MONADICAL

# Building AI-Powered Operational Assistants

## From Simple Automation to Strategic Implementation

# 👋 Hi there.

- Cofounder at Monadical (AI consultancy)
- Cofounder at DevCap (AI venture fund)
- Former professional poker player

MONADICAL

# This is a Series!

**Operational Assistants:**

Augment your team's capabilities

**Knowledge Systems:**

Transform how you manage information

**Customer Experience:**

Enhance service while maintaining control

**Analytics:**

Turn your data into actionable insights

MONADICAL

- Current state of AI - what's real, what's hype
- What operational assistants are
- Where they are useful
- Some helpful mental models for identifying opportunities
- A couple demos
- How to get started

MONADICAL

# Today in one slide:

- While “agents” in the literal sense are still not here, we are able to create “AI operational assistants”, which can augment your team in exciting ways.
- We’ve been seeing a ton of interest with our clients in this use case, and hope to demonstrate what’s possible.
- Fundamental to our approach is the belief that anything process-critical needs to be self hosted. We’ll discuss how to actually do that.

MONADICAL

# There are levels to “agency.”

| Agency Level | Description | How that’s called |
|---|---|---|
| ☆☆☆ | LLM output has no impact on program flow | Simple Processor |
| ★☆☆ | LLM output determines an if/else switch | Router |
| ★★☆ | LLM output determines function execution | Tool Caller |
| ★★★ | LLM output controls iteration and program continuation | Multi-step Agent |
| ★★★ | One agentic workflow can start another agentic workflow | Multi-Agent |

🤗 **Hugging Face**

MONADICAL

# There are 2 “classes” of LLMs that matter.

**Traditional RLHF-Tuned Chat Models:**

ChatGPT
Claude
Gemini
MISTRAL AI_
deepseek V3

**Chain-of-Thought Models**

OpenAI o1
deepseek R1

MONADICAL

# Chat models are seeing diminishing ROI in compute. But CoT models seem to be at the beginning of their curves.

**François Chollet**
@fchollet

People scaled LLMs by ~10,000x from 2019 to 2024, and their scores on ARC stayed near 0 (e.g. GPT-4o at ~5%). Meanwhile a very crude program search approach could score >20% with hardly any compute.

Then OpenAI started adding test-time CoT search. ARC scores immediately shot up.

MONADICAL

# The AI agent hype:

**sophia dew** @sodofi_ · Jan 27
**ai agents** will singlehandedly transform how people interact onchain

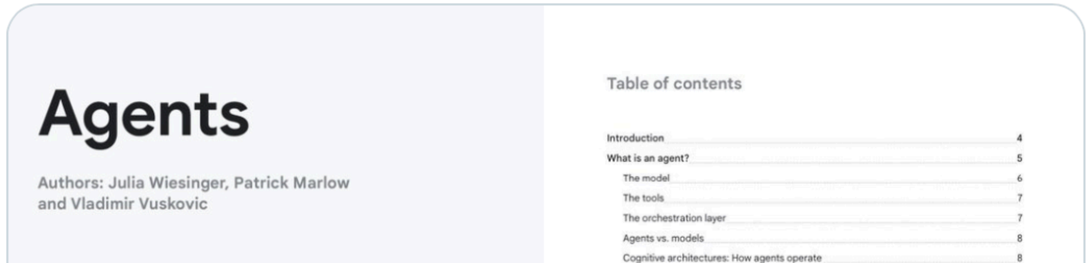
**Tomasz Tunguz** @ttunguz · Dec 2, 2024
In the bustling tech campuses of 2024, the age of passive **AI** – systems that merely respond to our queries – is giving way to something far more profound: the era of **AI agents**.

**Aadit Sheth** @aaditsh · Jan 21
ChatGPT was just step 1.

Google dropped a whitepaper on the next evolution of **AI**: **Agents**.

Now, everyone—from Satya Nadella to Jensen Huang—is talking about them.

Here's why **AI agents** are the next big thing. Let's break it down. 🧵


Agents

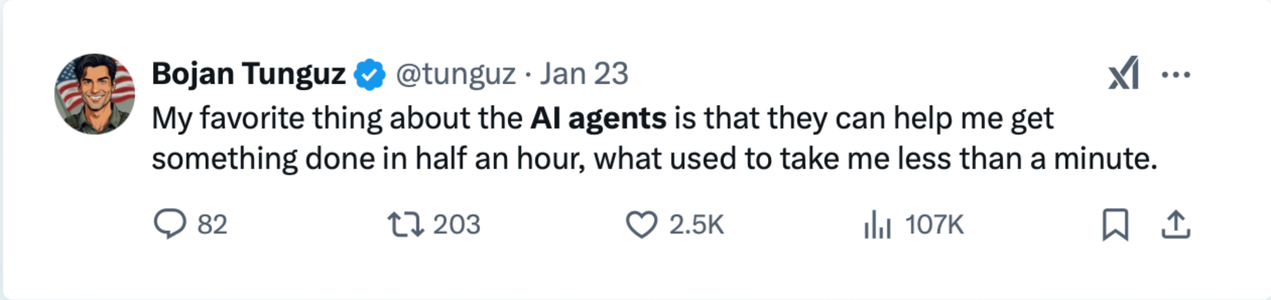
Authors: Julia Wiesinger, Patrick Marlow and Vladimir Vuskovic

Table of contents



MONADICAL

# The reality:




MONADICAL

# The *actual* reality:

**Igor_Katsai** Nov 2024

I am working on a simple workflow involving an AI agent that interacts with a PostgreSQL database to retrieve answers. The workflow operates as follows:

1. The AI agent receives a question from the chat.
2. It generates a query using the `{{ $fromAI("query") }}` template and sends it to the PostgreSQL tool.
3. The PostgreSQL tool executes the query and returns the result.

### Observed Issue

- During the **first iteration**, everything works as expected:
  - The AI generates an appropriate query.
  - The PostgreSQL tool executes it correctly.
  - The answer is returned without any issues.
- However, in many cases, the **initial answer is insufficient**, and the AI agent tries to refine its query and send additional iterations. Here's where the issue arises:
  - The PostgreSQL tool appears to **receive multiple inputs** with different queries from the AI.
  - Despite receiving these inputs, the tool **executes only the initial query repeatedly**, returning the same answer every time.
  - The process continues until the maximum number of iterations is reached, after which the workflow shuts down.

MONADICAL

# What is an operational assistant?

**AI extracts information** from one or more data sources

**AI processes that information** and prepares some kind of structured output

That output is **used by a human** being to complete a task more efficiently

MONADICAL

# Some examples to jog your intuition:

🏦 Loan application processing

🛫 Travel and expense reporting

💊 Clinical trial reporting on drug efficacy

📗 RFP response generation

🏥 Walk-in clinic assistant that summarizes patient history, family information, etc

MONADICAL

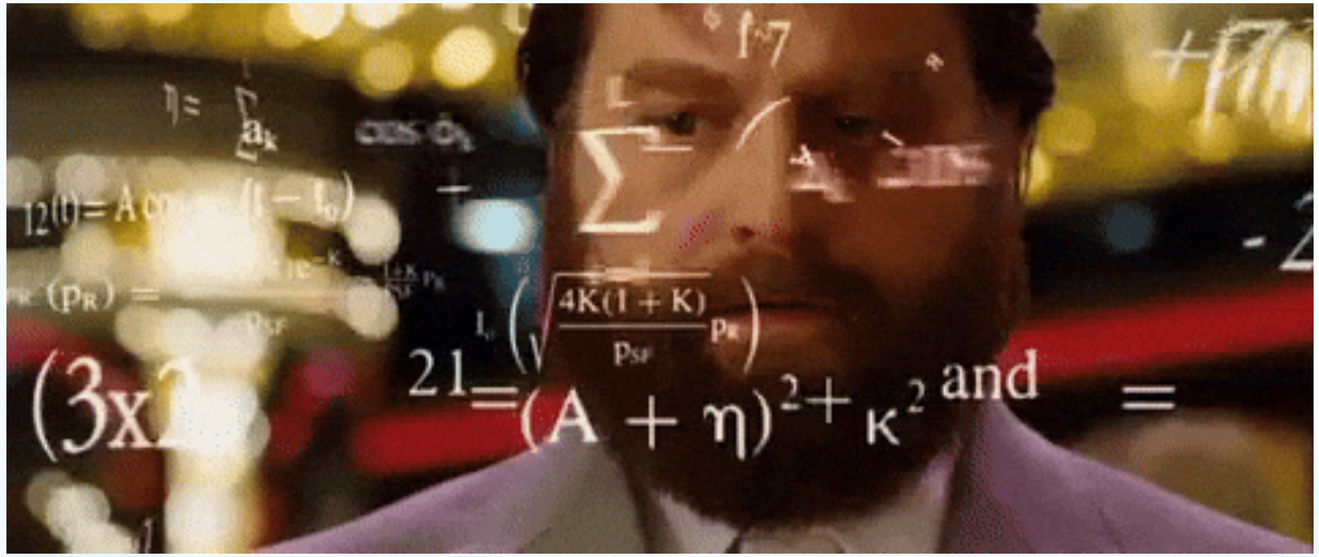
# Tasks that are well-suited for operational assistants:

- ✅ Data-intensive tasks
- ✅ Repetitive tasks
- ✅ Tasks where a “rough draft” output is a significant accelerator
- ✅ Tasks where iterative improvement is possible

MONADICAL

# Data-Intensive Tasks

The task heavily relies on gathering, processing, and synthesizing information from various sources.



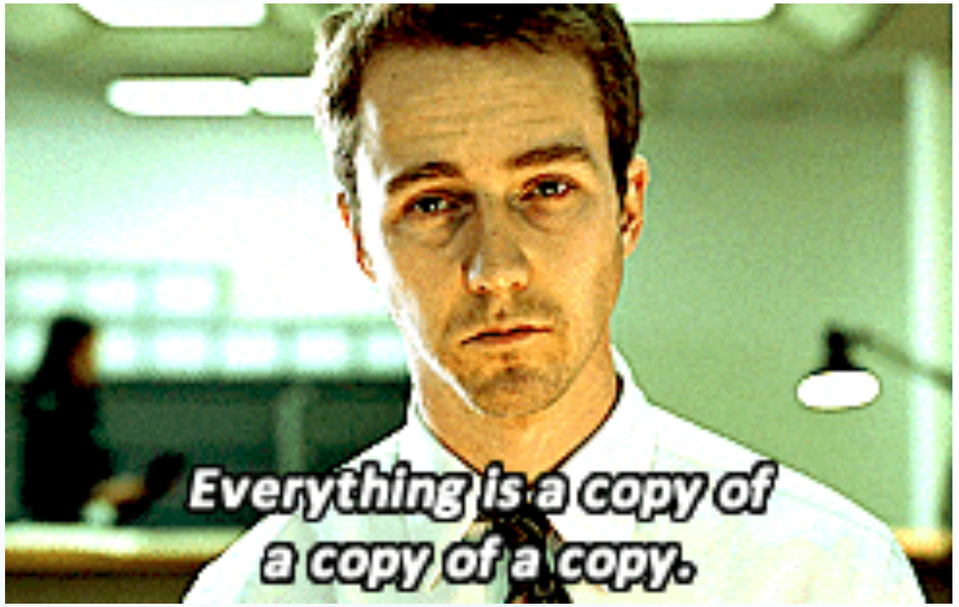

**Intuition**: AI excels at handling and processing large datasets.

MONADICAL

# Repetitive Tasks

The workflow can be, at least partially, defined by rules, or learned from data patterns, rather than requiring pure creativity or complex judgment calls beyond what current AI can handle.

**Intuition**: AI excels at learning rules from examples and applying them consistently.

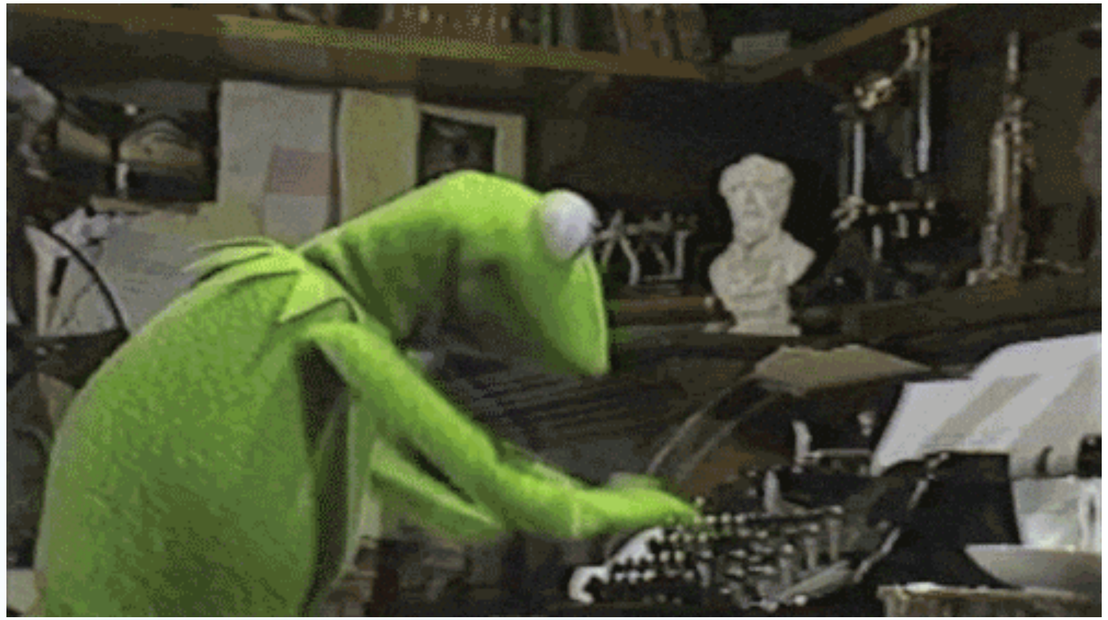



MONADICAL

# Tasks where a “rough draft” output is a significant accelerator

Even an imperfect output significantly speeds up the human’s workflow by providing a starting point, automating tedious steps, or surfacing key information.

**Intuition**: AI will make mistakes, but it can often produce a “good enough” version of an output much faster than a person can.



MONADICAL

# Tasks where iterative improvement is possible

The task allows for an iterative approach to building the AI assistant, where an initial “rough draft” capability can be progressively refined over time through feedback and further training.

**Intuition**: The operational assistant model lends itself to the generation of proprietary training data for the iterative improvement of a larger model.


MONADICAL

# It's a Great Time For Self Hosted

Proprietary made sense because it was so much cheaper than building on top of open source platforms. But  you can now build applications on top of open source tools, **tailored to your unique needs**, quickly and cheaply. And **stay in control of your data** in the process.

### Proprietary LLMs

ChatGPT  Claude  Gemini

### Open Source / Self-hosted LLMs

Llama 3  MISTRAL AI_ 7  deepseek

### Proprietary Applications

GitHub Copilot  CANVAS

### Open Source / Self-hosted Applications

LibreChat

MONADICAL

# Why self-host?

In the age of AI, **your data is an increasingly valuable asset**. When your team uses 3rd party APIs, there is no practical way to enforce how your data is being used. You want proprietary information shared and leveraged inside a system you control.


MONADICAL

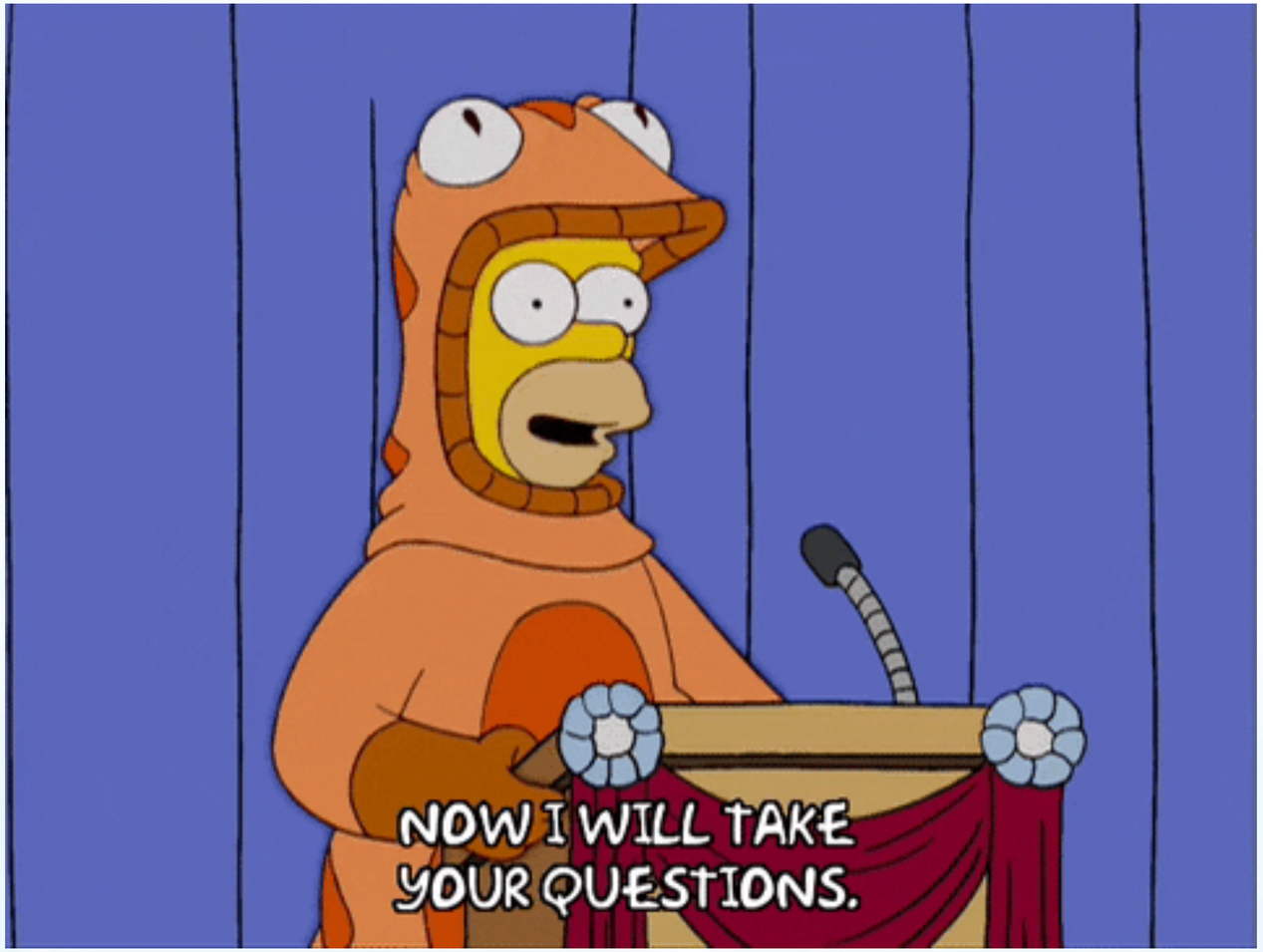
# Some demos!


MONADICAL

# How to get started:

- ✅ Identify potential use-cases
- ✅ Set up data pipeline & infrastructure
- ✅ Build proof-of-concept using open-source framework
- ✅ Iterate
- 👋 Lacking the in-house resources? We can help with all of this.

MONADICAL




MONADICAL

MONADICAL

# Let's **transform** your organization.

✉️ More AI questions? Email me at max@monadical.com

👍 Schedule a strategy call here: cal.com/monadical