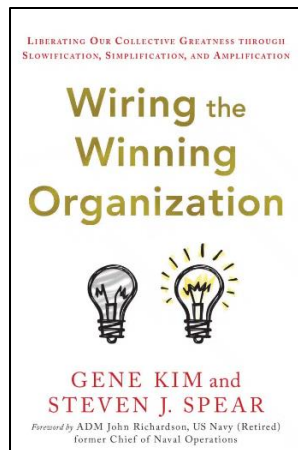
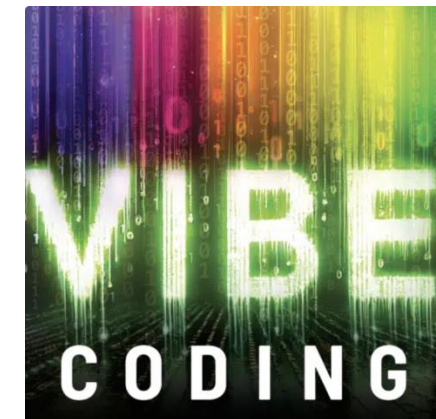


COORDINATION COSTS AND REWIRING ORGANIZATIONS TO WIN WITH AI

***Supercharging the learnings of DevOps
v2.0: Fast and Furious Edition***

#FAAFO: Fast, Ambitious, Autonomous, Fun & Optionality

***Scott Prugh
CTO: Biller Platform ACI Worldwide
scott.prugh@aciworldwide.com***



AI PRODUCT ENGINEERING

AI PROCESS ENGINEERING

“BE AMBITIOUS AND USE AI TO RE-WIRE YOUR PROCESSES AND YOUR ORGANIZATION’S PROCESSES”

OBSERVED PROBLEMS IN ENGINEERING, OPERATIONS & IT

TEAMS STRUGGLE TO MAKE PROGRESS

ESCALATIONS ARE THE NORM

REWORK OCCURS OFTEN

MEETINGS, MEETINGS, RE-MEETINGS

PROJECT MANAGERS OUTNUMBER WORKERS

AI WILL SOLVE EVERYTHING!

THE DEVOPS AND AI COORDINATION COST JOURNEY

**THE PHYSICS OF COORDINATION COSTS
& THE GOLDEN RULE OF DEPENDENCIES**

THE 3 LAYERS AND REWIRING ORGANIZATIONS

HOW DEVOPS SOLVED THE PROBLEM

HOW AI CAN REDUCE COORDINATION COSTS

AI PROCESS ENGINEERING & AI ENABLEMENT PROPERTIES

AI SILVER BULLET AND CULTURAL CHANGE

THE PHYSICS OF COORDINATION COSTS: DEPENDENCIES KILL

WaitTime	(%BUSY)/(%IDLE)	→ Phoenix Project					
CoordinationRisk	1 in 2^n	→ Troy Magennis <table><tr><td>0 100%</td><td>1 50%</td><td>2 25%</td><td>3 12%</td><td>4 6%</td></tr></table>	0 100%	1 50%	2 25%	3 12%	4 6%
0 100%	1 50%	2 25%	3 12%	4 6%			
KnowledgeLeft	1/(2^n)	→ Mary + Tom Poppendieck, Jon Smart <div><div>50%</div><div>25%</div><div>13%</div><div>6%</div></div> Tacit knowledge loss with handoffs					

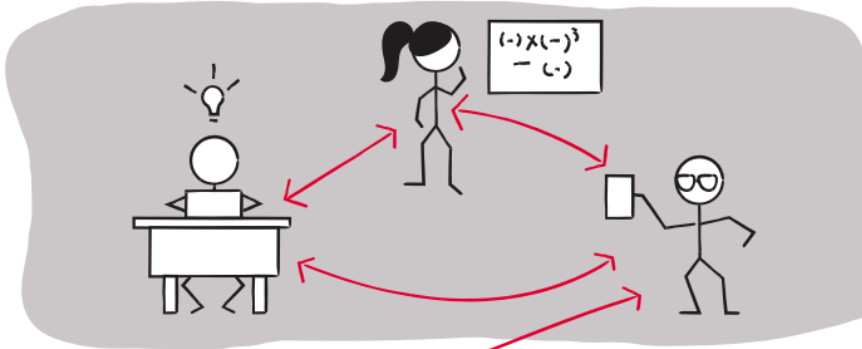
THE 3 C'S OF COORDINATION COSTS: CONTENTION, COUPLING, COHERENCE
→ MAKE IT UNLIKELY YOU WILL ARRIVE ON TIME WITH QUALITY ←

GOLDEN RULE:
REMOVING A DEPENDENCY DOUBLES YOUR ODDS



THE 3 LAYERS & REWIRING ORGANIZATIONS

LAYER 3
SOCIAL CIRCUITRY
FOR FLOW OF IDEAS
AND INFORMATION



LAYER 2
TOOLS AND
INSTRUMENTATION



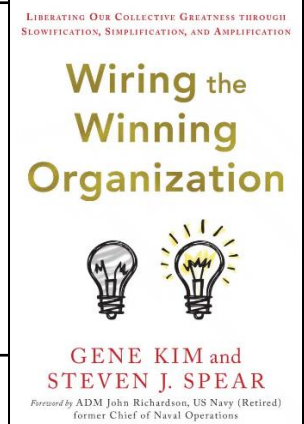
LAYER 1
TECHNICAL OBJECT



Organizational Architecture
System Architecture
Process Architecture
Information Flow / Ideas
Behavioral Norms

IDE
Version Control/CI/CD
IAC/Telemetry
Work Tracking
CoPilot & AI Assistance

Developers
Architects
Testers
"The Code"



THE 3 LAYERS & REWIRING ORGANIZATIONS: LAYER BAND-AIDS

Value Stream Flow Analysis: Common SDLC process

LAYER 3
SOCIAL CIRCUITRY
FOR FLOW OF IDEAS
AND INFORMATION

→ 281				
Fuzzy	Reqs	Solution	Dev	Install
93	41	34	65	48
33%	15%	12%	23%	17%

Developers are slow!
Let's add AI!!

If we make developers faster/more efficient
what is the best result, we can hope for?

$$\underline{50\% = 16 \text{ days} \rightarrow 281 - 16 = 265 = 6\%}$$

4hr/day in IDE(50%)
32 days coding
12% total lead time coding

AKA

Copilot might help
but it won't save you!

“Misattribution
of system problem”



developer

LAYER 2
TOOLS AND
INSTRUMENTATION

LAYER 1
TECHNICAL OBJECT

AI CAN IMPROVE INDIVIDUAL EFFECTIVENESS

AI ALONE WON'T REWIRE YOUR ORG

**YOU NEED TO APPLY SYSTEM THINKING TO YOUR
ORG AND PROCESSES**

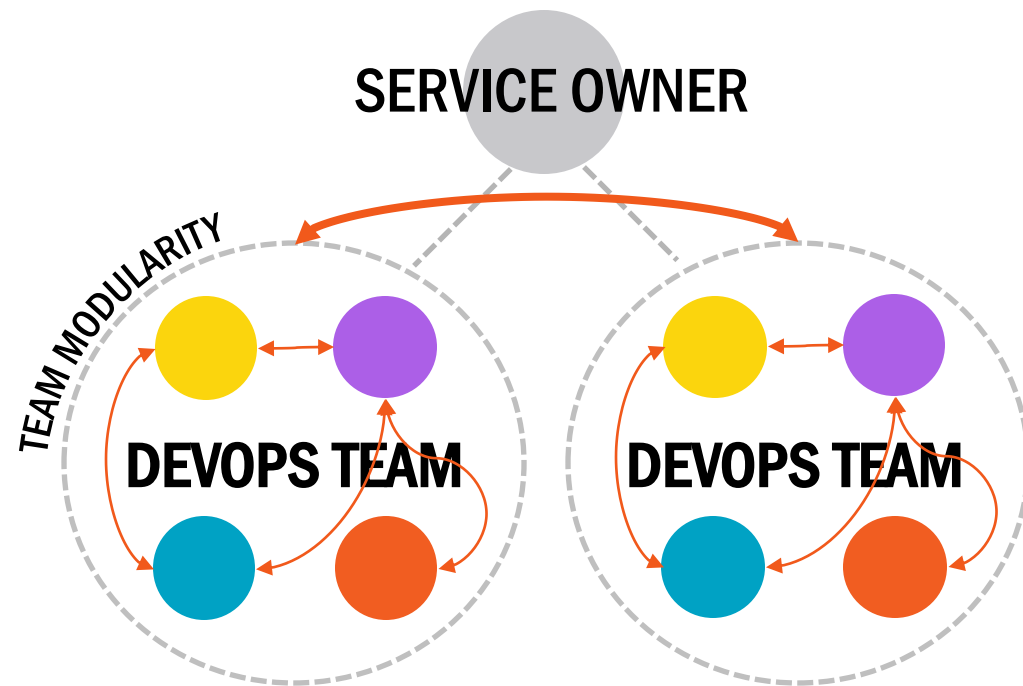
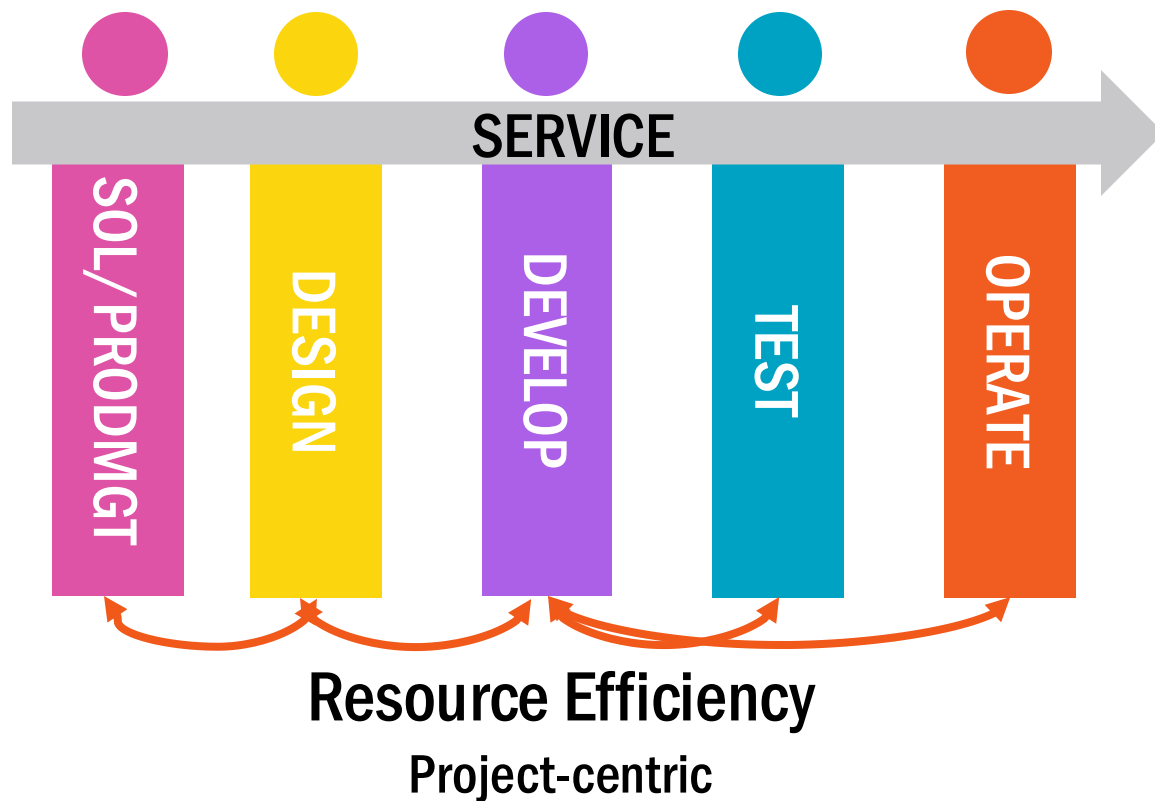
**“AI CAN ANNIHILATE COORDINATION COSTS
BETWEEN DEVS AND THE REST OF YOUR ORG”
– DR DANIEL BROCK**

HOW DEVOPS SOLVED THE PROBLEM

Technique	Results
LAYER 1-2 Technical Practices	Technical investments in CI/CD, testing, observability and cloud lowered variability and increased speed and quality.
LAYER 1-3 Work Practices	A focus on measurement, work visibility, small batches, feedback from production and customers greatly improved decision making, responsiveness and outcomes.
LAYER 2-3 Platform Enablement	Platform engineering lowered cognitive load and created self service at scale. This allowed teams to linearize and parallelize their work and focus on their business domain.
LAYER 2-3 Team Structure	Truly cross functional “T-Shaped” teams focused on the build → run lifecycle dramatically lowered coordination costs and reduced overall lead-time to deliver value.

The techniques in DevOps lowered coordination costs by re-wiring organizations at LAYER 2 and LAYER 3 by slowing down to invest in simplification and then amplified the results across organizations and the industry.

SUPERCHARGING FLOW AND KNOWLEDGE: DEVOPS



Flow & Knowledge Efficiency
Organized by product/service not projects

LAYER 1-2	Technical Practices
LAYER 1-3	Work Practices
LAYER 2-3	Platform Enablement
LAYER 2-3	Team Structure

DON'T PUT THE AI CART BEFORE THE CI HORSE



INTEGRATING AI IS SOFTWARE
ENGINEERING

TO BE GREAT AT THIS, YOU NEED
TO BE GREAT AT DEVOPS & CI

THIS IS YOUR SAFETY NET!

*“AI MAGNIFIES THE STRENGTHS OF
HIGH PERFORMING ORGS AND THE
DYSFUNCTIONS OF LOW-PERFORMING
ONES” – DORA 2025*

**THE FLOW AND SPEED OF WORK IN MOST ORGANIZATIONS
IS DOMINATED BY WAIT TIME AND COORDINATION COSTS**

**THESE COSTS OCCUR WHEN WAITING ON OTHERS FOR
CAPABILITY, KNOWLEDGE OR TASKS**

**USING AI TO REWIRE THE ORG TO REDUCE THESE COSTS
CAN HAVE AN OUTSIZED BENEFIT: 10X+ IMPROVEMENTS
IN LEADTIME AND QUALITY**

THE AI ENABLEMENT PROPERTIES V2.0

Property		Definition
FAST, FUN, AMBITIOUS, AUTONOMOUS	Knowledge	<i>Improve knowledge assimilation, management, and reasoning. Use AI to AMPLIFY and democratize knowledge and hone accuracy on conversations.</i>
	Capability	<i>Improve your capability and desire to do things you wait on others for. Turn yourself into an AI team.</i>
	Capacity	<i>Improve your ability to do work faster. Capability and capacity enablement with AI allow an individual to do the work of many.</i>
	Parallelism	<i>AI can be in multiple places at once creating parallel independence of action and removes COUPLING and facilitates SELF SERVICE.</i>
	Optionality	<i>Use AI to experiment, work incrementally and exploit optionality quickly.</i>

The flow and speed of work in most organizations is dominated by wait time and coordination costs.
Use AI to rewire the org to reduce these costs.

THE 3 DIMENSIONS OF ARCHITECTURE & SIMPLIFICATION

REMOVING A DEPENDENCY DOUBLES YOUR ODDS



LAYER 3 ARCHITECTURE IS YOUR TOOL

**LET'S RE-WIRE OUR
PROCESSES & THE ORG!**



ORGANIZATIONAL ARCHITECTURE

SYSTEM ARCHITECTURE

PROCESS ARCHITECTURE

SLOWIFICATION

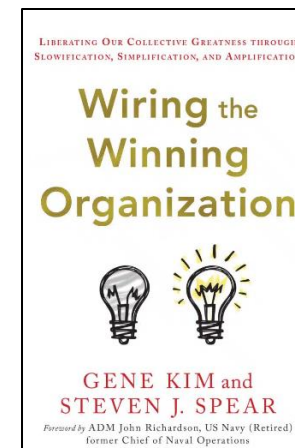
SIMPLIFICATION

MODULARIZATION

INCREMENTALISM

LINEARIZATION

AMPLIFICATION



APPLYING AI ENABLEMENT: CLIENT REQUIREMENTS ANALYSIS

Problem: Collecting client requirements and subsequent analysis is time consuming, inaccurate and requires handoffs between many groups.



Pretend you are a Cloud Migration Expert and create me a summary for a compelling presentation highlighting: Current State and Risks, Future State, Suggested Approach and Business Justification, Timeline and Approximate Costs to execute a migration to public cloud.

Please provide me a summary of data center renewal dates and contract terms to be aware of.

Please provide me a summary of DR and backup issues the client has.

Please generate me a document from this conversation that I can use as the outline to create the final deck for the client.

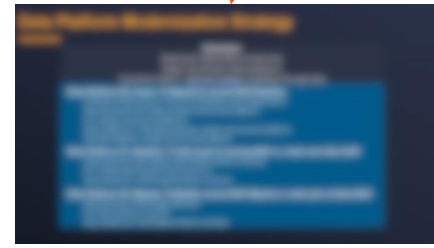
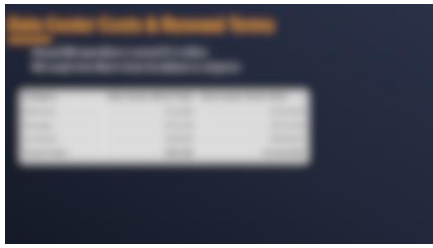
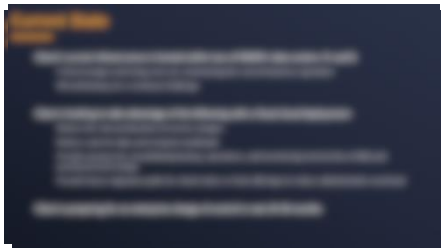
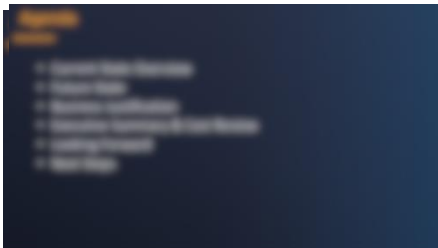


DOCS
NOTES
CONTRACTS

500+ PAGES

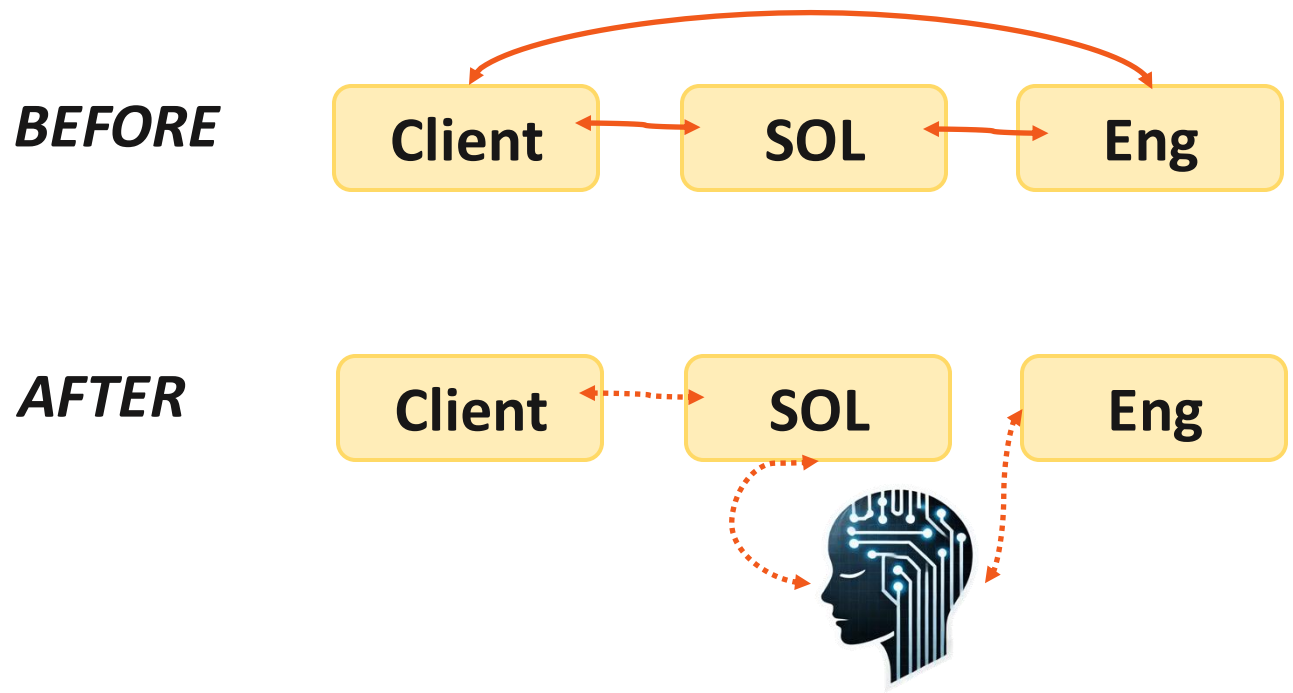


18 PAGES



Knowledge, Capability, Capacity, Parallelism

APPLYING AI ENABLEMENT: CLIENT REQUIREMENTS ANALYSIS



LeadTime: 10w → 2w

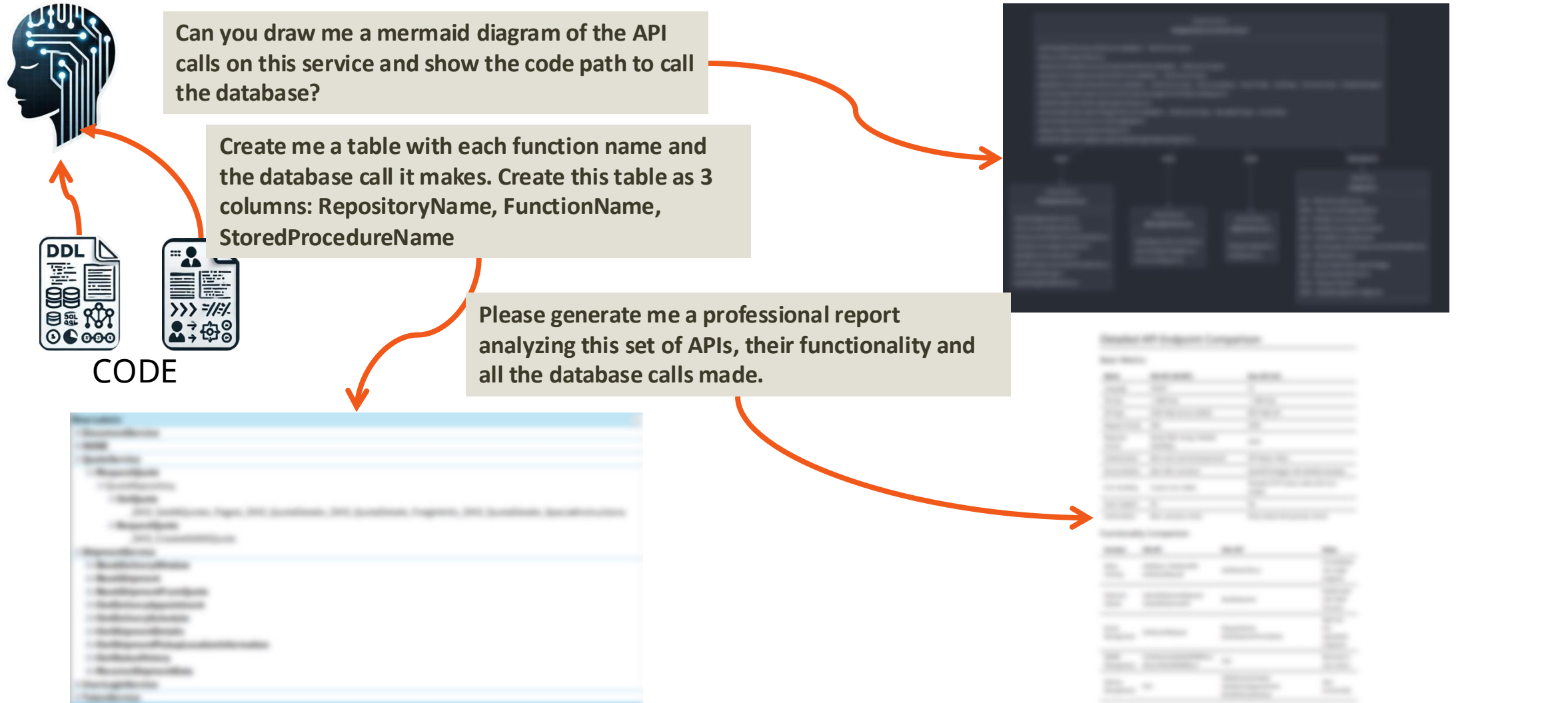
AI cleans up messy conversations and disparate docs.

AI enables independence of action!

Depends	Risk	Cost(h)	LT
3	1 in 8(12%) no delay	90	10w
1	1 in 2(50%) no delay	12	2w
3X	4x	7x	5x

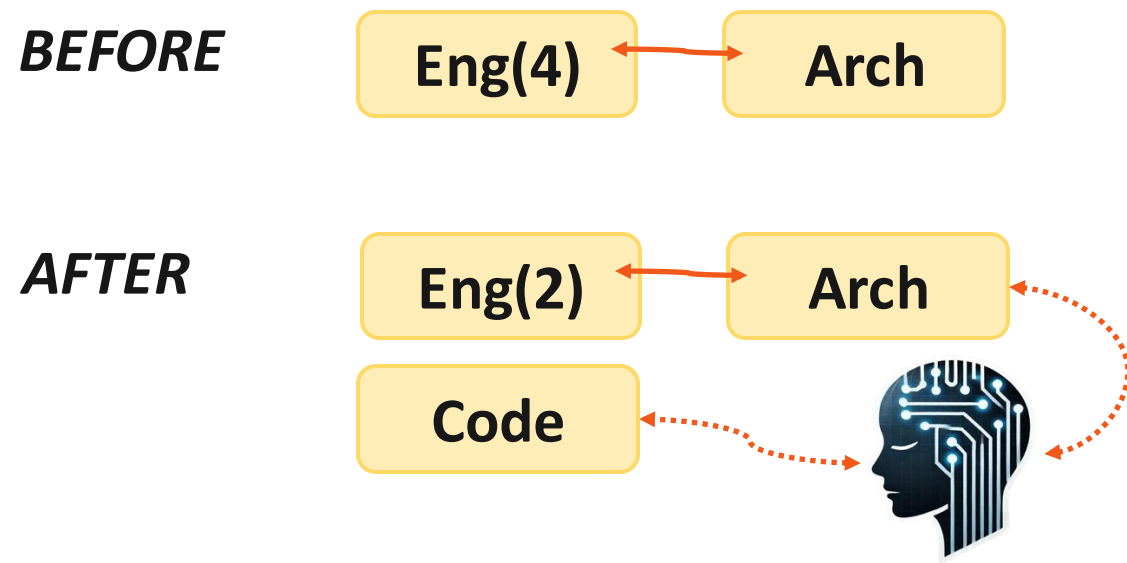
APPLYING AI ENABLEMENT: LEGACY SYSTEM ANALYSIS

Problem: Legacy system analysis required for new integrations. Traditional approaches require heavy coordination costs with key stakeholders.



Knowledge, Capability, Capacity, Parallelism

APPLYING AI ENABLEMENT: LEGACY SYSTEM ANALYSIS



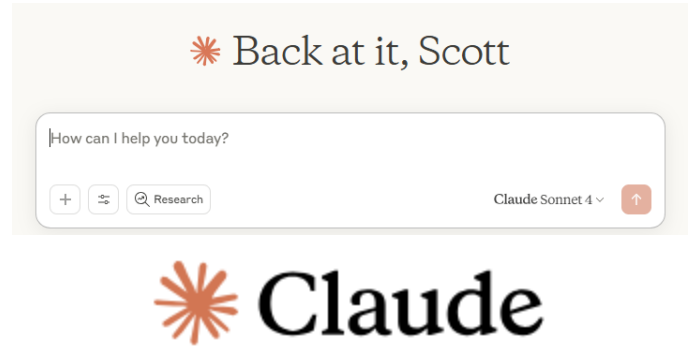
Cost(h): 50h → 5h

AI cleans up messy conversations.

AI code analysis is vastly superior to humans.

Depends	Risk	Cost(h)	LT
4	1 in 16(6%) no delay	50	10w
2	1 in 4(25%) no delay	5	2w
2X	4x	10x	5x

“Who is this guy, Claude?”
-David Rizzo



Knowledge, Capability, Capacity, Parallelism

APPLYING AI ENABLEMENT: E2E POC

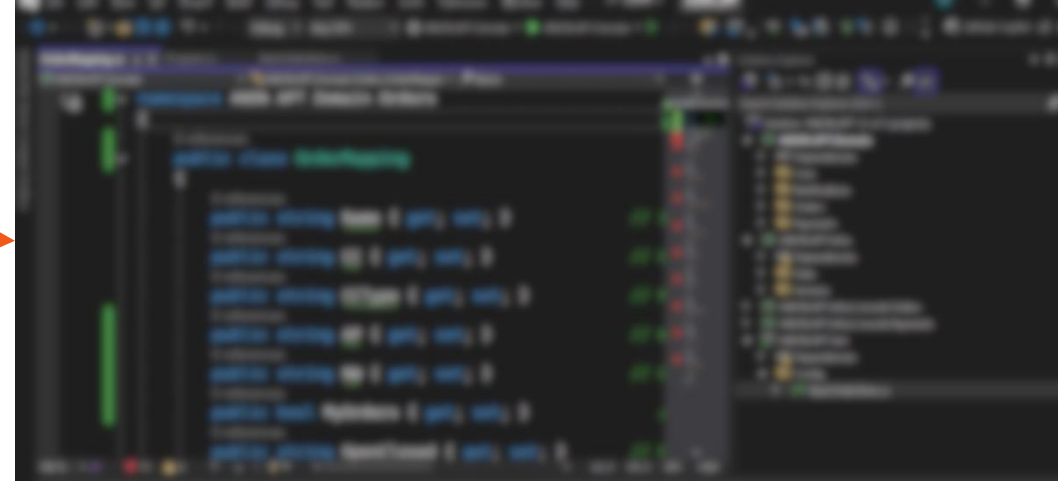
Problem: Building solutions and proof of concepts is costly in time and \$. Traditional approaches require coordination across many stakeholder and roles.

I've given you some sample database schemas of our current system and the API documentation of 3 systems we need to integrate with.

Please generate me a solution in C# that maps our orders to payments and then sends customer notifications via email and SMS.

Please add a set of tests that mock the 3 integrations and return test data for each one.

Please generate a dockerfile and OpenTofu to configure this solution on ECS.

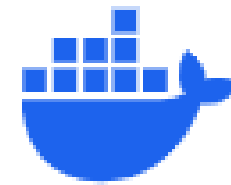


SCHEMAS



3 INTEGRATIONS

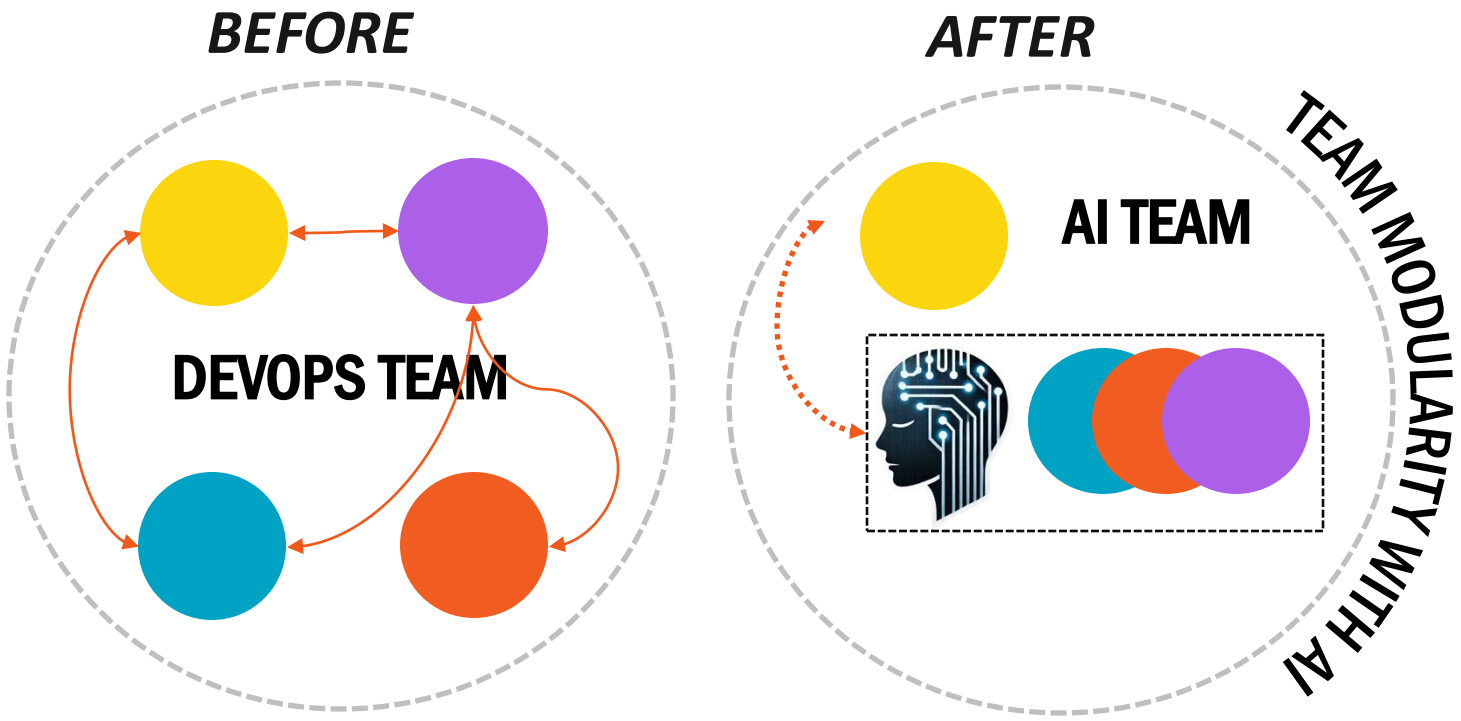
WSDL + OPENAPI



Knowledge, Capability, Capacity, Parallelism, Optionality

APPLYING AI ENABLEMENT: E2E POC

Knowledge, Capability, Capacity, Parallelism, Optionality



LeadTime: 12w → 2w

AI loves code & types way faster than you do.

Idea, code, test, deploy loop greatly collapsed.

Depends	Risk	Cost(h)	LT
4	1 in 16(6%) no delay	480	12w
1	1 in 2(50%) no delay	48	2w
4X	8x	10x	6x

AI: THE SILVER BULLET → WARNINGS AND BENEFITS

BEWARE OF SNAKE OIL:

WE DON'T NEED DEVS, JUST AI!

GIVING DEVS AI IS A SILVER BULLET TO PRODUCTIVITY!

BUT UNDERSTAND THE POWER:

AI MAKES EXPERTS DRAMATICALLY STRONGER

TECHNICAL MAESTROS AND ARCHITECTS JUST GOT BOOSTED

PEOPLE GREAT AT AI WILL WIN THE JOB WARS

AI LOVES CODE AND TYPES WAY FASTER THAN YOU

MOVE QUICKLY FROM CHOP TO THE AGENTS – HAVE THE IDE TALK

FINAL CAUTIONS AND THOUGHTS:

DON'T REMOVE YOUR CRITICAL THINKING

AI MOVES COGNITIVE LOAD IN WEIRD WAYS

GREAT ADVANCES LIE IN AUGMENTATION VS REPLACEMENT

SUMMARY: TACKLING COORDINATION COSTS

SLOWDOWN: CHANGE YOUR(YOU AND ORG) BEHAVIOR TO EMBRACE AI

THE PHYSICS: COORDINATION COSTS DEGRADE EXPONENTIALLY

THE GOLDEN RULE: REMOVING A DEPENDENCY DOUBLES YOUR ODDS

AI PROCESS ENGINEERING BEFORE AI PRODUCT ENGINEERING

REWIRE L2 & L3 OF YOUR ORGANIZATION TO EMBRACE AI
AI ENABLES INDEPENDENCE OF ACTION

DON'T SHORTCUT CI & DEVOPS FOR AI

BATTLE COORDINATION COSTS WITH AI ENABLEMENT

KNOWLEDGE, CAPABILITY, CAPACITY,

PARALLELISM,

OPTIONALITY

WHAT DOES THE NEW AI/AGENTIC ORG STRUCTURE LOOK LIKE?

