Adobe Tech Blog

Adobe PDF Extract: API Output Demystified

Joel Geraci Follow 5 min read · Jun 18, 2021

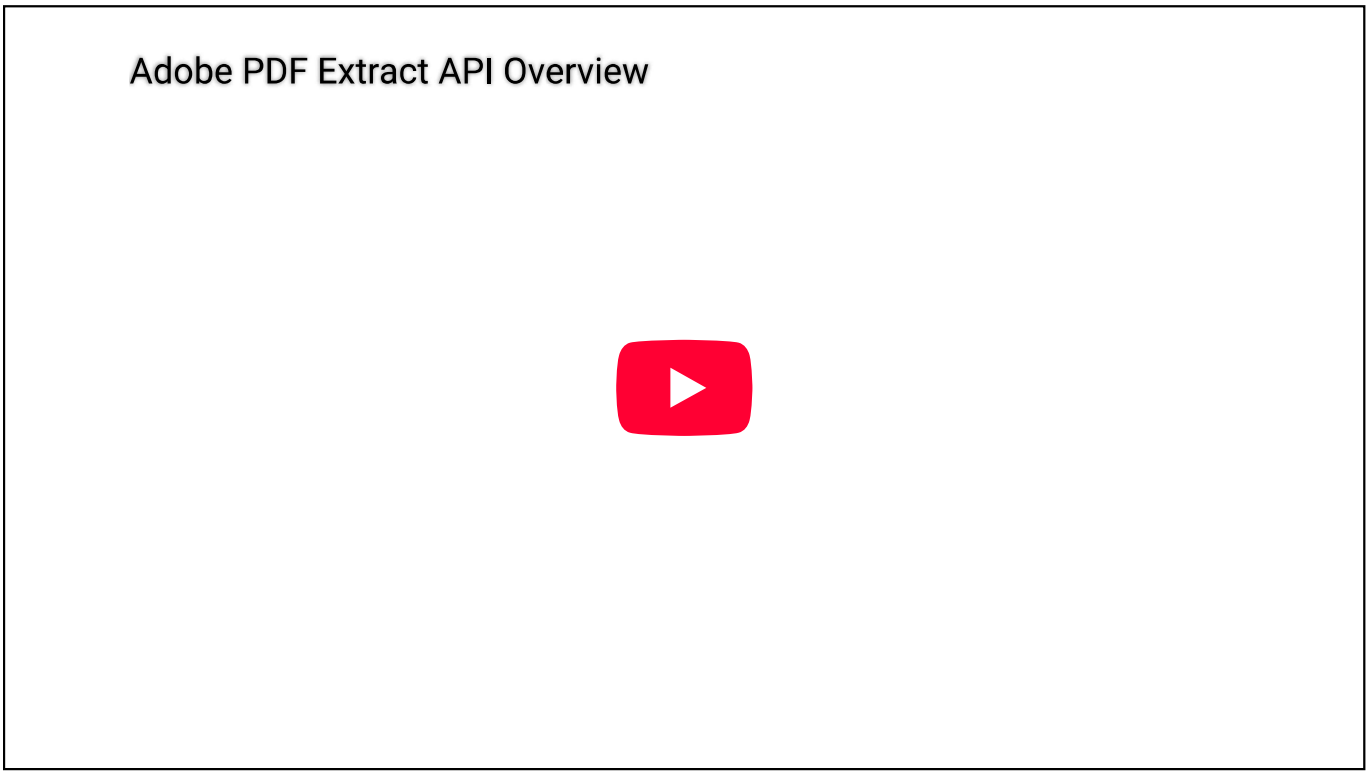
54

Since PDF's invention, getting text out of a PDF file, in the correct reading order, from any PDF, including tabular data, has been a challenge. Even the

best tools on the market were only good at one part of the problem or the other. Developers had to cobble together multiple tools depending on the type of PDF they had and the kind of data they needed to get out of it. The result was, they needed to know way too much about the PDF file before they knew which tool to use. A single API to extract the text to a usable form regardless of the actual content of the PDF simply didn't exist.

One of the main reasons that PDF can be difficult for computers to read is that there are many poor PDF renderers and engines out there. While the PDFs that these engines create might create an output that can be visually read, they often were not well-structured for computers to easily understand and glean the meaning of the reading order. They work off the philosophy "if it looks right, who cares how the code of the PDF looks underneath?"

This has continued to be a challenge, until now.



On the surface, the recent release of Adobe Extract API can be used to get the text content from a PDF file; just as the name implies. But along with that, PDF Extract API also:

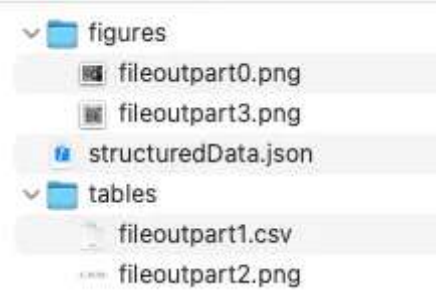
- Extracts data from the PDF in the correct reading order.
- Automatically performs OCR first if an image-only PDF is submitted.
- Extracts tables and figures separately.
- Provides detailed information about the precise position of every block of text in the PDF; even of every character if you need it.
- Provides block tags and font information in combination with the text block geometry can be used to infer content types and then develop heuristics to further parse the file and gain document intelligence.

PDF Extract API will always extract structured text from a PDF file as JSON even if the PDF is a scan of a document, but it can also optionally extract tables as separate CSV or XLS files and export images, and render illustrations and tables as PNG files.

Extracted Package

The output of Extract API is a ZIP package containing the following:

- The *structuredData.json* file with the extracted content & PDF element structure. See the JSON schema.
- One or two folders containing renditions for each element type selected as input. The folder name is either “tables” or “figures” depending on your specified element type. Each folder contains renditions with filenames that correspond to the element information in the JSON file.



This article will focus on the *structuredData.json* file and, hopefully, demystify its contents.

Content Structure

The *structuredData.json* file contains the structured information extracted from the PDF file. It contains 4 top level parts.

- **version**: The version information identifies the versions of the API components use to create and export the document's structure tree.
- **extended_metadata**: Contains metadata about the PDF document
- **elements**: An array of semantic elements (like headings, paragraphs, tables, figures) found in the document. The list is ordered based on the position of elements in the structure tree of the document.
- **pages**: A list of properties for each page of the PDF such as width, height, and rotation.

The rest of this article will discuss a few of the more interesting aspects of the elements in the elements property but if you are curious, the full JSON Schema is also available.

The elements property is an array of element objects. For developers who are just interested in the text of the PDF file, the two most interesting

properties are Text and Path. Text will be, as the name implies, the text content of the content block and Path contains the structure information where the terminal path part will look remarkably familiar to anyone with experience reading HTML code. Headings are shown by H1, H2, H3 etc. and paragraphs are indicated by P, tables, by Table and so on. A typical text element might look like this…

```
“Path”: “//Document/Sect/P”,
“Text”: “The quick brown fox jumps over the lazy dog “,
```

TextSize, LineHeight, and Font information is also available as well as the location information of the text block so the entire element might look like this…

```
{
“Bounds”: [
72.02690124511719,
719.9994964599609,
296.08961486816406,
734.8074951171875
],
“Font”: {
“alt_family_name”: “Clean”,
“embedded”: true,
“encoding”: “WinAnsiEncoding”,
“family_name”: “Adobe Clean”,
“font_type”: “Type1”,
“italic”: false,
```

```
“monospaced”: false,
“name”: “IBGRIB+AdobeClean-Regular”,
“subset”: true,
“weight”: 400
},
“HasClip”: false,
“Lang”: “en”,
“Page”: 0,
“Path”: “//Document/Sect/P”,
“Text”: “The quick brown fox jumps over the lazy dog “,
“TextSize”: 12,
“attributes”: {
“LineHeight”: 14.375
}
}
```


The Path property helps us but with the additional information about what page the block is on, font, text size and bounding box, we can infer a lot more about the semantic use of this block of text in the document. For example, I’d be able to know if the block was in the first or second column of a two-column layout or if the font were italic, it might be a caption or call out.

Once you’ve got the text plus this combination of detailed information, you can perform more in-depth document analysis, present the document in a variety of ways, and even integrate it with Adobe Embed API to create new PDF experiences.

How to get started

There is also documentation available to help you learn how to:

- Extract Text from a PDF
- Extract Text and Tables
- Extract Text and Tables (w/ Renditions)
- Extract Text and Tables and Character Bounding Boxes (w/ Renditions)
- Extract Text and Tables and Table Structure as CSV (w/ Renditions)
- (Beta Feature) Extract Text and Tables and Styling Info

PDF Extract API is part of Adobe PDF Services API, which also includes other services such as Document Generation, Create PDFs, OCR, and many other services. To learn more, go over to Adobe I/O to learn more. To get started with PDF Extract API, go to Get Started to sign up for your trial. There are SDKs available for Python, Java, and Node. There are also REST APIs available.

Pdf Extract Data Text Adobe Document Cloud

Joel Geraci is the Founder and Practice Lead of practicalPDF, a consulting company focused on helping Adobe customer unlock the full potential of PDF.

No responses yet

Write a response

What are your thoughts?

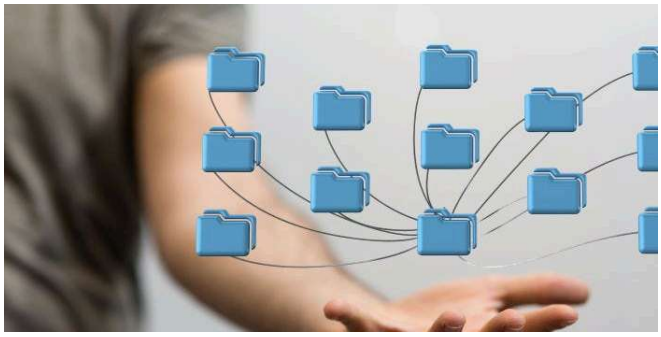
More from Joel Geraci and Adobe Tech Blog

In Adobe Tech Blog by Joel Geraci

Using the Search API in Adobe PDF Embed API

Suppress aspects of the native user interface while retaining the APIs necessary to replac...

Oct 13, 2021 2



In Adobe Tech Blog by Raymond Camden

Merging Documents Using Microsoft Power Automate

Learn how Microsoft Power Animate can be used to automate the process of merging...

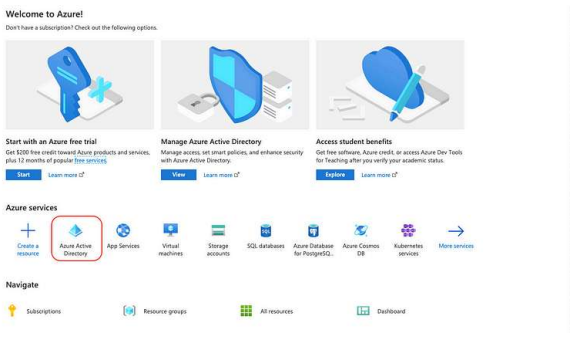
May 31, 2022 98 2

In Adobe Tech Blog by Jaemi Bremner

Taking Query Optimizations to the Next Level with Iceberg

How we are achieving read performance using Apache Iceberg.

Jan 15, 2021 260 1



In Adobe Tech Blog by Deepak Jain

SAML Authentication in AEM using Microsoft Azure Active Directory

Adobe Experience Manager has inbuilt support to use SAML based authentication...

Sep 1, 2020 20 2

See all from Joel Geraci

See all from Adobe Tech Blog

Recommended from Medium

In Artificial Intelligence in Plai... by Vivek Singh Pa...

Upload a CSV, Get a Dashboard—Building the AI Behind It (Part 1)

Dashboards shouldn't require a data team. No SQL. No BI tools. No config screens. Just...

Jun 19 137 10

In Towards AI by Lorentz Yeung

Fine-Tuning Open-Source LLMs for Text-to-SQL: Results and Key...

Analyzing 60+ runs: Performance metrics, learned patterns, and challenges in advance...

5d ago 21 1

In Generative AI by Rakib.ai

Unleash the Power of PaddleOCR: Your Guide to Best Open Source...

Want to find out about the best OCR that you can use to build AI applications at scale and...

Klippa

Mistral OCR for Document Processing: The Good, The Bad &...

Just this week, the France-based AI startup Mistral made waves in the LLM space after...

Feb 8 182

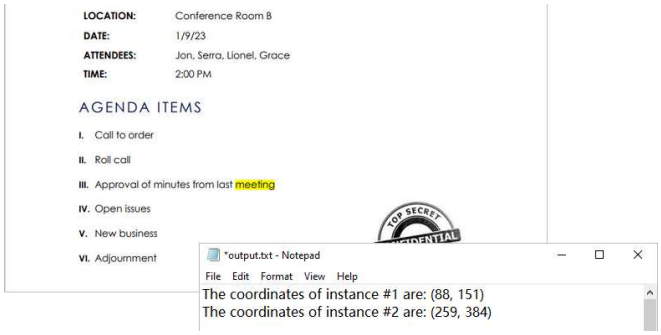
In I Am Datape... by Agha Mustafa Ali Khan Qiz...

Entity Recognition

Entity Recognition is heavily used in data modelling. Here we will discuss how it is use...

Jun 28

Mar 12 543 8



Alexander Stock

How to Extract Text and Image Coordinates from PDF Using...

PDF files are common in the digital world and often contain information that is challenging...

Mar 28 16

See more recommendations