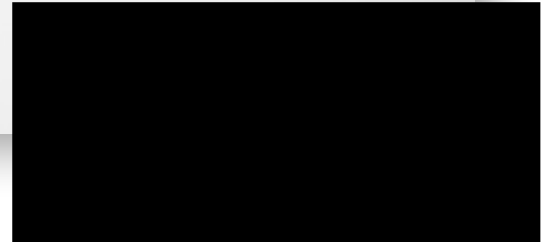


WebDriver

Server component of Selenium and Appium.
Specified by a W3C standard.

AT-SPI

De-facto standard accessibility framework on Linux.
Provides a toolkit-independent way of inspect and
controlling UIs.



Wait, Wait...
Accessibility?

Forestry

Putting all the Trees Together to Form a Forest

You can see the forest on the right ->



Client Libraries

Appium & Selenium

App-centric and supporting a multitude of languages.

HTTP/JSON

WebDriver

The client libraries talk over a HTTP REST protocol with the WebDriver.

AT-SPI

Application

The WebDriver talks over AT-SPI with the application to actually conduct the automation.

Automation



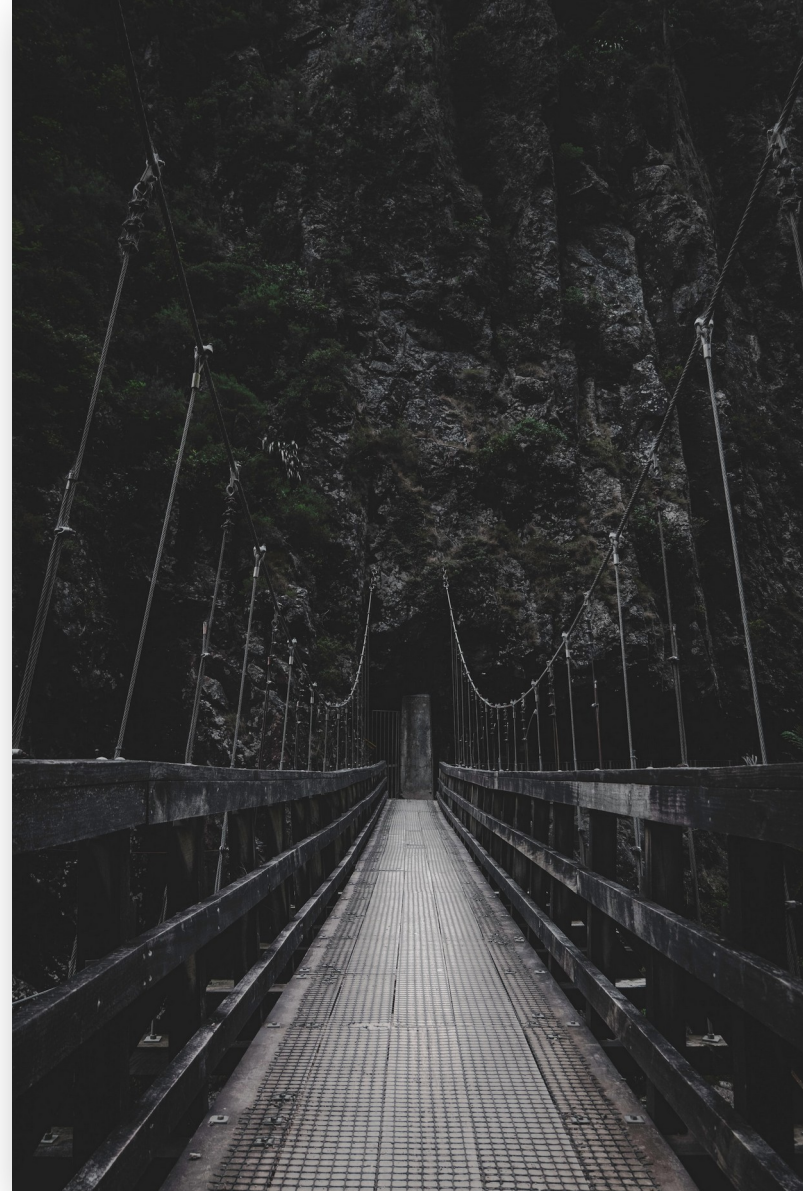
Automation



Bridge

Bridging the Gap

<https://invent.kde.org/sdk/selenium-webdriver-at-spi>





Framework

Simple wrapper tech to facilitate test running



Driver

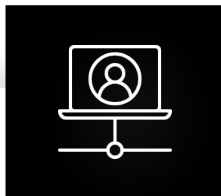
Written in Python with some assistance from C++

WebDriver



Wayland

Tests are run in a nested Wayland session



Screen Recorder

The Wayland session is screen recorded

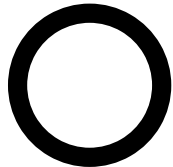
Our WebDriver is a Bunch of Things

LIVE
DEMO



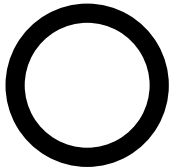
AT-SPI

The AT-SPI Ecosystem



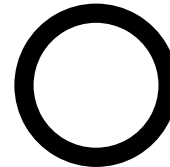
AT-SPI

The Protocol



Accerciser

A inspector tool for AT-SPI



Application

Needs to implement AT-SPI.
Effectively all do through
toolkits

Accerciser

The screenshot displays the Accerciser Accessibility Explorer application. The window title is "Accerciser Accessibility Explorer". The menu bar includes "File", "Edit", "Bookmarks", "View", and "Help". The main interface is divided into a tree view on the left and a detailed view on the right.

Tree View:

Name	Role	Children
> konsole	application	1
> kwin	application	1
> element-desktop	application	1
> TelegramDesktop	application	2
> KeePassXC	application	1
> Discord	application	1
> Firefox	application	1
> konsole	application	1
> code	application	2
> krunner	application	1
> KRunner	frame	6
> Configure	push button	1
> Search	text	4
> Show Usage Help	check box	1
> Pin	check box	1
> [Panel]	panel	3
> [Panel]	panel	2
> steamwebhelper	application	16
> soffice	application	2
> dolphin	application	1
> plasmashell	application	7

Detailed View: push button: Configure

Accessible

Child count 1

Description
Configure KRunner Behaviour

ID
QApplication.View.ToolButton_QMLTYPE_42

States	Attributes
enabled	
focusable	
sensitive	

Relations

Show

Action

Name	Description	Key binding
Press	Triggers the action	
SetFocus	Sets the focus	
Perform action		

> Application (not implemented)
> Collection (not implemented)

Path: 0 0

Accerciser



Code



Call to Action

I Need Your Help!

<https://invent.kde.org/sdk/selenium-webdriver-at-spi/-/wikis/writing-tests>

<https://apachelog.wordpress.com/2022/12/14/selenium-at-spi-gui-testing/>



Good News

The End!

Thanks for paying attention. If you haven't paid attention thanks for enjoying the AC.

It's a cruel joke that this slide contains a lake with what appears to be nippy foggy weather while we are suffering in this heat!