

Automated Extraction of Data from PDF Documents
using Customized JMP Add-ins

Peter Vogel, PhD
CSL Behring Innovation

Automated extraction of data from PDF documents using customized JMP Add-ins

- 01**
Introduction
- 02**
The Approach
- 03**
Creating an Add-in
- 04**
What's next?

01

Introduction

Introduction

Data is often not directly accessible due to

- Old software systems
- Proprietary software
- ...

yet highly relevant for analyses and decisions

Typical result

- Expensive manual data retrieval **OR**
- Lack of insights as data retrieval is too expensive

Example

- Structured data stored in PDF files
- JMP add-in to automatically parse the PDFs

Produkt: SFP RIASTAP 1G CA/CBS TOLL H69 P-699 6.0

Lot-Nr.: P100264126

LIMS Lot-Nr.: 589972

LIMS Produkt-Spezifikation: G5170

Stufe: NONE

LIMS Proben-Nr.	Start	Ende	Anforderung	OOX-Nr.	Abweichungs-Nr.	Resultat	Einheit	Status
END DATE 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: - WG: -	-	-	28.09.2020	-	PV
Lösezeit 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: <= 15 min WG: <= 10 min	-	-	3	min	PV
START DATE 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: - WG: -	-	-	28.09.2020	-	PV
END DATE 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: - WG: -	-	-	28.09.2020	-	PV
Organoleptische Prüfung 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: PASS WG: -	-	-	PASS	-	PV
START DATE 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: - WG: -	-	-	28.09.2020	-	PV
END DATE 1 / 04-010 / SOP:								
15324072	30-Sep-20	01-Oct-20	AG: - WG: -	-	-	01.10.2020	-	PV

Figure 1: Exemplary PDF with measurement data

02

Approach

The Approach

Guiding Principles

- Understand your question at hand
- Break it down into smaller modules
- Let JMP do the heavy lifting for you!
- Modules have defined in- and outputs
- Focus first on functionality, then appearance / UX

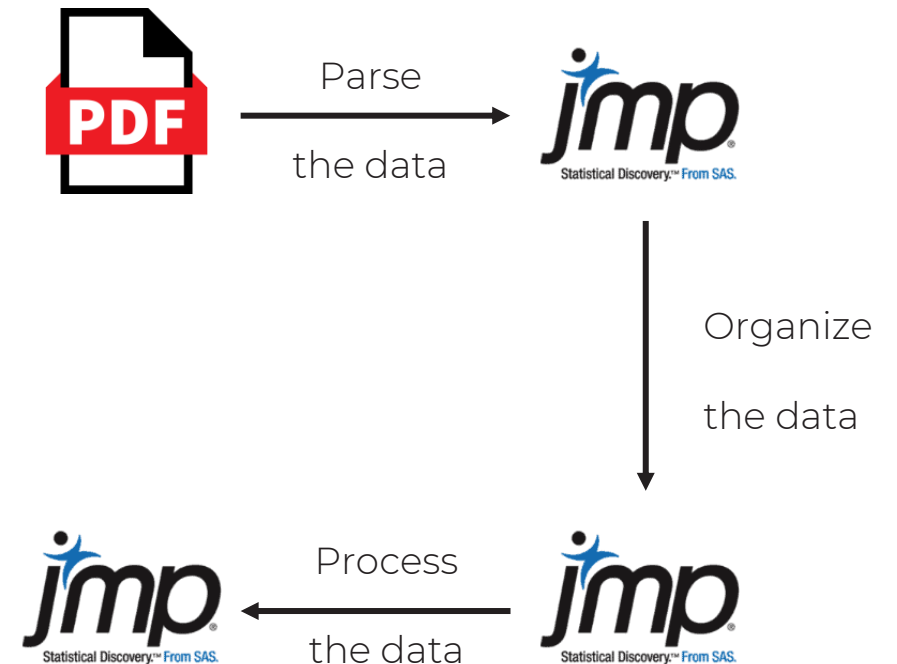


Figure 2: High-level workflow

The Approach - Understand your question

Let's inspect the structure of our PDFs

- First page
 - General header with project description
 - Header of the actual data table
 - Information on 1+ samples, typically with 4 lines per sample
- Interior pages
 - Same structure as first page
- Last page
 - Can contain sample information
 - Contains legend

General header

Sample information header

CSL Behring						Analysenbericht		
LOT REPORT Dieses Lot ist vollständig								
Produkt: SFP RIASTAP 1G CA/CBS TOLL H69 P-699 6.0								
Lot-Nr.: P100264126			LIMS Lot-Nr.: 589972			LIMS Produkt-Spezifikation: G5170		
Stufe: NONE								
LIMS Proben-Nr.	Start	Ende	Anforderung	OOX-Nr.	Abweichungs-Nr.	Resultat	Einheit	Status
END DATE 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: - WG: -	-	-	28.09.2020	-	PV
Lösezeit 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: <= 15 min WG: <= 10 min	-	-	3	min	PV

Figure 3: Page 1 of the PDF with annotations

The Approach - Understand your question

Let's inspect the sample data

LIMS Proben-Nr.	Start	Ende	Anforderung	OOX-Nr.	Abweichungs-Nr.	Resultat	Einheit	Status
END DATE 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: - WG: -	-	-	28.09.2020	-	PV
Lösezeit 1 / 04-003 / SOP:								
15324072	28-Sep-20	28-Sep-20	AG: <= 15 min WG: <= 10 min	-	-	3	min	PV

Figure 4: Exemplary sample data

The Approach – Break it down

Break it down into modules

- High-level workflow (revisited)

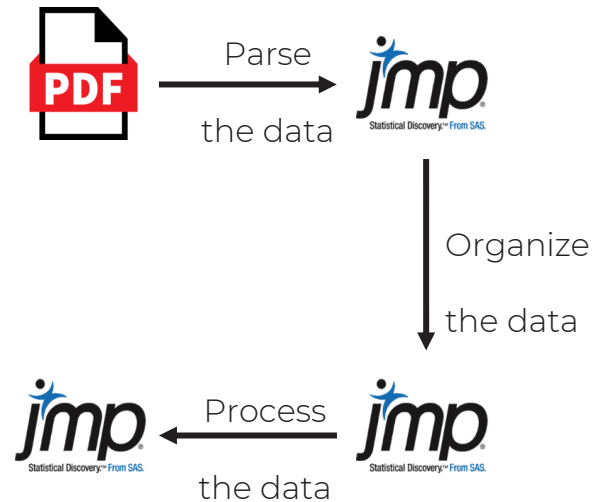
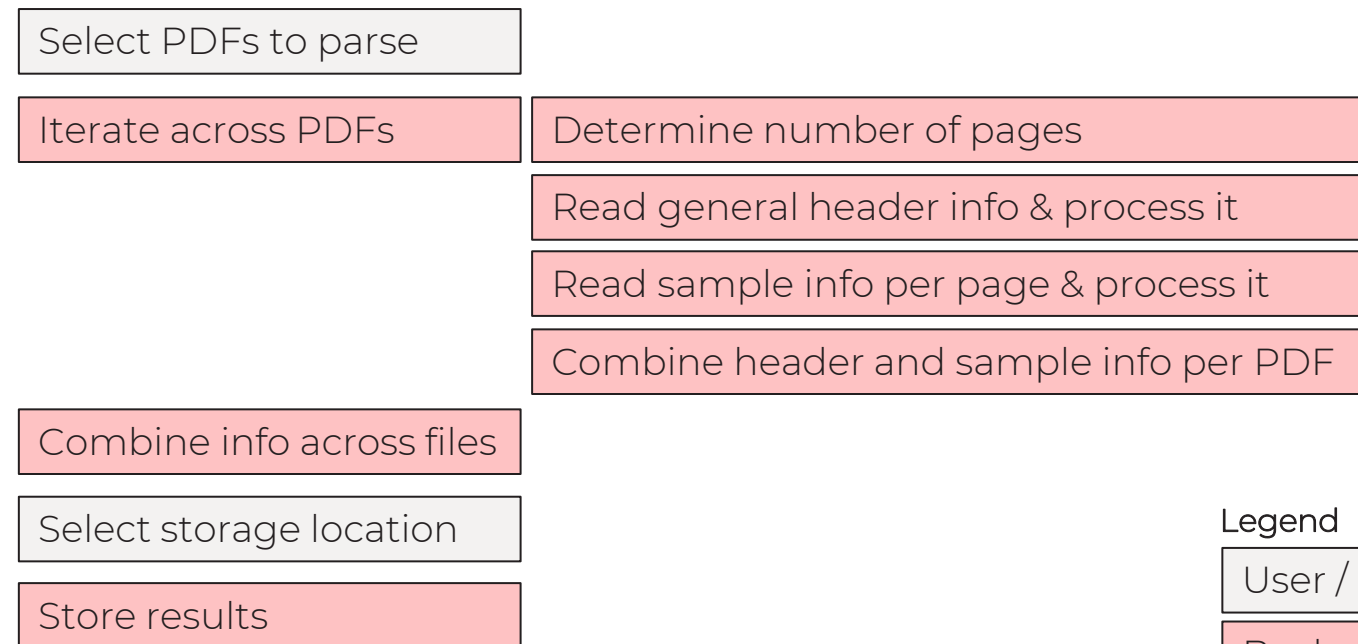


Figure 2 (revisited)

In smaller steps



Legend

User / on GUI

Background

Figure 5: Breaking down the workflow

The Approach – Utilize JMP

Utilize the PDF Wizard in JMP

- Interactive dialog to parse PDFs to JMP
- Simple configuration of relevant properties
- Creates scripts for you

Examples

- 1) General header on page 1
- 2) Sample information on all pages

→ Demonstration in JMP

Produkt: SFP RIASTAP 1G CA/CBS TOLL H69 P-699 6.0								
Lot-Nr.: P100264126			LIMS Lot-Nr.: 589972			LIMS Produkt-Spezifikation: G5170		
Stufe: NONE								
LIMS Proben-Nr.	Start	Ende	Anforderung	OOX-Nr.	Abweichungs-Nr.	Resultat	Einheit	Status

Figure 6:
Preview in PDF Wizard

	Column 3	Column 4
1	Produkt: SFP RIA...	
2	Lot-Nr.: P100264...	LIMS Produkt-Sp...

```
dat_header = Open(  
  filename,  
  PDF Tables(  
    Table(  
      table name( "Freigabedaten_Beispiel Page(1) Table(1)" ), // Table name  
      add rows(  
        page( 1 ), // Page  
        Rect( 1.0042, 1.7658, 7.8158, 2.205 ) // Rectangle to read from  
      )  
    )  
  )  
);
```

Figure 7: Code with minor modifications & comments

The Approach – Utilize JMP

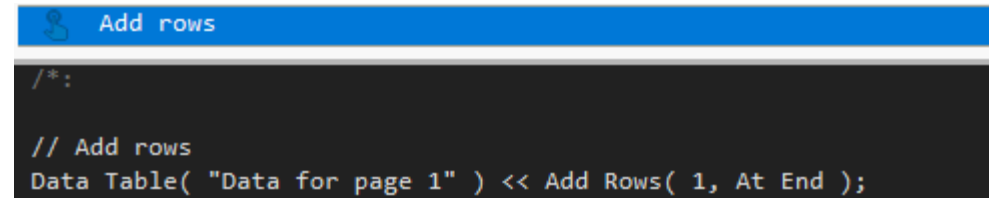
Utilize Log, Scripting Index, and other features

- Log - Records your actions on GUI as code
- Scripting Index lists functions and examples
- Formula editor for interactive formula creation
- Copy Table Script provides code for data table

Examples

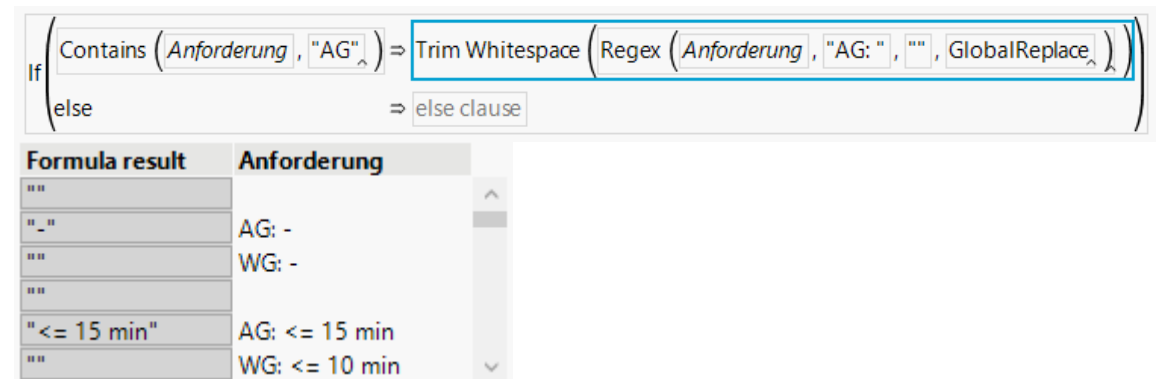
3) Process sample information for one page

→ [Demonstration in JMP](#)



```
/*:  
// Add rows  
Data Table( "Data for page 1" ) << Add Rows( 1, At End );
```

Figure 8: Add rows as captured in Log



Formula result	Anforderung
""	
"_"	AG: -
""	WG: -
""	
"<= 15 min"	AG: <= 15 min
""	WG: <= 10 min

Figure 9: Formula editor example for the AG component

The Approach – Defined inputs and outputs

Use functions to make your code modular

- Enforces defined in- and outputs
- Easier to maintain and debug code
- No need to copy & paste
- Motivates a good documentation of code

Examples

4a) Read sample data on page 1 as function

4b) Transform sample data as function

→ [Demonstration in JMP](#)

```
// Read the sample measurement information
// @param filename Character, specifies the PDF file to be parsed
// @note The page is currently still fixed to page 1
read_LIMS_sample_data_page = function({filename}, {Default Local},

    dat_content = Open(
        filename,
        PDF Tables(
            Table(
                table name( "Sample measurement information" ),
                add rows(
                    header rows( 1 ),
                    page( 1 ),
                    Rect( 1.0118, 2.5739, 7.7325, 10.5607 ),
                    column borders(
                        1.0118, 1.921, 2.6872, 3.2007, 4.4469, 5.2805, 6.2153, 6.8429,
                        7.3683, 7.7325
                    )
                )
            )
        )
    );
    dat_content = dat_content[1];

    // Return
    // Define what you want to return, can be one or multiple objects
    // Here the data table dat_content
    return(dat_content)
);
```

Figure 10: Read sample data (on page 1) as a function

The Approach – Improve UX once add-in is functional

UX improvement options

Examples

- All data tables are created as invisible only
- Meaningful instructions
- Helpful links & scripts
- Inform user about progress of code execution
- Intuitive and structured graphical user interfaces
- Collect additional information about issues of code execution

```
dat_page = New Table("dat_page",  
    New Column("A", Character),  
    invisible  
);  
  
// Or  
dat_page << Show Window(0);
```

Data successfully read and stored
The specified files have been successfully stored in the Jmp data table

Test.jmp

in the directory linked below.

[Link to directory](#)

Progress report - JMP

Progress report

Reading results from LIMS reports
Reading information from file [1/5]
Reading values on page [8/12]

Figure 12: A few options for UX improvements

03

Creating an add-in

Creating a JMP add-in

Why

- Easy to deploy
 - One file only
 - One-click installation
- Easy to utilize
 - Integrated in JMP GUI
 - Usage without interaction with scripts
- Metainformation at your fingertips
 - Author
 - Version number of add-in
 - JMP version requirements
 - ...

How

- JMP Add-in Manager
 - Guides user through add-in creation process
 - [Link](#) to JMP community post
- Manual or script-based
 - Higher effort, but deeper understanding
 - Can leverage JMP resources (File > New > Add-In)
 - Can leverage other add-ins (View > Add-Ins)
- Components
 - Definition file → addin.def
 - GUI integration file → addin.jmpcust
 - Actual JMP code

Creating a JMP add-in

Step-by-step

- addin.def
 - Create a unique ID for this add-in
- addin.jmpcust
 - Define integration in GUI, e.g., via File > New > Add-in
- JMP code
 - The core functionalities developed previously

→ [Demonstration in JMP](#)

The fully functional add-in

- Installation
- Example
 - Reading in 7 PDF files (here: 7 copies of same file)

→ [Demonstration in JMP](#)

04

What's next

What's next?

Celebrate!

Creating an add-in is truly an achievement!

Extend and maintain your add-in

- Code versioning & collaborative development
- Unit testing
- Deployment & updates at larger user base

[git](#)

[Hamcrest](#)

?

Love to hear feedback & questions

Peter Vogel, PhD

Global Digital Core, Plasma Product Development

Peter.Vogel@csllbehring.com