

WordCraft

ALPHA

clone of Microsoft Word

Write, restyle, cite, redline and mail-merge .docx files. You describe the document; Claude edits it through the wordcraft MCP server and renders the page so you can see it.

v0.7.0

Driven from Claude Code via MCP
Richard Young • richard@deepneuro.ai
2026-10-15

Alpha software. WordCraft is a pre-1.0 ArtCraft app. Expect missing features, rough edges and the occasional crash; keep originals safe and let Claude work on copies.

16

MCP tools

617

engine commands

9

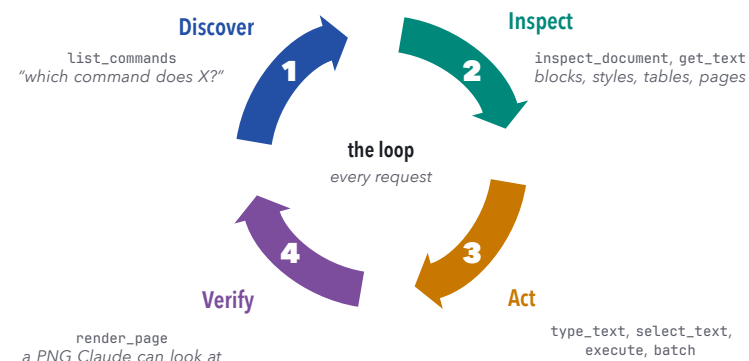
file formats written

5

starter templates

01 The idea in one picture

You never touch the ribbon. You type a goal in Claude Code; Claude turns it into MCP tool calls; the server runs engine commands on an in-memory document; and `render_page` hands back a PNG so Claude (and you) can see the page before it is saved.



i Check it is wired up

In Claude Code type `/mcp. wordcraft` should show *connected* with 16 tools. Smoke test: “Use wordcraft to open the sample document and tell me how many pages it has.” The document lives in the server process: nothing is on disk until Claude calls `save_document`, so always finish with “save it to ...”. Save format follows the extension: docx, pdf, odt, rtf, html, md, tex, txt, png.

02 First five minutes

Every image on this page is real output from WordCraft, captured through the MCP server while writing this guide. Paste the prompts as they are.

- 1 Start from a template** “Use wordcraft: new document from the report template and render page 1.” A PNG of the page comes back. That proves the connection works (storyboard 1).
- 2 Replace the placeholders** Tell Claude what the title, subtitle and summary should say; it runs `edit.replaceAll` for each (storyboard 2).
- 3 Add something visual** A native chart from numbers you paste, a table from CSV, or a picture with a caption.
- 4 Look, then fix** End every request with “render it and check”. `inspect_document` is cheap JSON; `render_page` costs image tokens, so use it when layout matters.
- 5 Save** “Save as ~/Documents/pilot.docx and also export a PDF.”

Storyboard 1: a report skeleton in one call

YOU, IN CLAUDE CODE

"Use wordcraft. Start a new document from the report template and render page 1 so I can see the layout."

MCP CALLS CLAUDE MADE

- 1 new_document {"template": "report"}
- 2 render_page {"page": 1, "scale": 1.2}

WHAT CAME BACK

Report Title

Subtitle or date

Summary

Summarise the findings in two or three sentences.

Background

Describe the context and the question the report answers.

Findings

First finding

Explain the evidence.

Recommendations

List what should happen next.

The report template as rendered: Title, Subtitle, Summary, Background, Findings, Recommendations, already styled with Heading 1/2 so a TOC works later.

Pilot Study Results

MIS 301 · Fall 2026

Summary

Students who used Claude Code with the Craft MCP servers finished the lab 38% faster and reported higher confidence. Error rates were unchanged.

Background

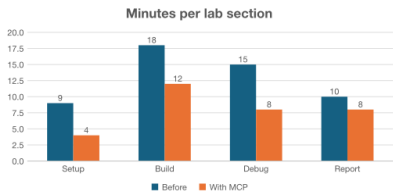
Describe the context and the question the report answers.

Findings

First finding

Completion time fell from 52 to 32 minutes (n = 41). The chart below shows the per-section breakdown.

Recommendations



The same page after storyboard 2. Other templates: blank, sample, letter, resume.

Storyboard 2: fill it in and add a native chart

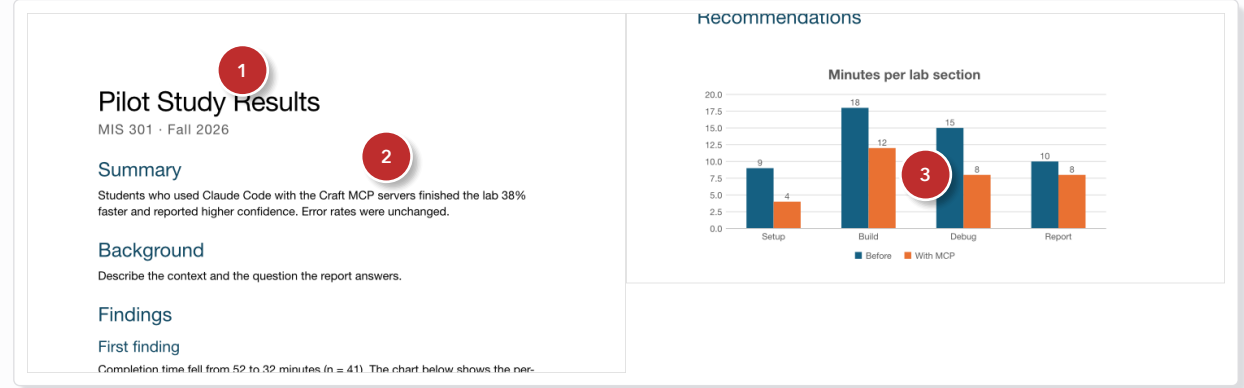
YOU, IN CLAUDE CODE

"Title it "Pilot Study Results", subtitle "MIS 301 · Fall 2026". Summary: students using Claude Code with the Craft MCP servers finished the lab 38% faster. First finding: completion fell from 52 to 32 minutes ($n = 41$). Under Recommendations add a column chart, minutes per lab section: Setup 9 vs 4, Build 18 vs 12, Debug 15 vs 8, Report 10 vs 8. Render it."

MCP CALLS CLAUDE MADE

- 1 `batch` 4 × `edit.replaceAll` {"text": "Report Title", "with": "Pilot Study Results"} ...
- 2 `select_text` {"text": "List what should happen next."}
- 3 `batch` `text.newParagraph`; `insert.chart` {"type": "column", "title": "Minutes per lab section", "data": {"categories": [...], "series": [{"name": "Before", "values": [9, 18, 15, 10]}, {"name": "With MCP", "values": [4, 12, 8, 8]}], "dataLabels": true}}
- 4 `render_page` {"page": 1, "scale": 1.2}

WHAT CAME BACK



- 1 Title and subtitle replaced by `edit.replaceAll`; styles untouched.
- 2 Summary text swapped in; the Background placeholder below is untouched because nobody named it.
- 3 Lower on the same page: a native, editable chart object placed where `select_text` left the caret, not a picture.

Prompts to explore with

●●● warm-up ●●● real work ●●● stretch

●●●

"Use wordcraft: list the commands that mention "toc" or "citation"."

See how Claude discovers commands with `list_commands`.

●●●

"Open ~/Documents/any.docx, give me its heading outline and word count, and render page 1."

`inspect_document` and `review.wordCount` read a real file; nothing is changed.

●●●

"Convert ~/thesis/ch3.md to Word, apply 1-inch margins and a 2-level hyperlinked TOC, save as ch3.docx."

Open, restyle, TOC, save, in one go.

●●●

"Replace every "Licensee" with "Customer" (whole word) and show me the page where the first one was."

`edit.replaceAll` plus a targeted `render_page`.



"Insert this CSV as a banded table sorted by Score descending, with a caption 'Table 1'."

insert.spreadsheet, table.sort,
references.caption.



"Compare syllabus-v1.docx and v2.docx as tracked changes, comment on every deleted paragraph, export a marked-up PDF."

review.compare, review.newComment, file.
exportPdf with markup.



"Mail-merge this letter with students.xlsx, one letter per advisor, then render the first page of each and tell me if any name overflows."

Stretch: Claude loops merge, render, inspect.

03 Goals → how

Each card is a prompt you can paste, the tools Claude reaches for, and the command ids doing the work.

Create, convert & automate

1 Draft a report, save DOCX + PDF

ASK CLAUDE

"Start a report with a title, Introduction and Methods from these notes; save as ~/Documents/report.docx and a PDF."

TOOLS `new_document` → `type_text` → `execute`
→ `save_document`

KEY `template: report;` `para.heading1`, `para.style` ;
`file.exportPdf`

2 Convert between formats

ASK CLAUDE

"Convert ~/thesis/ch3.md to Word, and proposal.docx to ODT and PDF."

TOOLS `open_document` → `save_document`

KEY reads docx, doc, odt, rtf, md, tex, html, txt;
writes by extension

● Bulk: `wordcraft-cli convert in out` in a loop.

3 Use a template's styles

ASK CLAUDE

"Create a memo from ~/templates/memo.dotx, or copy its Heading styles into this report."

TOOLS `execute`

KEY `file.newFromTemplate`, `styles.copyFrom`
(`path`, `styles`), `tools.attachTemplate`

4 Record a reusable macro

ASK CLAUDE

"Define a macro 'course-format' (Georgia headings, 1-inch margins) and run it on each syllabus."

TOOLS `execute` → `save_document`

KEY `tools.macros` (`define: name + steps`, `run`)

● Persistence across sessions not documented; a replayed `batch` works too.

Edit & format

5 Find and replace (regex too)

ASK CLAUDE

"Replace every 'Licensee' with 'Customer' (whole word) and reformat dates like 3/7/26 as 2026-03-07."

TOOLS `execute` → `get_text`

KEY `edit.replaceAll` : `text`, `with`, `wholeWord`, `matchCase`, `regex`

8 Page setup, headers, watermark

ASK CLAUDE

"1-inch margins, footer 'Page X of Y', course name in the header, diagonal DRAFT watermark."

TOOLS `batch`

KEY `layout.margins` (pt), `insert.header`, `insert.pageNumber` (format: x of y), `design.watermark`

- All lengths in points: 72 pt = 1 in.

6 Restyle the whole document

ASK CLAUDE

"Give this a cleaner look: new theme and style set, Georgia headings, open paragraph spacing."

TOOLS `batch` → `render_page`

KEY `design.theme`, `design.styleSet`, `design.themeFonts`, `design.paragraphSpacing`, `styles.modify`

7 Fix lists and numbering

ASK CLAUDE

"Number the steps under 'Procedure', make sub-steps lettered level 2, restart numbering in section 2."

TOOLS `select_text` → `execute`

KEY `para.numbering` (kind), `para.listLevel`, `para.restartNumbering`, `list.define`

Academic writing & references

9 Table of contents

ASK CLAUDE

"Check headings use Heading 1/2, then add a 3-level hyperlinked TOC after the title page."

TOOLS `inspect_document` → `execute`

KEY `references.toc` (levels, hyperlinks, tabLeader); `references.updateToc`

10 Citations and bibliography

ASK CLAUDE

"Switch to APA, cite Smith2024 (p. 12) after the 2nd Background sentence, add References at the end."

TOOLS `select_text` → `execute`

KEY `references.citationStyle`, `references.citation` (tag, pages), `references.bibliography`

- Source fields undocumented: let Claude read `references.sources` first.

11 Footnotes, captions, cross-refs

ASK CLAUDE

"Footnote 'IRB approval' with the protocol number; caption all figures; link 'see below' to Figure 2."

TOOLS `select_text` → `execute`

KEY `references.footnote`, `references.caption`, `insert.crossReference` (to, target, show)

12 Equations from LaTeX**ASK CLAUDE**

"Insert the cross-entropy loss as a displayed, numbered equation after the Method paragraph."

TOOLS `select_text` → `execute`

KEY `insert.equation (latex, display);`
`equation.number`

■ Tables, figures & mail merge**13 Formatted table from CSV****ASK CLAUDE**

"Insert this results CSV as a banded table with a header row, sorted by accuracy descending."

TOOLS `execute`

KEY `insert.spreadsheet (csv), table.look,`
`table.sort (column, descending)`

16 Mail merge from a spreadsheet**ASK CLAUDE**

"Merge this letter with ~/advising/students.xlsx (FirstName, Advisor) into merged.docx."

TOOLS `open_document` → `batch`

KEY `mailings.start, mailings.recipients (path,`
`sheet), mailings.insertField,`
`mailings.finish`

● Labels too: `mailings.labels (product: letter-3x10).`

14 Picture with alt text**ASK CLAUDE**

"Put ~/figs/arch.png at 5 in wide under Architecture, alt text, square wrap, caption it."

TOOLS `select_text` → `execute`

KEY `insert.picture (path, width, alt),`
`arrange.wrap, references.caption`

15 Native, editable chart**ASK CLAUDE**

"Add a column chart of enrollment: Fall 23 112, Spring 24 98, Fall 24 131, with data labels."

TOOLS `execute`

KEY `insert.chart (type, data: categories + series,`
`title, dataLabels).` Storyboard 2 shows one.

Review, audit & sharing

17 Tracked changes and comments

ASK CLAUDE

"Track changes as 'R. Young', tighten the abstract, comment on uncited claims, export a marked-up PDF."

TOOLS `execute` → `type_text` → `save_document`

KEY `file.setAuthor`, `review.trackChanges`,
`review.newComment`, `file.exportPdf` (markup)

18 Redline two versions

ASK CLAUDE

"Compare syllabus-v1.docx with v2 as tracked changes in a new document; save syllabus-redline.docx."

TOOLS `execute` → `save_document`

KEY `review.compare` (original, path, showIn: new)
● Moves, comments, footnotes options: "accepted; not honoured yet".

19 Scrub before sending

ASK CLAUDE

"Remove comments, revision authors and metadata, run the accessibility checker, make it read-only."

TOOLS `execute` → `save_document`

KEY `file.inspect` (remove), `file.privacyOptions`,
`file.accessibility`, `file.protect`

20 Audit without editing

ASK CLAUDE

"Give me the handbook's heading outline, word and page counts, styles used and proofing issues."

TOOLS `inspect_document` → `execute` → `render_page`

KEY `review.wordCount`, `review.issues`,
`styles.manage`, `file.info`

✓ Always add: "render it and check"

`render_page` returns a PNG Claude can actually see (page, scale pixels per point). Ask Claude to render after layout changes; for structure questions `inspect_document` is cheaper.

04 The command map (617 ids behind execute)

Every ribbon button is a dotted command id. Darker tiles are bigger families. Ask Claude to "list wordcraft commands matching word" to open any tile.

50

para.

headings, styles, numbering, spacing

48

table.

rows, merge, look, sort, borders

40

insert.

table, picture, chart, header, equation

39

format.

bold, font, size, color, highlight

35

review.

track changes, compare, comments, counts

34

view.

zoom, layout, rulers, panes

29edit.
find, replace, undo, paste, goTo

27file.
new, open, save, exportPdf, protect

26outline.
promote, demote, move, collapse

25references.
toc, citation, footnote, caption

21text.
insert, newParagraph, delete

20equation.
number, align, matrix, fraction

18layout.
margins, orientation, columns, breaks

17mailings.
start, recipients, fields, finish

16styles.
modify, copyFrom, manage

16draw.
ink pens, highlighter, shapes

16developer.
content controls, XML mapping

16caret.
move by word, line, paragraph

42select. picture. arrange.
selection; crop, wrap; align, group

74other
design, smartArt, chart, tools, help

05 Tool reference (all 16 MCP tools)

MCP tool	What it does
list_commands	Every command (id, label, ribbon location, shortcut, params); optional query filter.
execute	Run one command id with params.
batch	Run a list of commands in order; stops at first error unless keepGoing.
new_document	New doc; template: blank, sample, letter, resume, report.
open_document	Open path (.docx .doc .odt .rtf .md .tex .html .txt .json, or any text file).
save_document	Save; format from extension. No path = overwrite in place.
type_text	Type at the caret, replacing the selection; paragraphs: true splits lines.
select_text	Select the n-th (occurrence) match of text.
get_text	Plain text of the body.
inspect_document	Structure: blocks, styles, runs, lists, tables, sections, selection, pages.
render_page	Page to PNG (page, scale px per pt).
parity	Coverage of the word-processor feature catalog.
screenshot	Window screenshot (desktop app only).
click	Click at window x, y (desktop app only).
key	Key press with modifiers (desktop app only).
ui_inspect	Tab, panes, dialog, caret (desktop app only).

06 Example tool calls

Storyboard 2: fill the template (tool: batch)

```
{
  "commands": [
    {
      "command": "edit.replaceAll",
      "params": {
        "text": "Report Title",
        "with": "Pilot Study Results"
      }
    },
    {
      "command": "edit.replaceAll",
      "params": {
        "text": "Subtitle or date",
        "with": "MIS 301 - Fall 2026"
      }
    },
    {
      "command": "edit.replaceAll",
      "params": {
        "text": "Summarise the findings in two or three sentences.",
        "with": "Students who used Claude Code with the Craft MCP servers finished the lab 38% faster."
      }
    }
  ]
}
```

Storyboard 2: the native chart (tool: execute)

```
{
  "command": "insert.chart",
  "params": {
    "type": "column",
    "title": "Minutes per lab section",
    "data": {
      "categories": ["Setup", "Build", "Debug", "Report"],
      "series": [
        {
          "name": "Before",
          "values": [9, 18, 15, 10]
        },
        {
          "name": "With MCP",
          "values": [4, 12, 8, 8]
        }
      ]
    },
    "dataLabels": true
  }
}
```

Redline two versions into a new document (tool: execute)

```
{
  "command": "review.compare",
  "params": {
    "original": "~/courses/syllabus-v1.docx",
    "path": "~/courses/syllabus-v2.docx",
    "author": "R. Young",
    "level": "word",
    "showIn": "new"
  }
}
```

07 Limits and tips

! Alpha: work on copies

The README puts WordCraft at about 60% of the way to replacing Word for real work. Charts and SmartArt may not survive a .docx round trip yet, and pagination differs from Word's. Copy originals first, save to a *new* name, and re-check important output in Word or LibreOffice.

i Headless means in memory

Nothing hits disk until `save_document` or `file.exportPdf`. Ask Claude to save and report the path. `save_document` with no path overwrites the opened file in place.

! No file sandbox by default

If the server is registered without `--automation-read-root` / `--automation-write-root`, commands may read and write any file you can. Add those flags (e.g. your home folder) if you want a sandbox.

 Work efficiently

Selection first: `select_text` puts the caret where the next `type_text` or `execute` acts (that is why the chart in storyboard 2 landed under Recommendations). Lengths are points, colours RRGGBB; proofing is offline. One-shot jobs skip MCP: `wordcraft-cli convert / text / inspect / render / run`.