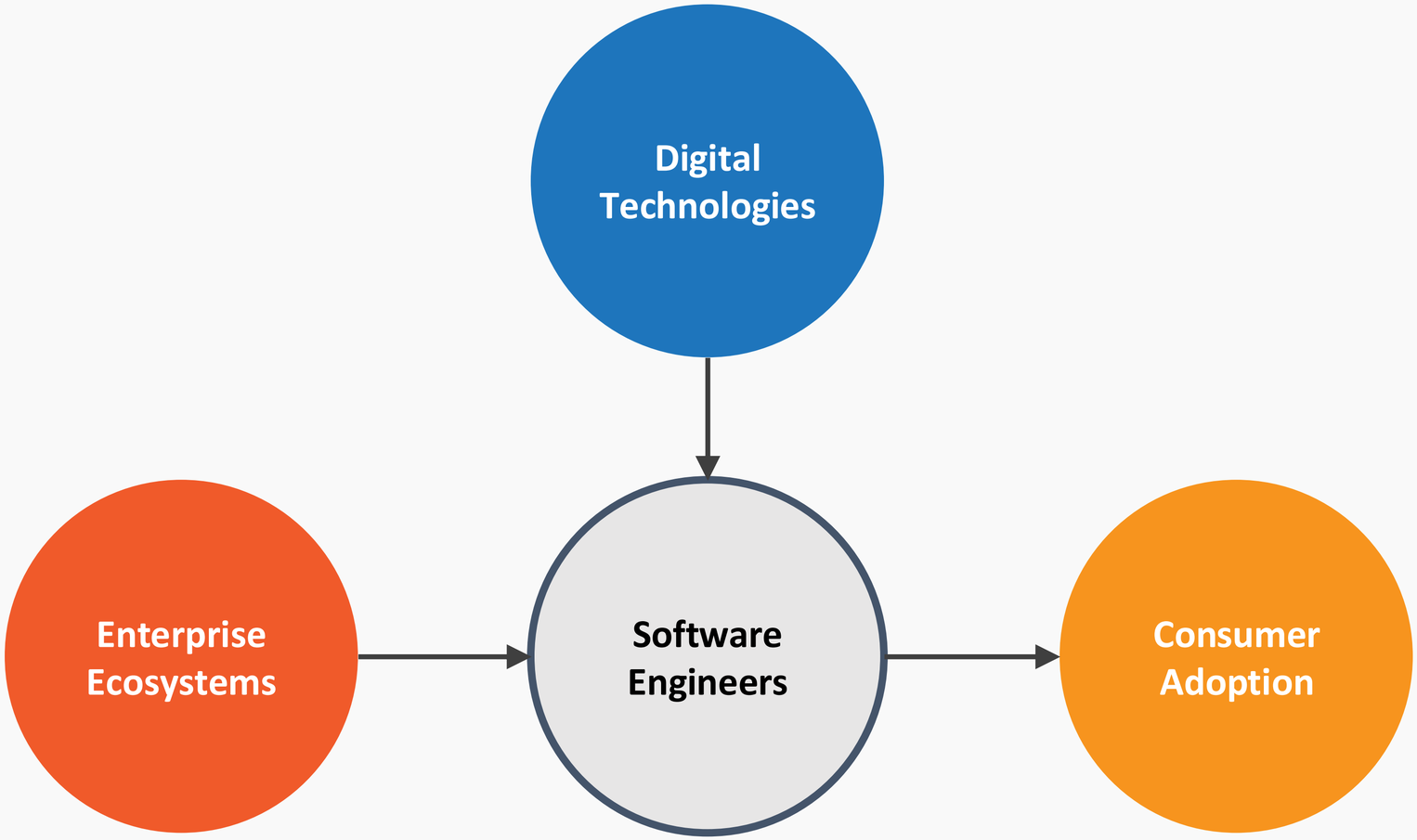
INNOVATION SUMMIT 2025

Ashutosh Bijoor
CTO, Accion Labs

Are Software Engineers Redundant?

Introducing
The Semantic Engineer

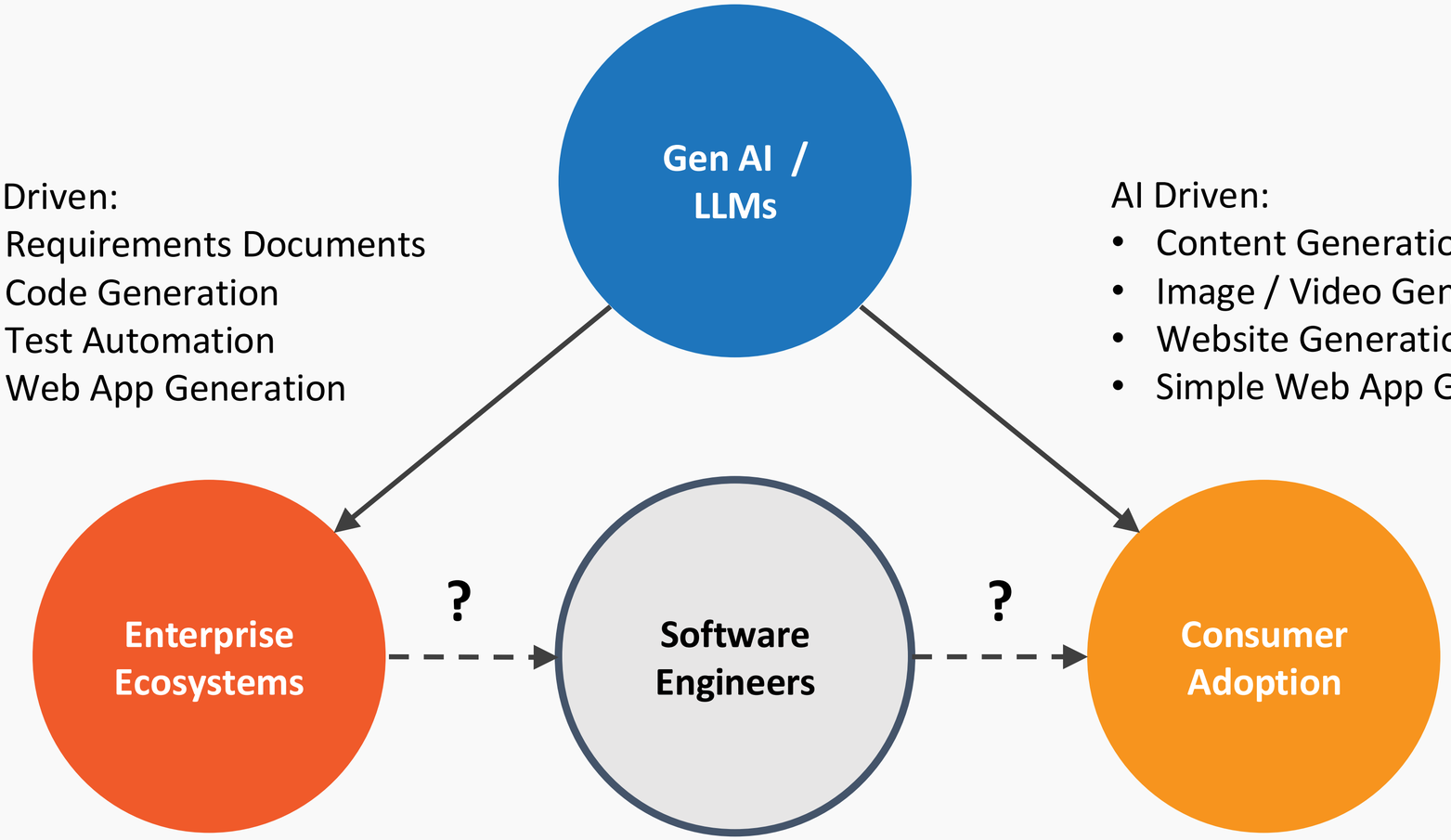
The Role of Software Engineers



Software engineers are the crucial intermediaries translating technology into enterprise solutions for consumers

The Emergence of Gen AI / LLMs

AI Driven:
- Requirements Documents
- Code Generation
- Test Automation
- Web App Generation

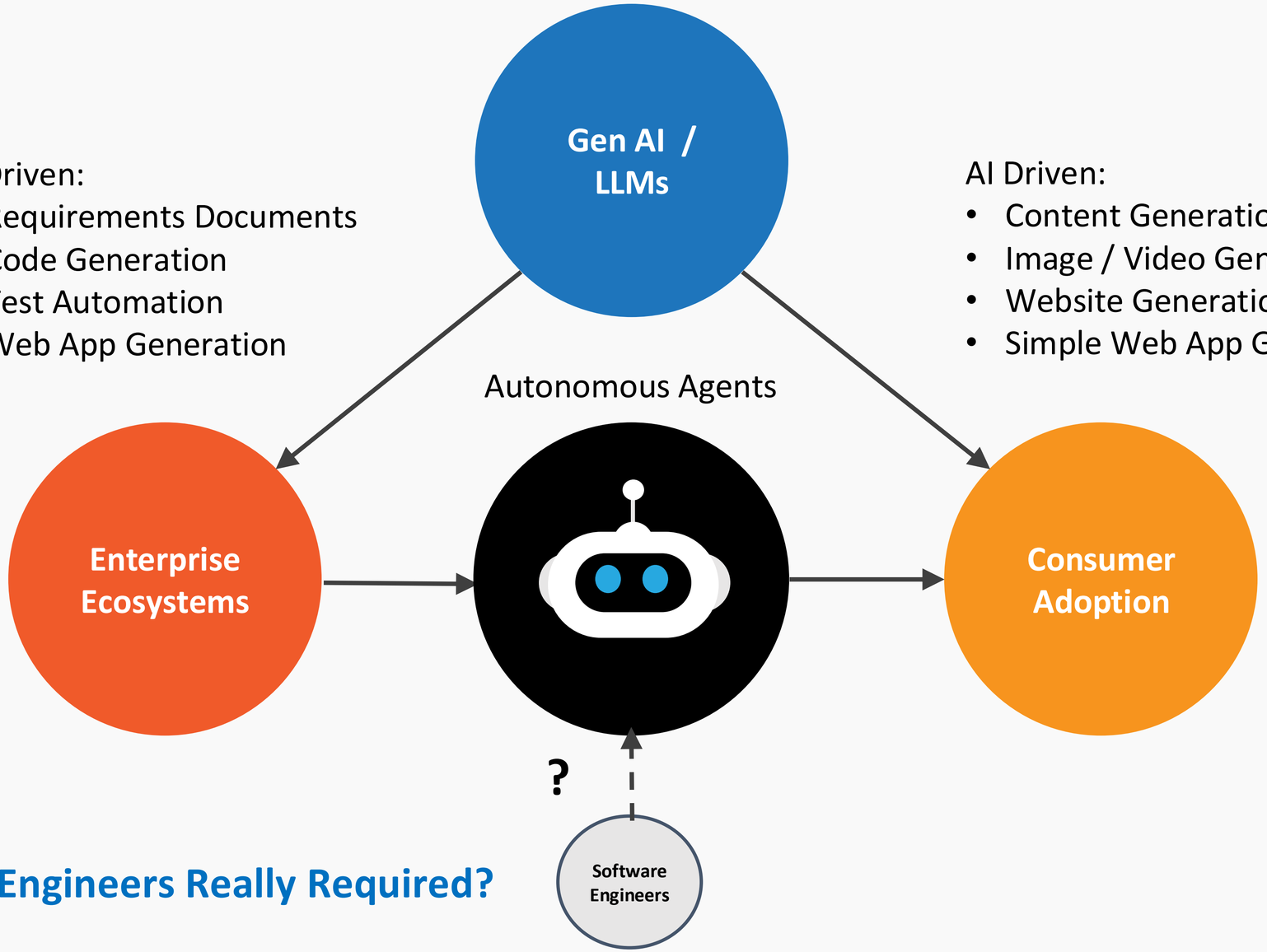


AI Driven:
- Content Generation
- Image / Video Generation
- Website Generation
- Simple Web App Generation

Are Software Engineers Disintermediated?

The Emergence of AI Agents

AI Driven:
- Requirements Documents
- Code Generation
- Test Automation
- Web App Generation



AI Driven:
- Content Generation
- Image / Video Generation
- Website Generation
- Simple Web App Generation

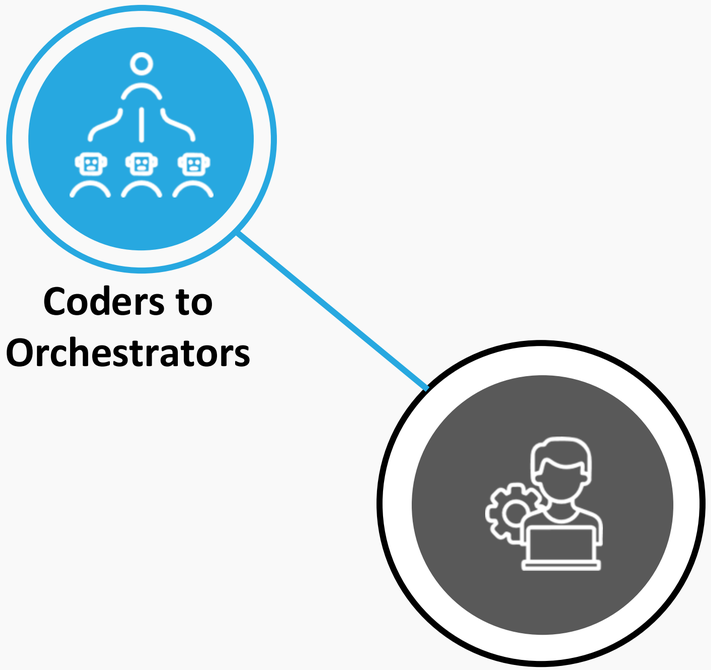
Are Software Engineers Really Required?

	Web Era 1990s - early 2000s	Mobile & Cloud 2000s - early 2010s	Platform Economies 2010s - early 2020s	Early AI 2020s - 2023	Agentic AI 2024 - 2030
Digital Technologies	**Static Information** HTML, CSS, JS; document-based web	**Interactive Applications** APIs, web services, responsive interfaces	**Intelligent Systems** Big data analytics, recommendation engines, open-source ML	**AI / ML / NLP / Gen AI** NLP interfaces, content & code generation, multimodal understanding	**Autonomous Agents** Multi-agent orchestration, reasoning systems, self-improvement
Enterprise Ecosystems	**Digital Publishing** Content focused websites, digital brochures, information repositories	**Digital Channels** Omnichannel experiences, integrated customer journeys	**Platform Businesses** API ecosystems, data marketplaces, digital partnerships	**Predictive Enterprises** AI-driven workflows, automated decision systems	**Autonomous Organizations** Self-optimizing value chains, emergent strategies
Consumer Adoption	**Content Readers** Passive information consumers, limited interaction	**Digital Consumers** Intuitive mobile-first customers	**Always-Connected Users** Digitally dependent, seamless interactions	**AI Enhanced Users** Personalized experiences, voice assistants	**Task Delegators** Proactive systems anticipating needs without explicit commands
Software Engineering	**Monolithic Systems** Waterfall process, manual coding, limited reuse	**Component-Based Systems** Agile methods, service-oriented architectures	**Distributed Architectures** DevOps, microservices, automated pipelines	**AI-Augmented Systems** MLOps, model-driven engineering, intelligent testing	**Agentic Ecosystems** Intent-to-code translation, ontology-driven development
Substance Complexity (Feature count)	**Low** Few features, simple relationships, limited scope	**Medium** More features, defined interactions, broader scope	**High** Many features, complex relationships, extensive scope	**Very High** Vast feature sets, intricate interdependencies enterprise wide scope	**Extreme** Unbounded features, emergent capabilities global scope
Dynamic Complexity (Rate of Change)	**Minimal** Static behavior, predictable patterns	**Low** Limited variability, controlled change	**Medium** Significant variability, regular adaptation	**High** Rapid evolution, contextual responses	**Very High** Continuous transformation, unexpected adaptations

The ~~Software~~ Semantic Engineer

Reinventing the role of the software engineer

The Semantic Engineer



Old Paradigm

Writing code to implementing specific functionality

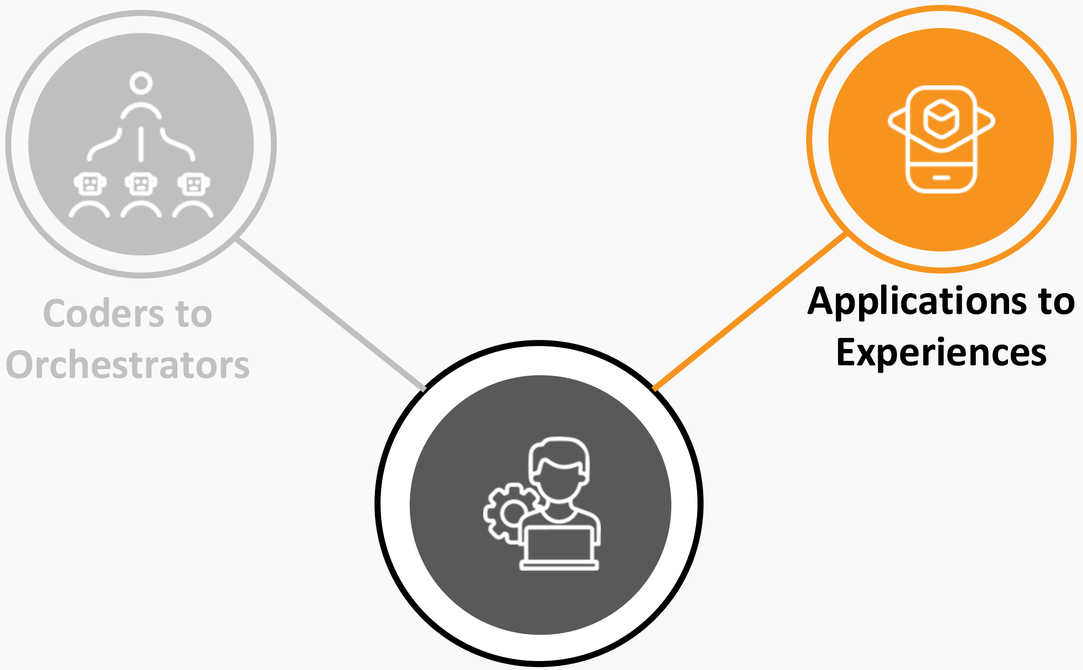
New Paradigm

Defining goals and acceptance criteria for AI to generate the code

Emerging Paradigm

Designing orchestration ecosystems of autonomous agents

The Semantic Engineer



Old Paradigm

Designing applications that solve specific problems

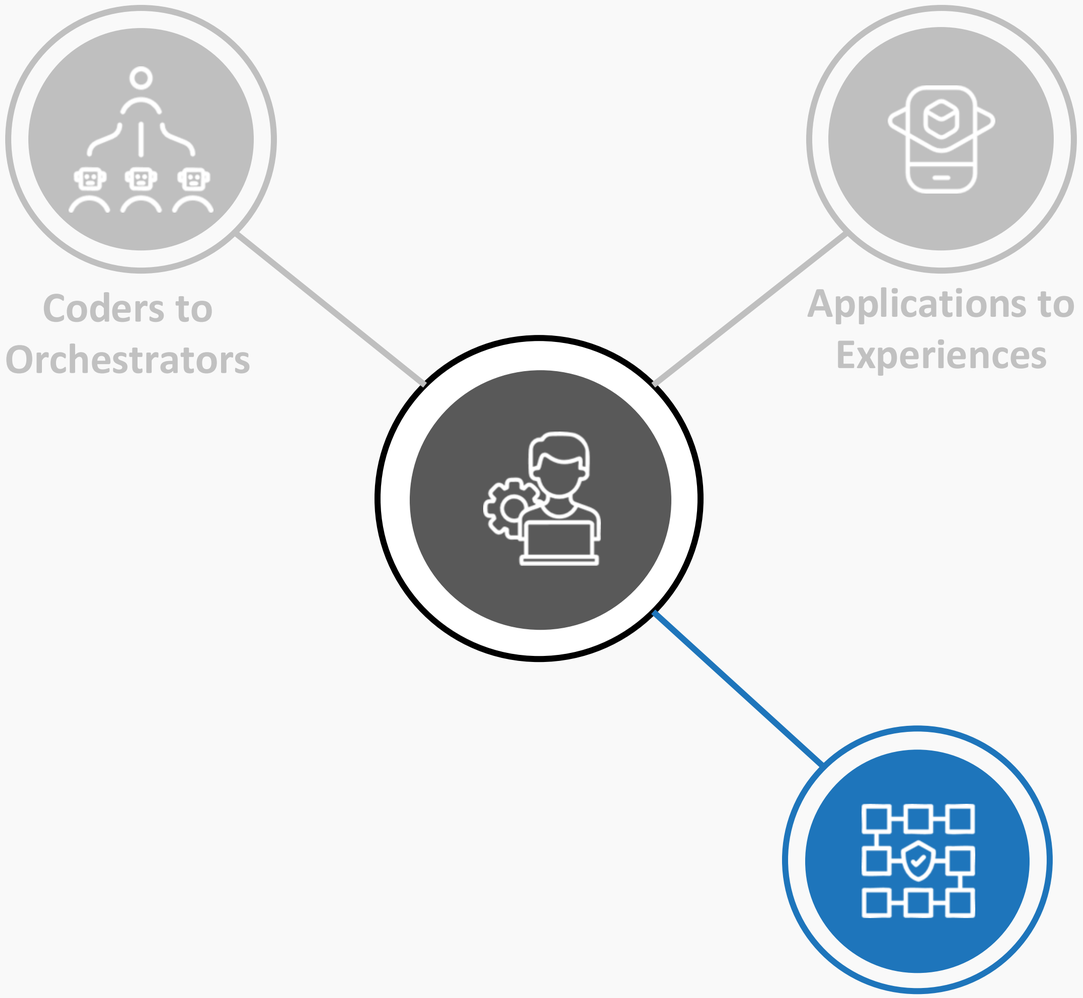
New Paradigm

Designing seamless experiences that address a set of problems

Emerging Paradigm

Designing dynamic experiences that address entire problem domains

The Semantic Engineer



Old Paradigm

Implementing secure features for well-defined risks

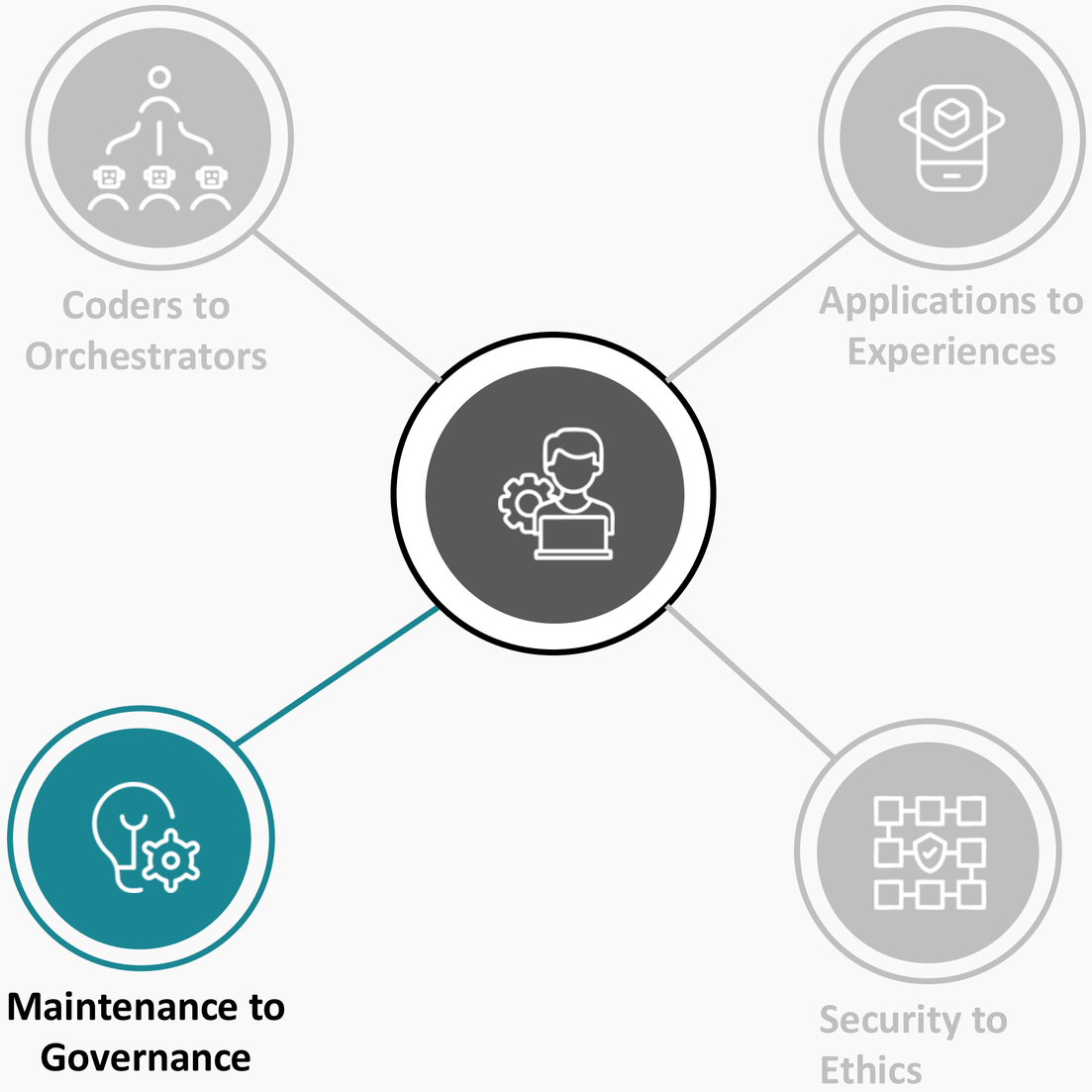
New Paradigm

Designing flexible and secure guardrails within which agents can operate safely

Emerging Paradigm

Creating adaptive ethical frameworks that secure dynamic agent ecosystems

The Semantic Engineer



Old Paradigm

Maintaining solutions for changing requirements

New Paradigm

Managing agents that automatically adapt to evolving needs

Emerging Paradigm

Creating governance frameworks that regulate entire agent ecosystems

The Semantic Engineer

THANK YOU

Accionlabs
INNOVATION
SUMMIT 2025