



Project Management in Technical Work

My **career journey** through project management — in construction, software, and everything in between.

Corey Yang-Smith

March 16 2026

Who Am I?

BSc. Civil Engineering



EIT @ Urban Systems

Construction and project management for municipal land development.



M.Eng Software Engineering

Transitioned my career towards tech, hoping to bridge the gap between domain-expertise and implementation.

SWE/ML Lead @ Urban Systems

Team management, stakeholder engagement, balancing time, budget, and scope.

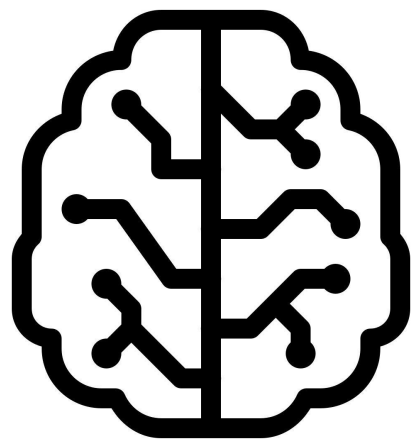


PhD Software Engineering

Now engaging in longer-term research projects.

ML Scientist @ SOVRA

More technical work, with "internal" project management.



Who Am I?

BSc. Civil Engineering

Managing multiple classes and projects

(Capstone), on top of Club responsibilities (budget-constrained).

Scope, deadlines, team coordination.

M.Eng Software Engineering

Multiple class projects, and capstone

now aligned with agile.

Agile thinking, requirements translation, and iterative delivery.

PhD Software Engineering

Longer term, larger scope research projects.

Working with long-term planning, milestones, ambiguity, and research risk.

EIT @ Urban Systems

Real construction projects, multiple stakeholders.

Through and through project management

Project planning, (external) stakeholder communication, and project delivery.

SWE/ML Lead @ Urban Systems

Team management, cross-functional teams,

engagement with executive stakeholders.

Balancing scope, schedule, budget, and (internal) stakeholder expectations while leading technical work.

ML Scientist @ SOVRA

Longer term, more complex projects. Self-directed time management.

Technical uncertainty, prioritized experimentation, and aligning work to business goals.



What I Learned Along the Way

University Projects

PM felt **informal and annoying**, but I learned it creates structure.



Construction Projects

PM became tangible because scope, stakeholders, and deadlines had real consequences.



Software Projects

PM became less about fixed plans and more about prioritization, alignment, and iteration.



AI/ML and Research Projects

PM became about managing uncertainty, experimentation, and risk.

Sound Familiar? **University Projects**



Unclear Roles

Everyone assumes someone else is handling it



Uneven Workloads

Two people do everything; two people do nothing



Miscommunication

Assumptions go unchecked; scope drifts without anyone noticing



Last-Minute Crunch

No milestone tracking means everything due the night before

❏ Even a three-person class project needs coordination. The habits you build now carry directly into industry.

Industry Projects



















Industry Projects

→ Multiple Teams, Multiple Companies

Competing interests, different incentives, shared deadlines.



Industry Projects

→ Multiple Teams, Multiple Companies

Competing interests, different incentives, shared deadlines.



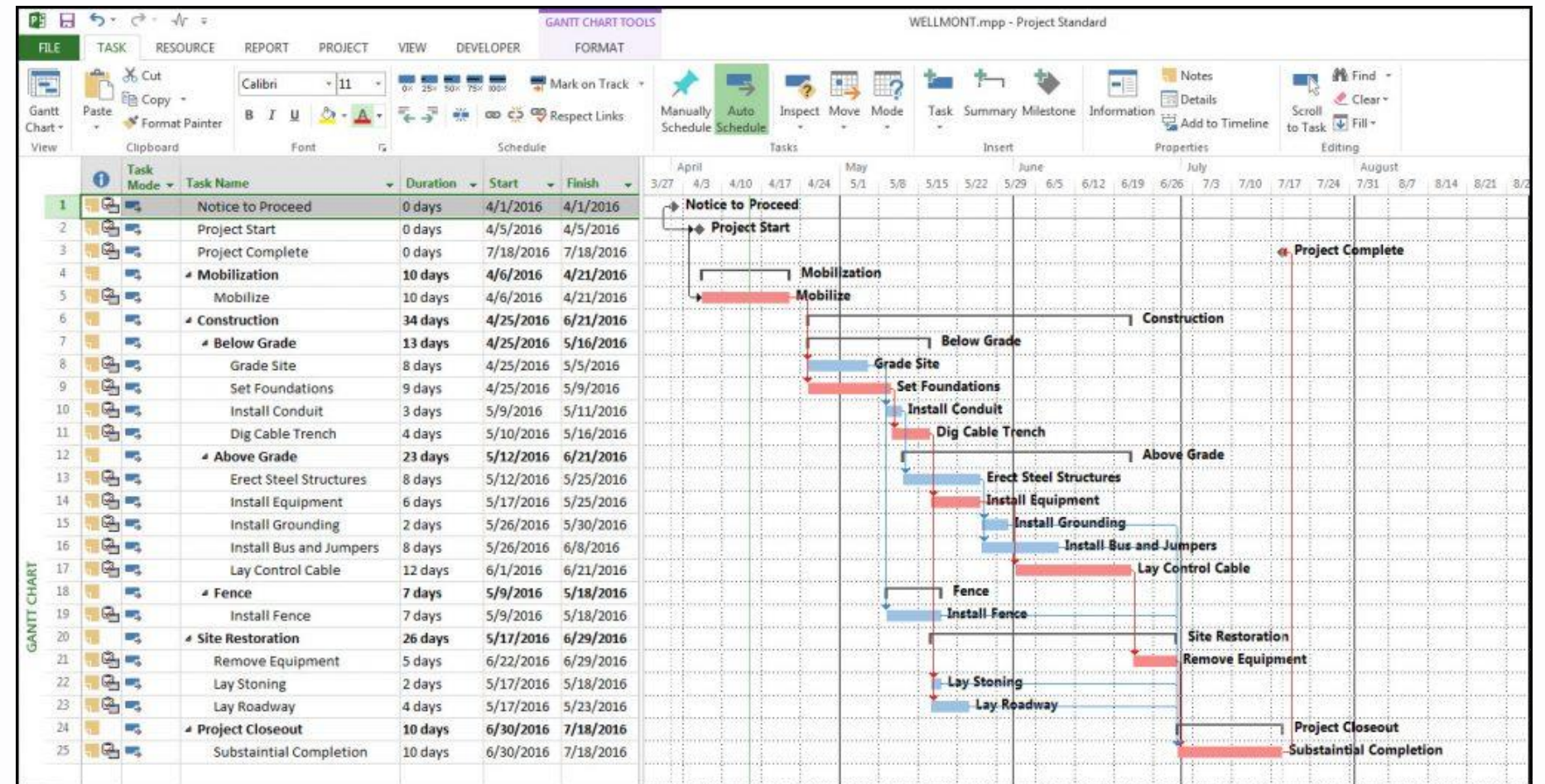
Industry Projects

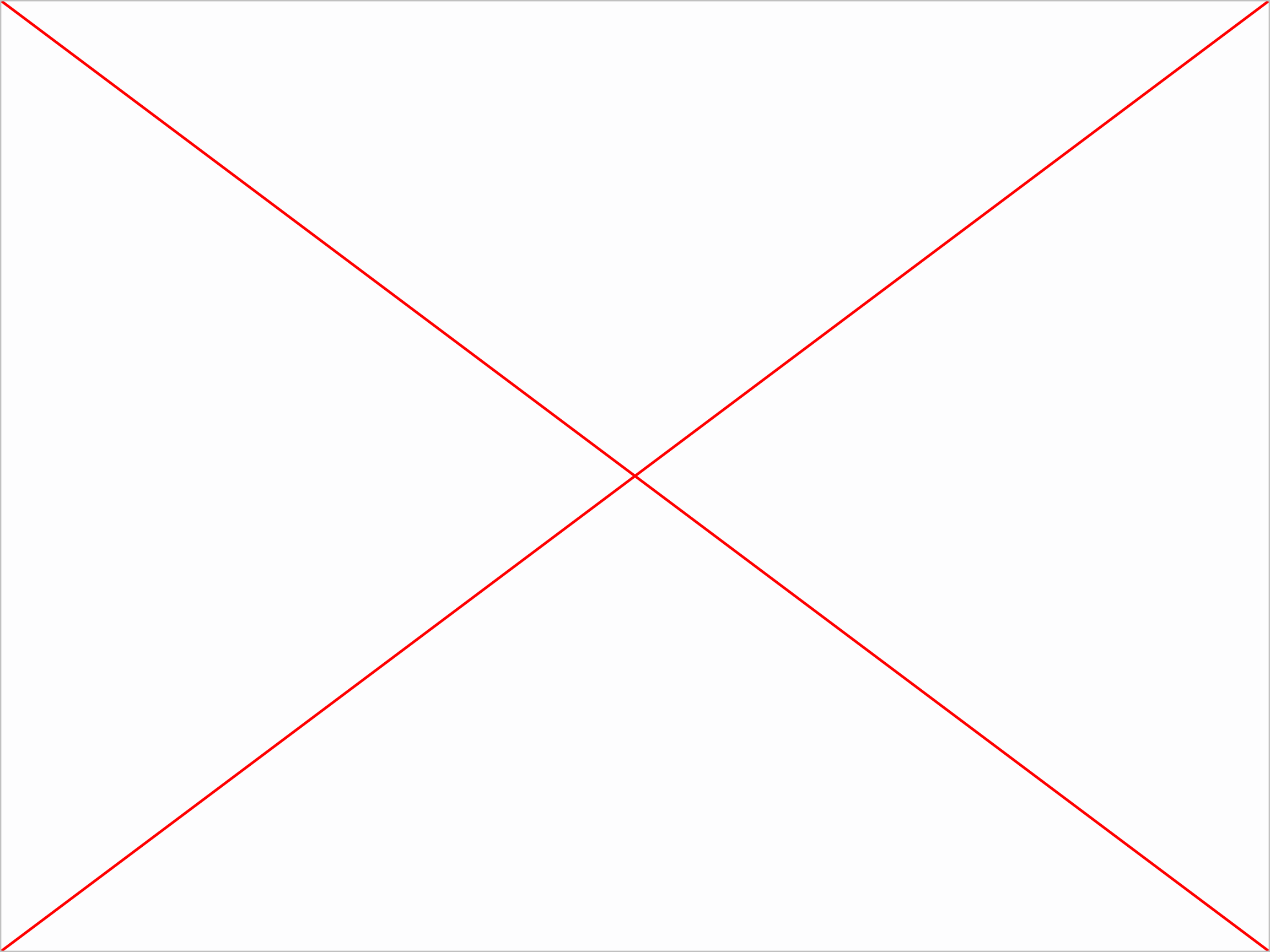
→ Multiple Teams, Multiple Companies

Competing interests, different incentives, shared deadlines.

→ Schedule Gambling

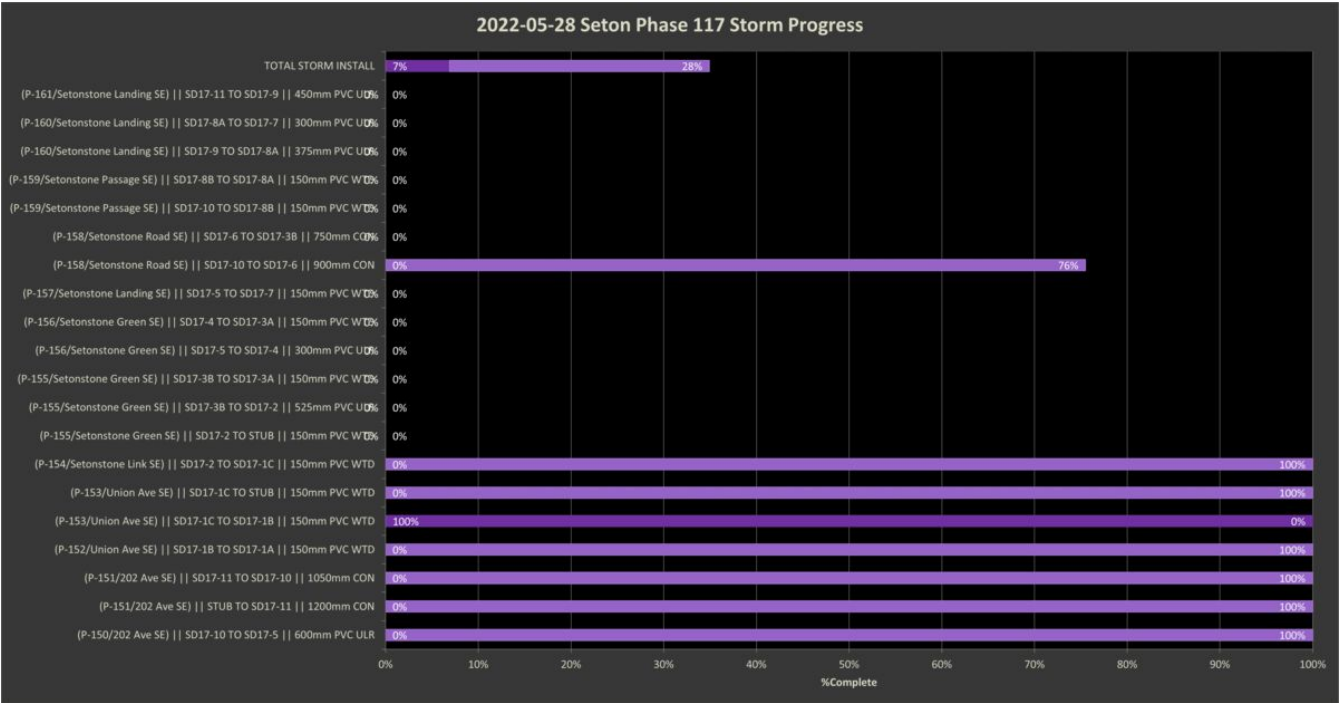
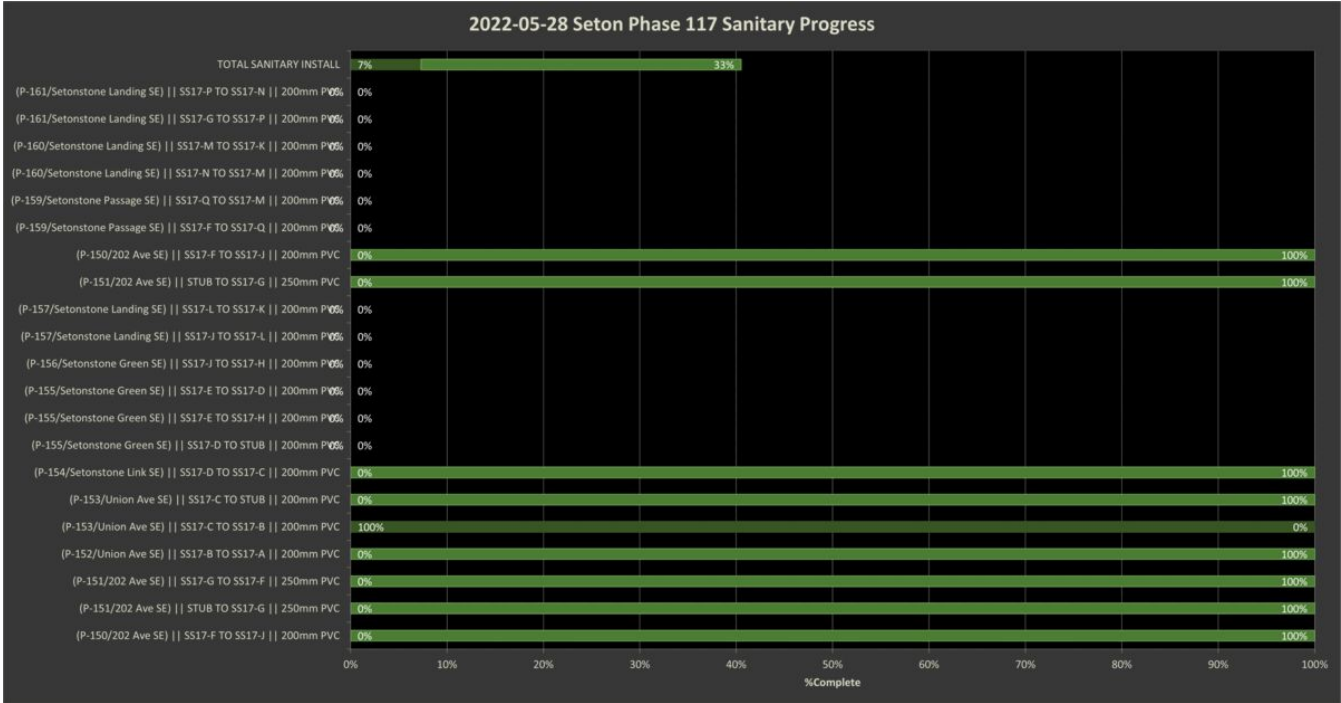
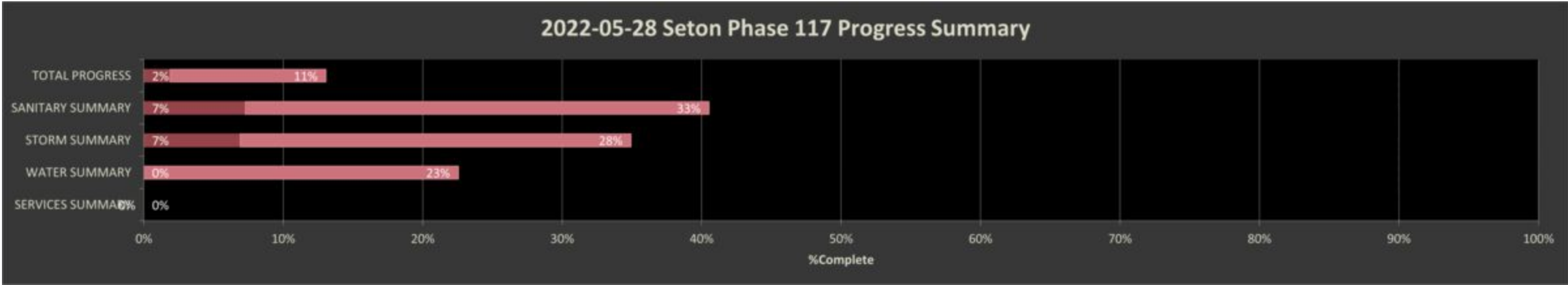
Gambling with project setbacks, weather conditions, and the city's review backlog.







Ready Accessibility: Investigate 82%



2022-05-28 Seton Phase 117 Weekly Report



Industry Projects

→ Multiple Teams, Multiple Companies

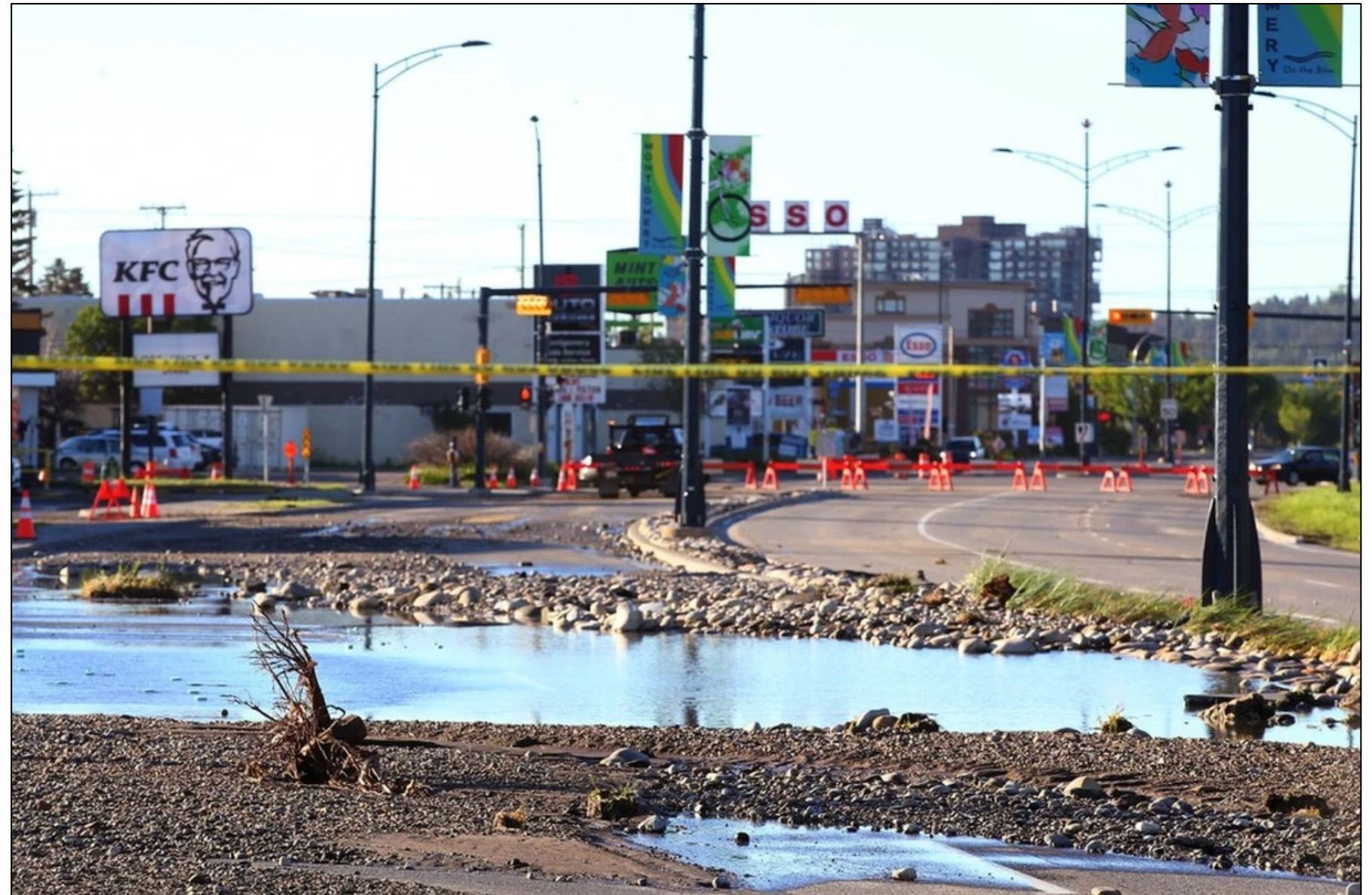
Competing interests, different incentives, shared deadlines.

→ Schedule Gambling

Gambling with project setbacks, weather conditions, and the city's review backlog.

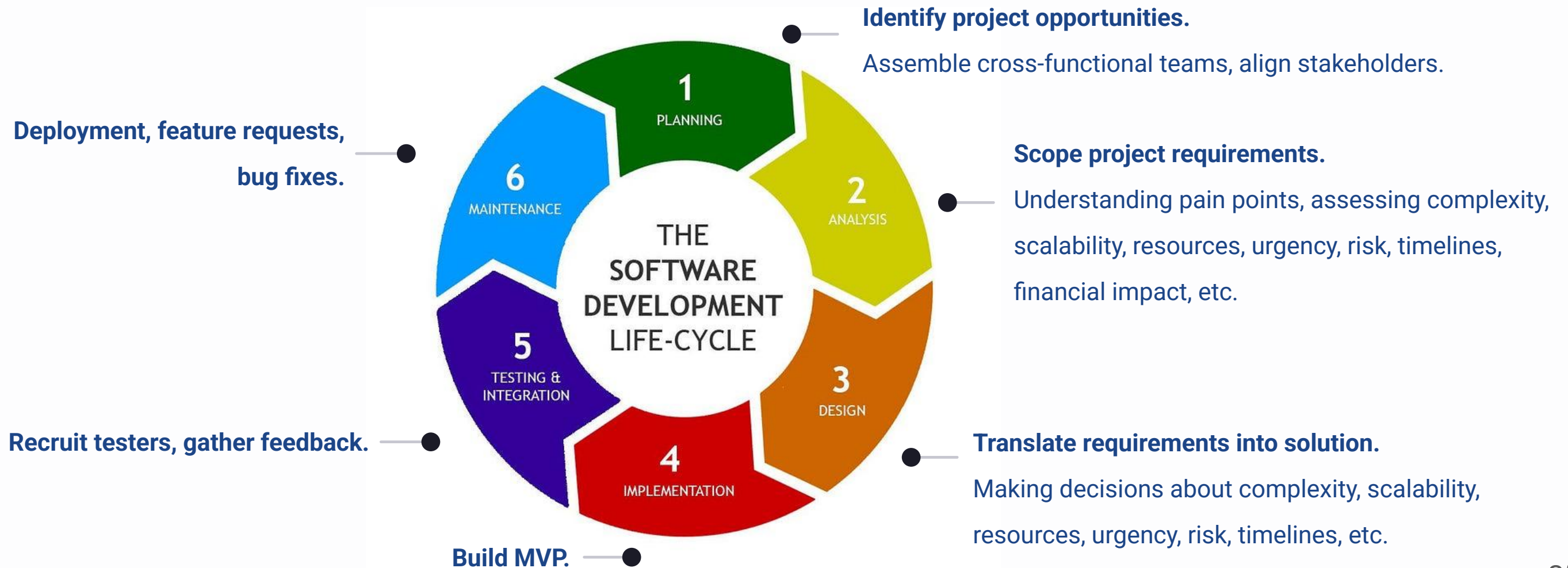
→ Real Consequences

People's livelihoods depend on delivery — focus sharpens fast



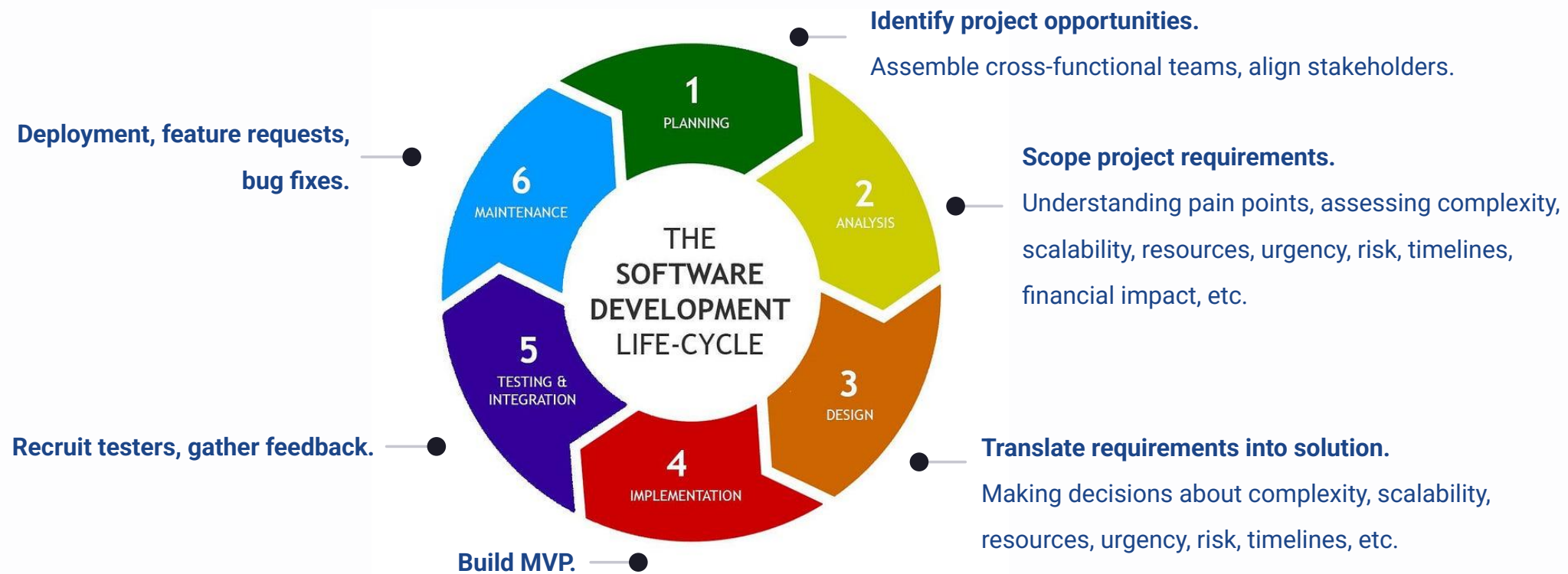
Building Software Products & Managing a Team

Shipping software isn't just writing code — it's translating, coordinating, and deciding under uncertainty.



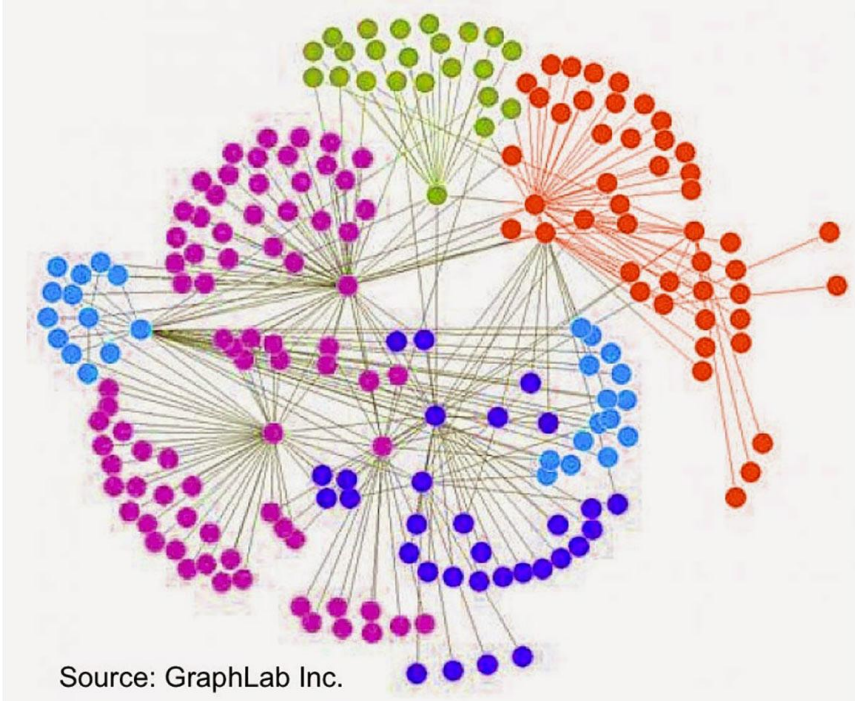
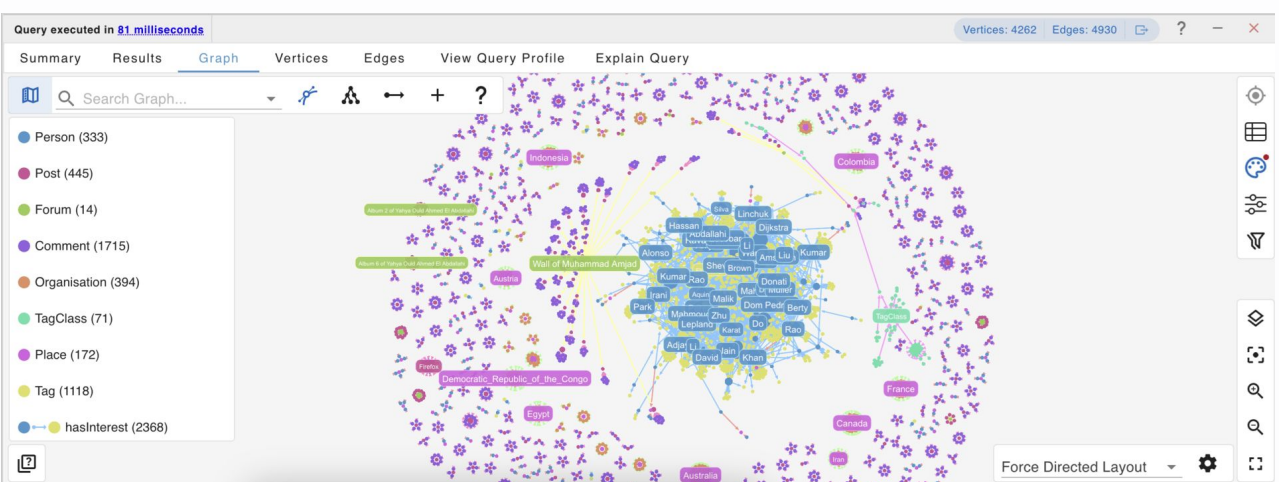
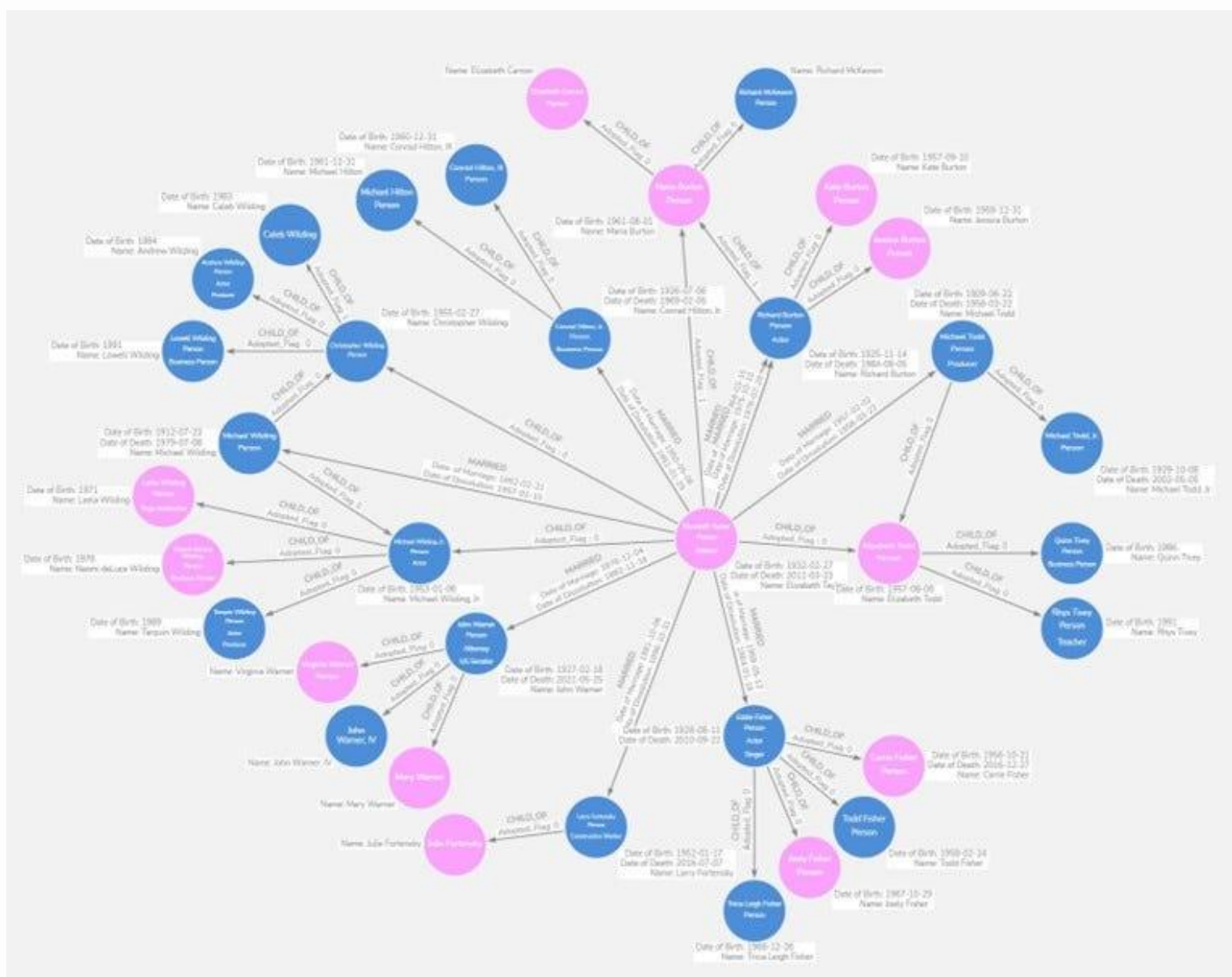
Building Software Products & Managing a Team

Shipping software isn't just writing code — it's translating, coordinating, and deciding under uncertainty.



❏ This is a **different kind** of project management — software evolves rapidly, and requirements rarely stay still.

Building Software Products & Managing a



Source: GraphLab Inc.

Building Software Products & Managing a Team



Working on Technical Projects Today

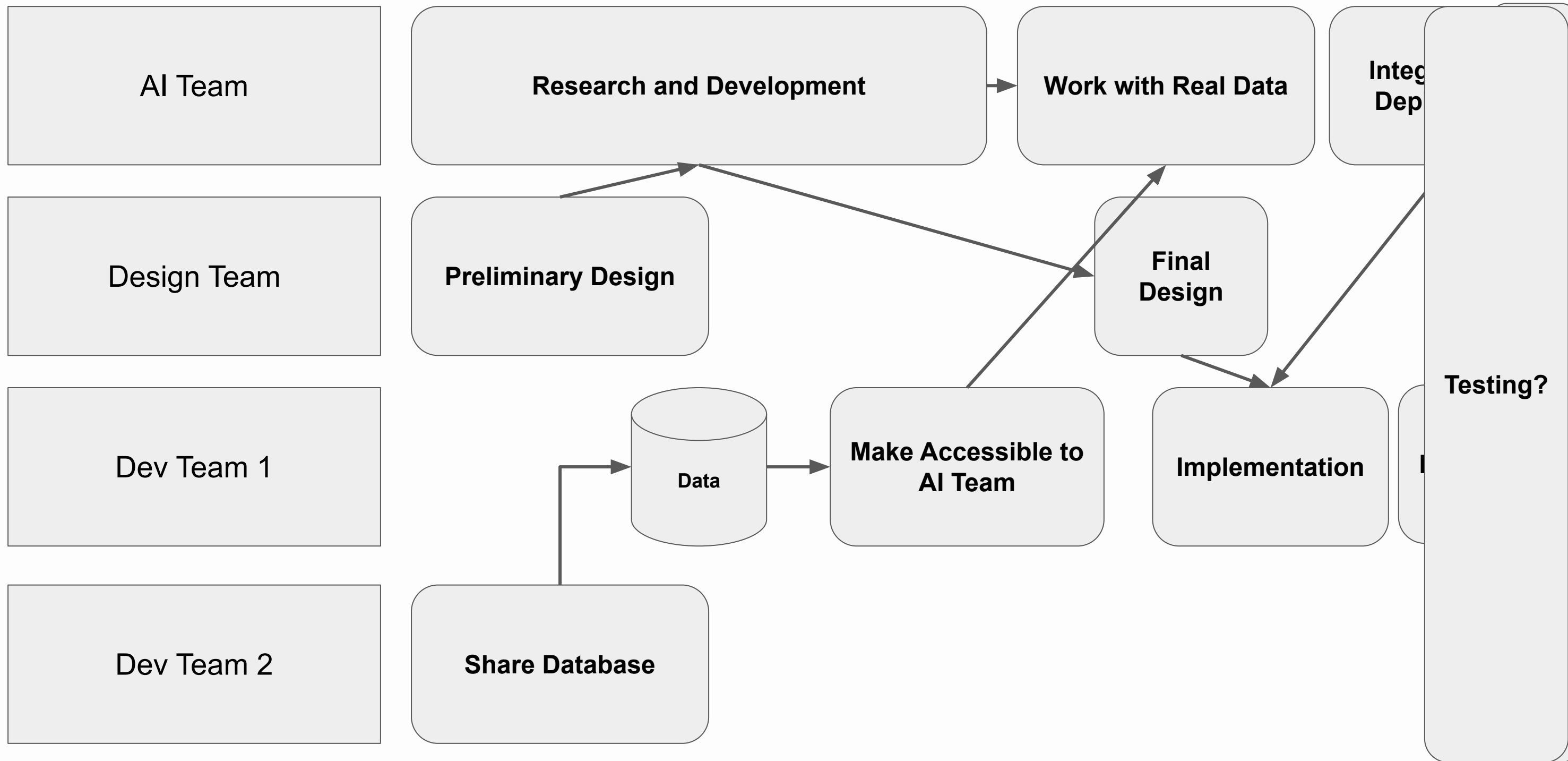
Complexity scales — and so does the need for coordination.

Requirements continue to evolve — even after development has begun.

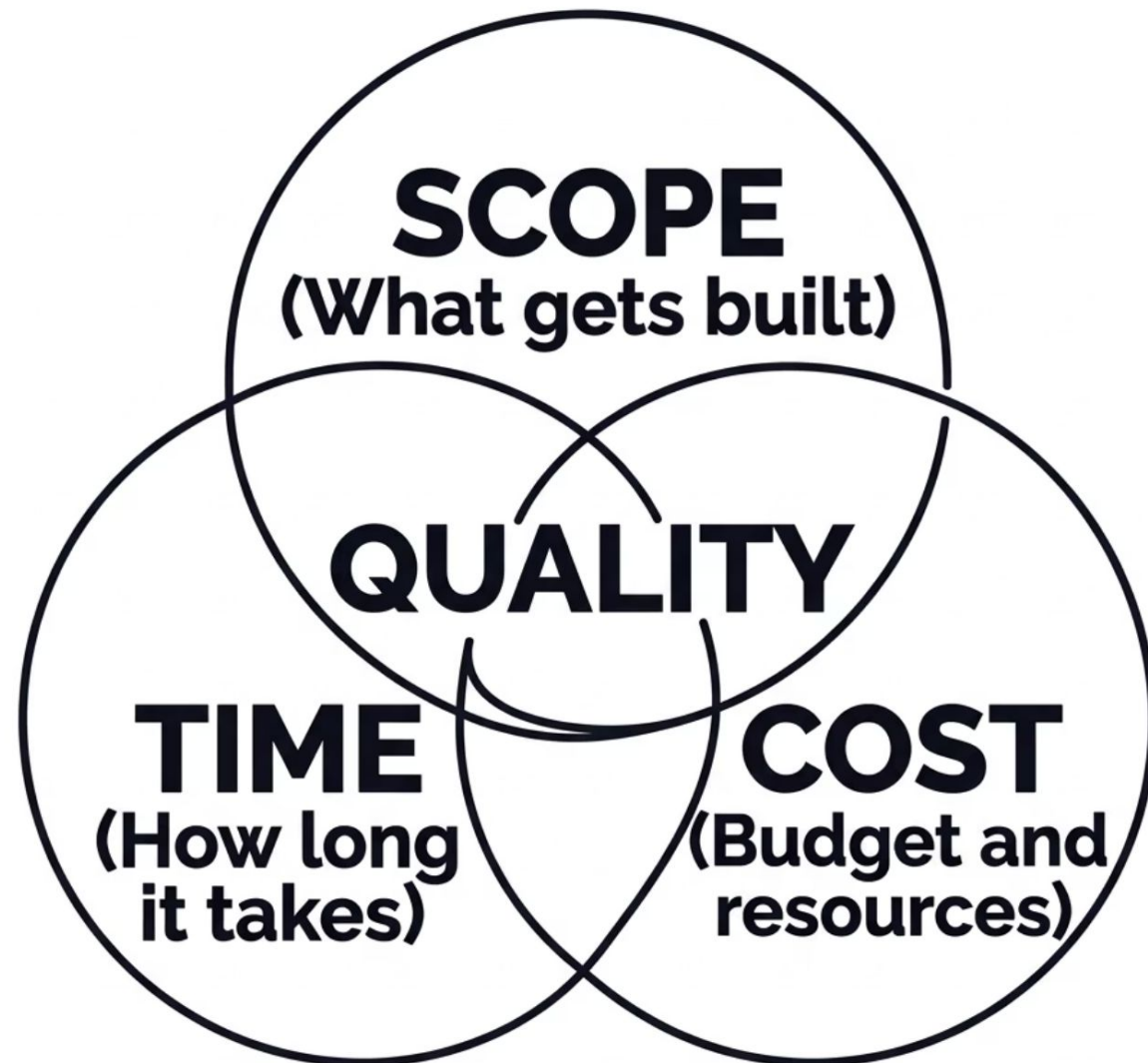
Integration challenges multiply with existing systems and growing internal teams.

Technical expertise alone isn't enough — knowing how to coordinate is the differentiator.





The Project Management Triangle



Changing one forces trade-offs in others.

Every project lives inside this triangle

You can have it **fast, cheap, or good** — pick two.

Every project decision is really a trade-off between these three constraints.

Based on your project, company, expertise, resources, demands, uncertainty, risk tolerance, etc - **which one of these can you afford to trade?**

Good PM then is deciding what is flexible and what is fixed, through understanding the business, the client, and the project objectives.

Lessons Learned & Key Takeaways

-  **Communication problems are often the real project problems.**
Unclear expectations, weak alignment, poor handoffs, and assumptions nobody surfaced.
-  **Technical Work only matters if it gets delivered.**
Great code is not enough on its own; it has to be coordinated, reviewed, communicated, and turned into something usable.
-  **Project Management is not a title, it's a responsibility.**
Even if you are a developer, researcher, or technical lead, you will end up managing scope, communication, risk, and delivery.

-  **Ambiguity is normal, especially in technical work.**
Good PM is not having things perfectly clear at the start, but it's creating enough structure to keep moving when things are uncertain.
-  **Alignment and prioritization matter more than just working hard.**
Teams can be busy and still make poor progress. Good PM is deciding what matters most right now.

Start Building These Skills Now

Every engineer eventually becomes a project manager. The question is whether you're ready when it happens. 🚀

Questions?

