

# **Stop Shopping. Start Building.**

OWN YOUR BUSINESS OPERATING SYSTEM

## The SaaS Trap

---

You're a builder. You can ship product. But running your business?

- 5 tools for customer lifecycle
- 3 tools for money
- 4 tools for team coordination
- **Zero** coherence between them

You didn't start a company to become a Zapier architect.

## The Real Problem

---

| What You Need                | What You Get                       |
|------------------------------|------------------------------------|
| Express your business truths | Learn each vendor's config UI      |
| One source of truth          | Data scattered across 12 SaaS apps |
| Modify when needed           | Locked into vendor roadmaps        |
| Audit trail                  | "Check the other system"           |

You're renting fragments. You should own the whole.

# Converge

YOUR BUSINESS LOGIC, YOUR PLATFORM

## What If You Could Just Say It?

---

Truth: Get paid for delivered work

Given work is marked as delivered  
And contract terms exist  
When the system converges  
Then invoice is issued  
And payment is tracked  
And late payment triggers follow-up

**That's it.** No tool-specific configuration. No brittle Zaps.

## How It Works

---

**01 Express** — Declare what must be true as Converge Truths

**02 Compose** — Pull from the Pack library or build your own

**03 Connect** — Providers handle the SaaS plumbing

**04 Run** — The engine converges to your Truths, every time

## The Stack You Own

---

| Layer      | What It Does           | You Control           |
|------------|------------------------|-----------------------|
| Packs      | Business truth modules | Fork and modify       |
| Blueprints | Composed workflows     | Customize freely      |
| Providers  | SaaS adapters          | Swap anytime          |
| Runtime    | Execution engine       | Self-host or use ours |

No vendor lock-in. Real portability.

## Packs: Install Behavior Like Libraries

---

Instead of configuring CRM workflows, install a Pack:

- Money — Invoice, collect, reconcile, close books
- Customers — Capture, qualify, engage, close, handoff
- Delivery — Scope, track, complete, learn
- People — Hire, onboard, pay, offboard
- Trust — Access, audit, compliance

Each Pack is Truths you can read, fork, and extend.

## Built for Builders

---

- **Open Truths** — Read exactly what it does
- **Fork anything** — Make it yours
- **Compose freely** — Mix Packs, add your own
- **Self-host option** — Your infra, your rules
- **No magic** — Deterministic, explainable, debuggable

You're not a customer. You're a co-builder.

## The 9 Axioms

---

Mathematical foundations that make Converge trustworthy:

| # | Axiom                | Formula  |
|---|----------------------|--|
| 1 | <b>Monotonicity</b>  | $\text{ctx} \subseteq \text{step}(\text{ctx})$                         |
| 2 | <b>Determinism</b>   | $\text{step}(\text{ctx}) = \text{step}(\text{ctx})$                    |
| 3 | <b>Idempotency</b>   | $\text{agent}(\text{ctx}) = \text{agent}(\text{agent}(\text{ctx}))$    |
| 4 | <b>Commutativity</b> | $a(b(\text{ctx})) = b(a(\text{ctx}))$                                  |
| 5 | <b>Termination</b>   | $\exists n: \text{step}^n(\text{ctx}) = \text{step}^{n+1}(\text{ctx})$ |

## The 9 Axioms (continued)

---

| # | Axiom                     | Formula   |
|---|---------------------------|---|
| 6 | <b>Consistency</b>        | $\neg\exists(f, \neg f) \in \text{ctx}$                   |
| 7 | <b>Starvation Freedom</b> | $\text{enabled}(a) \Rightarrow \diamond\text{runs}(a)$    |
| 8 | <b>Confluence</b>         | $\text{ctx}_1 \cup \text{ctx}_2 \rightarrow \text{ctx}^*$ |
| 9 | <b>Observability</b>      | $\forall \text{effect}: \text{logged}(\text{effect})$     |

These aren't guidelines — they're **guarantees enforced by the type system**.

## From Math to Business Value

---

| Axiom         | Business Guarantee                                |
|---------------|---|
| Monotonicity  | <b>Audit trail integrity</b> — nothing lost       |
| Determinism   | <b>Reproducible decisions</b> — debug anything    |
| Idempotency   | <b>Safe retries</b> — no double-charges           |
| Termination   | <b>Guaranteed completion</b> — no stuck workflows |
| Consistency   | <b>No contradictions</b> — ledgers balance        |
| Observability | <b>Full explainability</b> — compliance ready     |

## SaaS Shopping vs Converge

---

| SaaS Shopping                           | Converge            |
|---|---------------------|
| Find tool → Learn tool → Configure tool | Express need → Run  |
| 12 admin panels                         | One Truths repo     |
| Vendor decides features                 | You declare Truths  |
| Data export nightmare                   | You own the context |
| "It's on their roadmap"                 | Ship it yourself    |

## Example: The Money Loop

---

What takes 3 SaaS tools and 15 Zaps:

```
Truth: No invoice without delivery proof
Given work is marked complete
Then invoice must reference deliverables
And customer must have approved
```

```
Truth: Period close requires reconciliation
Given month-end is triggered
Then all transactions must be matched
And unmatched items block close
```

One file. Full audit trail. Your Truths.

## The Builder Path

---

- 1. Start** — Pick a Pack, run it on our hosted runtime
- 2. Customize** — Fork the Truths, adjust to your reality
- 3. Extend** — Add your own Packs and Truths
- 4. Own** — Self-host when you're ready

No permission needed. No sales calls. Just build.

# Your Business is Unique

YOUR PLATFORM SHOULD BE TOO

## What You Get

---

- **Packs** — Pre-built business logic you can inspect and modify
- **Runtime** — Hosted or self-hosted execution
- **Providers** — Adapters to the SaaS you already use
- **CLI + API** — Build your own interfaces on top
- **Community** — Other builders sharing Truths

## **Stop Renting. Start Owning.**

---

**"Your Truths are the asset."**

Tools are replaceable. Your business reality isn't.

With Converge, you express what must be true—and the platform makes it so.

**converge.zone**

BUILD YOUR BUSINESS OS

## Let's Talk

---



**Kenneth Pernyer** · Founder

[LinkedIn](#)

[Twitter](#)

[GitHub](#)