

ADAMS

Advanced **D**ata mining **A**nd **M**achine learning **S**ystem

Module: adams-pdf



Peter Reutemann

December 21, 2016

©2012-2016



THE UNIVERSITY OF
WAIKATO
Te Whare Wānanga o Waikato



Except where otherwise noted, this work is licensed under
<http://creativecommons.org/licenses/by-sa/4.0/>

Contents

1	Introduction	7
2	Flow	9
3	Tools	13
4	Troubleshooting	15
	Bibliography	17

List of Figures

2.1	Flow for creating a PDF file from various sources.	10
2.2	CSV files get added as tables.	11
2.3	Images can get inserted as well.	11
2.4	Plain text files get added as simple text.	12
3.1	Viewer for PDF files.	13

Chapter 1

Introduction

The *pdf* module adds PDF authoring capabilities to ADAMS. This is possible thanks to the iText [2] and jPod Renderer [3] libraries for creating, manipulating and viewing of PDF files.

Chapter 2

Flow

The following source actors are available:

- *PDFNewDocument* – creates an empty PDF document.¹

The following transformers are available:

- *PDFAppendDocument* – appends a PDF document.²
- *PDFCreate* – for creating PDFs, using text files, spreadsheets and images.³
- *PDFExtract* – extracts a range of pages.
- *PDFExtractImages* – extracts the images from a PDF.⁴
- *PDFExtractText* – obtaining the plain text of the PDF.⁵
- *PDFInfo* – outputs information about a PDF file.⁶
- *PDFMerge* – merges several PDFs into a single one.⁷
- *PDFPageCount* – determines the number of pages in a PDF.⁸
- *PDFRenderPages* – renders pages of a PDF as images.⁹
- *PDFStamp* – allows to add an overlay to a PDF.¹⁰

The following sinks are available:

- *PDFCloseDocument* – closes a PDF document, writes out the content to disk.¹¹
- *PDFViewer* – for viewing PDFs.¹²

Creating PDFs

Generating a PDF using the *PDFCreate* transformer is really easy. The transformer takes an array of file names as input, which will all get added to the

¹adams-pdf-create_pdf2.flow

²adams-pdf-create_pdf2.flow

³adams-pdf-create_pdf.flow

⁴adams-pdf-extract_images.flow

⁵adams-pdf-extract_text.flow

⁶adams-pdf-info.flow

⁷adams-pdf-page_count.flow

⁸adams-pdf-page_count.flow

⁹adams-pdf-render_pages.flow

¹⁰adams-pdf-page_overlay.flow

¹¹adams-pdf-create_pdf2.flow

¹²adams-pdf-view_pdf.flow

specified output PDF file. Basic options, like page size and orientation, can be set as well. How and what files get added to the PDF, is determined by the “proclets”, i.e., little processor classes, that you specify and configure:

- *SpreadSheet* – for adding spreadsheet files as tables.
- *Headline* – for adding a headline.
- *Image* – for adding images (GIF, JPEG, PNG).
- *PageBreak* – forces a pagebreak.
- *PlainText* – for adding plain text files as paragraphs.
- *Rectangle* – draws a rectangle at specified location.

Figure 2.1 shows a flow¹³ that adds all files (i.e., three) found in a directory to a single PDF. Figures 2.2, 2.3 and 2.4 show the resulting pages of the PDF in the viewer.

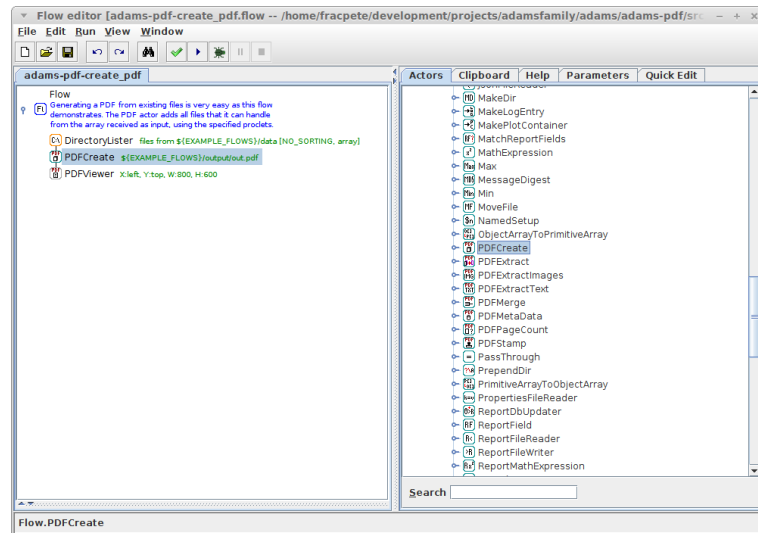


Figure 2.1: Flow for creating a PDF file from various sources.

¹³adams-pdf-create.pdf.flow

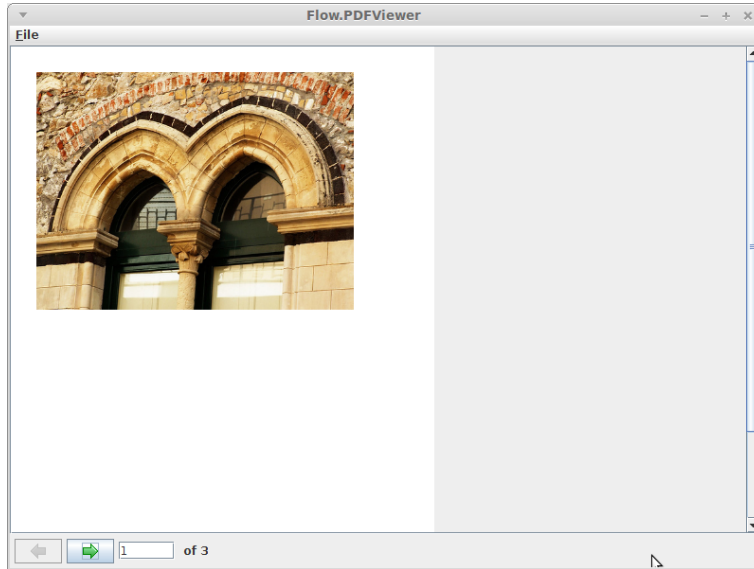


Figure 2.2: CSV files get added as tables.

 A screenshot of a window titled "Flow.PDFViewer". The window has a menu bar with "File" and standard window controls. The main content area displays a CSV table with 8 columns and 40 rows of data. The table is positioned in the top-left corner of the viewer. At the bottom, there is a navigation bar with left and right arrows, a text field showing "2", and "of 3".

RUN	SPEED1	TOTAL	SPEED2	NUMBE	SENS	TIME	T20BOL
25.0	2.0	10.0	1.5	0.0	5.0	5.7	11.4
24.0	2.0	10.0	1.5	0.0	10.0	17.8	35.1
30.0	2.0	10.0	1.5	2.0	5.0	11.3	22.6
2.0	2.0	10.0	1.5	2.0	10.0	8.4	16.8
40.0	2.0	10.0	2.5	0.0	5.0	16.7	33.3
37.0	2.0	10.0	2.5	0.0	10.0	12.0	24.1
16.0	2.0	10.0	2.5	2.0	5.0	9.5	19.4
22.0	2.0	10.0	2.5	2.0	10.0	3.9	7.9
33.0	2.0	30.0	1.5	0.0	5.0	27.0	18.0
17.0	2.0	30.0	1.5	0.0	10.0	19.5	13.0
28.0	2.0	30.0	1.5	2.0	5.0	18.5	12.4
27.0	2.0	30.0	1.5	2.0	10.0	25.7	17.1
14.0	2.0	30.0	2.5	0.0	5.0	19.9	12.7
13.0	2.0	30.0	2.5	0.0	10.0	22.4	14.9
4.0	2.0	30.0	2.5	2.0	5.0	23.9	15.9
21.0	2.0	30.0	2.5	2.0	10.0	30.1	20.1
23.0	5.0	10.0	1.5	0.0	5.0	13.4	26.8
35.0	5.0	10.0	1.5	0.0	10.0	34.3	68.5
19.0	5.0	10.0	1.5	2.0	5.0	39.7	79.5
34.0	5.0	10.0	1.5	2.0	10.0	10.6	21.2
31.0	5.0	10.0	2.5	0.0	5.0	28.9	57.8
9.0	5.0	10.0	2.5	0.0	10.0	35.6	71.2
38.0	5.0	10.0	2.5	2.0	5.0	17.6	34.4
15.0	5.0	10.0	2.5	2.0	10.0	6.0	12.0
39.0	5.0	30.0	1.5	0.0	5.0	129.4	86.3
8.0	5.0	30.0	1.5	0.0	0.0	107.4	71.6
28.0	5.0	30.0	1.5	2.0	5.0	111.7	74.4
11.0	5.0	30.0	1.5	2.0	0.0	109.1	72.7
6.0	5.0	30.0	2.5	0.0	5.0	100.4	67.0
20.0	5.0	30.0	2.5	0.0	0.0	109.3	72.6
10.0	5.0	30.0	2.5	2.0	5.0	106.5	71.0
32.0	5.0	30.0	2.5	2.0	0.0	134.0	89.3
1.0	4.0	20.0	2.0	1.0	5.0	10.8	10.8
3.0	4.0	20.0	2.0	1.0	5.0	9.4	9.4
5.0	4.0	20.0	2.0	1.0	5.0	9.8	9.8
7.0	4.0	20.0	2.0	1.0	5.0	13.9	13.9
12.0	4.0	20.0	2.0	1.0	5.0	12.3	12.3
18.0	4.0	20.0	2.0	1.0	5.0	9.3	9.3

Figure 2.3: Images can get inserted as well.

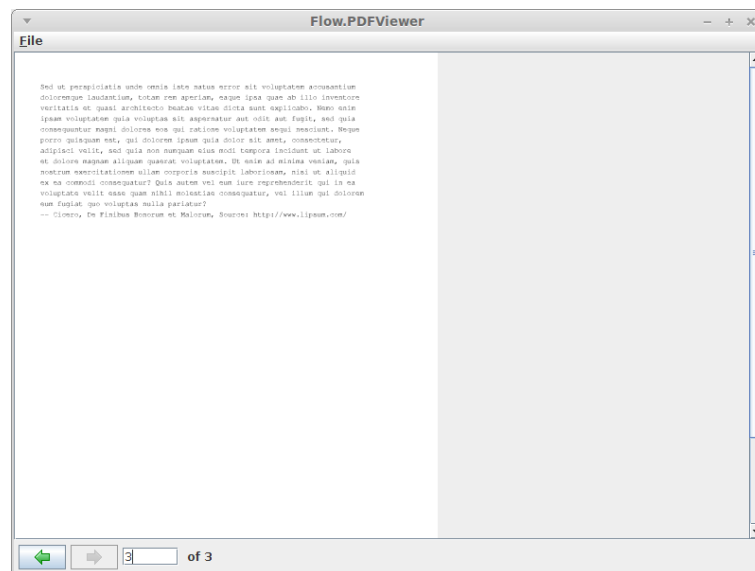


Figure 2.4: Plain text files get added as simple text.

Chapter 3

Tools

The *PDF viewer* can be used to load and browse PDF files. Figure 3.1 shows a screenshot of the viewer with a PDF file loaded.

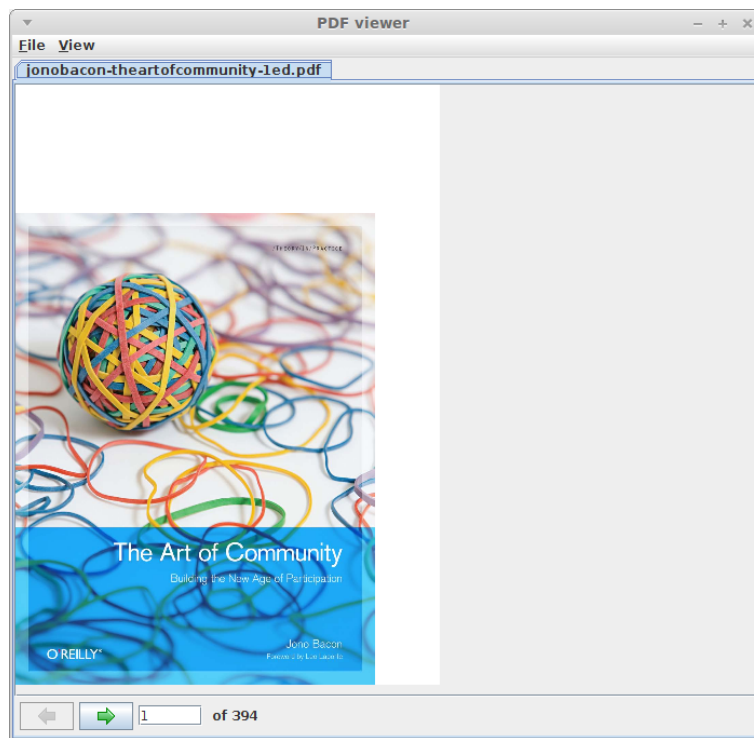


Figure 3.1: Viewer for PDF files.

Chapter 4

Troubleshooting

- **Problem:** 64bit Linux shows error message when viewing PDF that 'libfreetype.so' library was not found ("UnsatisfiedLinkError").

Solution: Make sure that /usr/lib/libfreetype.so exists. If not, add a symbolic link to the 64bit library, using a similar command as follows (for Kubuntu 11.10 or LinuxMint 11, 13):

```
sudo ln -s /usr/lib/x86_64-linux-gnu/libfreetype.so.6 /usr/lib/libfreetype.so
```

And restart the application.

Bibliography

- [1] *ADAMS* – Advanced Data mining and Machine learning System
<https://adams.cms.waikato.ac.nz/>
- [2] *iText* – A library that allows you to create and manipulate PDF documents
<http://itextpdf.com/>
- [3] *jPod Renderer* – A PDF rendering implementation for AWT and SWT.
<http://opensource.intarsys.de/home/en/index.php?n=JPodRenderer.HomePage>