



Scrum & AI: Partners or Rivals?

Alex Ballarin, PST

June 16th, 2026

Quick Guidelines

- Your microphones will be muted throughout
- This session is recorded. The recording and slides will be available after the webinar within 24 hours.
- Please ask questions!
 - Submit questions by selecting the Q & A icon:



Who is Scrum.org

Mission:
*Helping People and
Teams Solve
Complex Problems*



Ken Schwaber
Scrum.org Founder,
Chairman and
Co-creator of Scrum



Alex Ballarin, Professional Scrum Trainer

- Lives in Barcelona, Spain
- PST since 2016
- 22 years of experience helping teams and organizations to deliver better products
- Paying for ChatGPT Plus since the week it came out. 😊



• ALEX BALLARÍN · PROFESSIONAL SCRUM TRAINER

Scrum and AI Partners or Rivals?

How a Scrum Team's work changes when AI enters the development flow.

● QUICK POLL

What best describes your team today?

Before this session — where is your team with AI?

A We use AI individually, no shared workflow

B AI is integrated into how our team works

C We are experimenting / evaluating

D We have not started yet

● POLL RESULTS

Where are we as a group?

● THE DEBATE

"AI killed Scrum."

//

Two-week Sprints were designed for a world where writing code took time. That world is gone. With AI, a feature goes from idea to working code in minutes. You don't need ceremonies for that.

SE Senior Engineer · 4
000 likes on LinkedIn

Is this right? Let's find out.

WHAT WE WILL COVER

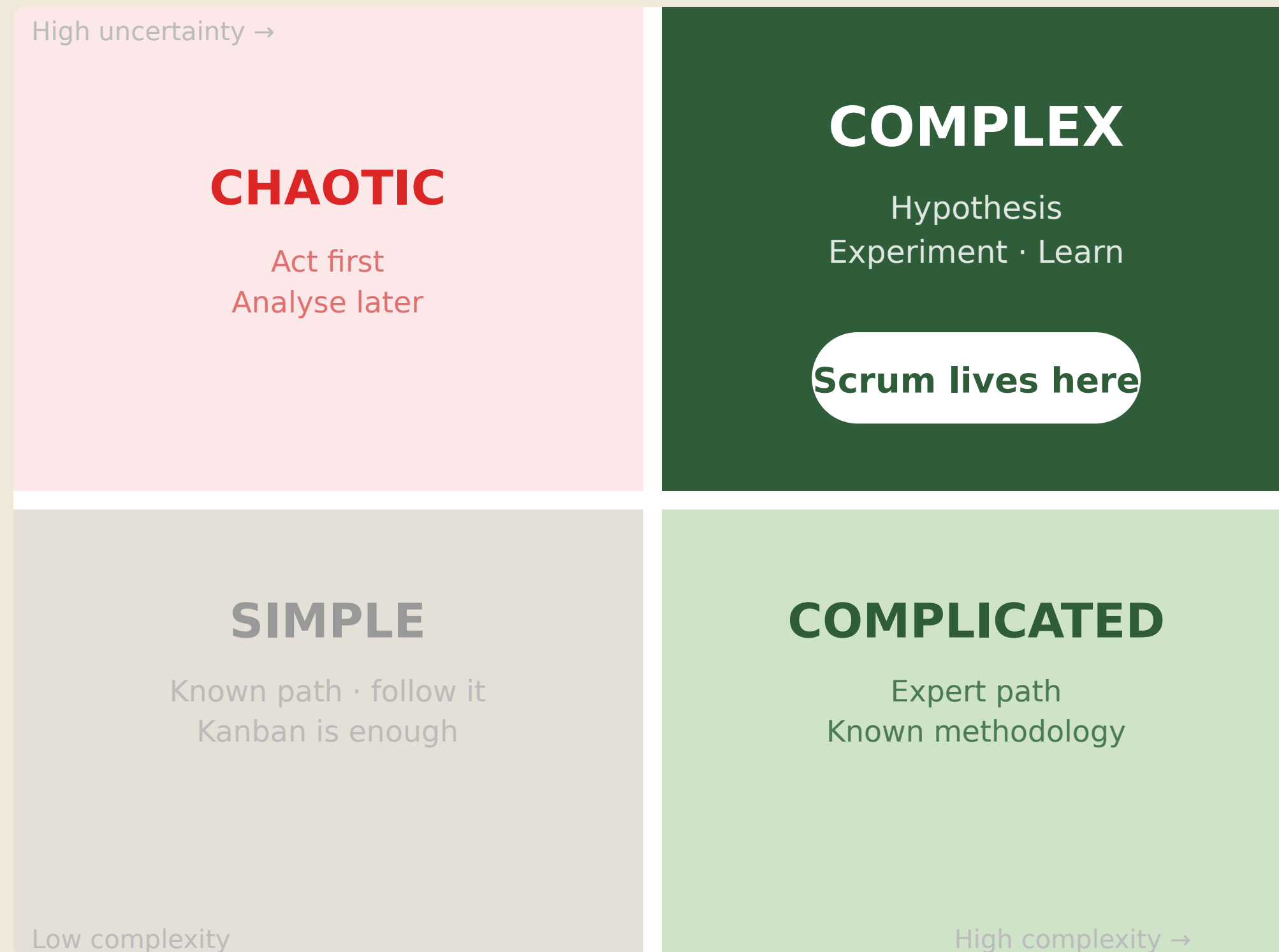
One question. Five answers.

01	What Scrum is actually for	8 MIN
02	What really changes with AI	10 MIN
03	The logical model as a compass	7 MIN
04	How Scrum adapts	12 MIN
05	Partners or Rivals?	10 MIN + Q&A

Guiding question: *What really changes when AI joins the team?*

● BLOCK 1 · WHERE SCRUM BELONGS

Not every problem needs Scrum.



Scrum was designed for **complex** problems — where neither requirements nor solution path are clear upfront.

● SCRUM LIVES HERE

- **Simple:** Known path → Kanban is enough.
- **Complicated:** Expert path → Known methodology.
- **Chaotic:** No path yet → Act first, analyse later.
- **Complex:** Unknown path → You *discover* the answer, not plan it.

- SCRUM AS A COMPLEX PROBLEM SOLVER

Not a delivery process. A learning machine.

Mechanical delivery

Sprints are two-week output cycles.
Stories go in, code comes out, Demo is a checkpoint.

Velocity is the north star.

"Done" means code is merged.
No question is being answered.

Result: fast output, unknown direction.

Outcome compass

Each Sprint starts with a **Sprint Goal** —
a hypothesis.
The Review is a feedback session, not a demo.

Impact is the north star.

"Done" means something was learned.
Every Sprint answers a specific question.

Result: slower on features, faster on value.

● THE KEY INSIGHT

AI does not remove uncertainty.

The reasons Scrum was born still hold: markets shift, users surprise us, technology behaves unexpectedly.

“

The biggest danger for Scrum is not Waterfall, not AI — it is Scrum reduced to a meeting cadence without empiricism.

If your Sprints are answering real questions, AI makes Scrum more powerful. If they are just output cycles, AI makes the problem bigger.

● BLOCK 2 · WHAT CHANGES WITH AI

Three models of AI-assisted development.

● VIBE CODING

Prompts → working code in hours

Ideal for exploration and rapid prototyping. The developer stays in a creative flow — no boilerplate, no context switching.

Trade-off: high potential for technical debt and undefined behaviours at the edges.

● SPEC-DRIVEN DEVELOPMENT

Structured specs → precise generation

The developer writes detailed specifications; the AI generates production-quality code. Less ambiguity, less debt.

Trade-off: requires disciplined thinking upfront — harder to start, much easier to maintain.

● AGENTIC AI

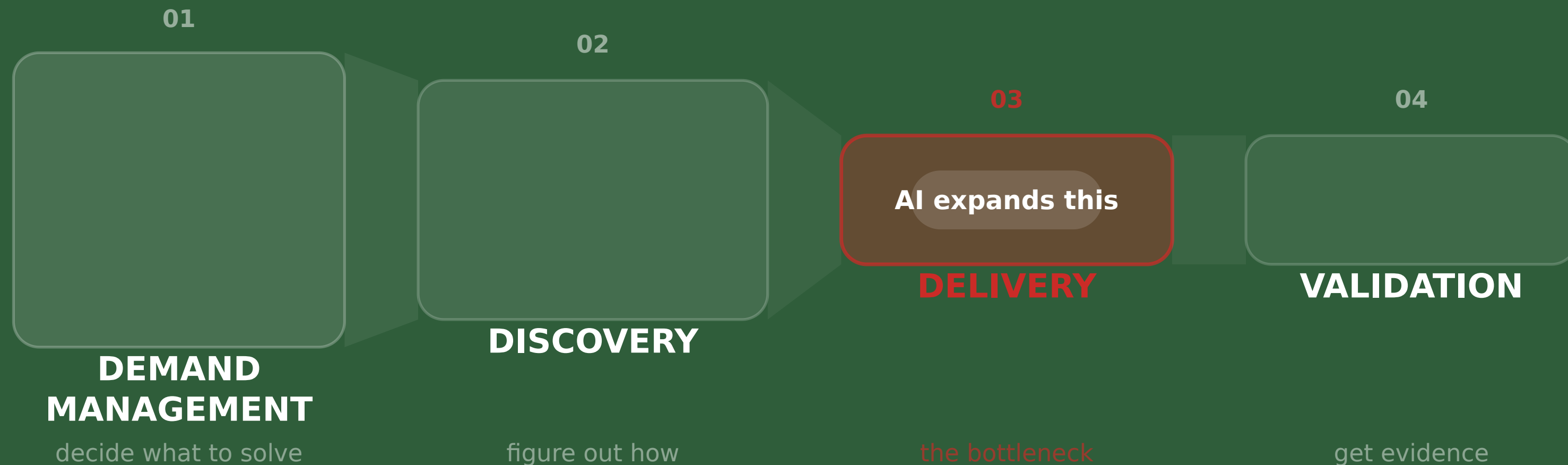
Autonomous flows → team supervises

The AI agent runs multi-step tasks end-to-end: write, test, fix, commit. The team reviews outputs and approves decisions.

Trade-off: highest leverage, but requires mature validation practices and clear boundaries of autonomy.

● THE SHIFT

The bottleneck moved.



AI has dramatically expanded the throughput of **Delivery** — what used to take weeks now takes hours.

The constraint is no longer writing code. It is **deciding what is worth building** — filtering demand, framing the problem, choosing what *not* to do.

● THE REAL RISK

Feature factory — on steroids.

×17

Code generated per developer

×1.3

Releases shipped

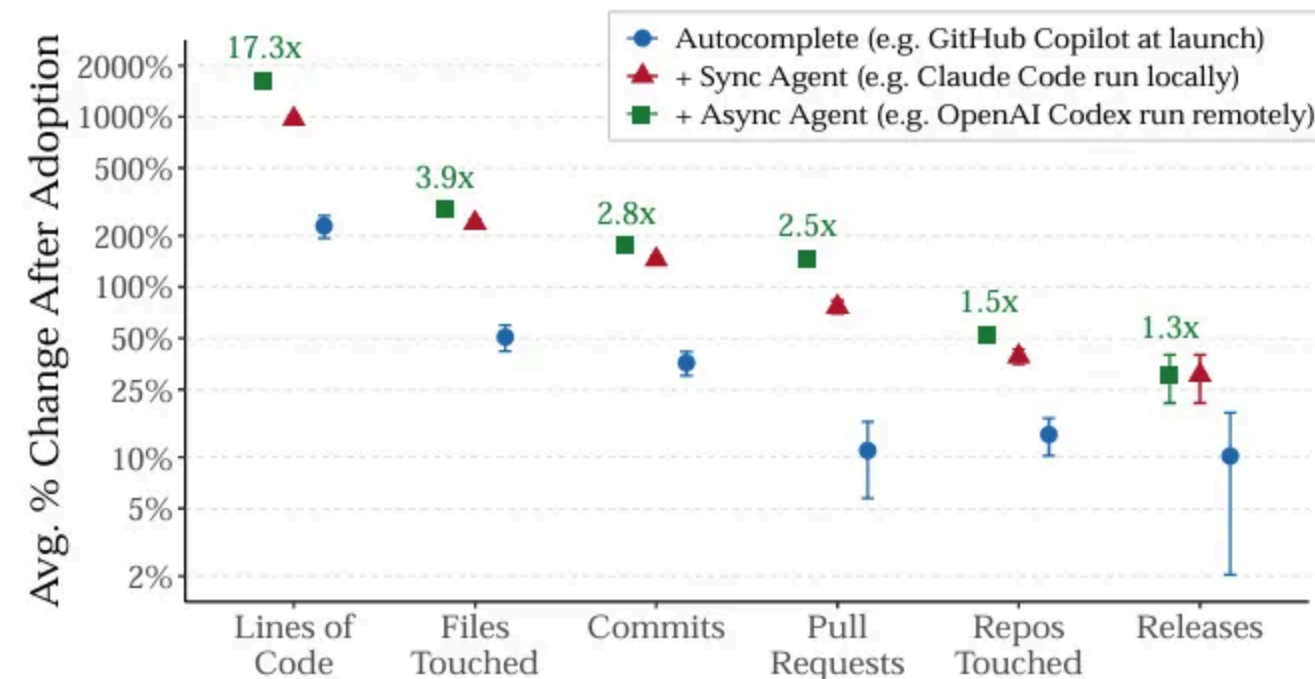
?

Validated outcomes

Speed is not the problem. **Strategic disconnection** is — and AI amplifies it.

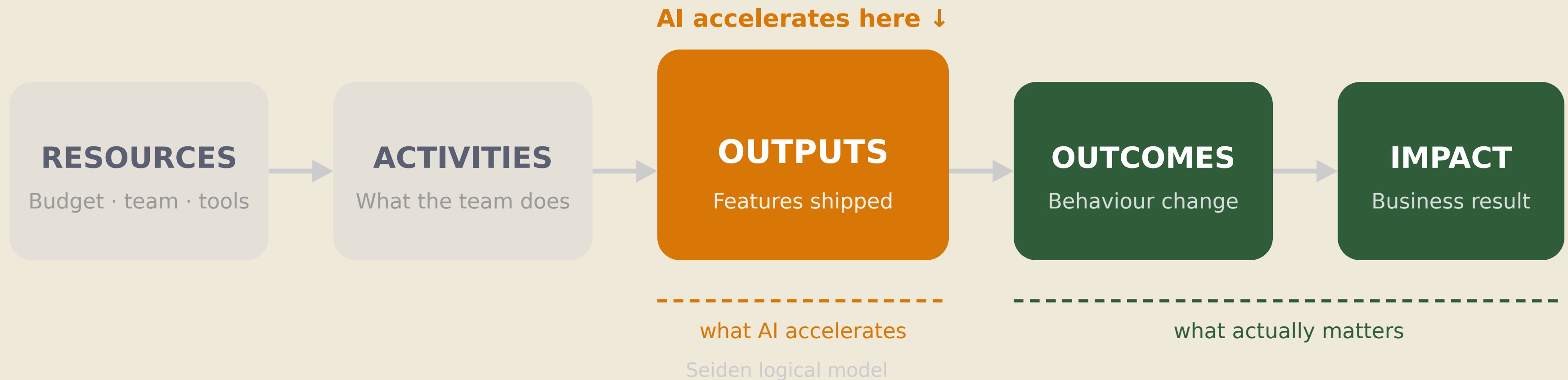
- Stakeholders push for more. Roadmaps expand.
- Teams execute without questioning value.
- Outcomes do not scale with velocity.

Figure 1: Overview of Productivity Effects of AI Coding Tools



● BLOCK 3 · THE LOGICAL MODEL AS COMPASS

AI accelerates one link. The chain has five.

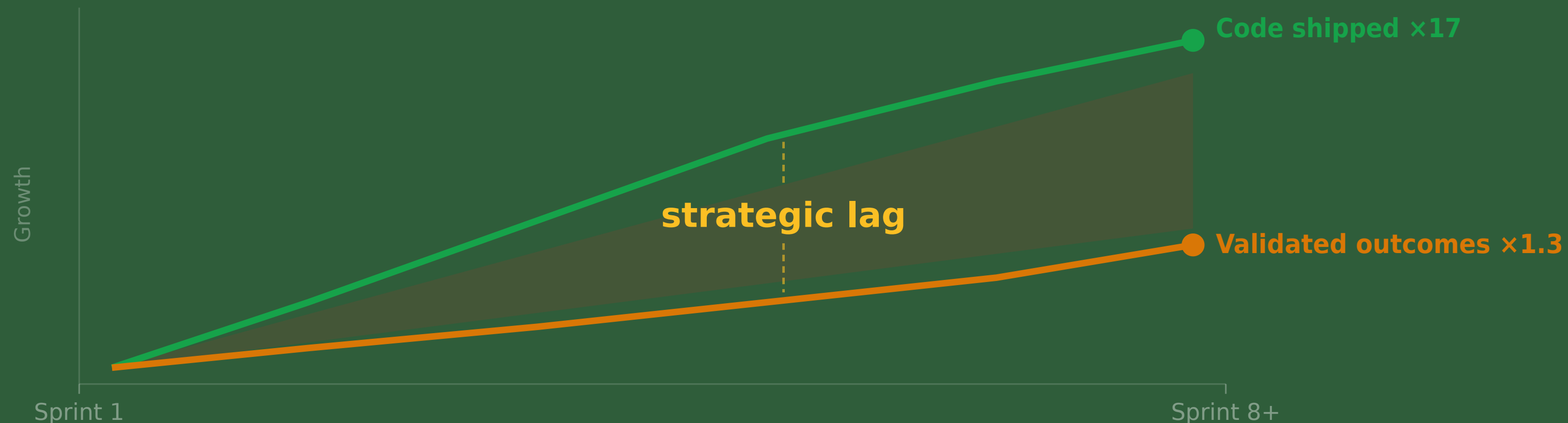


The **logical model** (Seiden) shows what connects effort to value. AI dramatically accelerates **Outputs**. The problem: teams celebrate output velocity as if it were outcome velocity.

Without clear Sprint Goals and Product Goals, more output just means more noise.

● THE LAG PROBLEM

When velocity goes up, the measurement gap widens.



As the pace of delivery increases, the lag between what we ship and what we measure — user behaviour change, retention, conversion — grows proportionally.

● OUTCOME-ORIENTED SCRUM

Product Goal and Sprint Goal as decision filters.

Product Goal

The strategic outcome the team is chasing in this phase of the product.

Not a feature list. Not a project scope.

A **hypothesis** about impact: *"If we improve retention in the first 7 days, we will reduce churn by X."*

Every backlog item is evaluated against this filter before it is even discussed.

Sprint Goal

The specific question this Sprint will answer.

Not a Sprint backlog. Not a commitment to features.

A **learning objective**: *"We believe that adding onboarding step Y will increase day-3 return rate."*

The Daily Scrum asks: are we still on track to answer this question?

● BLOCK 4 · HOW SCRUM ADAPTS

The empirical loop remains. The frequency and focus change.

What does NOT change

The three pillars: **Transparency, Inspection, Adaptation.**

The need for a clear Product Goal.

The value of a cross-functional, self-managing team.

Short cycles with real feedback.

Human judgment at the decision points.

What DOES change

Sprint length can shrink — code generation is no longer the constraint.

Daily focus shifts from progress reporting to output validation.

Refinement becomes specification work — the input to AI generation.

Review shifts from feature demo to outcome conversation.

Definition of Done must rise with output speed.

- SPRINT LENGTH AND DEFINITION OF DONE

When code generates in hours, two weeks is too long to wait for feedback.

Shorter Sprints are not about moving faster. They are about learning faster — reducing the time between a hypothesis and its first real signal.

- **Sprint Goal** → a *falsifiable hypothesis* , not a feature commitment.
- **Product Goal** → the strategic filter, visible at every Planning, Review and Daily.
- **Definition of Done rises** → automated tests, security review, observability. The bar cannot fall because delivery got cheaper.

If your DoD is the same as before AI, you are accumulating invisible risk.

● ADAPTED EVENTS

The events evolve. Their purpose stays the same.

Daily Scrum

Shifts from **progress sync** to **output validation** and **AI decision review**.

Questions that matter now:

- Does yesterday's output actually meet the Sprint Goal?
- What agent decisions must the team review or override?

Sprint Review

Shifts from **feature demo** to **outcome conversation**.

Questions that matter now:

- What did we learn about real user behaviour?
- Do the signals support or challenge the Product Goal?
- What should we stop building?

Retrospective

Shifts from **process friction** to **human-AI collaboration quality**.

Questions that matter now:

- Which specs produce the best AI output?
- Where are we over- or under-supervising the agent?
- What validation steps are we skipping?

● THE HYBRID SCRUM TEAM

The roles evolve. The accountability stays the same.

Product Owner

Becomes a **hypothesis validator**, not a feature approver.

The backlog is a set of bets, not a to-do list. The PO's job is to maximise the signal-to-noise ratio: which bets are worth running this Sprint?

The key skill: translating business outcomes into testable Sprint Goals.

Scrum Master

Becomes a **facilitator of human-AI learning**, not a meeting scheduler.

Helps the team build the discipline to specify well, validate rigorously, and improve the collaboration loop.
Removes impediments to effective supervision of AI output.

Developers

Become **architects and supervisors of AI output**, not prompt operators.

They design the system, write the specifications that produce quality code, review what the AI generates, and own the Definition of Done. AI executes; developers judge.

● THE BACKLOG EVOLVES

From user stories to value hypotheses.

Before

"As a user I want to receive a weekly digest so that I stay informed."

- Captures **request** — assumes the feature creates a value.
- Says nothing about what we will measure, or when to stop.
- Output is a **feature** → success = *shipped* .

Now

"We believe a personalised weekly digest will make users return 2x more often. We will test it with 20% of users in Sprint 8."

- Captures **hypothesis** — defines the outcome and a the signal.
- Includes a **stopping condition** if it fails.
- Output is **learning** → success = *validated* .

Spec-Driven Development changes the *format* of the specification. The shift above changes its *purpose*.

● THE NEW BOTTLENECK

When building costs almost nothing, the decision is the scarce capital.

More ideas than ever. More stakeholder requests. More technical possibilities. The ability to filter, prioritise, and say no is now the team's most valuable — and most scarce — skill.

“

The Product Goal is not a planning artefact. It is the filter that protects the team from going very fast in the wrong direction.

Teams that master demand management in the AI era will build ten times more value with the same people. Teams that don't will build ten times more waste.

● PARTNERS OR RIVALS? — THE ANSWER

It was designed to learn fast.

Scrum was not designed to produce fast — AI handles production now. What Scrum provides is a framework for **learning fast**: clear hypotheses, short cycles, validated outcomes.

The AI-era team that wins is not the one that ships the most. It is the one that answers the right questions fastest.

● THREE THINGS TO DO THIS WEEK

Start here.

-
- | | | |
|----|------------------------------------|-------------------------------------|
| 01 | Define a clear Product Goal | BEFORE YOU USE AI TO BUILD ANYTHING |
|----|------------------------------------|-------------------------------------|
-
- | | | |
|----|-------------------------------|---|
| 02 | Audit your Daily Scrum | IS IT VALIDATING VALUE — OR REPORTING ACTIVITY? |
|----|-------------------------------|---|
-
- | | | |
|----|-------------------------|-------------------|
| 03 | Measure outcomes | NOT JUST VELOCITY |
|----|-------------------------|-------------------|
-
- If you do one thing: write a Product Goal before your next Sprint Planning — the outcome you are chasing, why it matters now, and how you will know you succeeded.

“

Speed is not the problem. The problem is going very fast in the wrong direction.

• THANK YOU · Q&A

Let's talk.

Alex Ballarín · Professional Scrum Trainer · ITNOVE

- **Scrum.org** · scrum.org/alexballarin
- **Web** · itnove.com
- **Email** · alex@itnove.com

Questions

Connect with the Scrum.org community



Forums
Scrum.org



X
@scrumdotorg



LinkedIn
LinkedIn.com/
company/
Scrum-org



Facebook
Facebook.com
/Scrum.org



RSS
Scrum.org/RSS



YouTube
@ScrumOrg





Thank you!