Recovering structure from PDF

Key takeaways

1. A PDF paints character shapes (glyphs), lines, and images at coordinates. A tool must recover rows, columns, merged cells (spans), and reading order when the PDF has no labels for that structure.

2. Different tables can leave the same marks on a page. Base each structural decision on a mark that supports it, and record unresolved structural alternatives in warnings.

3. A checkable recovery keeps cell structure, uncertainty, and evidence tied to the original file. A plain-text Markdown table or a careless rewrite can discard them.

Common misunderstandings of recovering tables from PDF

Which claim would you defend right now? Choose before the evidence.

"A PDF stores a document's text and tables; extraction mostly means parsing them out."

"If the extracted Markdown table looks right, the recovery is right."

"When two extraction tools agree on a cell, the cell is probably correct."

Keep your choice. Look for the mechanism or counterexample that would make it fail.

Instapoll

Instapoll · Review · Oct 22 A

A LaTeX paper’s untagged PDF shows a table whose header “Latency (ms)” spans two columns. What does the PDF file itself record about that header?

- A. A table cell holding the text, with a column span of two
- B. The text, tagged as the header of the two columns beneath it
- C. Glyphs painted at page coordinates, and a short line drawn under them
- D. The LaTeX command that produced the span, kept for later editing

A PDF page is a drawing program

PDF drawing commands from the pdflatex compiler; numbers are rounded

Text at coordinates, and lines

```
BT /F32 9.96 Tf 231.4 395.2 Td
% one operator paints a whole row
[(System)-6005(p50)-2714(p99)] TJ
ET
% a short line under the header
q 1 0 0 1 321.6 406.5 cm
0.299 w 0 0 m 60.0 0 l S Q
BT /F32 9.96 Tf 231.4 378.3 Td
% numbers adjust character spacing
[(Lo)-28(c)28(k-free)-2208(ring)] TJ
% down a line, before the numbers
0 -11.96 Td [(queue)] TJ
93.92 11.96 Td [(1.2)-2992(8.4)] TJ
ET
```

Table 1: Queue latency under load.

	Latency (ms)	
System	p50	p99
Lock-free ring queue	1.2	8.4
Mutex	1.9	31.0

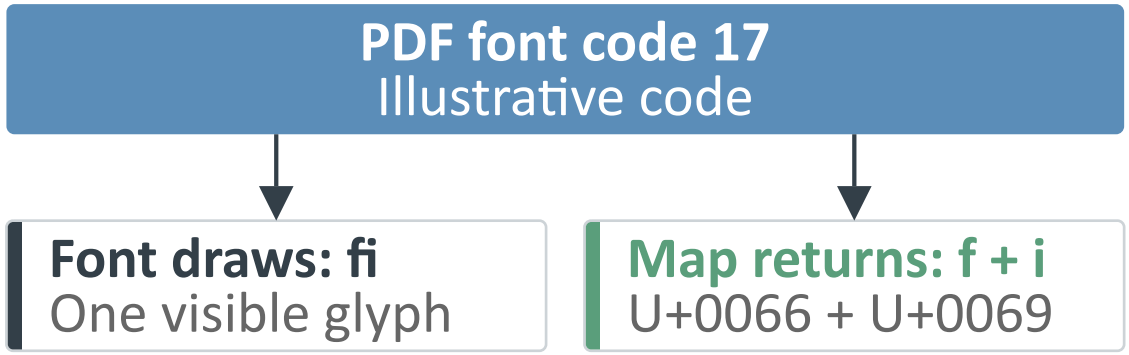
Structural tags label rows and cells; this PDF has none. A rule is a drawn line.

Latency is response time in milliseconds (ms). p50/p99 mark the 50th/99th percentiles.

A Unicode map connects codes to characters

The code used to draw a shape need not be the standard number for its character

One stored code, two uses



A font defines drawn shapes. One shape is a glyph; a ligature such as fi combines letters in one glyph.

What the terms mean

Unicode	Standard character numbers, called code points. U+0066 names f.
Unicode map	Lookup from PDF font codes to Unicode characters, used for copying and extraction.

Look-alikes: − is minus (U+2212); - is hyphen (U+002D).

A missing or incorrect map can leave the page looking right while extracted text is wrong.

Recovering characters, reading order, and table structure

1 **Glyph → characters**
ﬁ → f + i
A glyph is a font's drawn shape.

2 **Characters → words**
r i n g → ring
Gaps group characters into words.

3 **Words → lines**
Lock-free ring
Aligned bottoms of letters suggest one line.

4 **Lines → reading order**
① ↓ ② ③ ↓ ④
Read down columns, or across the page?

5 **Lines → cells**
System | p50 | p99
Alignment and rules suggest columns.

6 **Pages → one table**
[A] + [B] → [A, B]
Join continuations; keep one header.

An error low on the ladder surfaces higher up as a wrong table.

Table extraction failures

Two PDF readers and optical character recognition (OCR), which guesses text from pixels

PDFPLUMBER
Rules

Table 1: Queue latency under load.

		Latency (ms)	
System		p50	p99
Lock-free queue	ring	1.2	8.4
Mutex		1.9	31.0

→ no table

No vertical rules are detected, so this configuration finds no table.

PDFPLUMBER
Whitespace

```
Latency │(ms)
queue   │·│·
```

The header splits into two cells. “queue” becomes its own row.

Rules also produce two blank rows (not shown).

PYMUPDF
Text stream

```
ring
 ↓ queue
 ↓ 1.2 → 8.4
```

No table is found. “queue” arrives before the numbers beside “Lock-free ring”.

TESSERACT
OCR

p50 → po0

The 5 becomes an o. Plain text also drops OCR’s confidence score (its estimate of certainty).

The marks allow more than one table

Recovery works backward from the drawn page; different tables can produce the same marks

Reading A: what the author wrote

```
3 columns; header spans 2
0  ·              | Latency (ms)
1  System         | p50 | p99
2  Lock-free ring | 1.2 | 8.4
   queue          |     |
3  Mutex          | 1.9 | 31.0
```

Reading B: also consistent with the marks

```
4 columns
0  ·          | ·    | Latency | (ms)
1  System     | ·    | p50     | p99
2  Lock-free  | ring | 1.2     | 8.4
3  queue      | ·    | ·       | ·
4  Mutex      | ·    | 1.9     | 31.0
```

Many logical tables paint the same page, and the page alone does not always say which one it came from. Justified text stretches spaces to align both edges of a line; the stretched gap makes “ring” look like a column.

Which marks support the reading?

Schematic cell boxes; the PDF has none. Measured distances use points (pt), 1/72 inch.

Spanning header

Latency (ms)	
p50	p99

partial rule

The header and the rule extend over both numeric columns. This supports a spanning header.

An isolated “ring”

Lock-free	ring
queue	
Mutex	

Only “ring” occupies this column. This favors justified text; an otherwise empty column is still possible.

Row or wrap?

Lock-free	→ ↓ 12.0 pt
queue	→ ↓ 11.9 pt
Mutex	

Almost equal gaps cannot distinguish a wrapped cell from a new row. Record both interpretations in a warning.

Record the ambiguity in warnings: “queue may wrap the previous cell or start a new row.”

Two readers can copy the same OCR error

A searchable scan can store both the page image and hidden text guessed from it

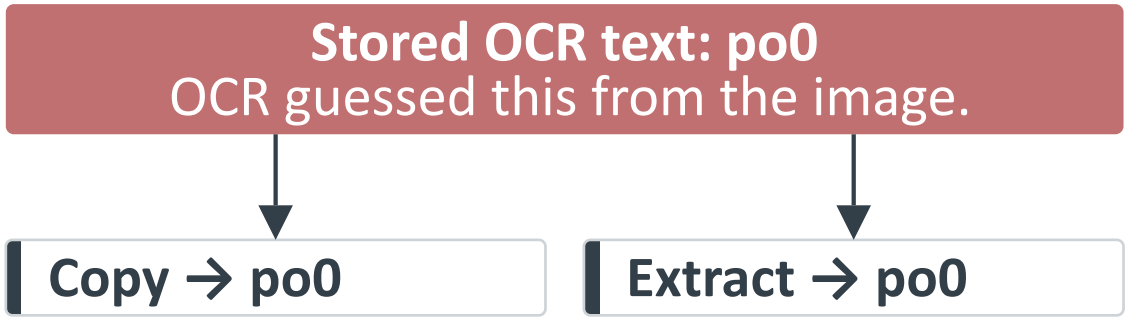
The page image you see

Table 1: Queue latency under load.

		Latency (ms)	
System		p50	p99
Lock-free queue	ring	1.2	8.4
Mutex		1.9	31.0

The printed heading is p50.

The hidden text tools read



Both operations can reuse the stored guess without reading the pixels.

Agreement between readers of the same stored text is not independent confirmation.

TableBundle records the recovered tables

HW4 input: a PDF + a request naming the tables. Required output: a TableBundle JSON file.

Your output file

answer.json

```
SIMPLIFIED OUTLINE; SELECTED FIELDS
source: PDF identity
tables: [
  dimensions: rows, columns
  cells: [
    row, column, spans
    text, status
    evidence: page, rectangle
    warnings: structural alternatives
  ]
]
provenance: methods used
```

Cell status describes text

Report unknown explicitly; an omitted cell has no status.

known	Text is determined. text: “” means verified empty.
unknown	text: null. The cell exists, but its text cannot be determined.
conflict	text: null. Keep each candidate and its evidence.

Positions and spans are concrete. Warnings record structural uncertainty.

Identify the file, then locate the evidence

source.sha256 identifies the file; page rectangles locate its evidence

Which exact PDF?

PDF bytes → **SHA-256** file fingerprint

SHA-256 computes a fingerprint of the whole file. It lets the reviewer detect a different PDF, even with the same filename. Keep the PDF; its hash cannot recover it.

Where on its page?

Page 1 **Rectangle**

Bounding box (bbox): upper-left and lower-right corners in points (1/72 inch) from the full page's top-left. x right; y down.

Use page rectangles, not byte offsets: a cell can span commands or exist only as pixels.

Recover the cell, then record how to check it

One worked example, from source marks to a cell record; HW4 does not mandate this sequence

1. Infer from the page

Table 1: Queue latency under load.

		Latency (ms)	
System		p50	p99
Lock-free queue	ring	1.2	8.4
Mutex		1.9	31.0

The partial rule supports a two-column header. The blue box locates the text.

2. Record the answer

```
text: "Latency (ms)"
row: 0
column: 1
column_span: 2
status: "known"
```

Rows and columns start at 0; pages start at 1.

3. Attach evidence

```
page: 1
bbox: [322.6, 372.2,
       380.6, 382.2]
```

Save the source rectangle with the cell. The renderer highlights it so a reviewer can check the reading.

The rectangle locates evidence. It does not prove that your proposed cell is correct.

Markdown export loses structure and evidence

Earlier: infer the structure. Now: preserve it when exporting the result.

Required structured record

Latency (ms)	
p50	p99
1.2	8.4

One header spans two columns. Its record also carries a source box and a status.

Optional Markdown view

```
Latency (ms)

| p50 | p99 |
|-----|-----|
| 1.2 | 8.4 |
```

The shared header becomes a caption. A plain pipe table has no standard fields for the span, status, or source box.

Text cleanup must also preserve spans, status, and evidence. Validate the complete bundle.

Key Takeaways

1. A PDF page is a drawing program: it paints glyphs, lines, and images at coordinates. Without structural tags, an extractor must infer rows, columns, spans, and reading order. An error low in that chain surfaces as a wrong table.

2. Different tables can leave the same marks. Base each structural decision on a mark that supports it — a partial rule, an aligned extent, a repeated header. Record unresolved structural alternatives in warnings; use unknown or conflict for uncertain cell text.

3. Make the recovered table checkable: identify the exact PDF and attach page rectangles to the cells. Keep structure, uncertainty, and evidence in the structured result when cleaning text or exporting Markdown.

Write the Markdown table

10 min

Table in the PDF

	Elapsed time (s)	
Workload	1 worker	4 workers
Build index	12.0	4.0
Query	6.0	2.0

Illustrative values.

Your task

- Write one Markdown pipe table for the table shown. Preserve the labels, values, and units. Use one header row and no HTML.
- Identify one structural decision you made and one piece of information your Markdown cannot preserve.