Diving into PDF.js

The advantages of the web platform

Calixte Denizet <calixte@mozilla.com>
Marco Castelluccio <marco@mozilla.com>

FOSDEM 2024

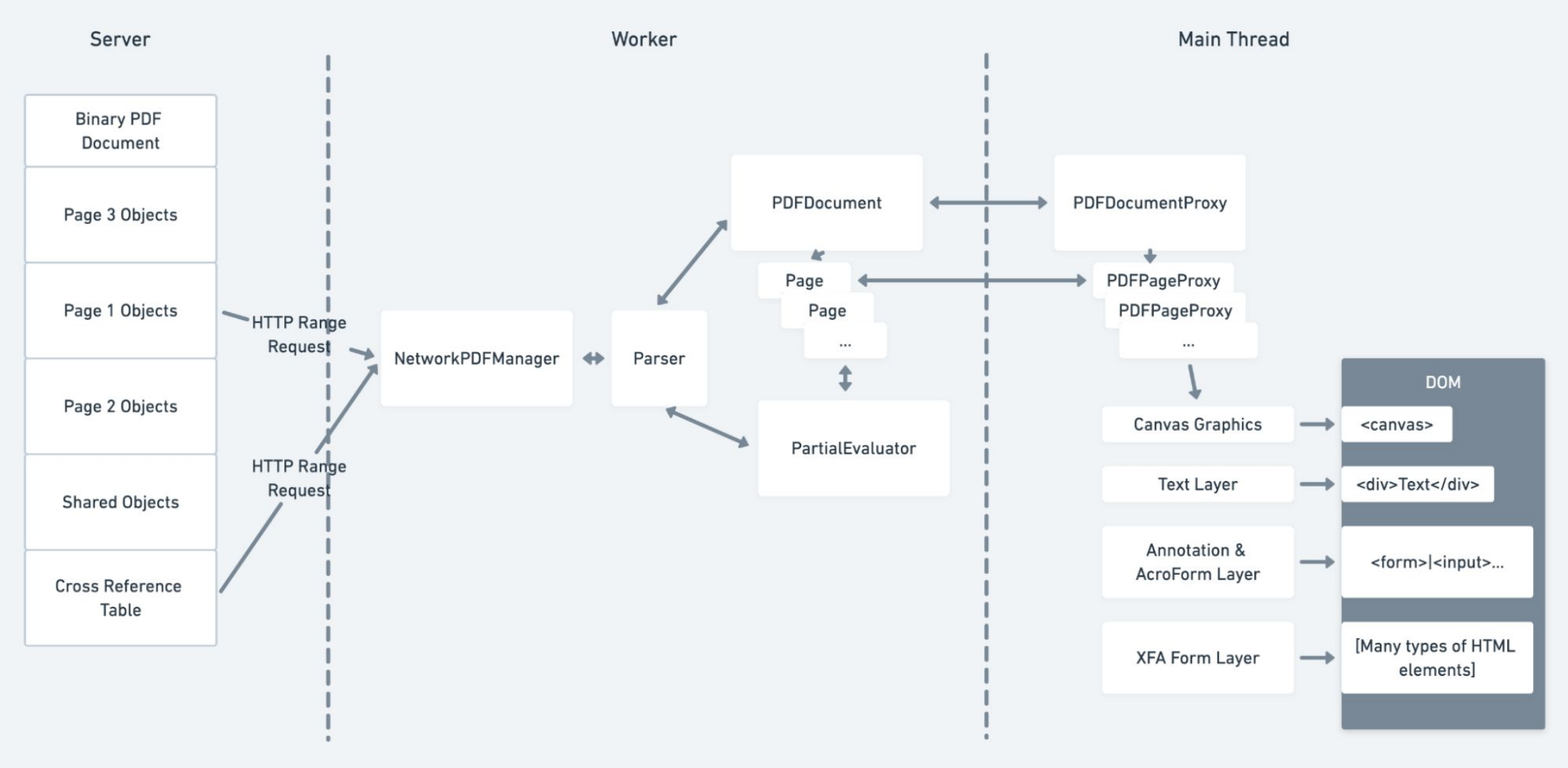
PDF.js

PDF format

- A PDF is a sequence of objects identified by a number and a table mapping identifiers and object offsets in the file
- Objects can be dictionaries, arrays, binary streams, …
- PDF files can embed images, fonts, drawing instructions, xml, …

PDF.js

Global architecture



How does it work?

- Data streamed into a worker

- The worker has a lot of parsers:
  - parser for the **pdf**
  - parsers for the embedded **images** (jpeg, jpeg2000, jbig2, ...)
  - parsers for the different **font** formats
  - **xml** parser

- Objects are parsed and sanitized (and fixed!)

- A drawing instructions list is generated and sent back to the main thread to be drawn on a HTML5 canvas.

Et voila !

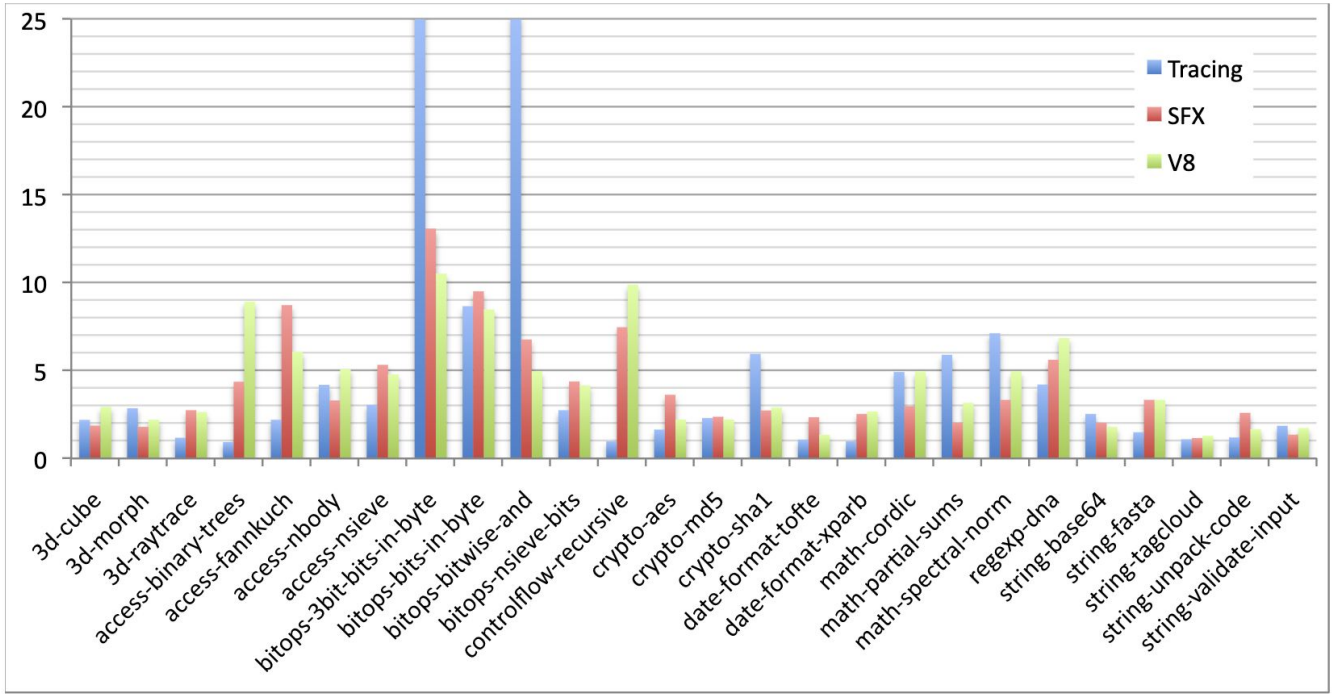


Figure 10. Speedup vs. a baseline JavaScript interpreter (SpiderMonkey) for our trace-based JIT compiler, Apple's SquirrelFish Extreme inline threading interpreter and Google's V8 JS compiler. Our system generates particularly efficient code for programs that benefit most from type specialization, which includes SunSpider Benchmark programs that perform bit manipulation. We type-specialize the code in question to use integer arithmetic, which substantially improves performance. For one of the benchmark programs we execute 25 times faster than the SpiderMonkey interpreter, and almost 5 times faster than V8 and SFX. For a large number of benchmarks all three VMs produce similar results. We perform worst on benchmark programs that we do not trace and instead fall back onto the interpreter. This includes the recursive benchmarks `access-binary-trees` and `control-flow-recursive`, for which we currently don't generate any native code.

In particular, the `bitops` benchmarks are short programs that perform many bitwise operations, so TraceMonkey can cover the entire program with 1 or 2 traces that operate on integers. TraceMonkey runs all the other programs in this set almost entirely as native code.

- Two programs trace well, but have a long compilation time. `access-nbody` forms a large number of traces (81). `crypto-md5` forms one very long trace. We expect to improve performance on this programs by improving the compilation speed of nano-jit.

Text layer

- To make the text selectable, there is a **text layer** with all the text

Trace-based Just-in-Time Type Specialization for Dynamic Languages

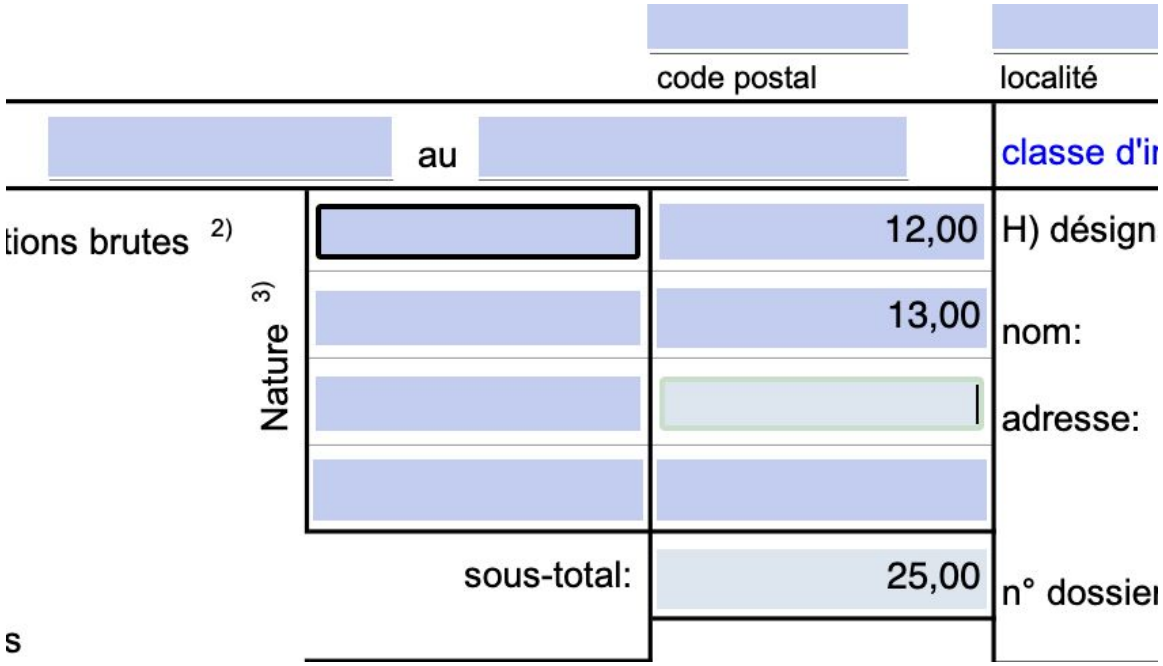
Andreas Gal*+, Brendan Eich*, Mike Shaver*, David Anderson*, David Mandelin*,
Mohammad R. Haghighat$, Blake Kaplan*, Graydon Hoare*, Boris Zbarsky*, Jason Orendorff*,
Jesse Ruderman*, Edwin Smith#, Rick Reitmaier#, Michael Bebenita+, Mason Chang+#, Michael Franz+

Mozilla Corporation*
{gal,brendan,shaver,danderson,dmandelin,mrbkap,graydon,bz,jorendorff,jruderman}@mozilla.com

Adobe Corporation#

Annotation layer

- To render forms, an annotation layer is added with HTML form elements



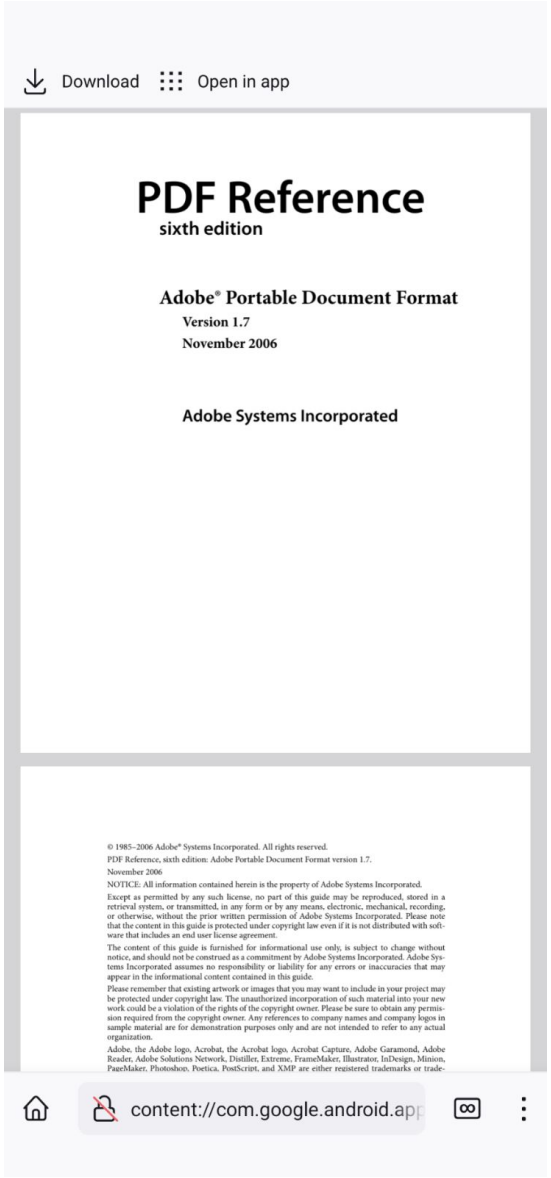
Editing layer

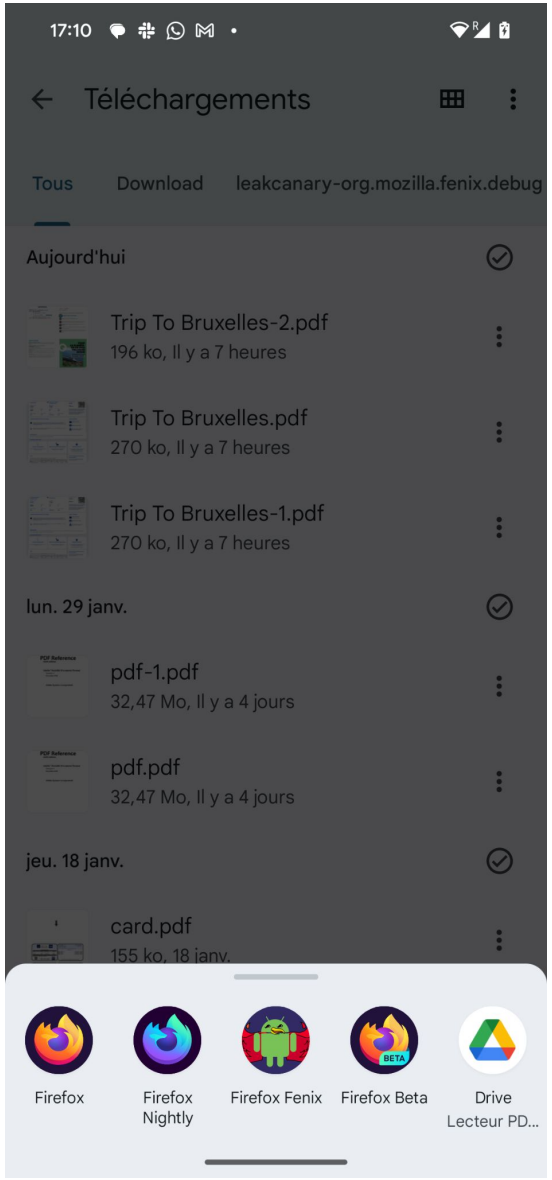
- To render added annotations, another layer is added

PDFs can embed JS

- To validate or compute form data, JS code can be embedded in PDFs.
- For the Firefox viewer, JS code is executed within a **sandbox**, i.e. a special isolated environment which has been used for years in WebExtensions.
- For the non-Firefox viewer, JS code is executed with **QuickJS** (a JS engine written in C) compiled in WASM.

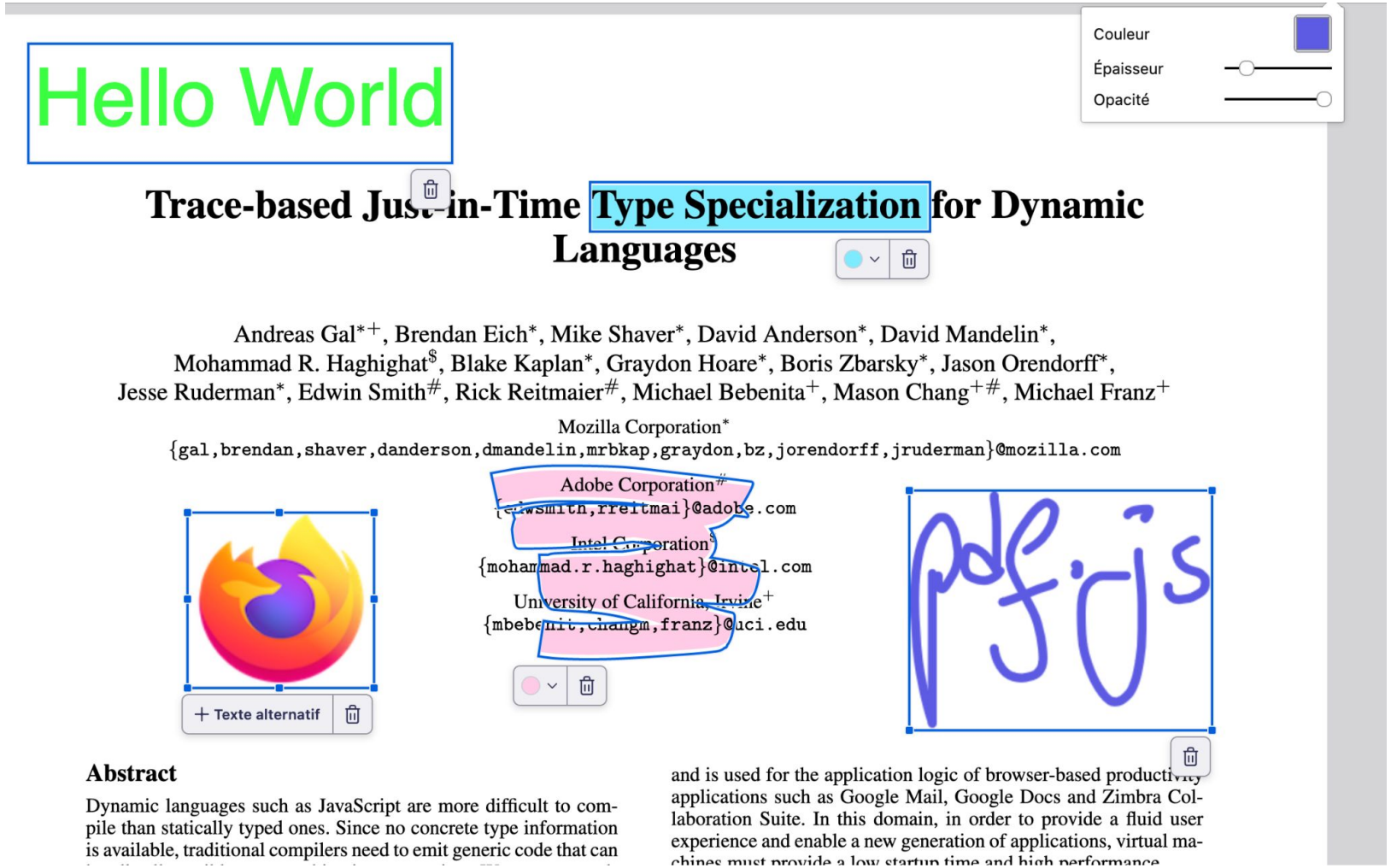
Android


Download
Open in app
PDF Reference
sixth edition
Adobe® Portable Document Format
Version 1.7
November 2006
Adobe Systems Incorporated
content://com.google.android.app


17:10
Téléchargements
Tous
Download
leakcanary-org.mozilla.fenix.debug
Aujourd'hui
Trip To Bruxelles-2.pdf
196 ko, Il y a 7 heures
Trip To Bruxelles.pdf
270 ko, Il y a 7 heures
Trip To Bruxelles-1.pdf
270 ko, Il y a 7 heures
lun. 29 janv.
pdf-1.pdf
32,47 Mo, Il y a 4 jours
pdf.pdf
32,47 Mo, Il y a 4 jours
jeu. 18 janv.
card.pdf
155 ko, 18 janv.
Firefox
Firefox Nightly
Firefox Fenix
Firefox Beta
Drive
Lecteur PD...

What we are doing

- Editing features



What we are planning

- Merging and splitting, helping users not compromise their PDFs

- Signatures

- Reader mode

moz://a

Thank You