

PDF Viewer Research

- 2020.01.22
- anakin@boinit.com

현재까지 검토된 사항으로는 pdf.js로 진행하는것이 좋을것으로 보임.

Common Issues

- Tip 연동이 필요한가?
 - page 이동 event 처리 필요(가능은 할것으로 보임)
- 전자책판과 교사PC 의 완벽한 연동이 필요한가?
 - 아마 이부분도 epub viewer와 마찬가지로 적당한 수준에서 끊어야 할것으로 보임.
 - 특히 확대/축소 세밀한 이동 등은 제외해야 할듯
 - 페이지 이동 정도는 가능해 보임
- 보기 모드는?
 - 일단 세로 스크롤 보기 형태로 생각하는 것이 좋음
 - 물론 조금 더 파면 페이지별 넘김 등도 가능함.
- 검색 UI/UX에 따라 일이 조금 많아질 수 있음
- 제공되는 기본 viewer 모듈을 사용할지 core 모듈만 사용하여 커스터마이징 할지를 결정해야 함.

pdf.js

- <https://mozilla.github.io/pdf.js/>
- 라이선스: Apache License 2.0

Summary

- 기본적으로 느림.(그럴 수 밖에... 특히 IE...)
- 썸네일 / 목차 기능도 제공 (하지만 백그라운드로 열일하여 만들어서 하기 때문에 느림)
- **annotation 기능 없음**
- **링크 이동 기능 없음**
- 고해상도이며 페이지가 많은 문서는 권장하지 않음: 메모리 및 생성하는 캔바스의 수가 많아지기 때문에...
- 2.0 부터는 IE10 이하 지원 중단됨.

For the (upcoming) release of PDF.js version 2.0, see <https://github.com/mozilla/pdf.js/projects/5>, we'll no longer support older browsers such as e.g. IE 9/10.

- pdfjs-1.10.100 으로 IE9(IE11에서 simulating) 테스트 성공

IE9에서 텍스트 선택 및 검색 안됨

- pdfjs-1.3.91 으로 IE9(IE11에서 simulating) 테스트 성공

IE9에서 텍스트 선택 및 검색 됨

- 기능 및 UI Customizing을 위한 기간이 필요함
- 폰트 문제 추가 검토 필요할것으로 보임

Which browsers are supported?

Browser	Supported	Notes
Firefox	Yes	
Chrome	Yes	
Opera	Yes	
IE 11/Edge	Mostly	Some missing features/defects have been reported, but no problems in general.
Safari 9+	Mostly	Some missing features/defects have been reported, but no problems in general. For mobile only iOS 10 and higher are supported.
IE 10 and below	No	Too many features are missing to be supported.

What types of PDF files are slow in PDF.js? Can I optimize a PDF file to make PDF.js faster?

Typically, PDFs with a smaller file size will be rendered faster and it depends on how big a single page is. The amount of pages does not affect the performance. It's essential that you optimize your documents for the web. See [Optimize a PDF](#) from Adobe's website for more information. There are more improvement techniques that we can suggest:

1. Avoid using high resolution images – 150 dpi resolution for scanned images shall be enough for screens, especially for low powered devices;
2. Try to use JPEG encoding for color images/photos in RGB colorspace when possible;
3. Avoid using expensive compositions/effects such as transitions/masking – flatten transparency;
4. Avoid using PDF generators (or don't create content) that produce ineffective PDF output (e.g.

LibreOffice creates a lots of tiny images for vector elements/pictures it does not understand);

5. If there is such a setting, use web-optimized PDF output / linearization;
6. Fix or don't produce corrupted PDFs that do not conform to the PDF32000

PDFObject

- <https://pdfobject.com>
- 라이선스: MIT license.

Summary

- 기본적으로는 브라우저의 기능을 이용하여 embedding하고 없을 경우 pdf.js 를 사용
- 안봐도 될듯...
- 테스트 결과

pdfjs forced mode가 있음

PDFJS.version = '1.3.91';

Which browsers are supported?

PDFObject 2.x is designed for modern browsers, and has been successfully tested in Chrome, Firefox, Safari (macOS and iOS), IE 9-11, and MS Edge.

IMPORTANT: Browser support for PDFObject does not mean the browser supports PDF embedding!
PDFObject only embeds PDFs in browsers that support PDF embedding. Some browsers do not natively support PDF embedding, including Safari for iOS and Internet Explorer [chart](#). For browsers that do not support PDF embedding, PDFObject provides a way for you to display alternate content. See below for details.

PDFTRON WebViewer – JavaScript PDF Library

- <https://www.pdftron.com/blog/webviewer/pdfnetjs-html5-pdf-viewer-and-editor/>
- 라이선스: 상용
- demo: <https://www.pdftron.com/webviewer/demo/>

Summary

- IE9 등을 지원하기 위해서는 별도의 서버를 설치 해야 함.
- 데모서버라서 그런지 속도는 느림.
- 각종 edit가 가능한 점은 좋음