Collabora Productivity

LibreOffice Online
–
client side development

by Mihai Varga

Consultant Software Engineer Intern

+MihaiVarga13 mihai.varga@collabora.com

A brief introduction

A brief introduction

- LibreOffice Online:
  - development started in April 2015
  - consists of two major parts:
    - the server
    - the client
      - Leaflet
      - JavaScript

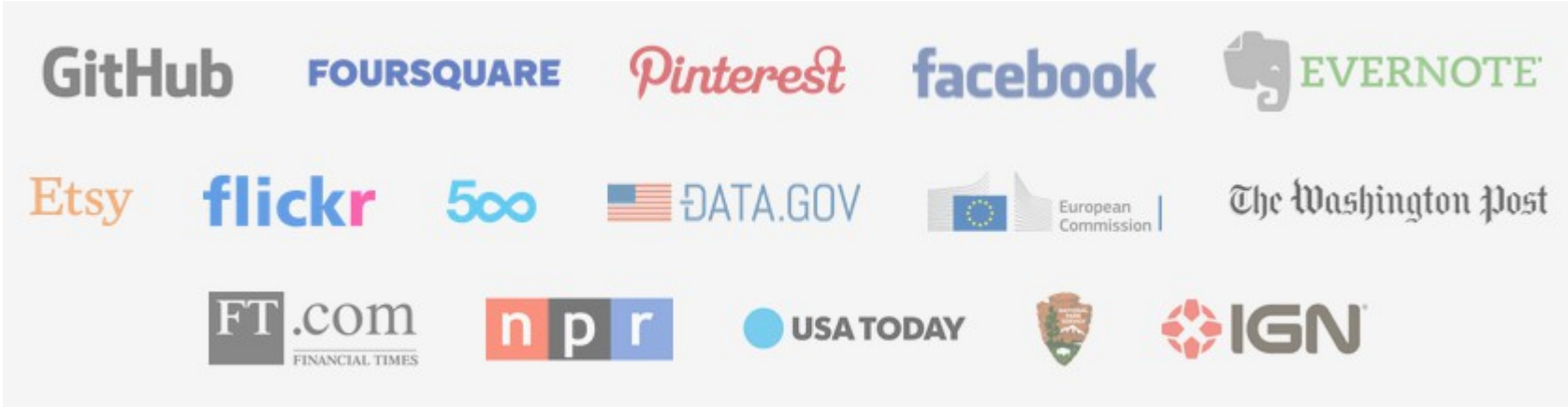
Leaflet

A brief introduction

- how Leaflet usually looks like

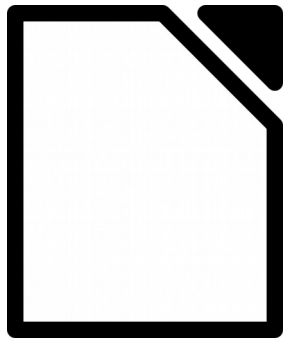
Leaflet

- an open-source JavaScript library
- interactive maps
- tile-based implementation
- used by many big companies



Leaflet

- What have we changed ?
  - added web socket for communication with the server
  - binary images
  - caching more tiles



Loading the document

Loading the document

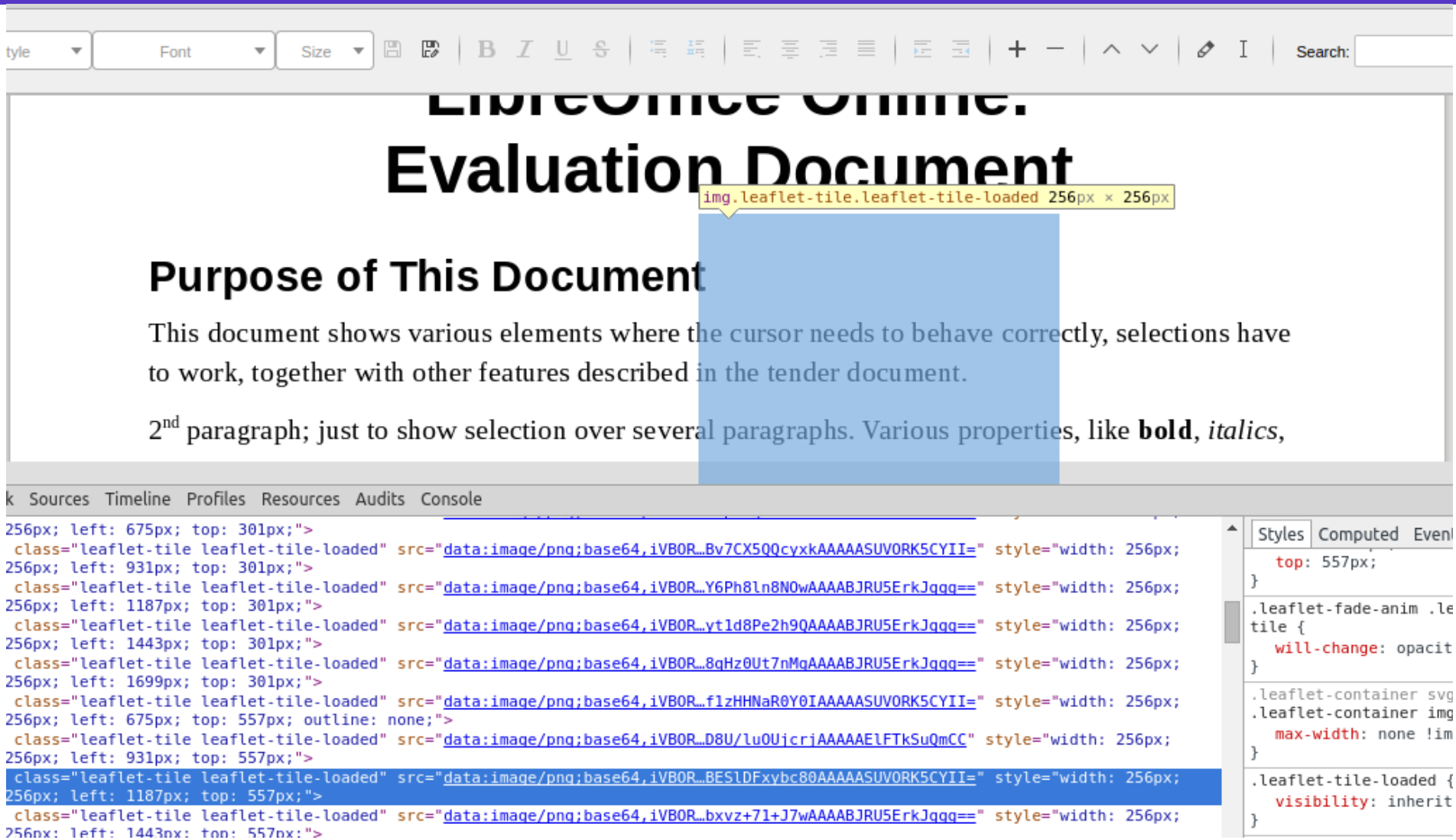
- used Leaflet's simple CRS (Coordinate reference system)
- the tile at (0, 0) is placed in the left top corner
- request images from server based on the coordinates of the visible area

Loading the document

- the server sends binary .png images → these are transformed into data URIs
- example: An HTML fragment embedding a picture of a small red dot:

```
<img src="
AAAFCAYAAACNbyblAAAAHElEQVQI12P4//8/w38GIAXDIBKE0DHxgljNBAAO
9TXL0Y4OHwAAAABJRU5ErkJggg==" alt="Red dot" />
```

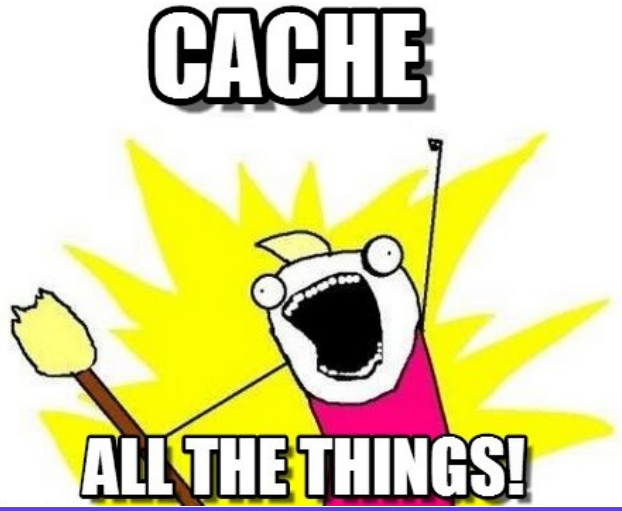
Loading the document



Caching

Caching

- tiles outside of the viewing area are pre-fetched and cached
- tiles are not deleted when removed from the DOM tree



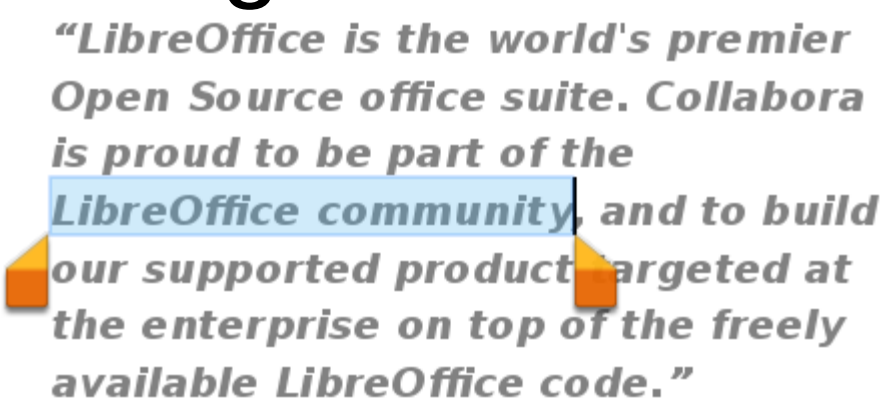
Viewing the document

Viewing the document

- implemented a scroll API → scroll bars can be easily plugged in
- based on Leaflet's panning methods
- the document can also be panned with the mouse
- different zoom levels

Viewing the document

- selections are implemented as a SVG overlay
- text can be selected and copied from the document
- the user can shrink / enlarge the selection



Editing the document

Editing the document

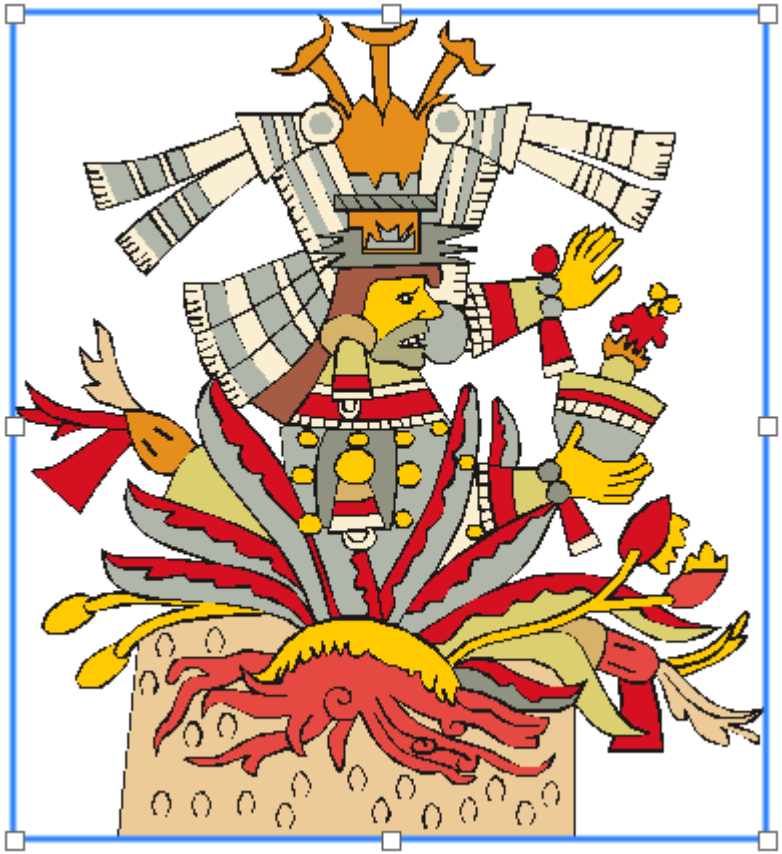
- typical scenario: key strokes are sent to the server → tiles are invalidated → new tiles are requested and repainted → cursor is moved, etc.

Editing the document

- how are key strokes captured?
  - we have a hidden text area in which the user types
  - keyboard events are intercepted
  - browser compatibility issues, the keyboard event is different across browsers

Editing the document

- images and shapes can be moved and resized



Editing the document

- copying
  - the ClipboardEvent is captured in the hidden text area
  - due to security issues, access to the user's clipboard is quite restricted
  - the event must be handled synchronously when captured, else it will 'expire'

Editing the document

- what's next for copying
  - there is ongoing work for a Clipboard API (current status: Working Draft)
  - very little support for rtf
  - plain text or html currently works better
  - alternatives: zeroclipboard library
    - but this uses flash which will soon be deprecated

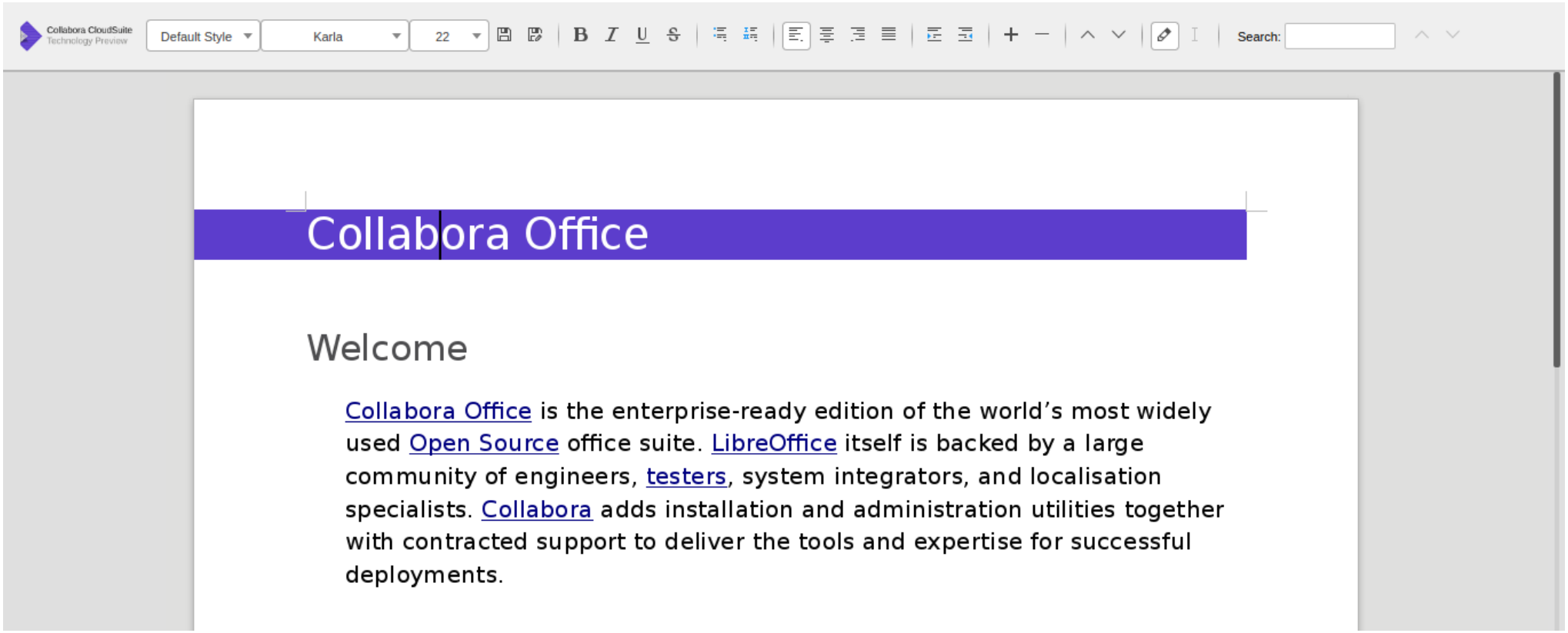
The toolbar

The toolbar

- we've extended Leaflet's API to provide methods for building / using the toolbar
- most of the methods work through a UNO command
- it's easy to plug in and out different toolbar components
- can be easily integrated in an already existing JavaScript application

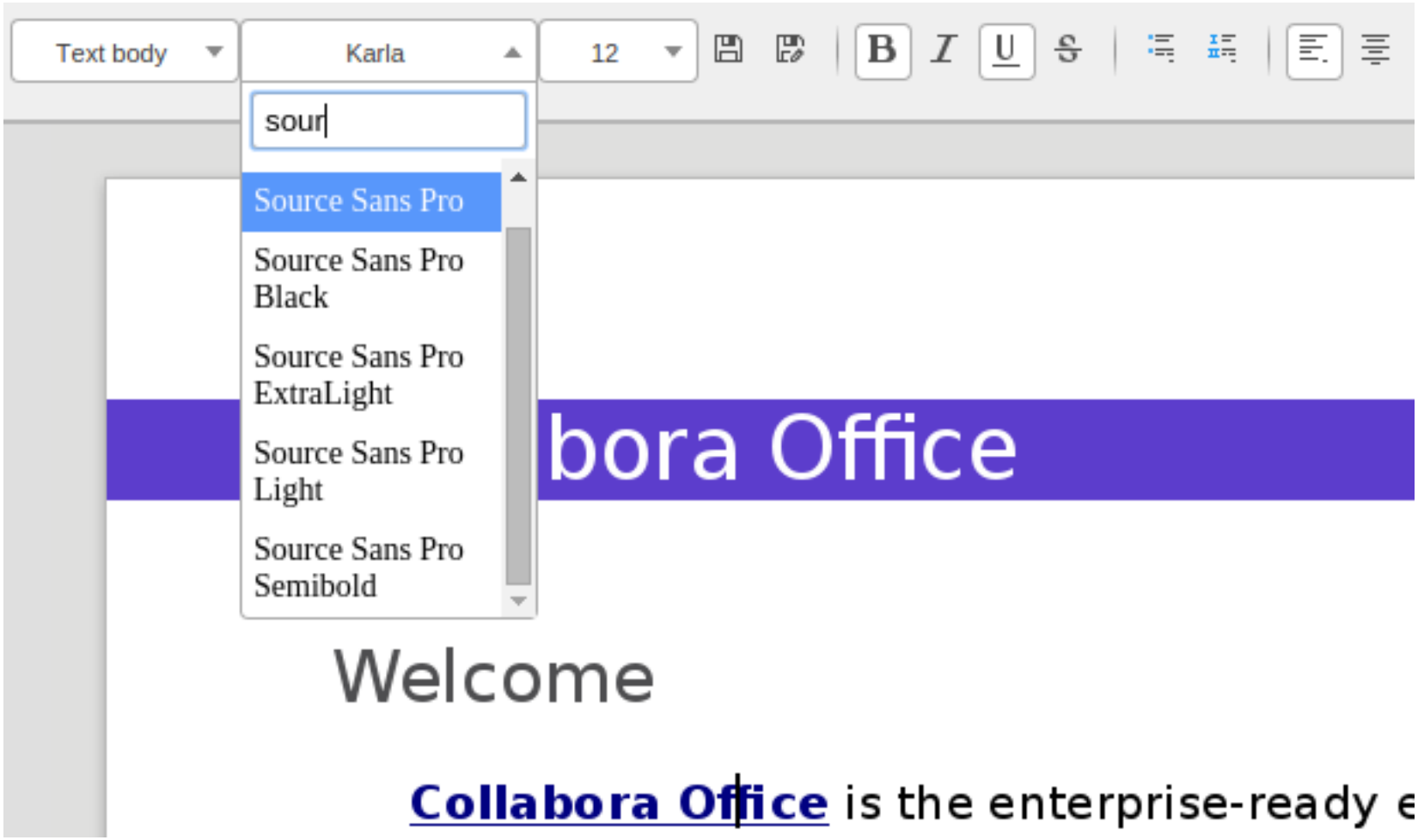
The toolbar

Cloudsuite toolbar



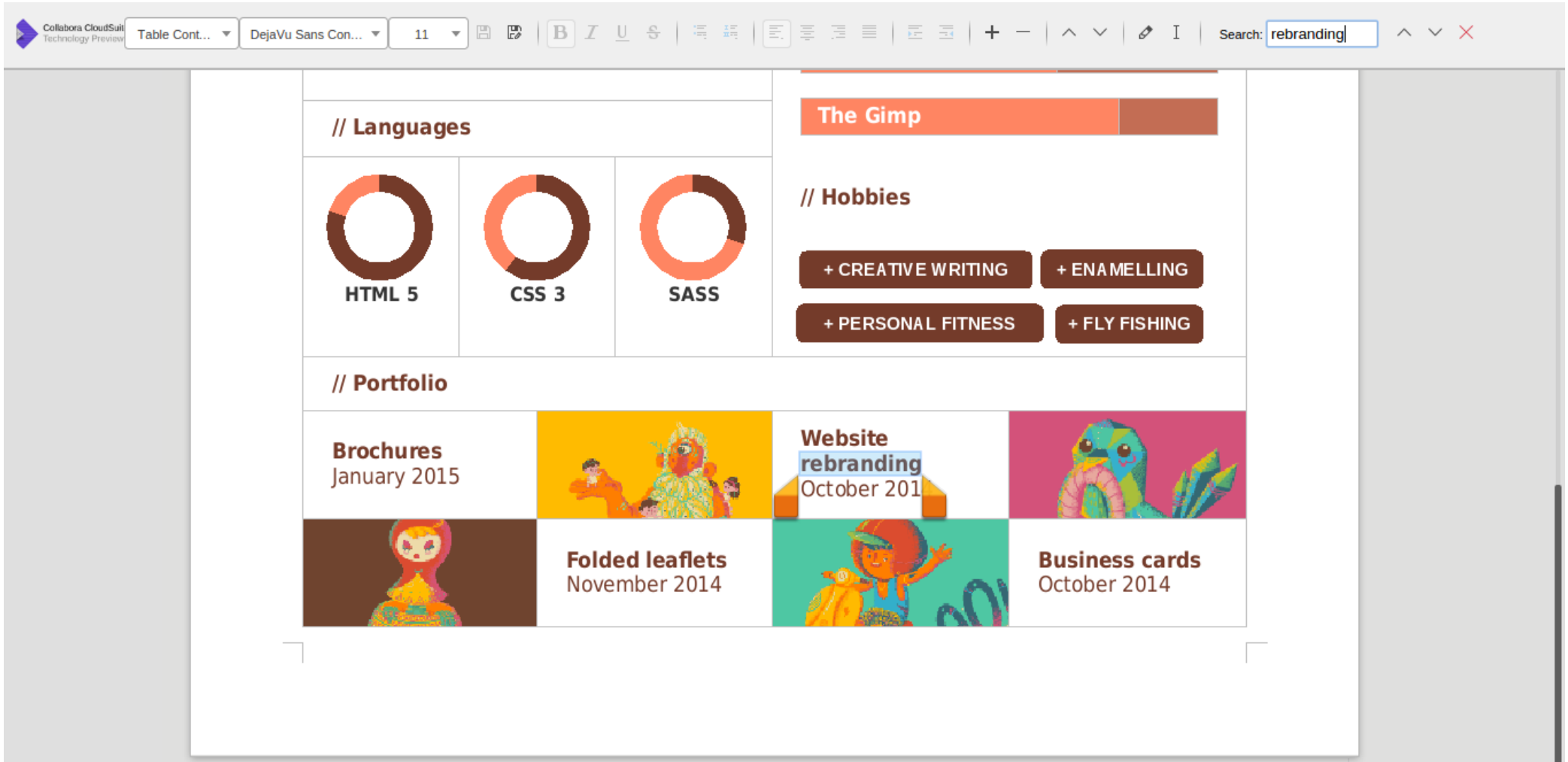
The toolbar

A closer look at the toolbar



The toolbar

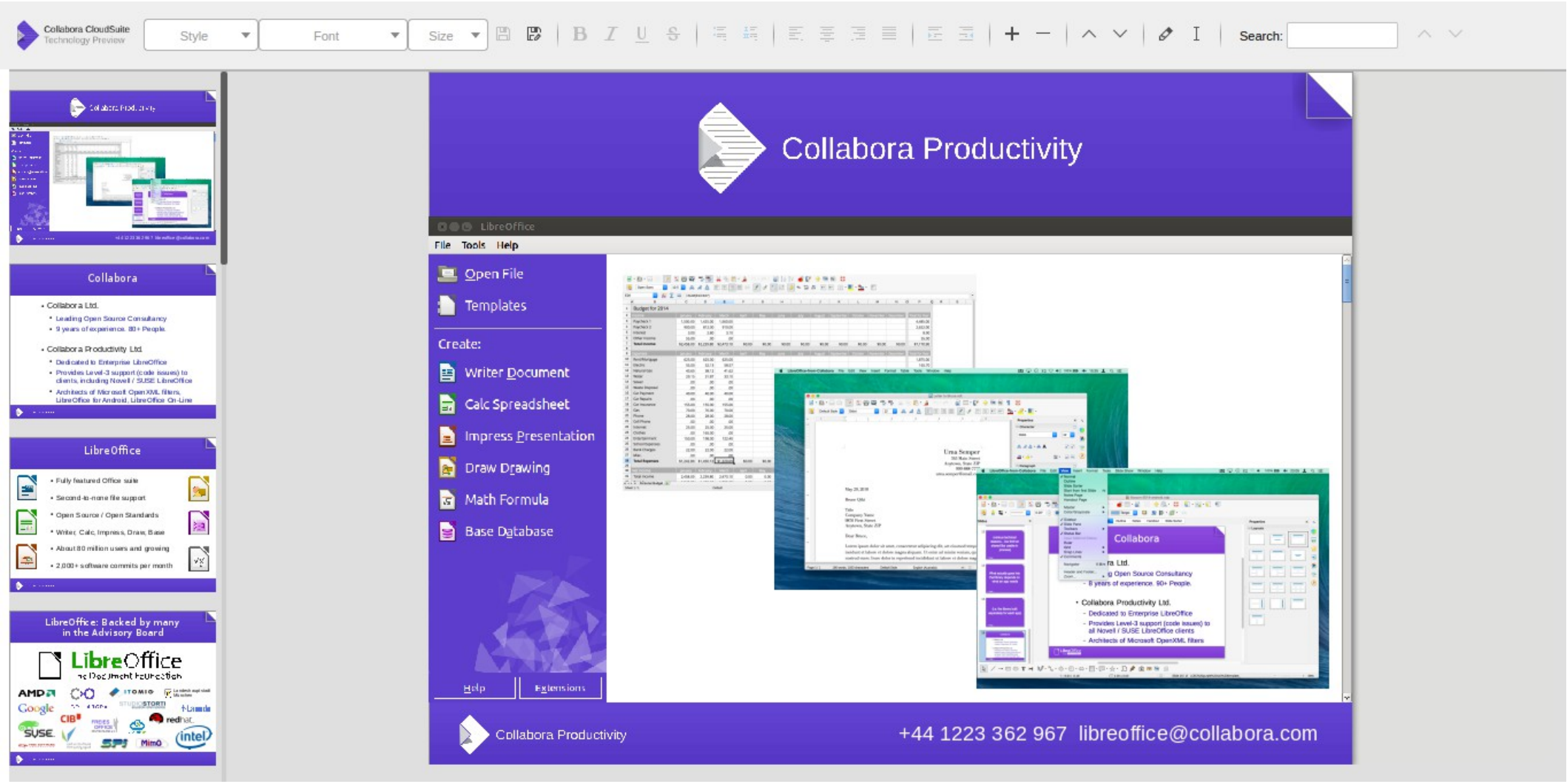
Searching



The toolbar

- a few other features not available through UNO commands:
  - goToPage / goToPart
  - Enable editing / viewing mode
  - Zoom in / out
  - Enable mouse selection or panning

The toolbar

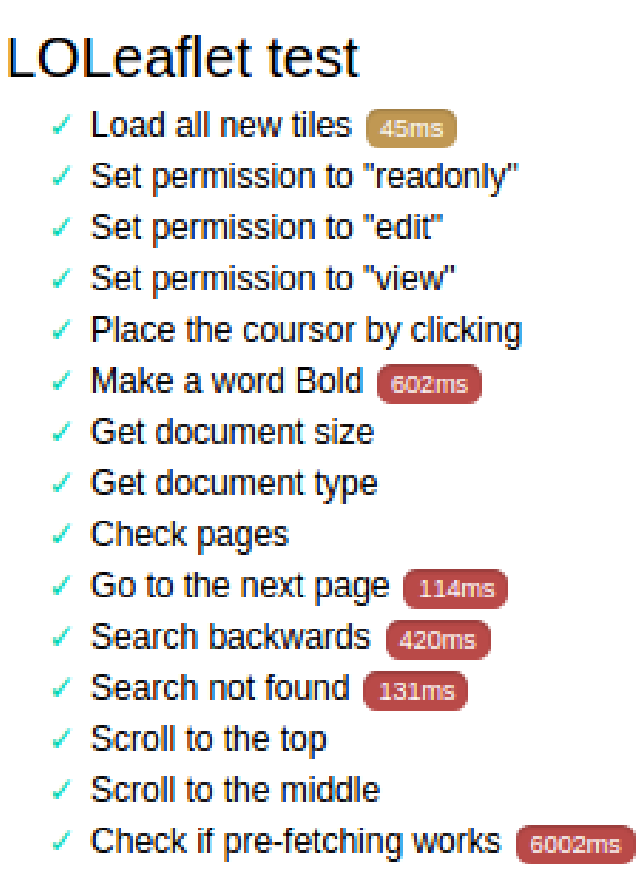

Collabora CloudSuite
Technology Preview
Style
Font
Size
Search:
Collabora Productivity
LibreOffice
File Tools Help
Open File
Templates
Create:
Writer Document
Calc Spreadsheet
Impress Presentation
Draw Drawing
Math Formula
Base Database
Help
Extensions
Collabora Productivity
+44 1223 362 967 libreoffice@collabora.com
Collabora
Collabora Ltd.
Collabora Productivity Ltd.
LibreOffice
LibreOffice: Backed by many in the Advisory Board

Testing

Testing

- automated testing is possible
- Leaflet was already using the mocha JavaScript framework
- nice way of testing the new and the old features of Leaflet
- we can replay an editing session

Testing

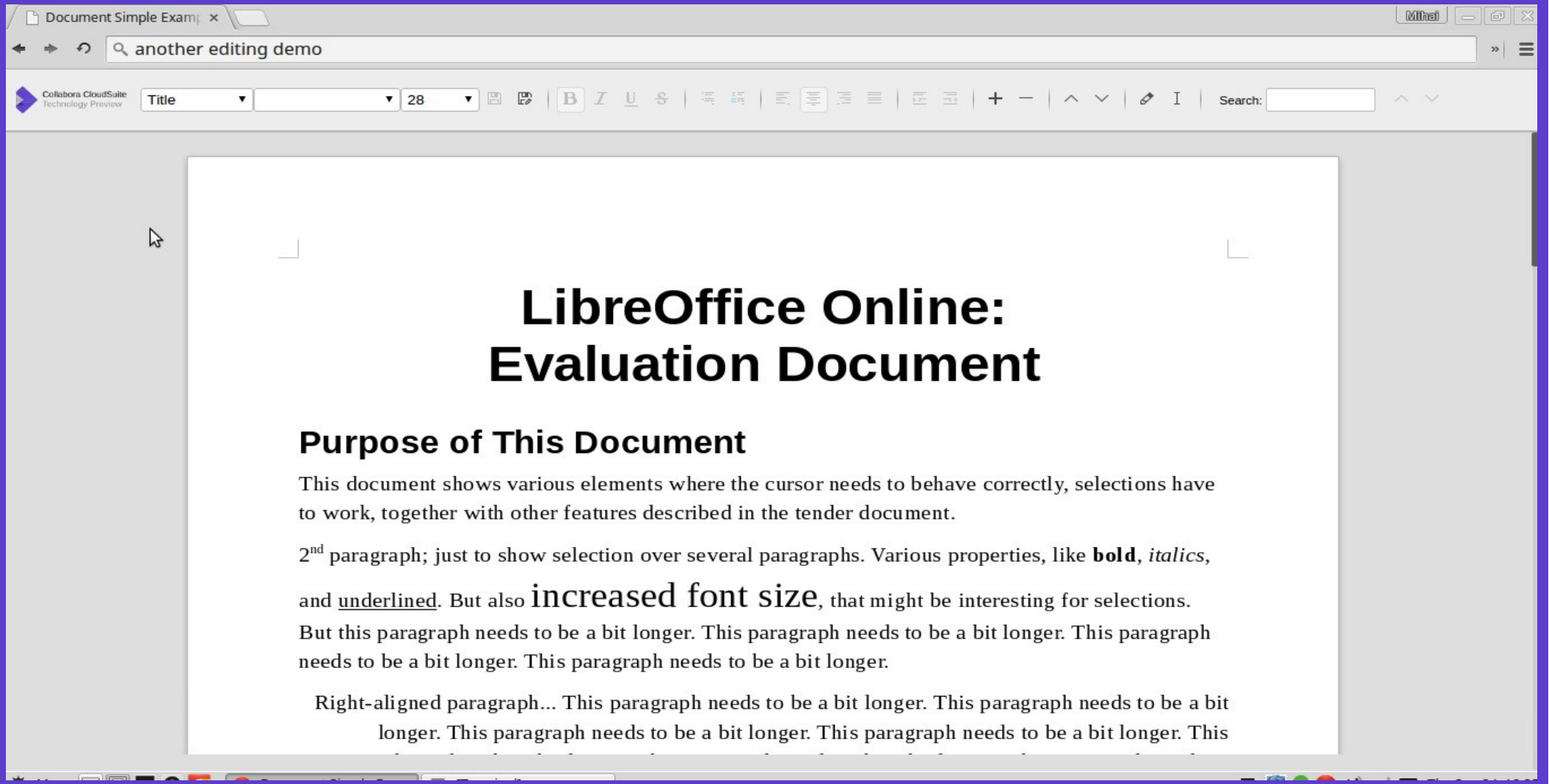


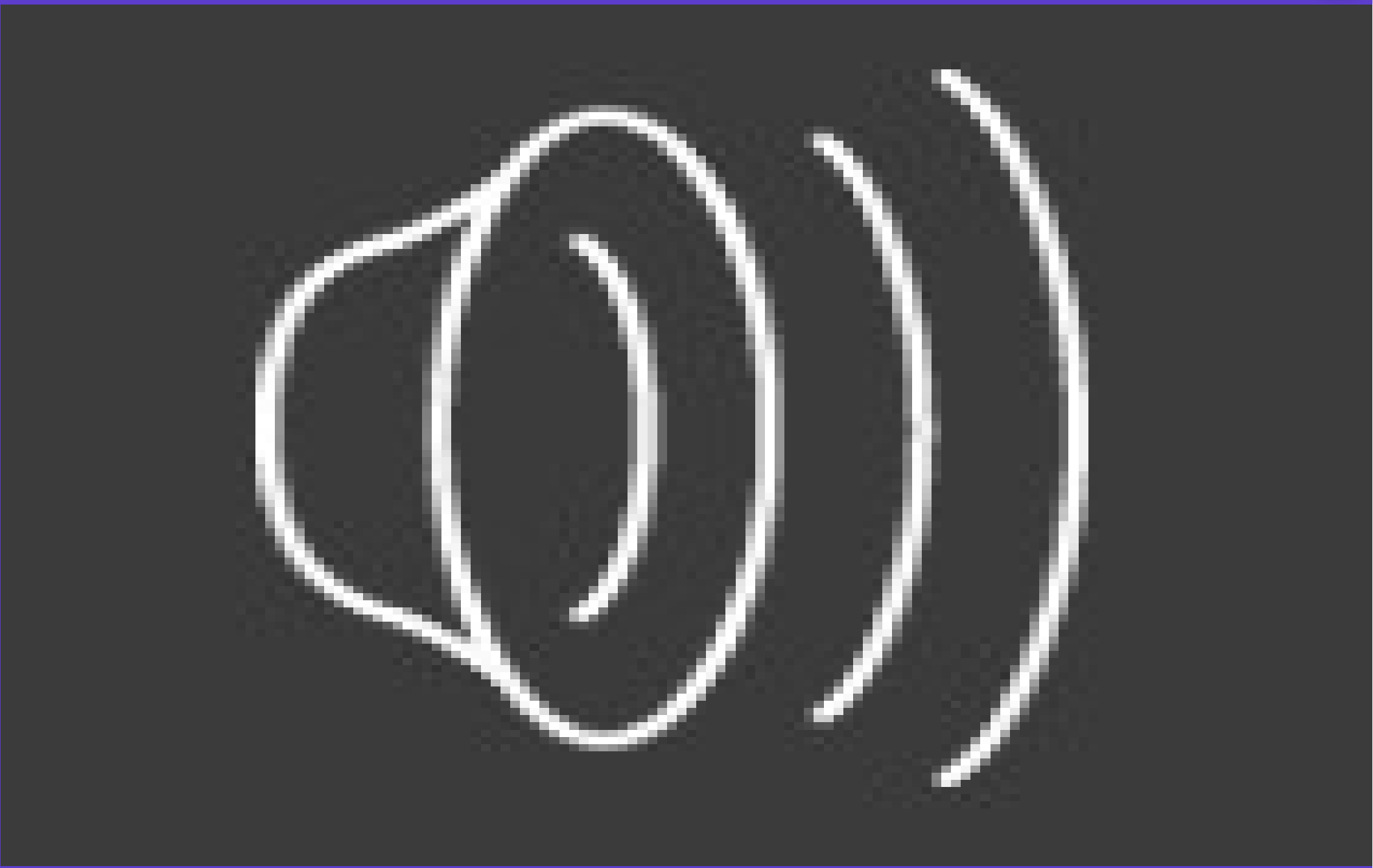
- the tests have to be run in the browser
- an alternative is to use PhantomJS, but it currently only has hixie-76 websockets
- PhantomJS 2.x promises RFC 6455 websockets, so this will be the way to go

Testing

- performance while editing
- tile loading time it's really good on average (< 100 ms)
- we are still working on improving it
- might have to do with how binary images are loaded in the browser

Demo


Document Simple Exam
another editing demo
Collabora CloudSuite
Technology Preview
Title
28
Search:
LibreOffice Online:
Evaluation Document
Purpose of This Document
This document shows various elements where the cursor needs to behave correctly, selections have to work, together with other features described in the tender document.
2nd paragraph; just to show selection over several paragraphs. Various properties, like bold, italics, and underlined. But also increased font size, that might be interesting for selections. But this paragraph needs to be a bit longer. This paragraph needs to be a bit longer. This paragraph needs to be a bit longer. This paragraph needs to be a bit longer.
Right-aligned paragraph... This paragraph needs to be a bit longer. This paragraph needs to be a bit longer. This paragraph needs to be a bit longer. This paragraph needs to be a bit longer. This



Thank you!

- Slides will also be available on http://mihai-varga.blogspot.com/ (sometime soon)