Proceeding Operations

The Unprecedented Process

Mandate: **Ready Agile Systems**

Ordinance: **Deploy, Twin, Control, Scale, Change (Details, Systems, Duplication, Maintenance)**

Solution: **Mandate Risk Coherence, Requirement Based Assurance, Narrative Control, Exception Prevention, Automation and Anticipation**

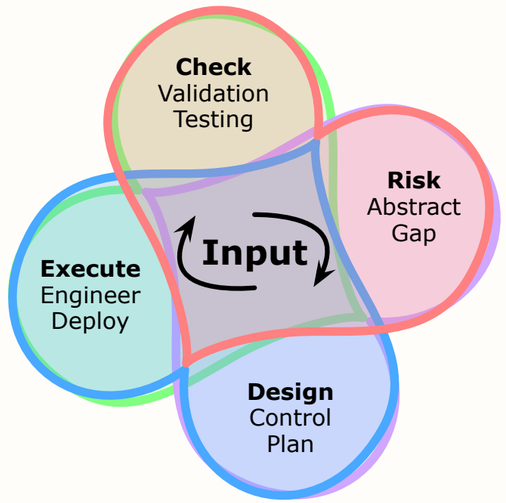
Value: **Efficiency, Improvement, Elastic Scope, Quality Validation and Robustness, Expectedness**

Eliminate Exceptions Success System

Tracing Minimum Objective Plan

Input/Mandate (**Objective**)

Design/Check (**Development**)



Execute/Risk (**Operation**)

Product/Operation Engineering

Outcome Fast-Forward

* **Successive Measurement Improvement . . . . . .**
  (Quantifiable Gain)

* **Proceeding Ongoing Production . . . . . .**
  (Successive Yield Improvement)

* **Production Mandate Improvement . . . . . .**
  (Quality Control / Risk Prevention)

Little's Law Overall improvement is the limit of process development value

Theory of Constraints Unconstrained systems have unlimited performance

Conway's Law Organization engineering twins the systems within the organization

Gold’s Law Nothing goes as planned with contemporaneous production development

Campbell's law Indicators subject to influence, become initiators of change

Cobra Effect A solution attempt that makes the problem worse

Alan Kay Oxymoron

Minimum Adverse / Scientific

Solutions Integration
(Extemporaneous Best)

ABC Evaluation (Risk Adverse)
- Empirical Plan (Crowdsource, Multi-perspective)
- Solution Reduction (Scope/Goal Alignment)
- Last/Loudest Consensus (Best/Sum Reconciliation)

“Quick Work” Engineering (Minimum Viable)
- Operational Control (Objective Definition)
- Tool Alignment (Development)
- Purpose Built (Production)

Minimum Objective
(Production Operations)

Scientific Method (Risk Engineering)

1) Observe	(-> Mandate)
2) Assert	(-> Plan)
3) Hypothesis	(-> Execute)
4) Predict	(-> Check)
5) Test	(-> Risk)
6) Iterate	(-> Product)

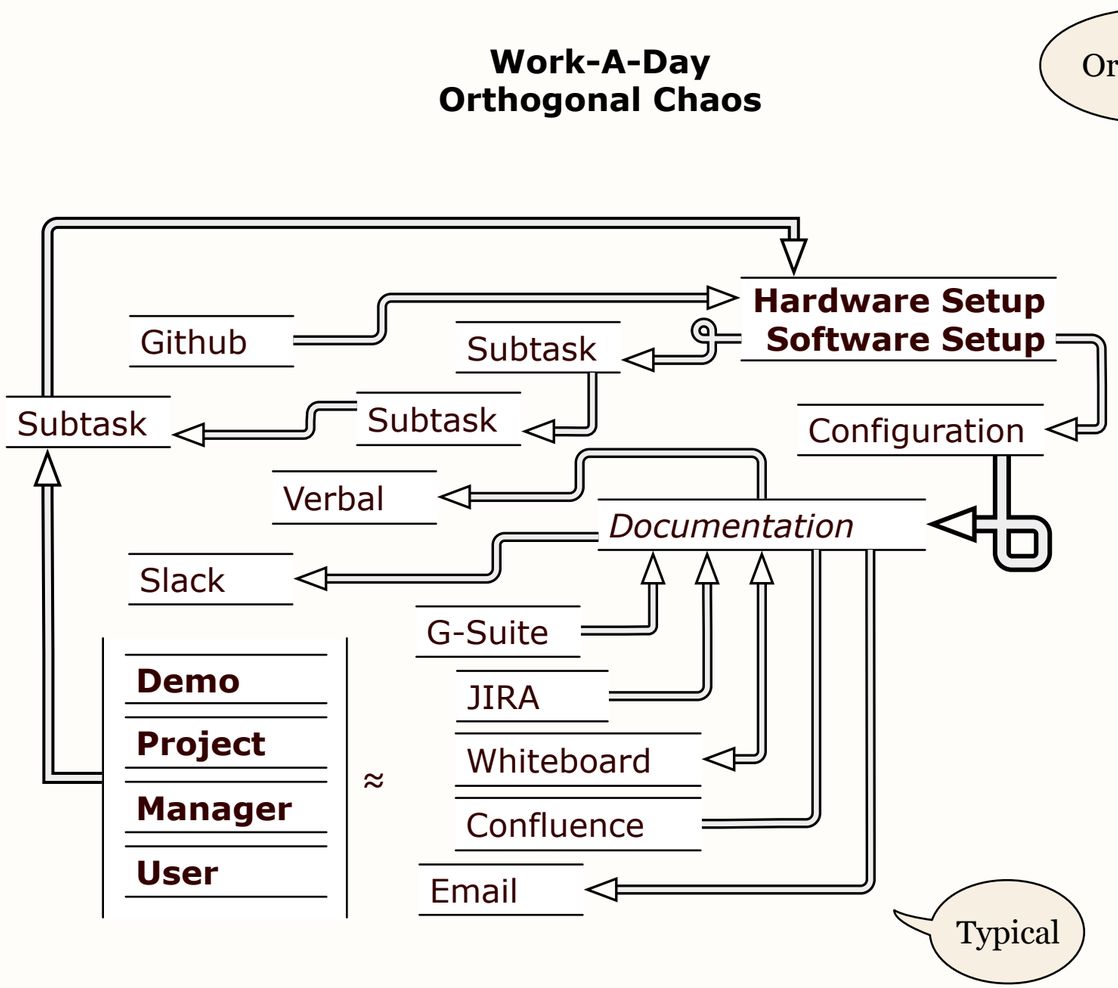
Bootstrap Condition

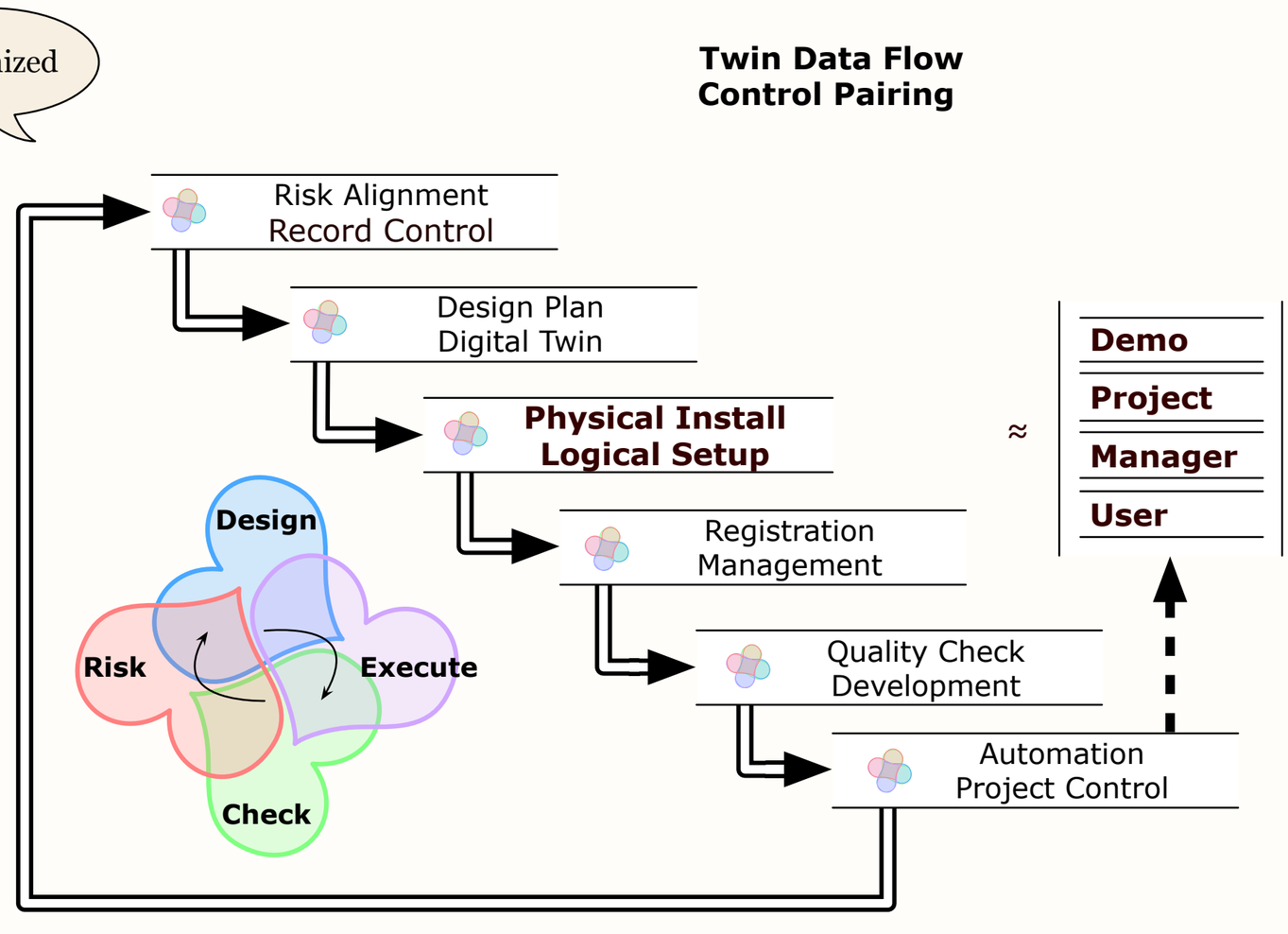
Deployment Evolution

Work-A-Day
Orthogonal Chaos

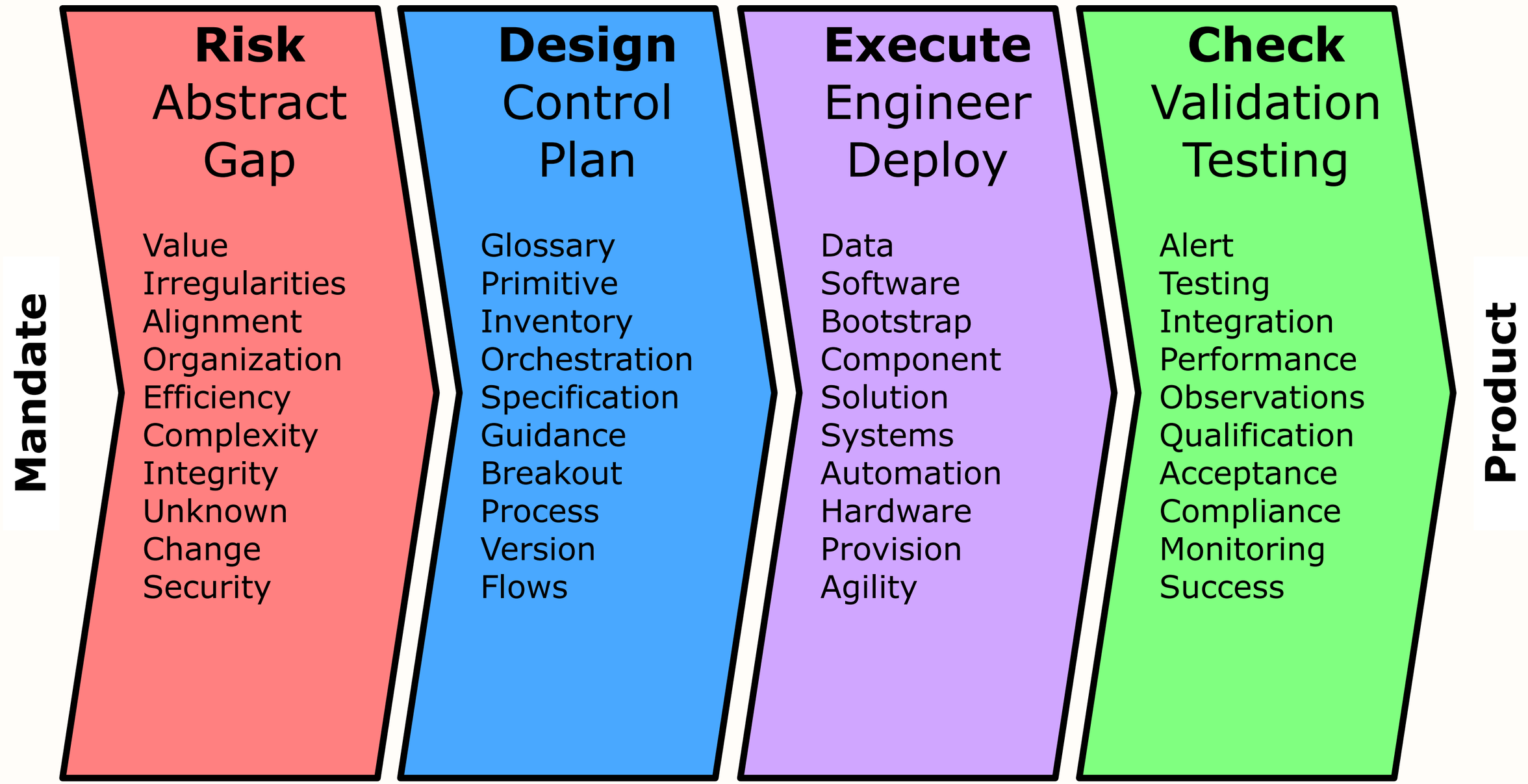
Organized

Twin Data Flow
Control Pairing

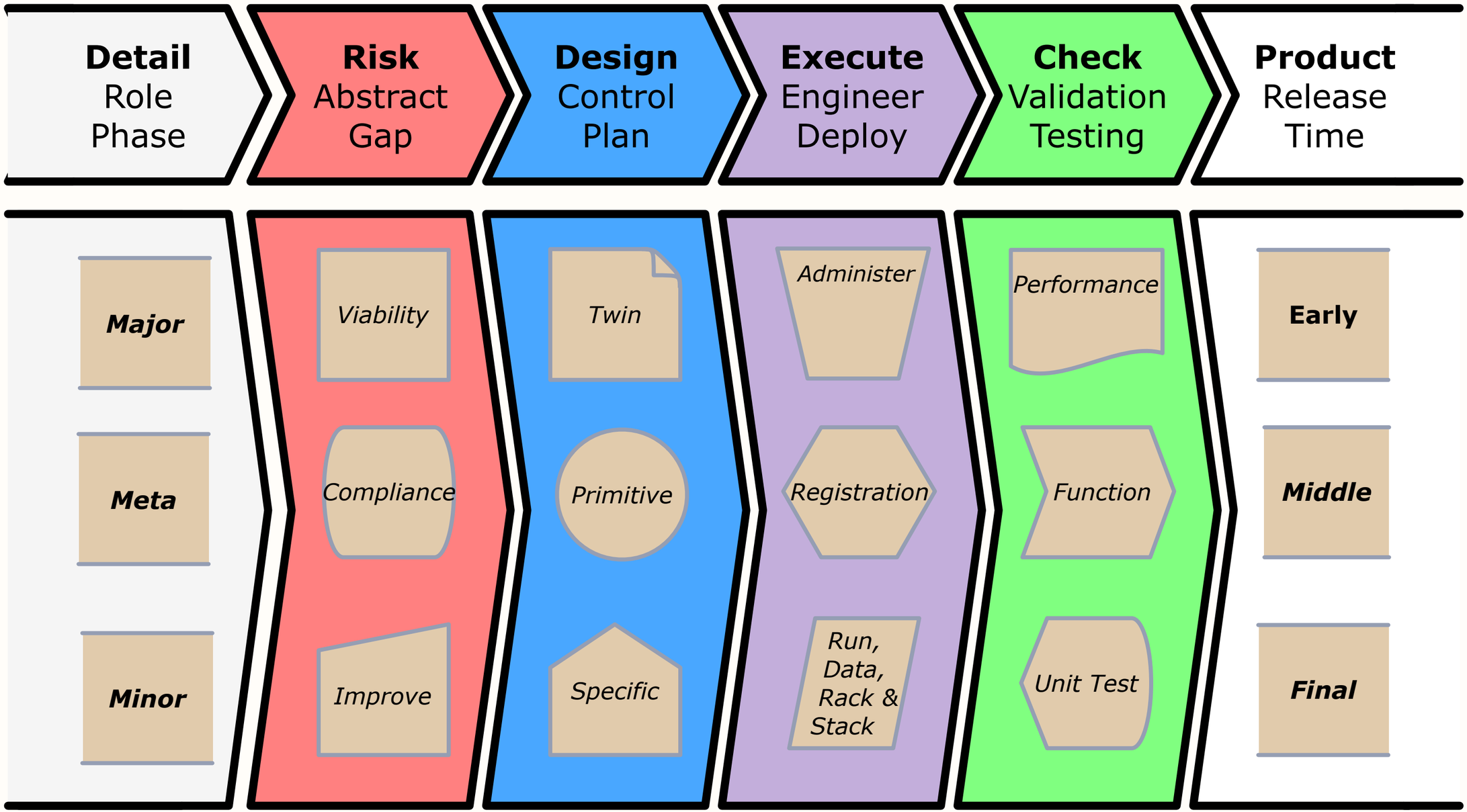




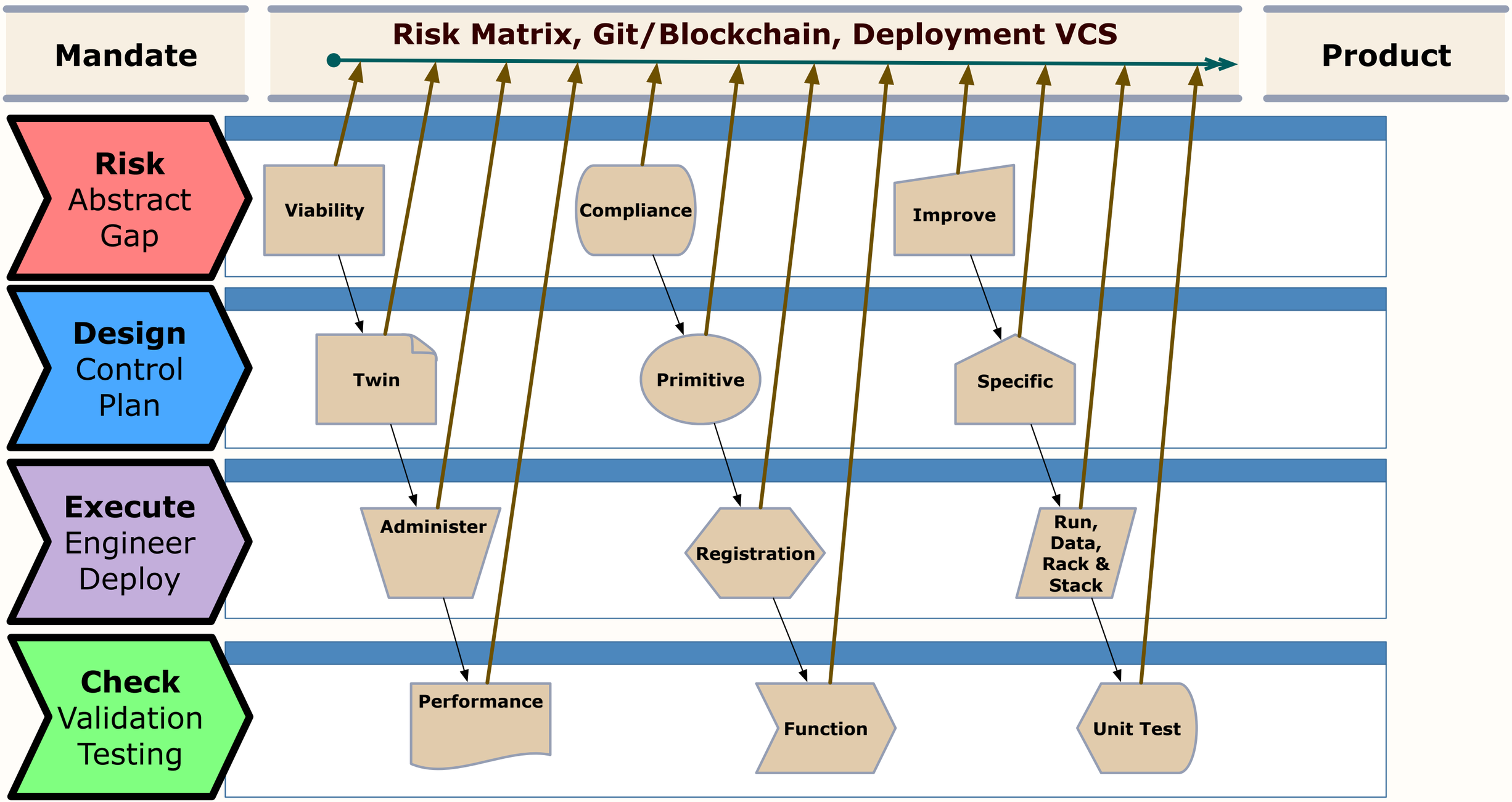
Four Phase Process



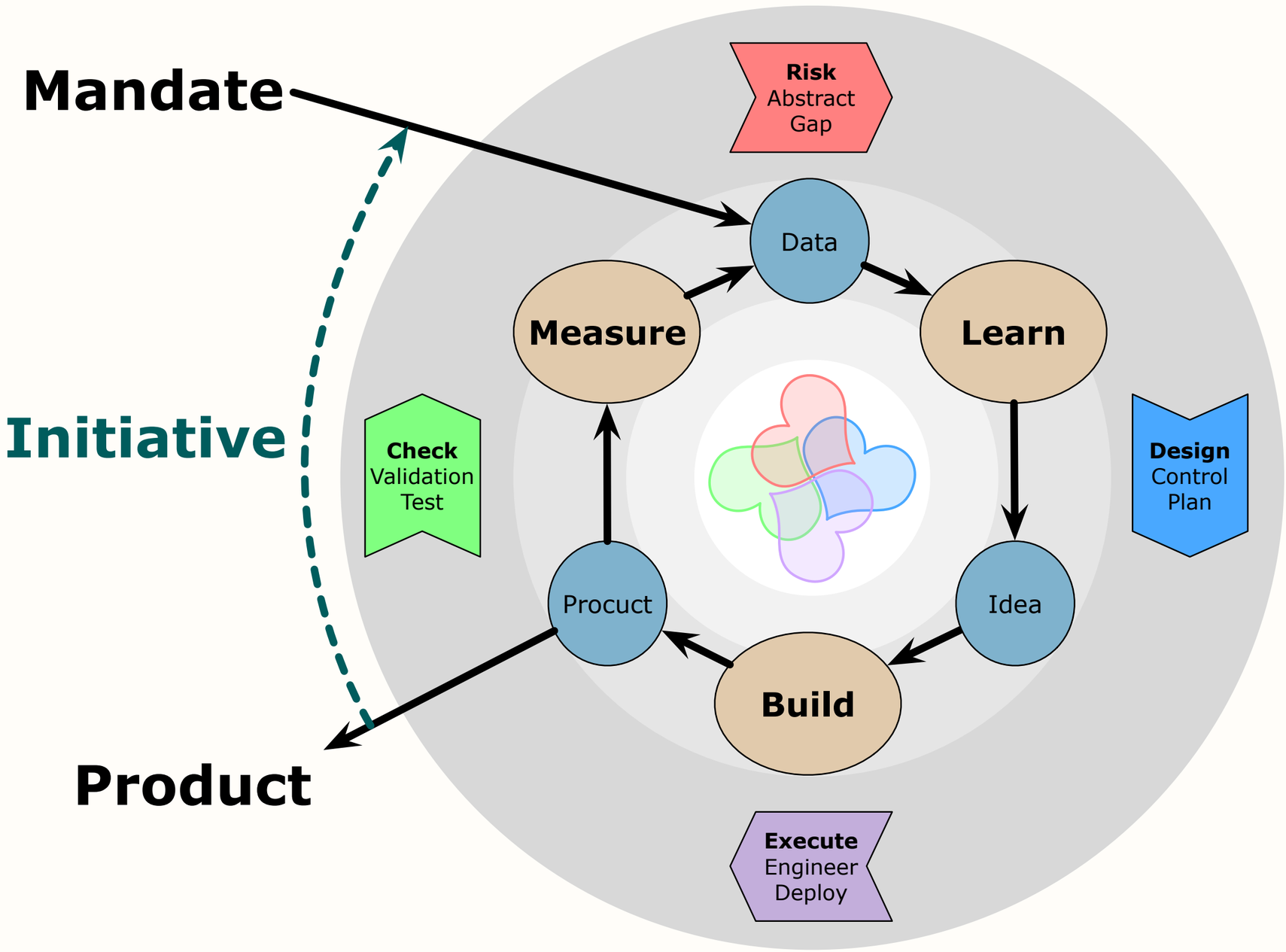
Delivery Controls



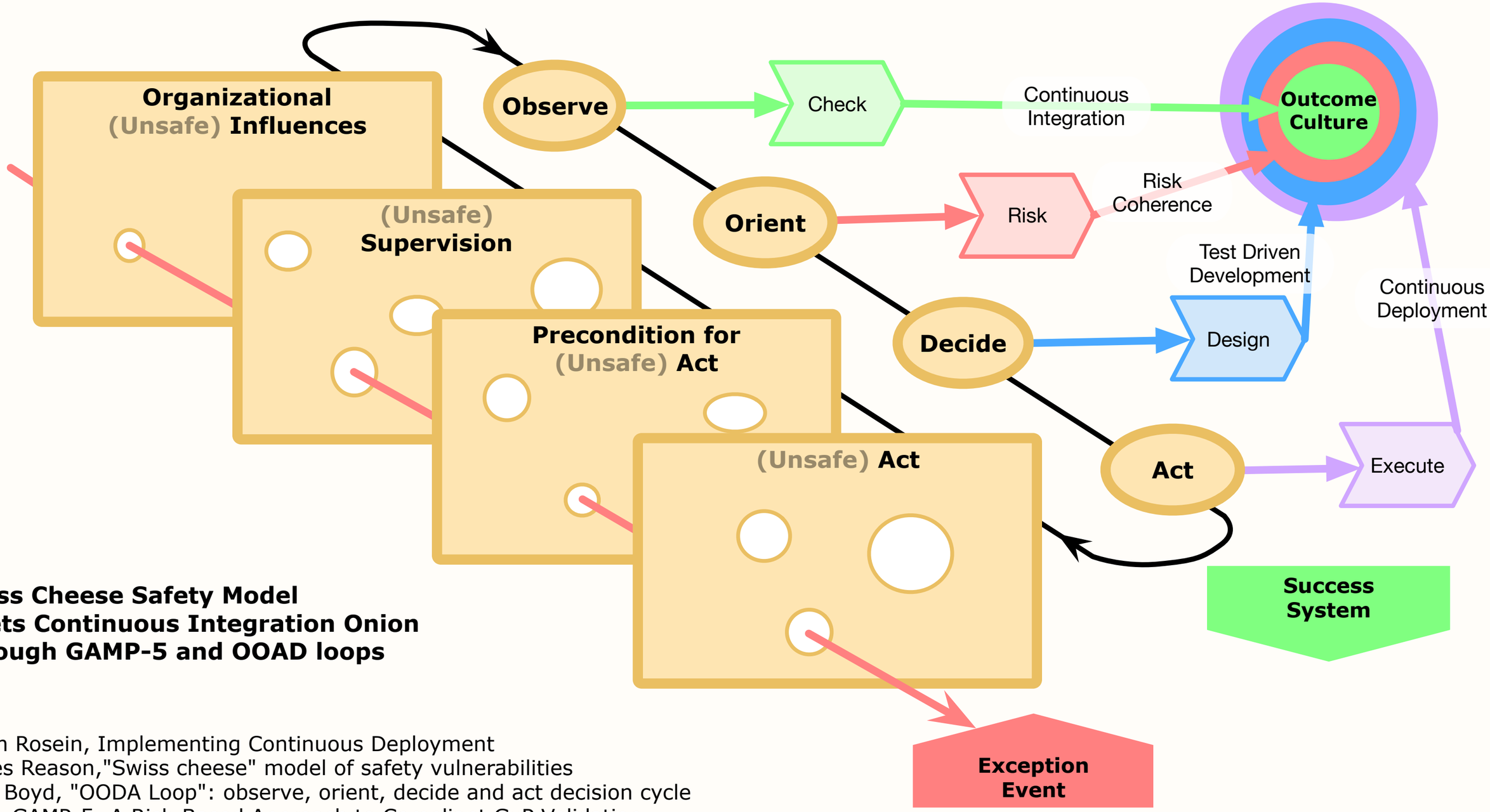
Objective Improvement



Loop Cycle



Unprecedented Cheese and Onions



Swiss Cheese Safety Model
Meets Continuous Integration Onion
Through GAMP-5 and OOAD loops

Adam Rosein, Implementing Continuous Deployment
James Reason,"Swiss cheese" model of safety vulnerabilities
John Boyd, "OODA Loop": observe, orient, decide and act decision cycle
ISPE, GAMP-5: A Risk-Based Approach to Compliant GxP Validation