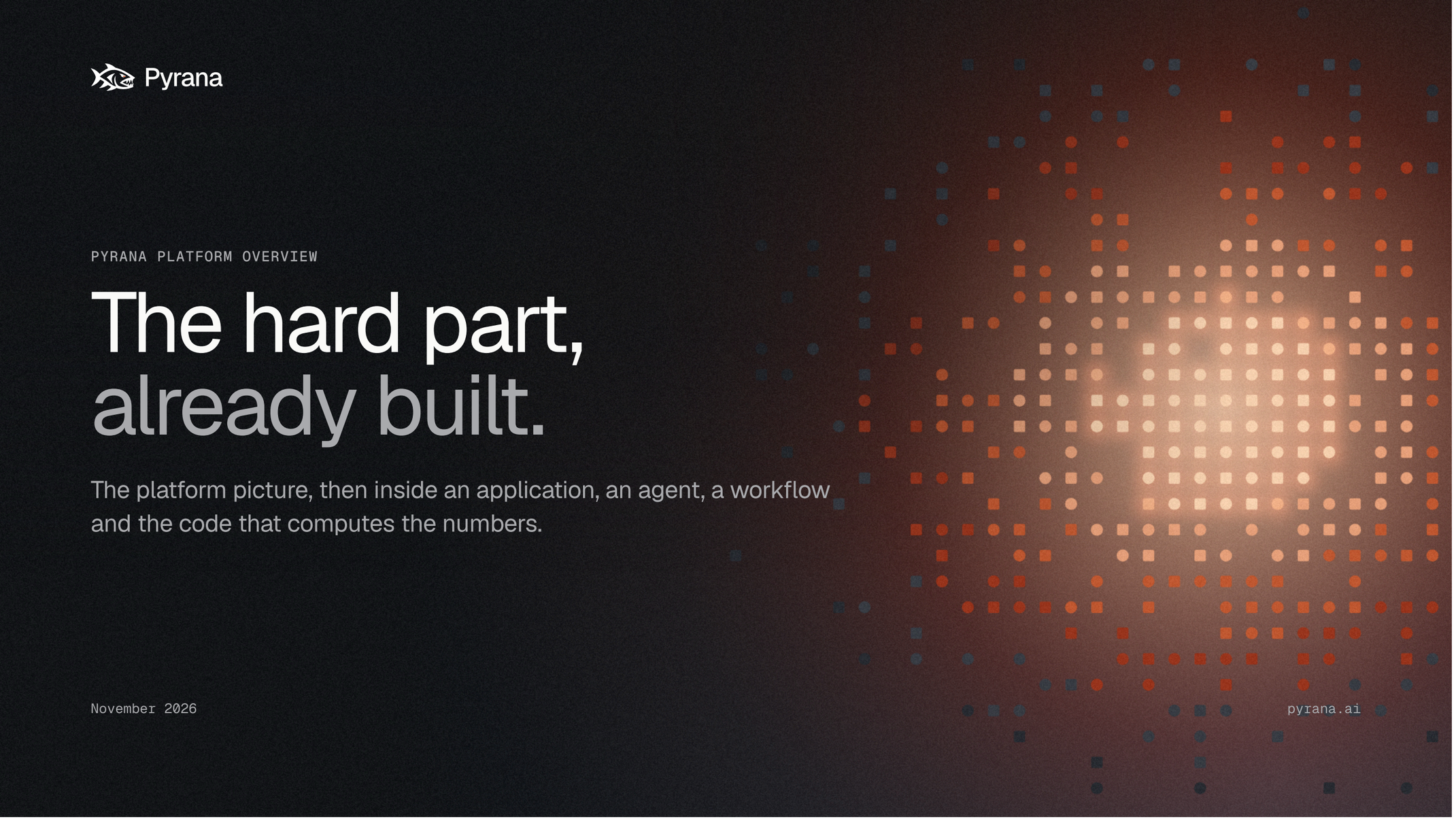
Pyrana

PYRANA PLATFORM OVERVIEW

The hard part, already built.

The platform picture, then inside an application, an agent, a workflow and the code that computes the numbers.



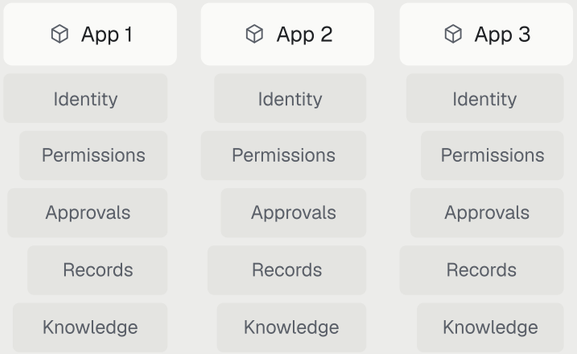
November 2026

pyrana.ai

THE ANSWER

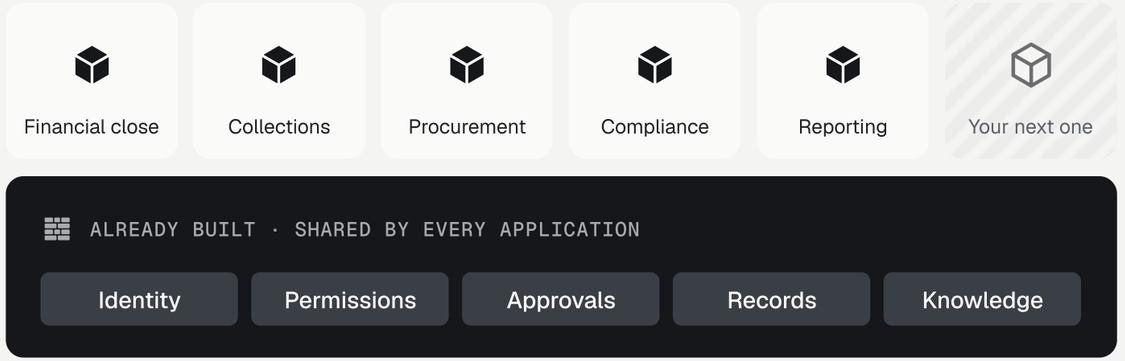
Your teams should build business applications, not AI infrastructure. The foundation is the hard part, and it is already done.

BUILDING IT YOURSELF



Every application rebuilds the foundation, each one a little differently.

WITH PYRANA

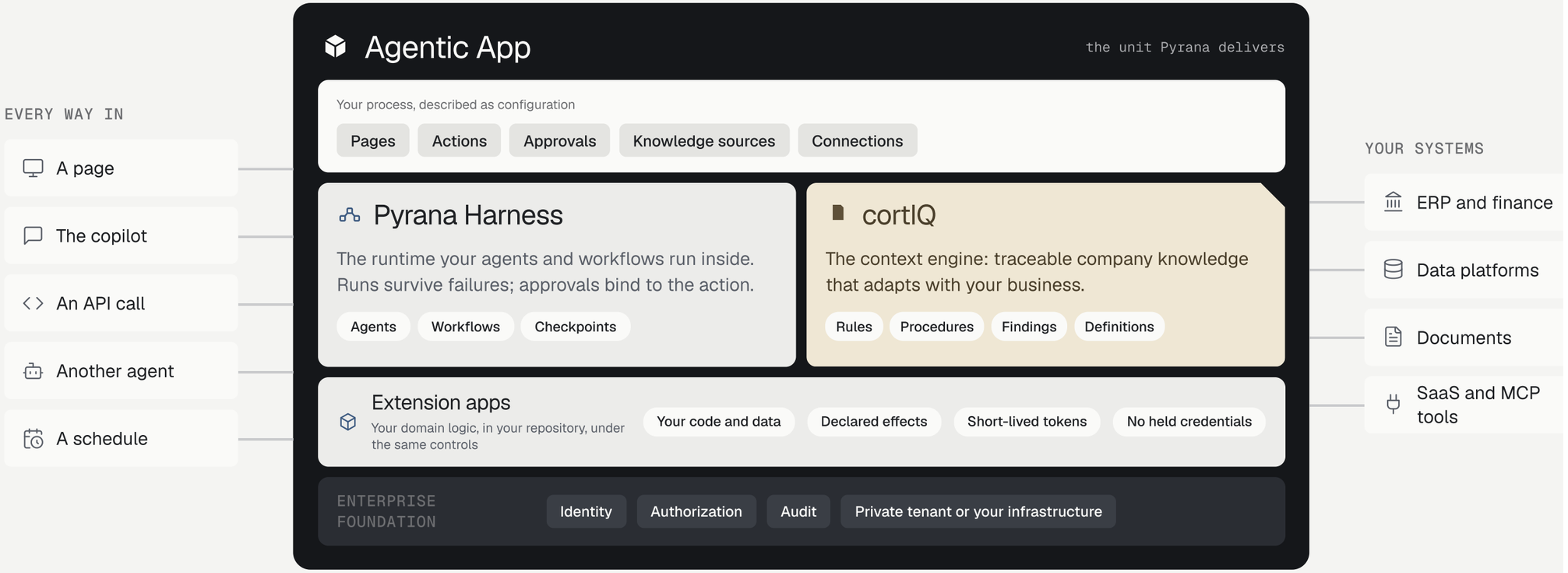


Each application is configuration on a foundation that already exists. None of them starts from zero.

ARCHITECTURE

Production-grade AI-native apps built for your business.

Pyrana delivers interfaces, workflows, agents, context, storage, and controls via configuration, not code.



Already built in every application: one identity, one authorization model, one record, one knowledge layer.

Already built. Configured per application.

Agentic App the unit Pyrana delivers

Your process, as configuration: Pages · Actions · Approvals · Knowledge sources · Connections

Every way in

- A page
- The copilot
- An API call
- Another agent
- A schedule
- An event

People

- Business users use
- Builders teach
- Operators approve
- Other assistants act as the user

Automation

- Scheduled daily, weekly, custom
- API-driven other systems call
- Background no prompt needed

5 Applications

- **Pages and copilot** — one unit a business owns
- **Roles and scopes** — each person sees their part

Inside view — Open →

6 Application state

- **Plans and decisions** — where no system has a home
- **Extensions** — add what the business needs

12 Push, activate, deploy

- **Versioned releases** — the owner activates each

Pyrana Harness · the runtime your agents and workflows run inside

3 Agents

- **Specialists** — own tools and skills
- **Finish checks** — done means verified

Inside view — Open →

4 Workflows

- **Hierarchies** — level by level, to the top
- **Durable runs** — wait, resume, retry

Inside view — Open →

8 Code

- **Tested code** — named data calls, scripts
- **Numbers by code** — agents write words

Inside view — Open →

1 cortIQ context engine

- Rules
- Procedures
- Findings
- Definitions

Sourced — every statement cited
Connected — related knowledge linked
Versioned — history kept
Governed — who may change it

Change the knowledge and the results change. No code change.

2 Connections

- **APIs** — any REST or HTTP service
- **Databases** — SQL and ODBC sources

9 Models and hosting

- **Right model per agent** — from approved providers
- **Retries and failover** — runs keep going

10 Observability

- **Every run and tool call** — step by step
- **Cost and lineage** — of every result

11 Build

- **By conversation** — copilot designs and tests
- **In code** — SDKs and MCP

7 Enterprise foundation: security and governance around every block

- **Company sign-in** — your single sign-on
- **Fine-grained access** — per App, entity, role
- **Approval by a person** — policy decides when
- **Audit trail** — who, what, when, why
- **Tenant isolation** — each client separate

Your systems of record

- ERP
- HR
- CRM
- Commerce
- Inventory
- Documents
- Risk
- Tickets
- Data lake
- SaaS

They stay the record. Pyrana works on top, reads and writes back.

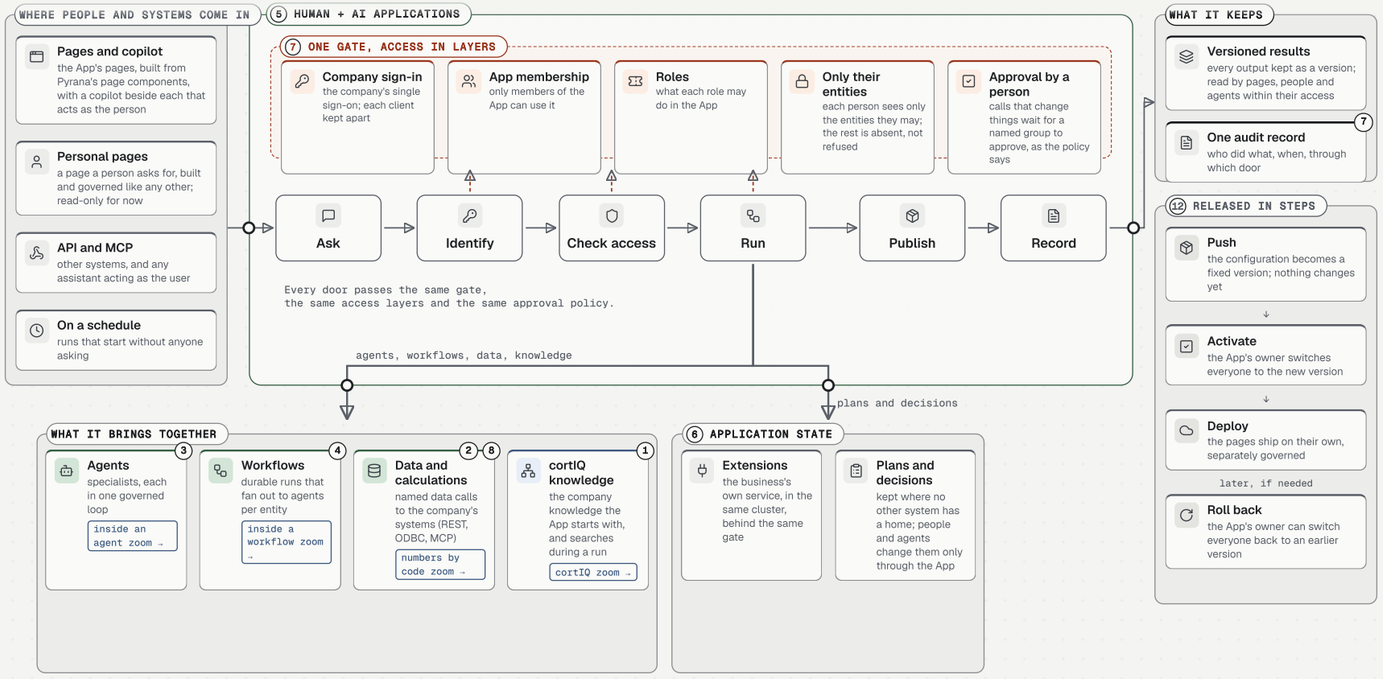
BLOCK 5 · INSIDE AN APP

One configuration. Everything the platform enforces.

5 INSIDE THE BLOCK

Apps: full-stack applications, not just agents.

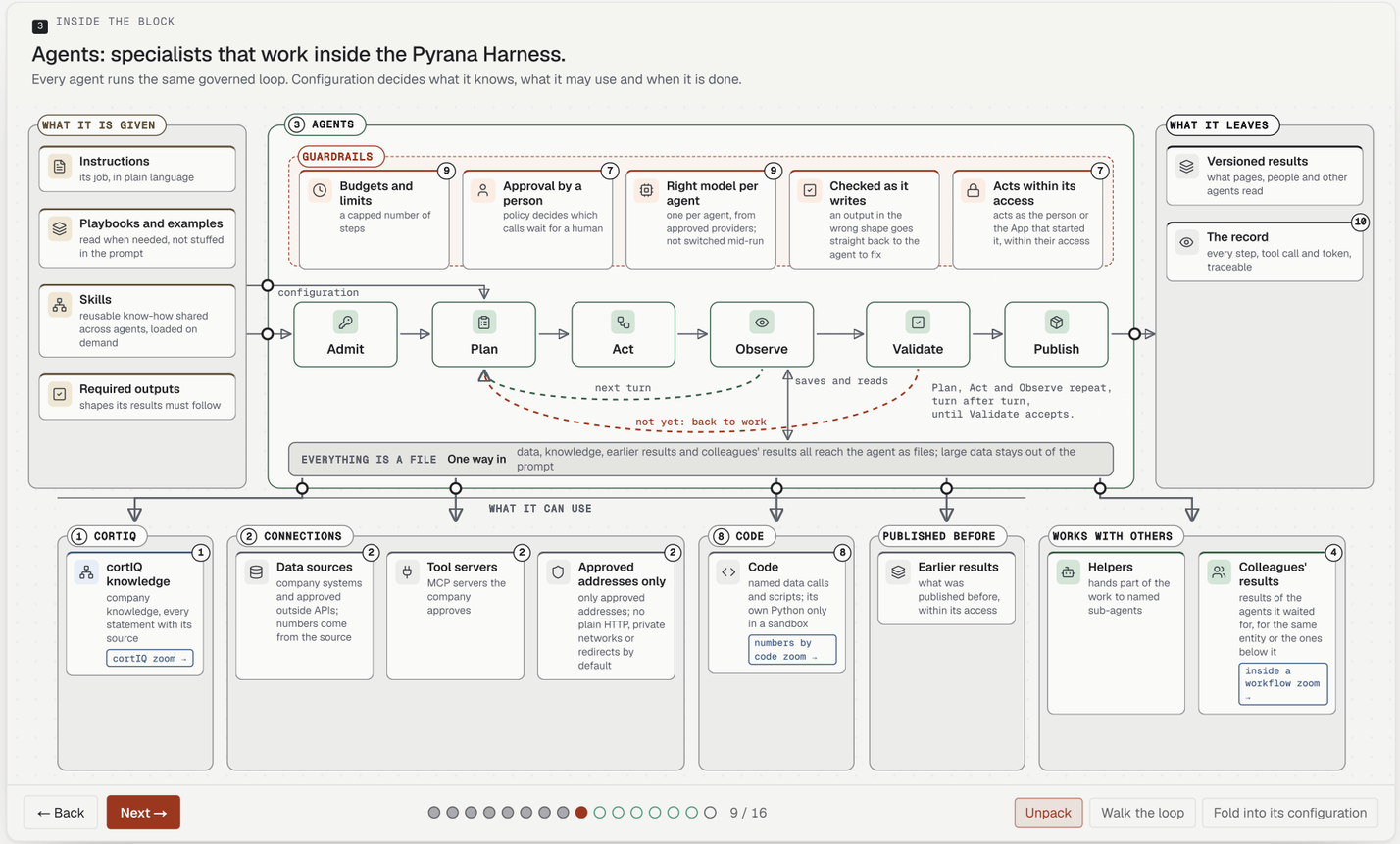
Pyrana runs whole business applications, not just agents: the pages people work in, who they are, what they may see and do, and the agents, data and results behind them. People and agents work over the same pages, data and results, through one gate.



← Back | Next → | 9 / 16 | Unpack | Follow a request | Fold into its configuration

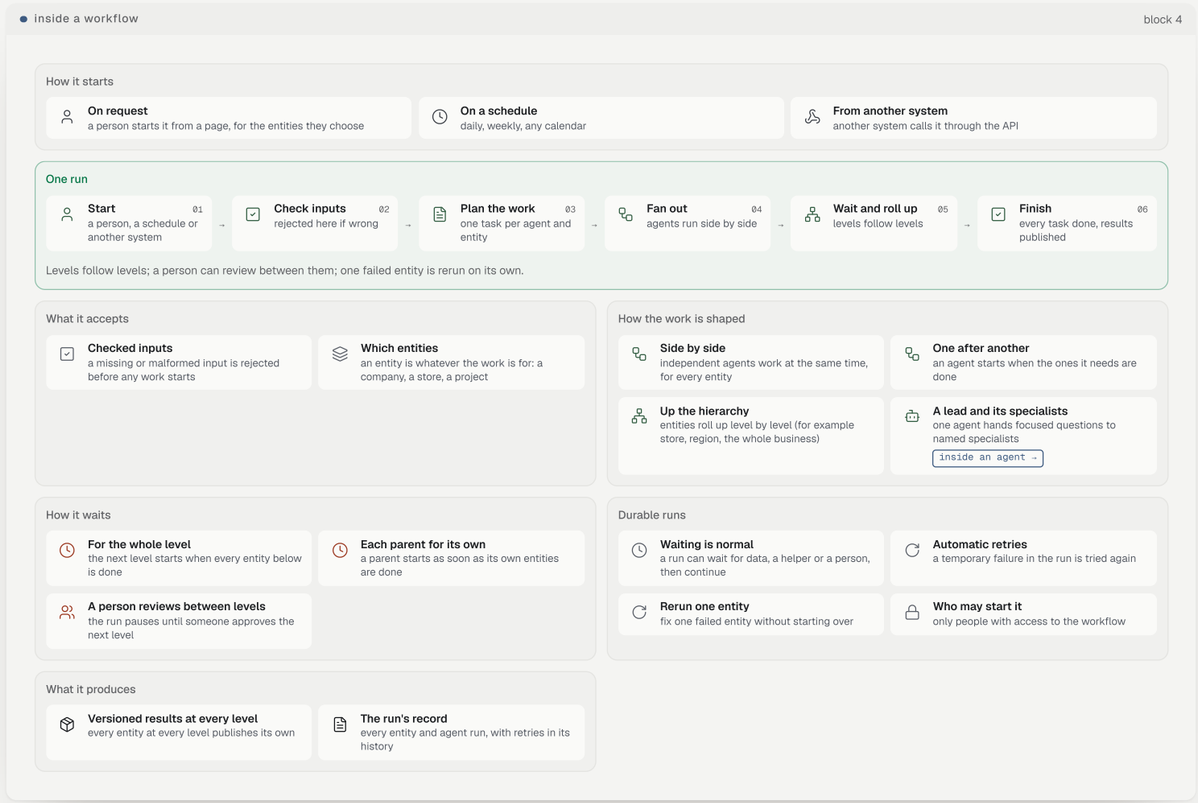
BLOCK 3 · INSIDE AN AGENT

An agent works only with what it is given and allowed. Every run leaves a record.



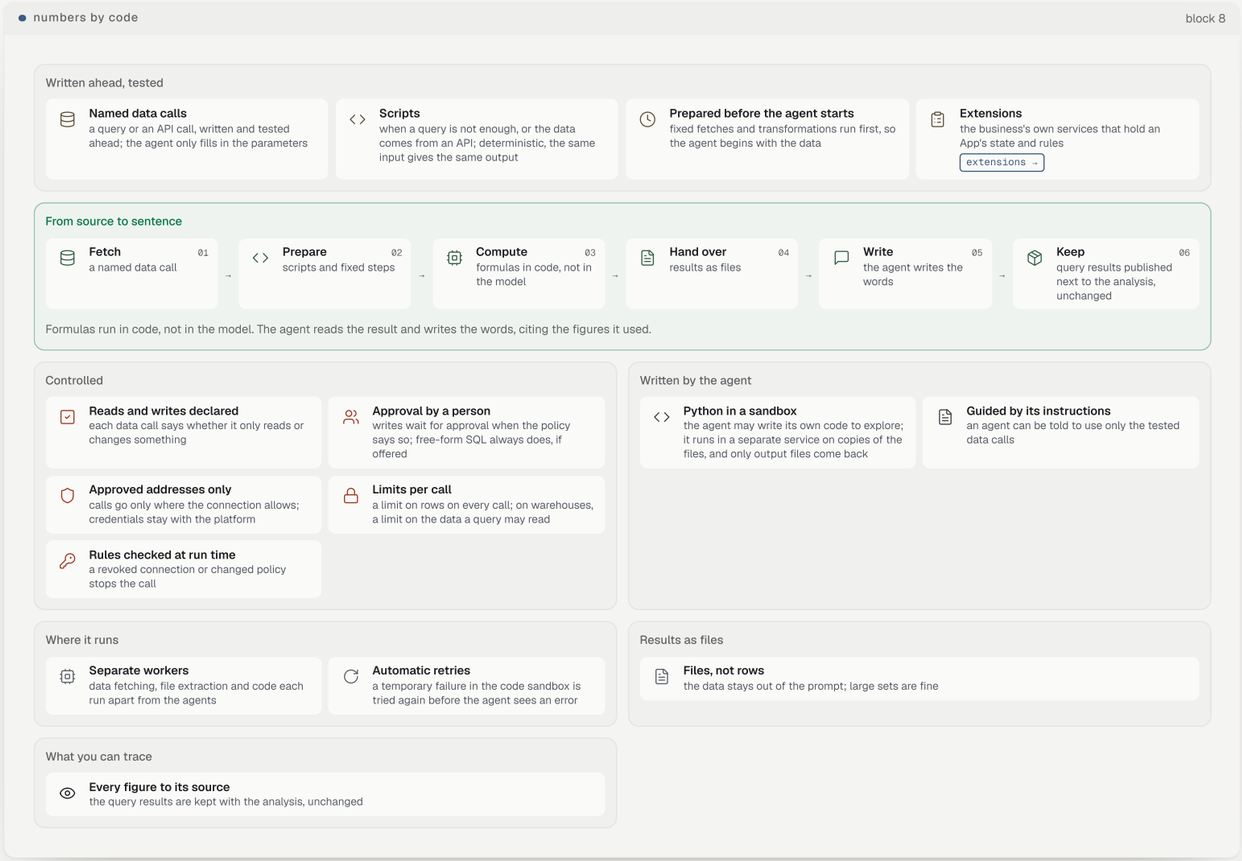
BLOCK 4 · INSIDE A WORKFLOW

Workflows decide who works and when a person checks.
The harness keeps every run going until it finishes.



BLOCK 8 · NUMBERS BY CODE

Numbers by code. Words by agents.



NEXT STEP

Start with one process. Prove it in production, then scale.
Six weeks to the first application; the second starts on the same foundation.

Day 1
A 45-minute working session on the process

Day 2
A written plan for the first application

Week 1
The application configured, with clean data

Week 6
The first application in production

Next
The second starts with the foundation in place

START WITH A PROCESS THAT IS

Regulated · Differentiating · Cross-system · Living in spreadsheets

Have a stalled pilot?
Bring it. Pyrana rebuilds it on the platform and shows the difference.

Pyrana