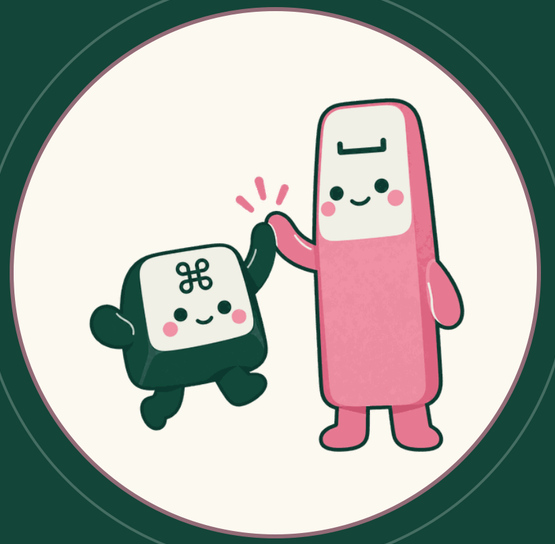
The Knowledge System of Leading Researchers

Yohan Koo's research WIKI and agents. On real screens, see how papers turn into research ideas and reports.



01

Part 1

Compile papers with an LLM Wiki

02

Part 2

Find research questions in the main vault

03

Part 3

Run it with research agents

Based on my own framework and the system I actually run.

CMDSPACE

You collected the papers.
Why hasn't the research moved?

Collecting, reading, and managing are different jobs. When one folder carries all three, all three stall.



01 **Collect** PDFs pile up in the Downloads folder

02 **Read** summaries are made on the spot, then lost

03 **Write** you can't find them again when needed

KEY POINT

Count how many papers you actually reused last month.

Today's map on one page

Four boxes from source to output. Each box does a different job, and today we go from left to right.

STEP 01
Source library
Zotero keeps PDFs and citations, as is

STEP 02
LLM Wiki
literature compiled with one template

STEP 03
Main vault
my projects and my interpretations

STEP 04
Research agents
run collection, checks, briefings

KEY POINT
Mark which of the four boxes your materials are stuck in now.

ROLE BOUNDARY
Sources: never edited / **Wiki**: AI organizes / **Vault**: I interpret / **Agents**: propose, human confirms

Opening the system I use every day

Today's screens were not built for a demo. They are the vault and wiki I actually run.

01
Main vault
more than 10,000 notes

02
LLM Wiki
about 1,100 pages

03
Paper analyses
16 papers with source quotes

04
In use
over 4 years since the first daily note

KEY POINT

Every screen today comes from a system in real use.

Part 1
Building the LLM Wiki

We compile papers and technical documents with the same template.

Zotero guards the source library

Citations, PDFs and highlights live in Zotero alone. The wiki and vault point to sources by a single citekey.

STEP 01
Zotero
keeps citations, PDFs and highlights as originals

STEP 02
Citekey
Better BibTeX gives each paper a stable key

STEP 03
Plugin
CMDS Zotero inserts citations, imports notes, opens the PDF page

STEP 04
Wiki and vault
only citekeys in the text, citations read from Zotero

KEY POINT

Keep citations in Zotero only, and leave just the citekey in your notes.

ROLE BOUNDARY

Zotero: the citation source, the plugin only reads / **Wiki and vault**: point to sources by citekey

A summary compresses.
A wiki compiles.

A summary is written once and then disappears. A wiki leaves the source intact and builds knowledge by splitting it by rules.



01 **Summary** made in a chat, buried with the chat

02 **Search** digs through sources again for every question

03 **Wiki** organized in advance and kept growing

KEY POINT

Make your next question start on top of your last notes.

ROLE BOUNDARY

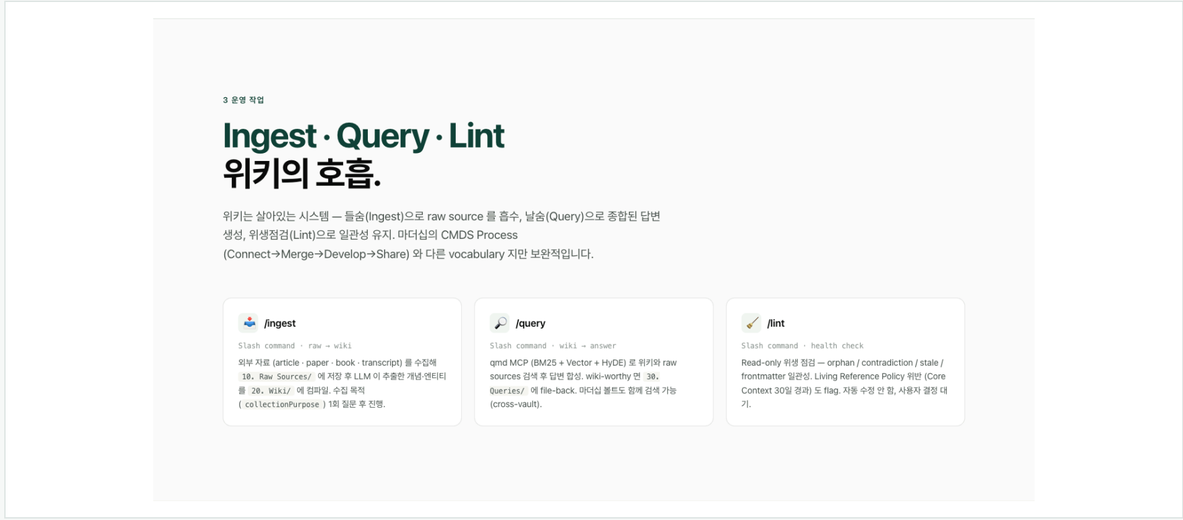
Sources: never changed / **AI**: splits and links by rules / **Researcher**: sets the rules

Based on my own framework and the system I actually run.

Karpathy, LLM Wiki · LLM Wiki public guide · Jisan: Between raw and wiki

The three layers of an LLM Wiki

Sources, wiki, rules. The rules file, more than the model, decides wiki quality.



LLM Wiki public guide: operations overview (captured 2026-09-10)

01 Sources papers and materials stored unchanged

02 Wiki concepts, claims, methods linked as pages

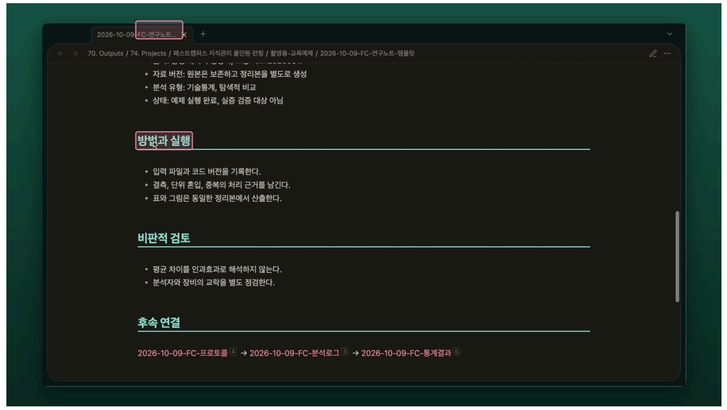
03 Rules a file stating what to keep and how

KEY POINT

Before building a wiki, write down what it should keep.

Every paper kept under the same fields

The template is your research standard. Shared fields are what let you line up many papers side by side later.



Research note template (teaching example, synthetic data)

01 Question and method what was asked, and how

02 Claims and evidence source location for each claim

03 Limits and relations supports, refutes, uncertain

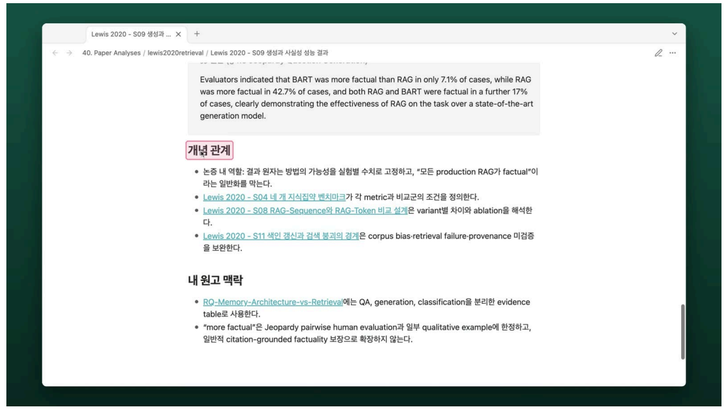
04 My research where can I use it

KEY POINT

Pick five fields that fit your field today.

Keep the source location for every claim

Where the evidence sits outlasts any summary. When you cite later, you won't have to dig through the source again.



Results note for Lewis 2020 RAG (wiki paper analysis, marked unverified)

01 Source quote the exact sentence beside each claim

02 Concept links how it connects to other papers

03 My manuscript where it could be cited

KEY POINT

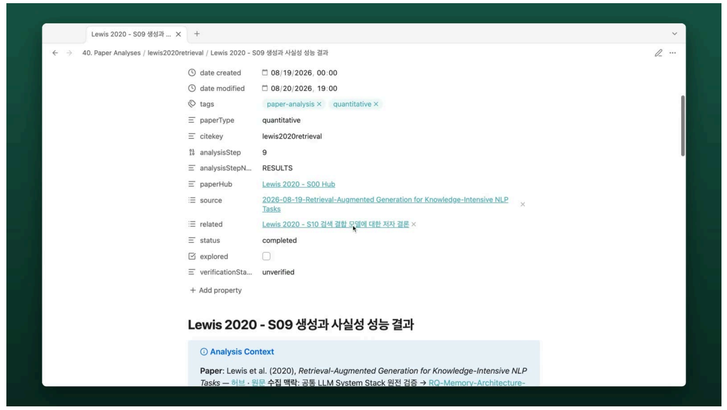
Record one source location before writing one summary line.

Still frame of the real wiki / unverified labels left as is.

Lewis et al. 2020, RAG LLM Wiki public guide

Split each paper into a hub and stage notes

Instead of one long summary per paper, we split it into one hub and stage notes, and pull out only the stage we need.



Stage note properties. Hub, source, and related notes are linked

01 Hub the entry point to the whole paper

02 Stage notes method, results, limits kept apart

03 Properties record source, hub, review status

KEY POINT

Many reusable pieces beat one long summary.

Still frame of the real wiki / unverified labels left as is.

Start tonight
LLM Wiki Starter Kit

The three-layer structure you saw today is public as an empty vault. Download it, open it in Obsidian, and start.

Scan with your phone camera to open the starter kit page.

01 **Structure** sources, wiki, queries, paper analyses

02 **Rules** the rules file agents read

03 **Commands** ingest, query, and lint steps

KEY POINT

Today, add just one paper and see if it is organized by the rules.

Part 2
Linking the Main Vault

We connect the wiki's literature knowledge to my projects and interests.

The wiki goes wide, the vault goes deep

The wiki is a broad reference layer organized by AI. The main vault holds my own interpretation. Link them; don't mix them.



01 **Reference layer** the wiki, many papers organized broadly

02 **Interpretation layer** main vault, my words and projects

03 **Connection** links and questions that move between them

KEY POINT

Don't mix AI-written notes and your own interpretation in one folder.

ROLE BOUNDARY

Wiki: AI writes, human reviews / **Main vault**: human writes, AI assists

Three devices that link the two layers

One link is not enough. Questions, two-way links, and cross-search together keep the connection alive.

01

Research question page

the knot tying evidence to projects

02

Two-way links

vault to wiki, and wiki to vault

03

Cross-search

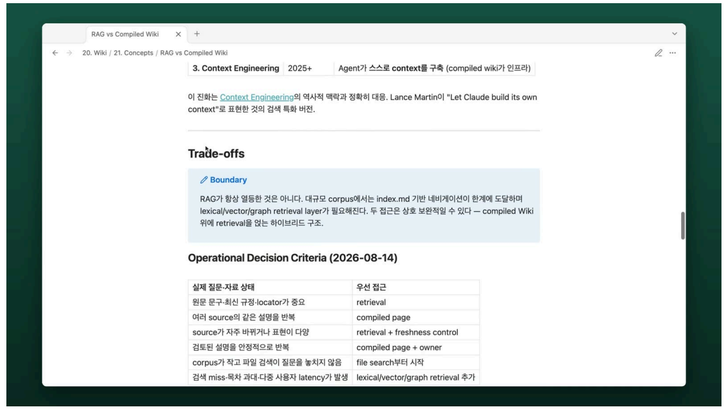
one question across both vaults

KEY POINT

Turn one research question into a page and gather related papers there.

One question ties the two layers

The question "When do you need search, and when is a wiki enough?" grew into a concept page and a decision table.



Wiki concept page 'RAG vs Compiled Wiki': boundaries and decision criteria

01 Question does a growing wiki need search

02 Evidence claims from many sources on one page

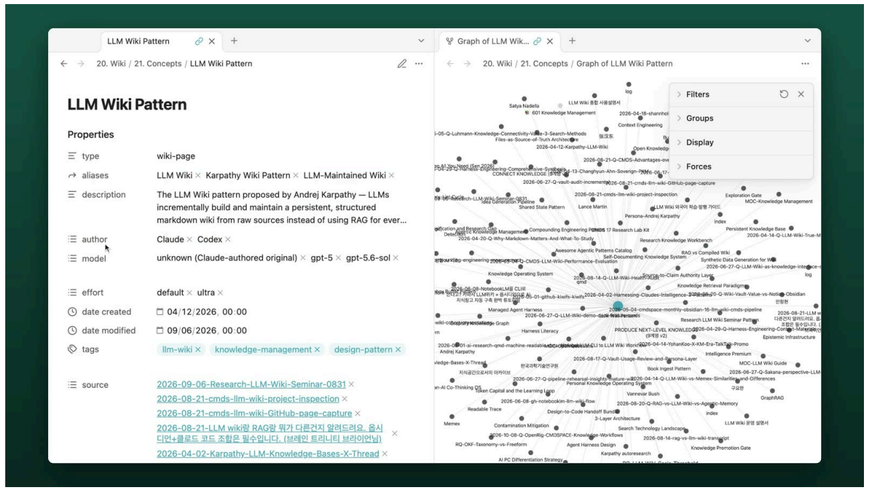
03 Criteria a decision table by situation

KEY POINT

Write the question as a sentence, and evidence starts to gather under it.

Follow the graph to keep researching

Start from one page and follow links to nearby concepts, and you can see what to read next.



LLM Wiki Pattern page and its local graph (real wiki)

01 Start the concept page you are reading

02 Nearby linked concepts and sources

03 Next thin links point to what to read next

KEY POINT

Pick your next paper from the thin edges of the graph.

Still frame of the real wiki / unverified labels left as is.

LLM Wiki public guide Jisan: Topology of knowledge 1

Live: one question searches both vaults

We give an agent one research question and have it find wiki evidence and main vault notes together.

질문: 위키가 커지면 언제부터 검색이 필요해지는가

어디서	찾은 노트
위키 (AI가 정리)	Wiki Search Methods Comparison
위키 (AI가 정리)	RQ: LLM Wiki 패턴이 수백 소스·수천 페이지에서도 index.md 네비게이션만으로 충분한가
위키 (AI가 정리)	LLM Wiki 운영 설명서
위키 (AI가 정리)	Ingest-Query-Lint Cycle
메인 볼트 (내가 씀)	**검색은 휘발성, 위키는 자산**
위키 (AI가 정리)	LLM Wiki Pattern
위키 (AI가 정리)	Obsidian Tooling for LLM Wiki
위키 (AI가 정리)	MOC-LLM Wiki Knowledge Interface Lecture

두 볼트를 한 번에 검색한 사전 실행 결과 (2026-10-15, qmd 교차 검색, 상위 8건)

Pre-run of the same question (2026-10-15). Fallback if the live demo stalls

01 Question one research question set in advance

02 Search wiki and main vault at the same time

03 Result evidence pages and my notes together

KEY POINT

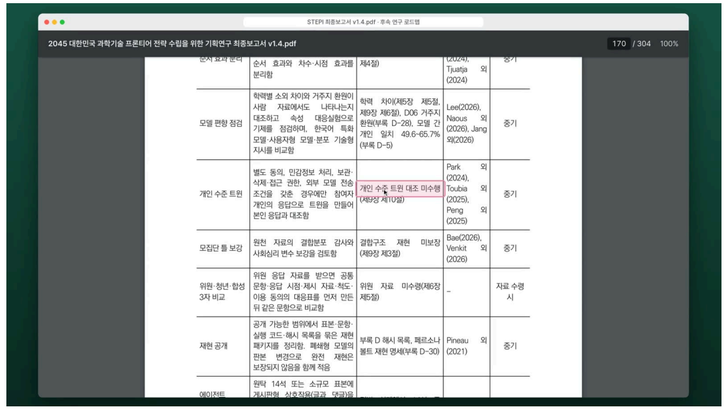
Check that wiki evidence and your own notes appear side by side.

ROLE BOUNDARY

Agent: searches and groups both vaults / **Researcher**: chooses what to use

Use the same columns, and the gaps show

Putting 8 prior studies and this one into the same 7 columns exposed unverified cells, which became follow-up research tasks.



Follow-up research roadmap table in the STEPI final report (by CMDSPACE)

01 Same columns prior and current studies aligned

02 Unverified not "none," just not checked this time

03 Follow-up blank cells become roadmap items

KEY POINT

Treat blank cells as candidates to check, not confirmed gaps.

From blank cells to follow-up research

We check the blank cells against a map of claims and evidence, then turn them into a research plan you can carry out.

STEP 01
Comparison table
literature side by side on shared fields

STEP 02
Evidence map
sorted into supports, refutes, uncertain

STEP 03
Gap candidates
where evidence is thin or conflicting

STEP 04
Follow-up design
question, method, data needed

KEY POINT
Type the one question you're holding into the chat in one line.

Part 3
Research Agents

From literature review to report drafts, we look at agents that run your research records.

My agents work on a harness

An agent is a model plus a harness (loop, tools, context, permissions). A skill is a procedure the agent uses.

01
Harness
Claude Code and Codex, 8 work rigs on two Macs

02
Agents
9 9Yohan specialists and one conductor

03
Skills
about 120, from decks to paper intake to inboxes

04
Highlights
policy-study support, blind model review, daily briefings

KEY POINT

Before building an agent, write the repeated procedure down as a skill.

ROLE BOUNDARY

Harness: runtime and permissions / **Agent**: plans and runs / **Skill**: procedure / **Human**: approves

Based on my own framework and the system I actually run.

Anthropic: agent harness design · Claude Code skills docs · 9Yohan agent design · Agents and harness overview

A skill ledger for every kind of work

I keep all 121 skills in a ledger by kind of work. The 18 base skills that enforce vault rules are shared by all four areas.

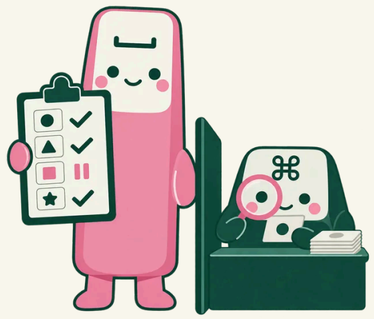
01	**Research, 20**	multi-research, research-verify
02	**Teaching and writing, 30**	cmds-lecture-builder, thebetter-writer
03	**Projects, 31**	cmdspace-web-builder, ask-inbox
04	**General work, 22**	cmds-email, meeting-minutes

KEY POINT

Pick one task you repeat and write it down as your first skill.

Three agents and one promise

Agents don't do the research for you; they run the research records. Changes happen only after human approval.



01 Ingest gather new material into the wiki

02 Maintenance check errors, gaps, changed evidence

03 Briefing report current status and next steps

KEY POINT

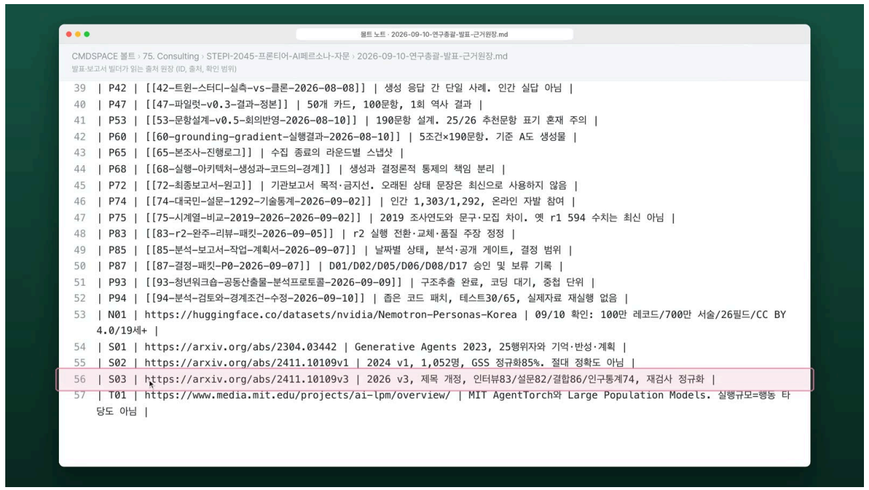
Let agents read freely, but write only after your approval.

ROLE BOUNDARY

Agents: find, check, propose / **Researcher**: confirms and approves

From one number back to the source abstract

An evidence ledger traces a number in the report back to the exact source sentence in three steps.



STEPI evidence ledger. Source URL, check date, and scope for each report figure

01 Report sentence the number and its citation

02 References go to the source URL

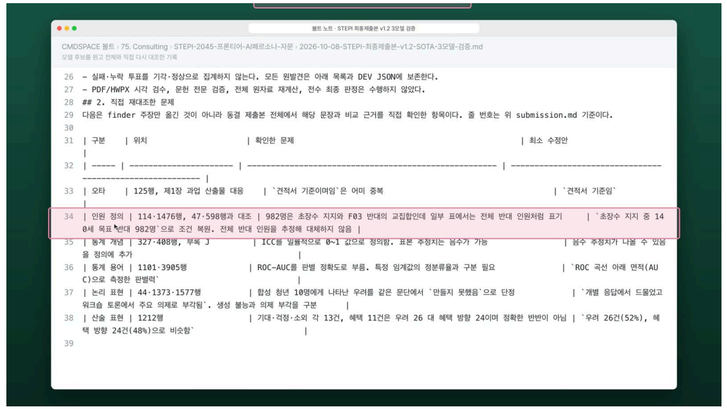
03 Evidence ledger date and scope of the check

KEY POINT

Leave a path back to the source for every number in your report.

Agents flag first, people confirm

Before submission, several models read the draft and flagged candidates. Candidates aren't errors; people confirmed them.



STEPI pre-submission review note. Model-flagged candidates rechecked by a person

01 Candidates several models flag the same sentence

02 Distinction candidate count is not error count

03 Confirmation a person rechecks the full draft

KEY POINT

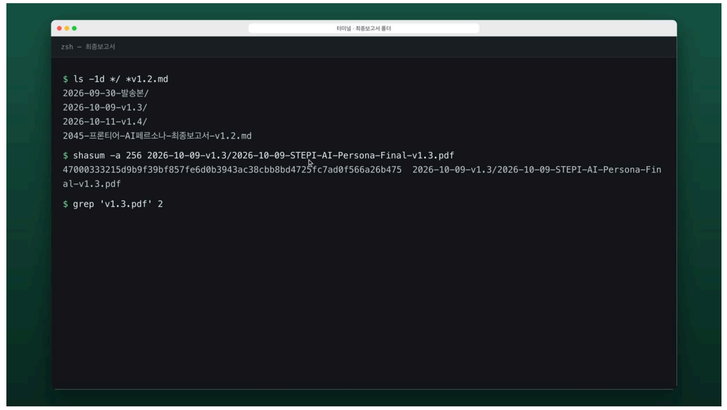
Treat model flags as candidates; confirm only after checking the source.

ROLE BOUNDARY

Models: flag candidate issues / **Researcher**: confirms against the manuscript

Accept the point, decide the fix, keep the original

We accept the critique, but people decide how to fix it, and earlier versions are kept with fingerprints, not deleted.



STEPI report version folders and SHA-256 fingerprint check (real terminal run)

01 Accept valid points go into the next version

02 Not followed keep the numbers, add missing steps

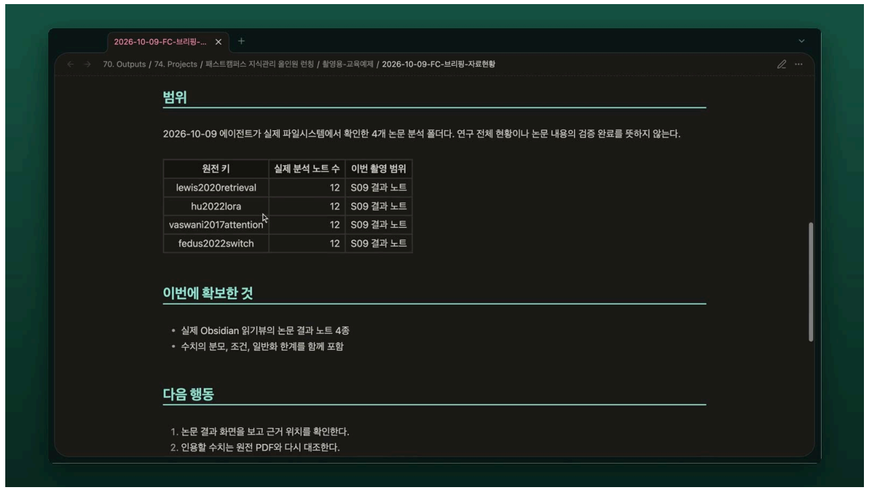
03 Preserve earlier versions verified by fingerprint

KEY POINT

Keep the previous version each time you revise, so you can always go back.

A daily briefing tells you what's next

The agent checks the actual files and reports what exists, what's missing, and what to do next.



Research status briefing (course example; agent checked real files)

01 Scope state what was checked first

02 Secured list only what actually exists

03 Next steps proposed in priority order

KEY POINT

Make every briefing state its scope and what it could not check.

Building the three stages yourself

Today you saw a finished system. In the course, you start from a starter vault and build the same structure for your research.



01 **Repository** designing an Obsidian research vault

02 **Research WIKI** an AI-readable wiki and comparison tables

03 **Agents** ingest, maintenance, and briefing

04 **Finish** carry your own research question through

KEY POINT

Run one research question of yours through all four boxes once.

What to open tonight

Along with the course materials, here are four public resources behind today's talk.

01

Wiki starter kit

an empty three-layer wiki

02

System files

public rules that agents read

03

AKM Index

self-assess your system

04

Jisan blog

the essays behind today's argument

KEY POINT

Open just one of these four tonight.

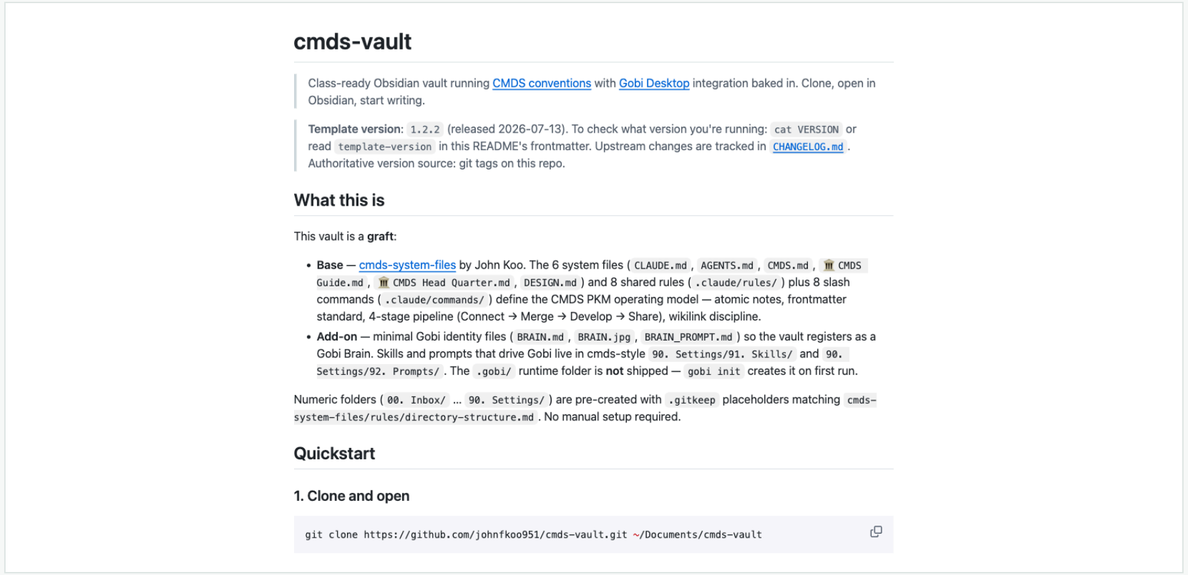
Your questions

I'll take the research questions you left in the chat along with questions about today's talk.

01	**Research questions**	where to start on the map
02	**Today's talk**	wiki, vault, agents
03	**Setup**	cost, security, operating system

Appendix: start with a vault that has operating rules

Start with a vault that has operating rules, not an empty folder. Two are public: one for the main vault, one for the LLM Wiki.



cmds-vault

> Class-ready Obsidian vault running CMDS conventions with Gobi Desktop integration baked in. Clone, open in Obsidian, start writing.

> **Template version**: `1.2.2` (released 2026-07-13). To check what version you're running: `cat VERSION` or read `template-version` in this README's frontmatter. Upstream changes are tracked in `CHANGELOG.md`. Authoritative version source: git tags on this repo.

What this is

This vault is a **graft**:

- **Base** — cmds-system-files by John Koo. The 6 system files (`CLAUDE.md` , `AGENTS.md` , `CMDS.md` , `🏛 CMDS Guide.md` , `🏛 CMDS Head Quarter.md` , `DESIGN.md`) and 8 shared rules (`.claude/rules/`) plus 8 slash commands (`.claude/commands/`) define the CMDS PKM operating model — atomic notes, frontmatter standard, 4-stage pipeline (Connect → Merge → Develop → Share), wikilink discipline.
- **Add-on** — minimal Gobi identity files (`BRAIN.md` , `BRAIN.jpg` , `BRAIN_PROMPT.md`) so the vault registers as a Gobi Brain. Skills and prompts that drive Gobi live in cmds-style `90. Settings/91. Skills/` and `90. Settings/92. Prompts/` . The `.gobi/` runtime folder is **not** shipped — `gobi init` creates it on first run.

Numeric folders (`00. Inbox/` … `90. Settings/`) are pre-created with `.gitkeep` placeholders matching `cmds-system-files/rules/directory-structure.md` . No manual setup required.

Quickstart

1. Clone and open

```
git clone https://github.com/johnfkoo951/cmds-vault.git ~/Documents/cmds-vault
```

CMDS Starter Kit public GitHub page (captured 2026-09-10)

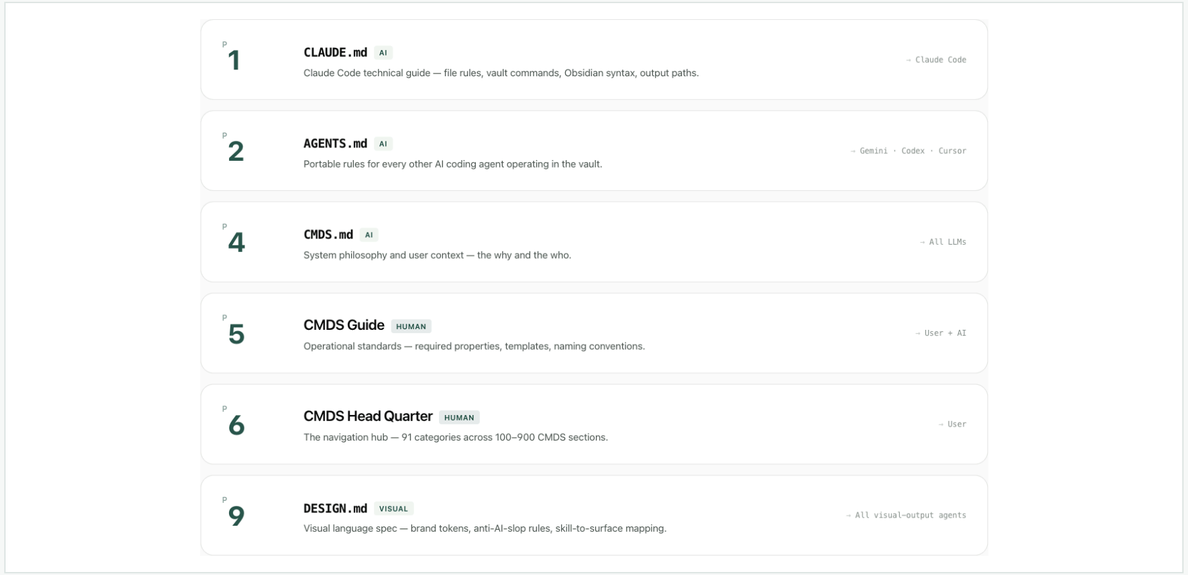
01 CMDS vault main vault folder structure and rules

02 LLM Wiki sources, wiki, and queries layers

03 Adjust change the fields to fit your research

Appendix: the rules files agents read

Operating rules that people and AI follow together, published as separate files: a real example of rules that set wiki quality.



System Files public site (captured 2026-09-10; counts as shown then)

01 Technical rules how agents work

02 Philosophy purpose and context of the system

03 Standards note properties and naming rules

Appendix: AKM Index self-assessment

It checks whether your knowledge system actually works, not how many notes you have, with five pillars and 25 criteria.



AKM Index public site (captured 2026-09-10)

01 Five pillars context, harness, loop, and more

02 Evidence self-report kept apart from real files

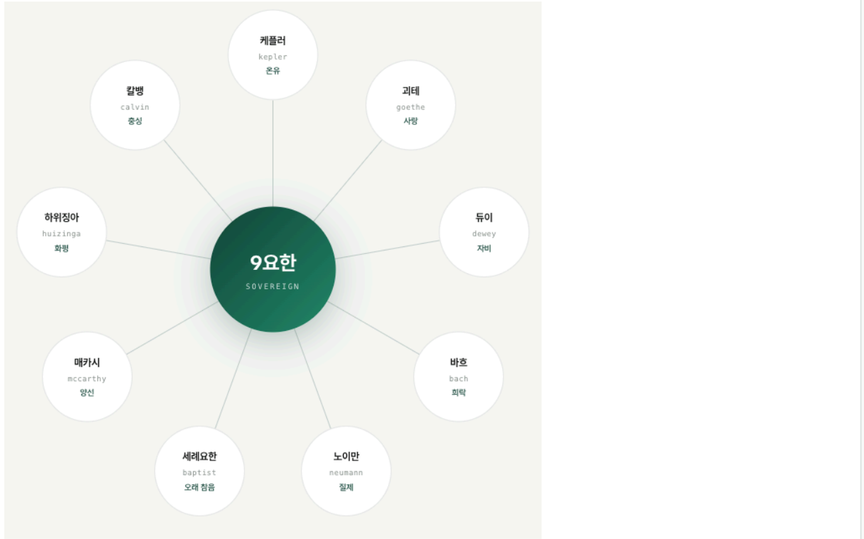
03 Board others' assessment results made public

Reference appendix opened on request / not added to the main session.

AKM Index · Jisan: No measure, no compounding · Jisan: 14th on my own rubric · Jisan: What 77 systems told me

Appendix: agents designed with divided roles

Work is split into nine roles such as research, writing, and teaching, while final judgment and coordination stay in one place.



9Yohan public page (captured 2026-09-10)

01 Roles nine, including research, writing, teaching

02 Coordination split and merged at the center

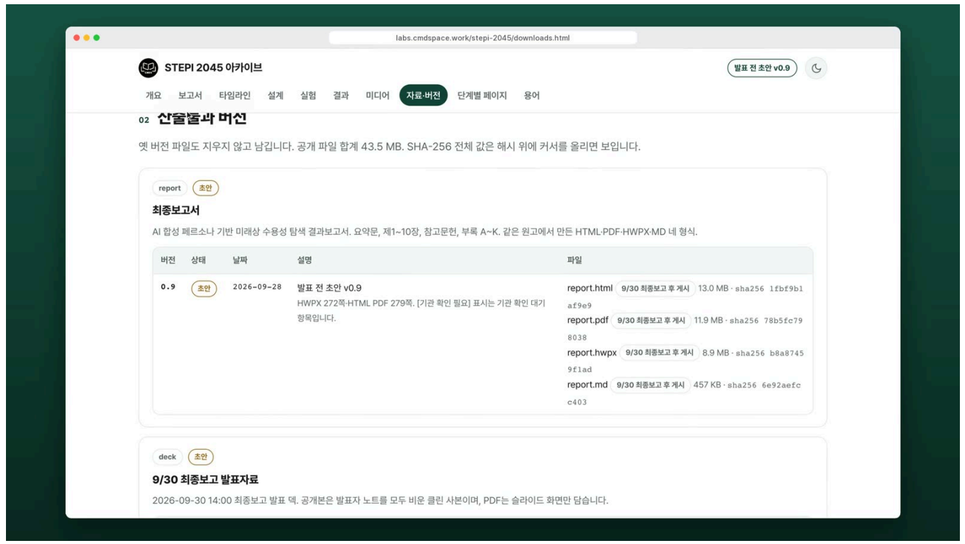
03 Accountability a person gives final approval

Reference appendix opened on request / not added to the main session.

9Yohan Jisan: Nine Yohans

Appendix: the research process as a public archive

Report versions, file fingerprints, and the design process of the STEPI planning study are kept in a public archive.



Deliverable versions in the STEPI 2045 public archive

01 Versions files and fingerprints for each report

02 Process design and experiment stages in order

03 Principle versions are kept, never deleted

Appendix: a question desk that answers from public material

Ask a question and it points you to published material. If nothing fits, it doesn't make things up; it hands off to a person.



Real Ask CMDSPACE answer (captured 2026-10-02; wording varies each time)

01 Select the right item from the public list

02 Answer only from the selected material

03 Hand off if none fits, pass it to Yohan Koo

Appendix: cost, security, setup

Short answers to the three questions that come up most in Q&A.

01

Cost

paid AI subscription;
Obsidian and Zotero are free

02

Security

files stay on your computer;
your institution's rules come first

03

Setup

macOS and Windows,
some terminal use

Appendix: skills, agents, harness

What we call an agent has its own context, plans, uses tools and runs in a loop.

01	**Skill**	a procedure package that does not run on its own
02	**Specialist agent**	runs its task in its own context, returns results
03	**Conductor agent**	plans, splits and merges
04	**Harness**	the runtime for loop, tools, context and permissions

Appendix: schema and harness ledger

The schema is the rules agents follow; the harness is where they run. Measured on 15 October 2026.

01	**Schema**	9 system files, 9 shared rules, 7 required properties
02	**Harness**	Claude Code and Codex, two Macs, 8 work rigs
03	**Agents**	9 9Yohan specialists plus a conductor, 10 rig seat roles
04	**Automation**	32 commands, 6 hooks, 16 scheduled jobs

Thank you

Public resources on today's structure and what comes next are at the links below.



Course materials

labs.cmdspace.work/
fc-seminar-1015/

CMDSPACE

cmdspace.work

About the speaker

bio.cmdspace.work