

Browser document pipeline review

A four-page sample PDF shipped with the Codepool file viewer demo. Every page carries a different layout so page navigation, text extraction and zoom controls can be exercised without hunting for a real document.

1. Scope

The viewer has to answer two questions in one screen: what is this file, and what does it contain. The first is metadata — size, type, page count, producer. The second is content, and it has to work for documents that were never authored for the web: scans, exports from office suites, camera photos.

2. What the browser can do on its own

Images, video and plain text need no library at all. PDF needs a parser and a renderer, TIFF needs a decoder because no browser ships one, and the three office formats are ZIP containers around XML, so they need a schema-aware renderer rather than a byte reader.

Format	Container	Renderer	Pages
PDF	one file, xref table	pdf.js	per page
TIFF	IFD chain	UTIF	per IFD
DOCX	ZIP + OOXML	docx-preview	per section
XLSX	ZIP + OOXML	SheetJS	per sheet / row block
PPTX	ZIP + OOXML	built-in	per slide

3. Page two: the metadata question

A file's extension is a claim, not a fact. The viewer sniffs magic bytes before it trusts the name, because a .jpg that is really a PNG will otherwise render as a broken image with no explanation. The same check separates a DOCX from a XLSX when both arrive as PK zip archives.

3.1 Why EXIF deserves its own panel

A photograph carries a surprising amount of structured data: the body, the lens, the exposure triangle, the orientation the camera was held at, and often the exact coordinates where the shutter fired. Reading it is a hundred lines of byte parsing and no network access at all.

3.2 Multi-page handling

Pagination is where most browser viewers give up. A TIFF can hold a hundred IFDs, a PDF a thousand pages, a spreadsheet a million rows. The viewer keeps one page mounted at a time and treats page count as data, not as UI state.

4. Page three: a wide table

Long tables are the easiest way to break a viewer's layout. This one is deliberately wider than the reading column of the pages around it.

#	Check	Expected	Observed
1	extension disagrees with magic bytes	content wins	content wins
2	EXIF present but no orientation tag	default to 1	default to 1
3	TIFF with three IFDs	3 pages	3 pages
4	PDF with a broken xref	repair or refuse	repair
5	CSV with quoted newlines	one field	one field
6	DOCX with section breaks	one page per section	one page per section
7	XLSX with merged cells	rowspan honoured	rowspan honoured
8	PPTX with an off-slide shape	clipped, not shifted	clipped
9	binary with no renderer	metadata only	metadata only

5. Page four: closing notes

If you are reading this page, page navigation, zoom and the metadata panel all worked. That is the whole test.

Sample data only

Nothing on this page describes a real service, invoice or person. The document exists so that a viewer can be demonstrated without depending on a third-party file, and it is regenerated by `_scripts/build_file_viewer_samples.py`.