cognee

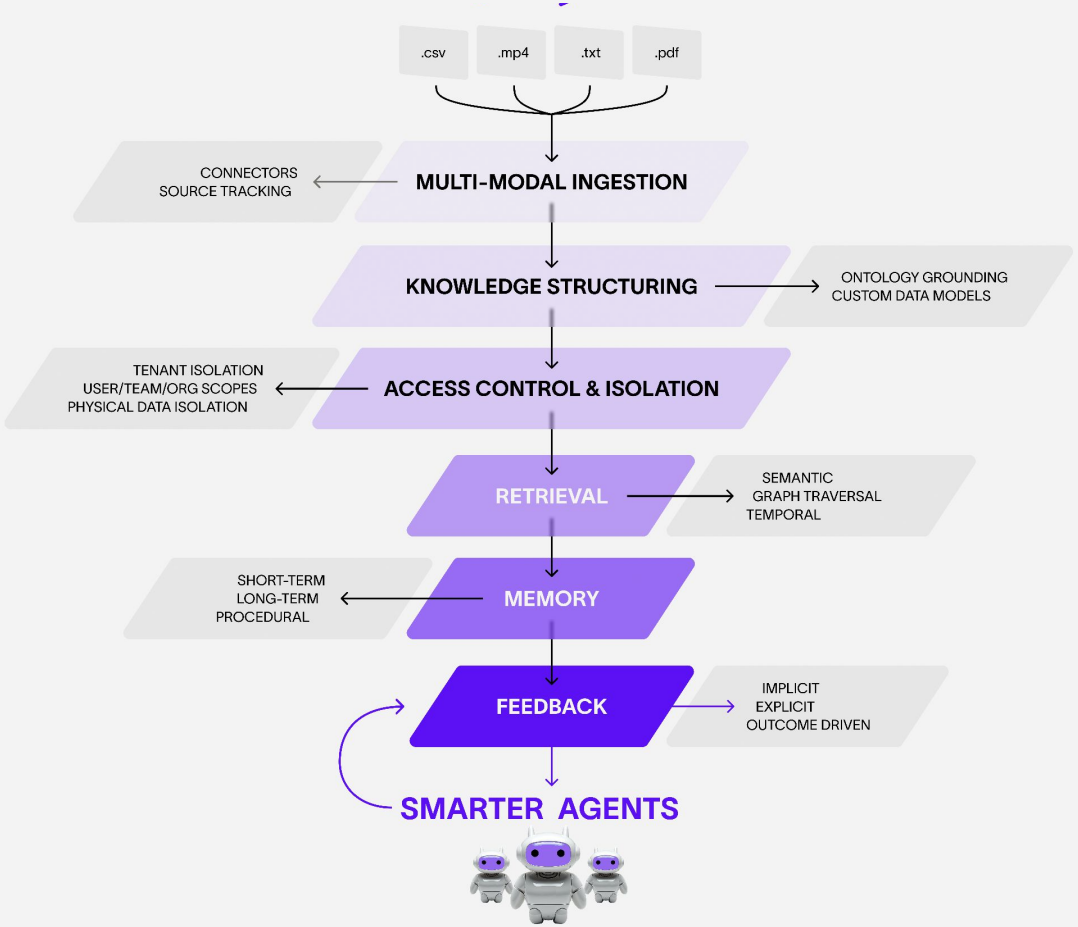


BRAIN BEHIND
YOUR AGENTS

NY · 2026

cognee

From Data Ingestion To Smarter Agents With Cognee



cognee

TRUSTED BY ENGINEERS IN

Software

splunk> · Redis · aws · Infosys · botpress · studocu · kovai.co

Blended · ATLASSIAN · AUTODESK · ribbon · REPLY · rentpost

eCommerce

Alibaba

Hardware

intel

Finance

QRT · DBS · Royal Bank of Canada · ALTUM

Energy

raizen · PEERS

Academia

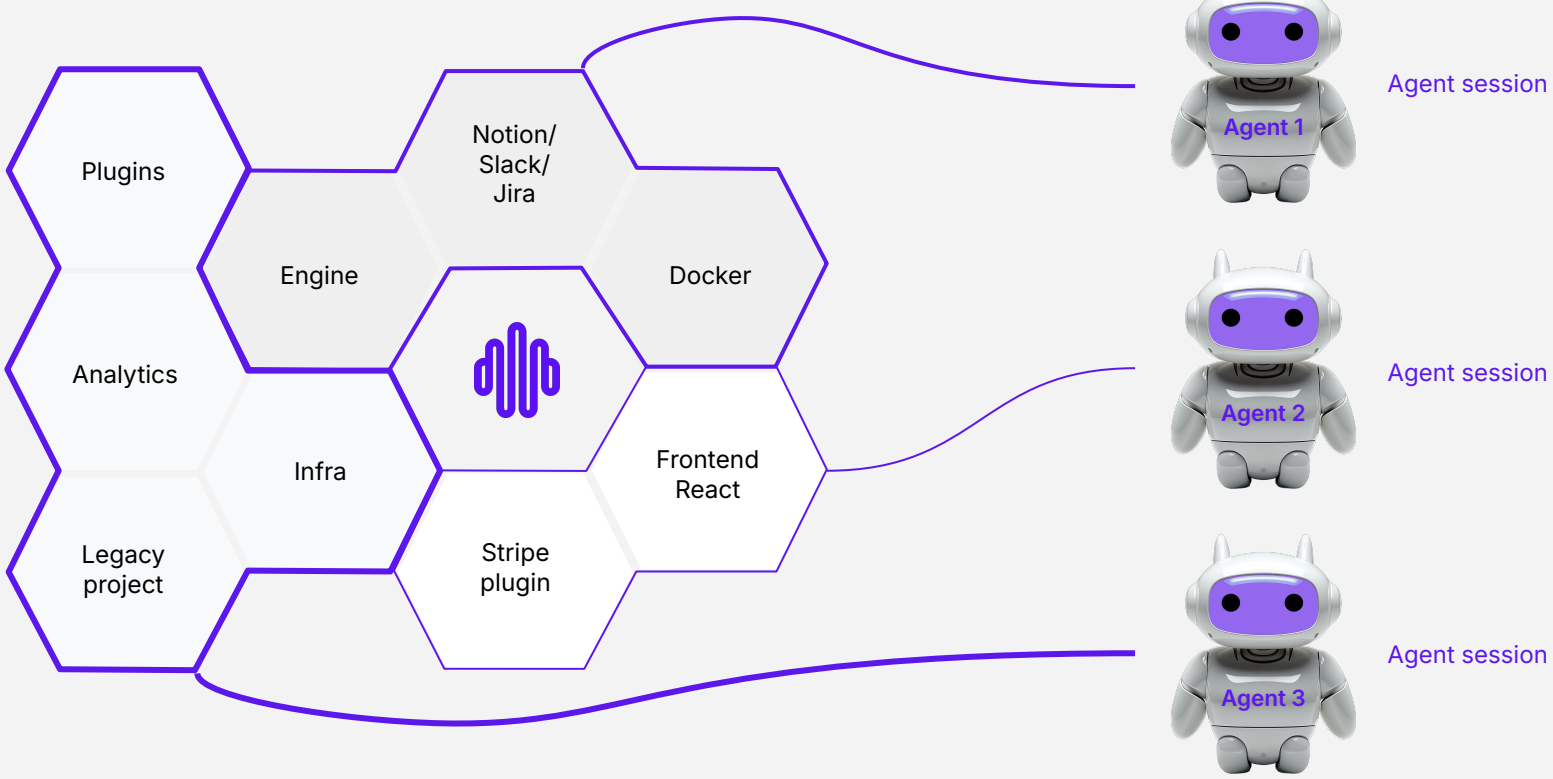
TECHNISCHE UNIVERSITÄT BERLIN · Berkeley UNIVERSITY OF CALIFORNIA · GESIS Leibniz-Institut für Sozialwissenschaften · BROWN UNIVERSITY · PUC RIO

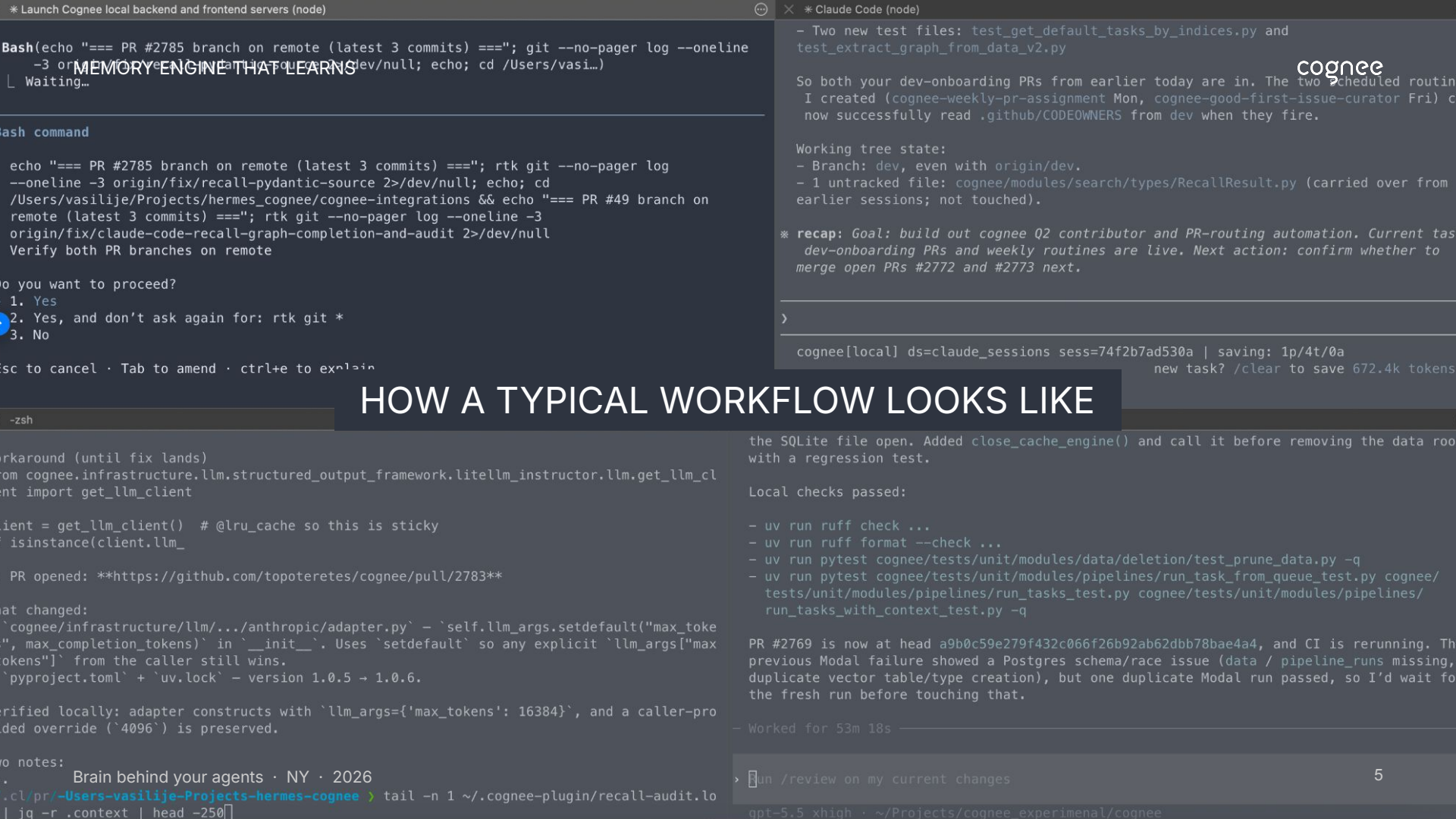
+80 other companies and thousands of open source developers

★ **16.8k** GitHub stars ➢ **5M+** SDK runs/month

cognee

TYPICAL WORKFLOW — MANY REPOSITORIES AROUND THE ORGANIZATION




✳ Launch Cognee local backend and frontend servers (node)
Bash(echo "=== PR #2785 branch on remote (latest 3 commits) ===" ; git --no-pager log --oneline
-3 origin/fix/recall-pydantic-source 2>/dev/null; echo; cd /Users/vasi…)
Waiting…
MEMORY ENGINE THAT LEARNS
Bash command
echo "=== PR #2785 branch on remote (latest 3 commits) ===" ; rtk git --no-pager log
--oneline -3 origin/fix/recall-pydantic-source 2>/dev/null; echo; cd
/Users/vasilije/Projects/hermes_cognee/cognee-integrations && echo "=== PR #49 branch on
remote (latest 3 commits) ===" ; rtk git --no-pager log --oneline -3
origin/fix/claude-code-recall-graph-completion-and-audit 2>/dev/null
Verify both PR branches on remote
Do you want to proceed?
1. Yes
2. Yes, and don't ask again for: rtk git *
3. No
Esc to cancel · Tab to amend · ctrl+e to explain
✳ Claude Code (node)
- Two new test files: test_get_default_tasks_by_indices.py and
test_extract_graph_from_data_v2.py
cognee
So both your dev-onboarding PRs from earlier today are in. The two scheduled routin
I created (cognee-weekly-pr-assignment Mon, cognee-good-first-issue-curator Fri) c
now successfully read .github/CODEOWNERS from dev when they fire.
Working tree state:
- Branch: dev, even with origin/dev.
- 1 untracked file: cognee/modules/search/types/RecallResult.py (carried over from
earlier sessions; not touched).
※ recap: Goal: build out cognee Q2 contributor and PR-routing automation. Current tas
dev-onboarding PRs and weekly routines are live. Next action: confirm whether to
merge open PRs #2772 and #2773 next.
❯
cognee[local] ds=claude_sessions sess=74f2b7ad530a | saving: 1p/4t/0a
new task? /clear to save 672.4k tokens
HOW A TYPICAL WORKFLOW LOOKS LIKE
-zsh
orkaround (until fix lands)
rom cognee.infrastructure.llm.structured_output_framework.litellm_instructor.llm.get_llm_cl
ent import get_llm_client
lient = get_llm_client() # @lru_cache so this is sticky
isinstance(client.llm_
PR opened: **https://github.com/topoteretes/cognee/pull/2783**
hat changed:
`cognee/infrastructure/llm/.../anthropic/adapter.py` — `self.llm_args.setdefault("max_toke
", max_completion_tokens)` in `__init__`. Uses `setdefault` so any explicit `llm_args["max
okens"]` from the caller still wins.
`pyproject.toml` + `uv.lock` — version 1.0.5 → 1.0.6.
erified locally: adapter constructs with `llm_args={'max_tokens': 16384}`, and a caller-pro
ded override (`4096`) is preserved.
wo notes:
.cl/pr/-Users-vasilije-Projects-hermes-cognee ❯ tail -n 1 ~/.cognee-plugin/recall-audit.lo
| jq -r .context | head -250
the SQLite file open. Added close_cache_engine() and call it before removing the data roo
with a regression test.
Local checks passed:
- uv run ruff check ...
- uv run ruff format --check ...
- uv run pytest cognee/tests/unit/modules/data/deletion/test_prune_data.py -q
- uv run pytest cognee/tests/unit/modules/pipelines/run_task_from_queue_test.py cognee/
tests/unit/modules/pipelines/run_tasks_test.py cognee/tests/unit/modules/pipelines/
run_tasks_with_context_test.py -q
PR #2769 is now at head a9b0c59e279f432c066f26b92ab62dbb78bae4a4, and CI is rerunning. Th
previous Modal failure showed a Postgres schema/race issue (data / pipeline_runs missing,
duplicate vector table/type creation), but one duplicate Modal run passed, so I'd wait fo
the fresh run before touching that.
Worked for 53m 18s
› Run /review on my current changes
gpt-5.5 xhigh · ~/Projects/cognee experimenal/cognee
Brain behind your agents · NY · 2026
5

cognee

Hooks are shell commands Claude Code runs at *fixed* *lifecycle moments, memory, observability, or any cross-session behavior.*

IT GETS

The triggering event payload on stdin (JSON)

IT GETS

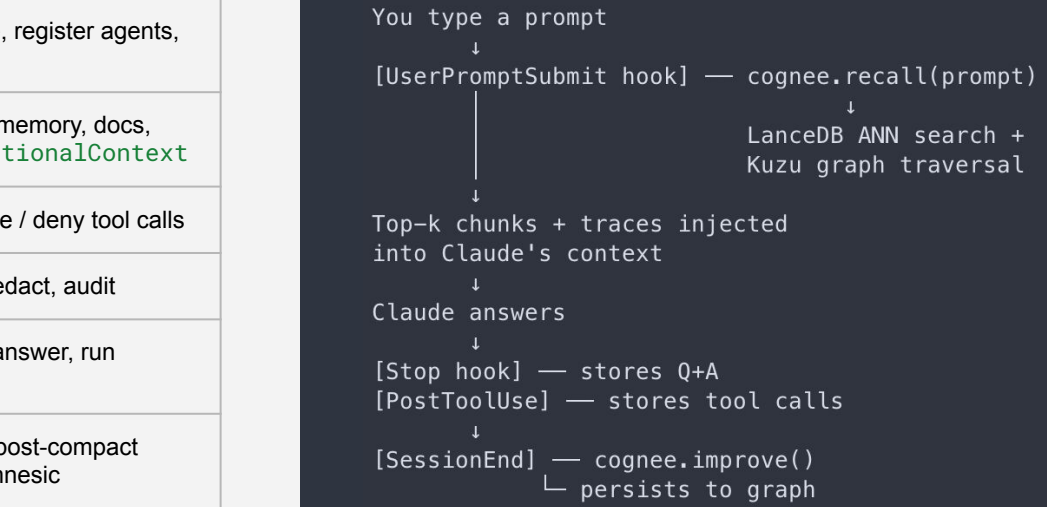
An environment including and the cwd

IT GETS

Do its work and return — with a JSON response on stdout

SOLUTION — HOOKS

EVENT	WHEN IT FIRES	TYPICAL USE
SessionStart	Once when Claude opens a session	Bootstrap state, register agents, print a banner
UserPromptSubmit	Every user prompt, before model call	Inject context (memory, docs, RAG) via `additionalContext`
PreToolUse	Before any tool runs	Guard / approve / deny tool calls
PostToolUse	After a tool returns	Capture, log, redact, audit
Stop	After the assistant finishes responding	Save the final answer, run analytics
PreCompact	Before context-window compaction	Distill state so post-compact Claude isn't amnesic
SessionEnd	When the session closes	Flush, persist, summarize



cognee

Append only →
How do you control and version edits? What about conflicting instructions?

⚠
Doesn’t scale — files are files.

⚠
Doesn’t contain traces and sessions — you need to distill data manually

cognee

Databases as files instead of files

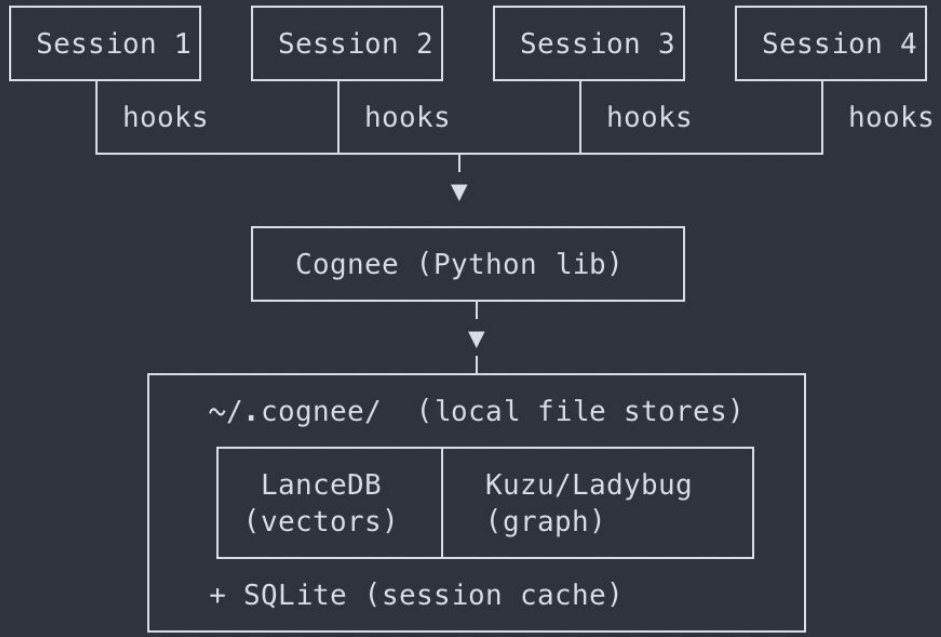
	WHAT IT IS	WHAT IT STORES
LanceDB	Local file-based vector DB	Embeddings of every chunk, every Q&A, every tool trace
kùzu	Local file-based graph DB	Entities, relationships, session knowledge
Hooks	6 Claude Code lifecycle events	Capture turns automatically, inject context on each prompt

cognee

Storing sessions, traces and memory in file stores *enables many deployment patterns*

	SHIP	SCALE	DEPLOY
Desktop	Nothing	One user, every session	Build/test on push; optional release packaging
Lambda	DB-as-Lambda-layer or EFS	Request volume	CI package layer; deploy Lambda; update EFS artifact
Sandbox	Tarball in S3	Concurrent agents, ephemeral lifetimes	Build tarball; upload to S3; tag/version artifact
Sidecar	Container + volume	Mixed-language agent fleets	Build/push container; run integration tests; deploy manifests
Replicas	Snapshot pipeline	Reads to thousands of consumers	Generate snapshots; validate; publish; roll forward/rollback

cognee

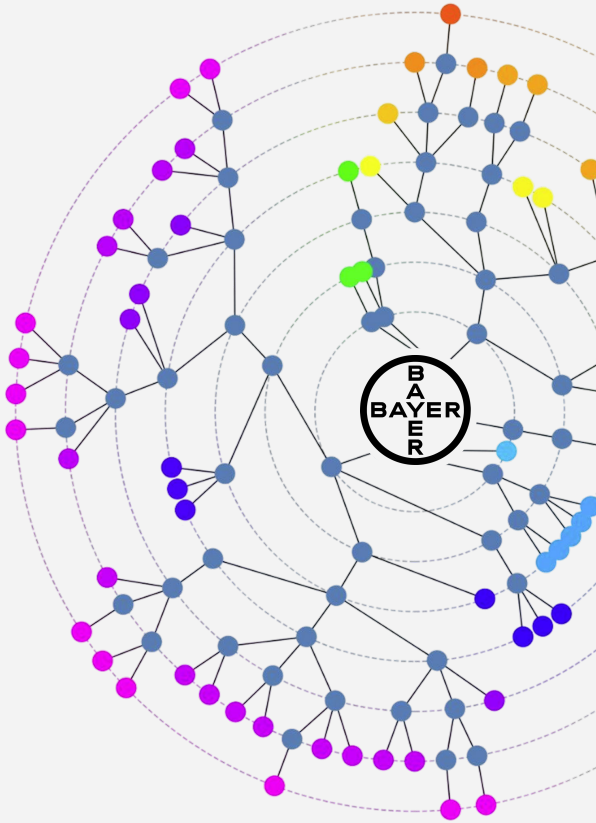

Session 1
Session 2
Session 3
Session 4
hooks
hooks
hooks
hooks
Cognee (Python lib)
~/.cognee/ (local file stores)
LanceDB
(vectors)
Kuzu/Ladybug
(graph)
+ SQLite (session cache)

cognee

BAYER · PHARMACEUTICAL R&D

Agentic research system for hypothesis generation in drug discovery

Global life science company Bayer carried out an extensive R&D project together with Cognee, which utilized graph-vector reasoning to generate, rank and explain novel research hypotheses inside Bayer's internal AI Scientist platform.



cognee

vasilije@cognee.ai



Vasilije Markovic
CEO