
Proceeding Operation of the Unprecedented Process

Mandate: **Ready Agile Systems**

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

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

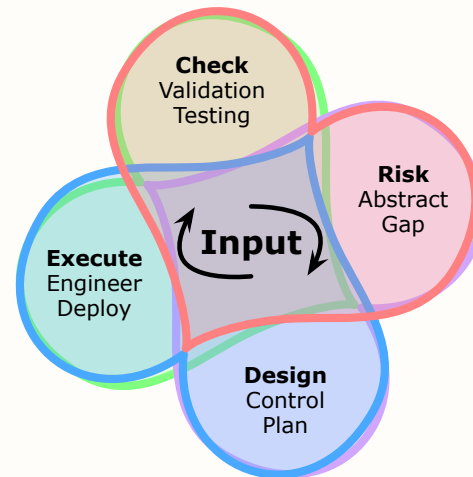
Value: **Efficiency, Improvement, Elastic Scope,
Quality Validation and Robustness, Expectedness**

Eliminate Exceptions Success System (Tracing Minimum Objective Plan)

Input Drives (**Objective**)



Design/Check (**Development**)



Execute/Risk (**Operation**)

Repeating Patterns of Input Cycles

* **Objective Improvement is Success**

* **Successive Input yields Continuous Outcome**
(due to Ongoing Improvements Limitations)

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

Theory of Constraints Unconstrained systems have unlimited performance

Conway's Law Organization designs are copies of the the systems within the organization

Gold's Law Nothing goes as planned, because outcome is contemporaneous

Campbell's law Indicators subject to influence, become initiators

Cobra Effect A solution attempt that makes the problem worse

Computer Science Oxymoron (Alan Kay)

ABC Option Evaluation (Risk Adverse)

Empirical Plan (Crowdsource, Multi-perspective)
Last/Loudest Reconciliation (Best/Average Consensus)
Options Contention (Goal Alignment)
Integration Iteration (Vector Addition)

Computer Engineering Design (Risk Plan)

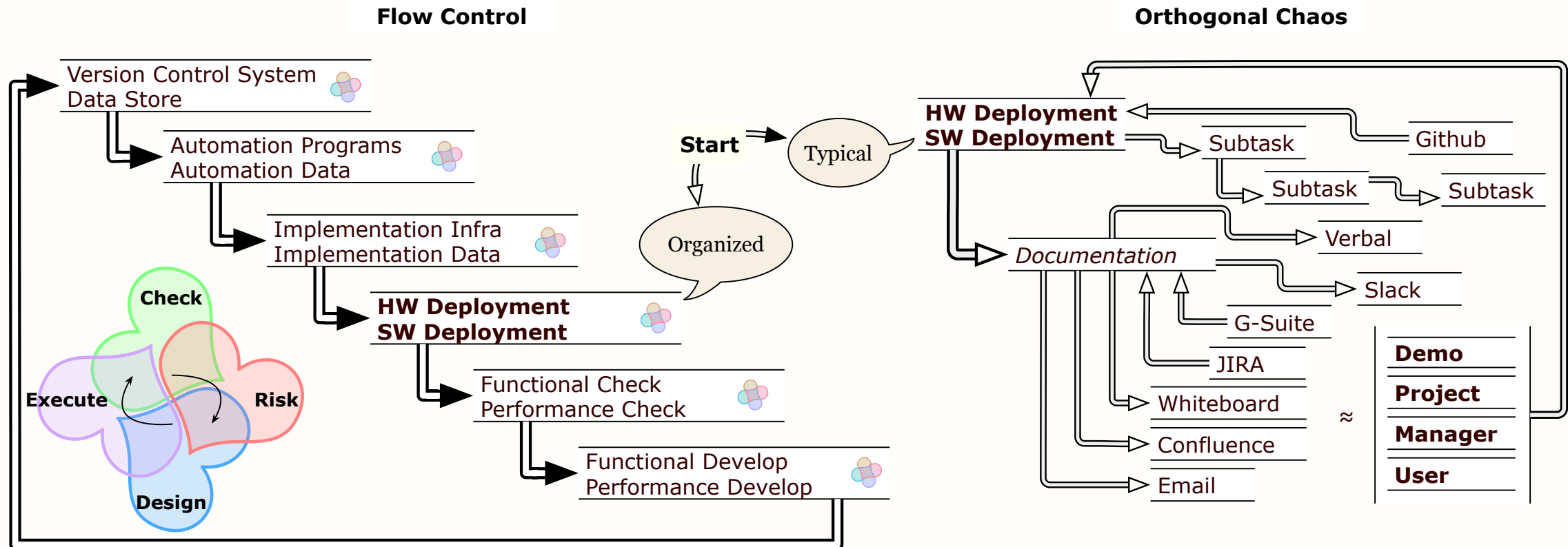
Operational Control (Objective Requirement)
Tool Alignment (Development)
MVP/Purpose built (Operation)

Scientific Method (with Risk Engineering)

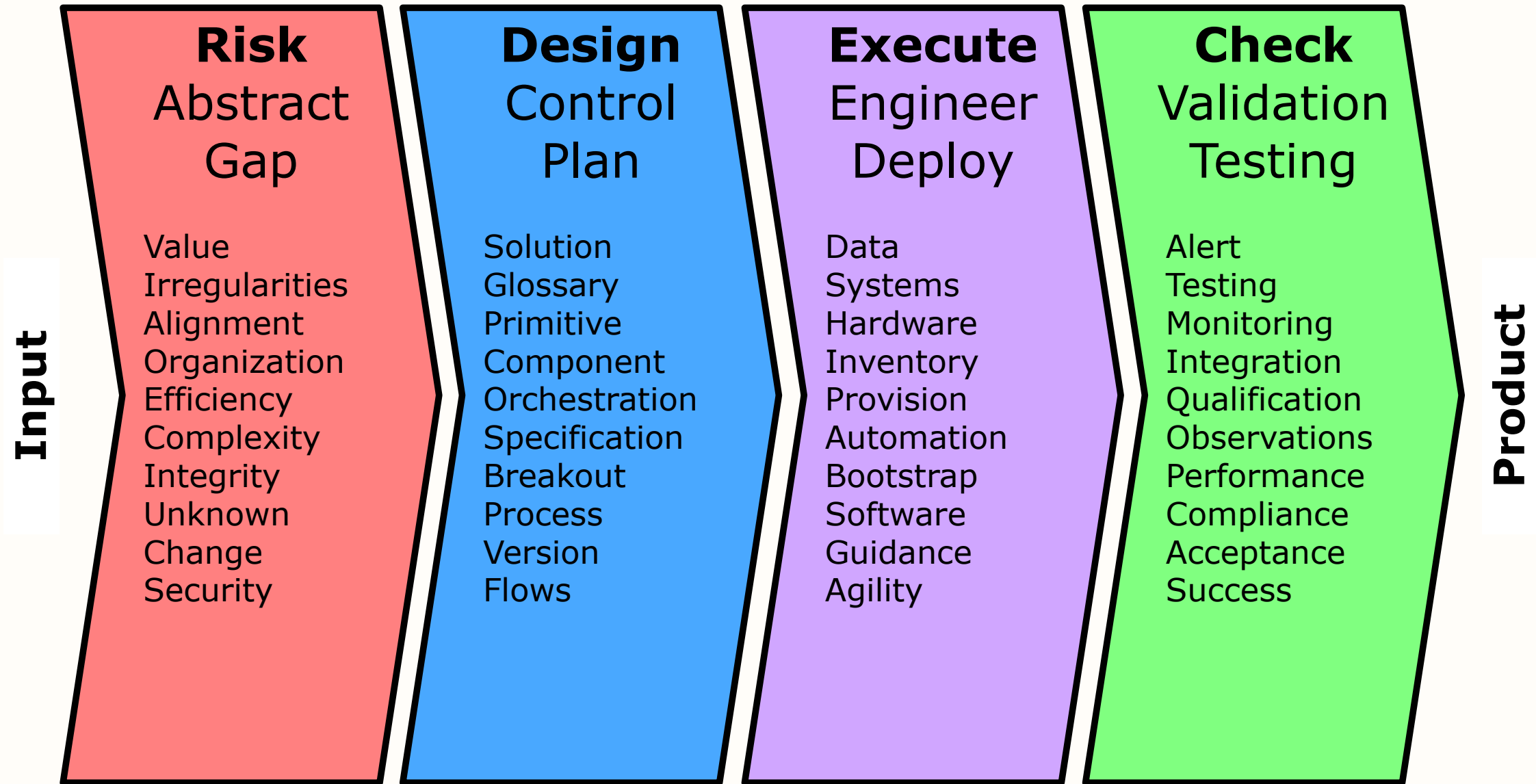
- 1) Observe (-> Input)
- 2) Assert (-> Plan)
- 3) Hypothesis (-> Execute)
- 4) Predict (-> Check)
- 5) Test (-> Risk)
- 6) Iterate (-> Product)

Bootstrap Deploy Iteration

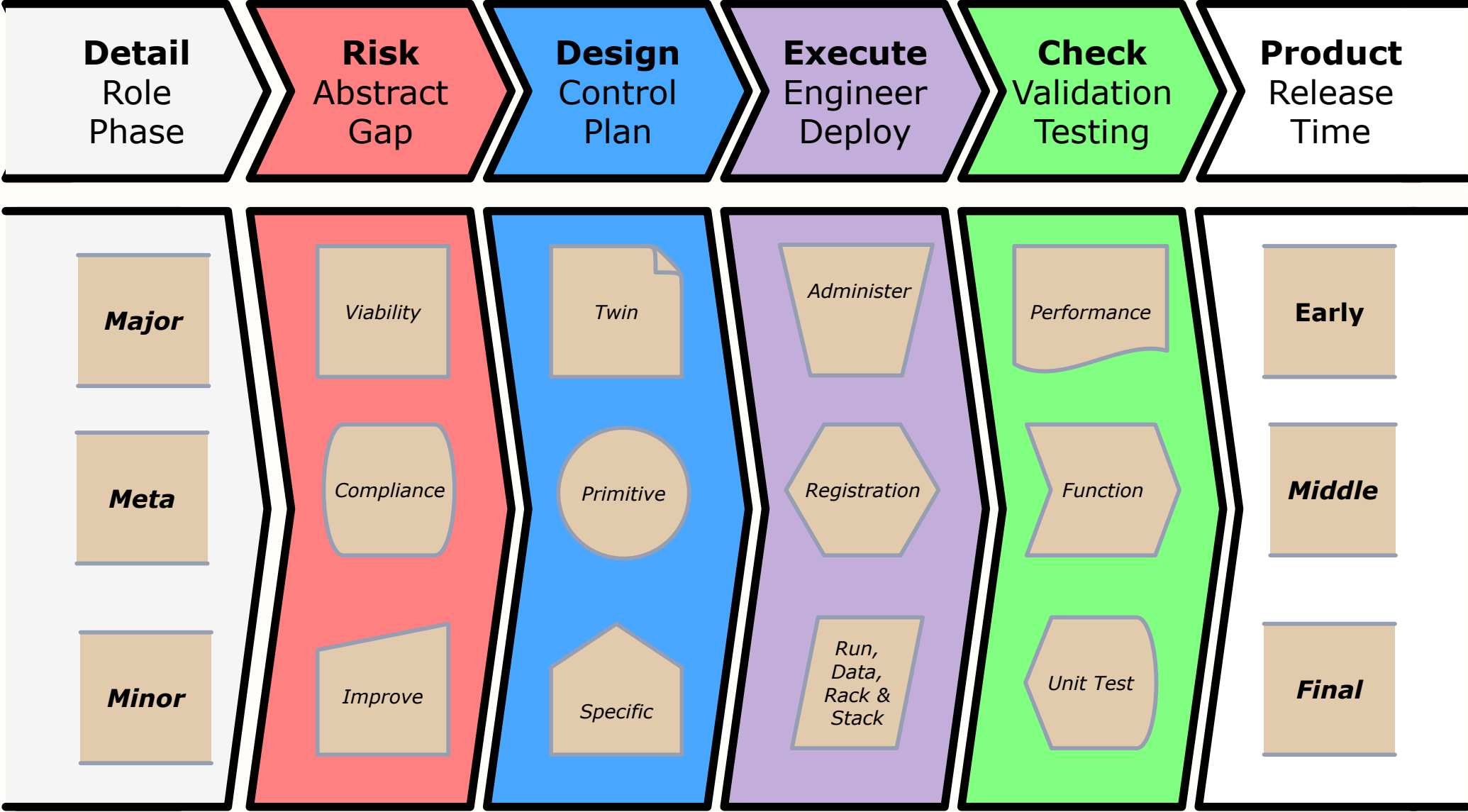
Flows, Configuration, Performance, Checks



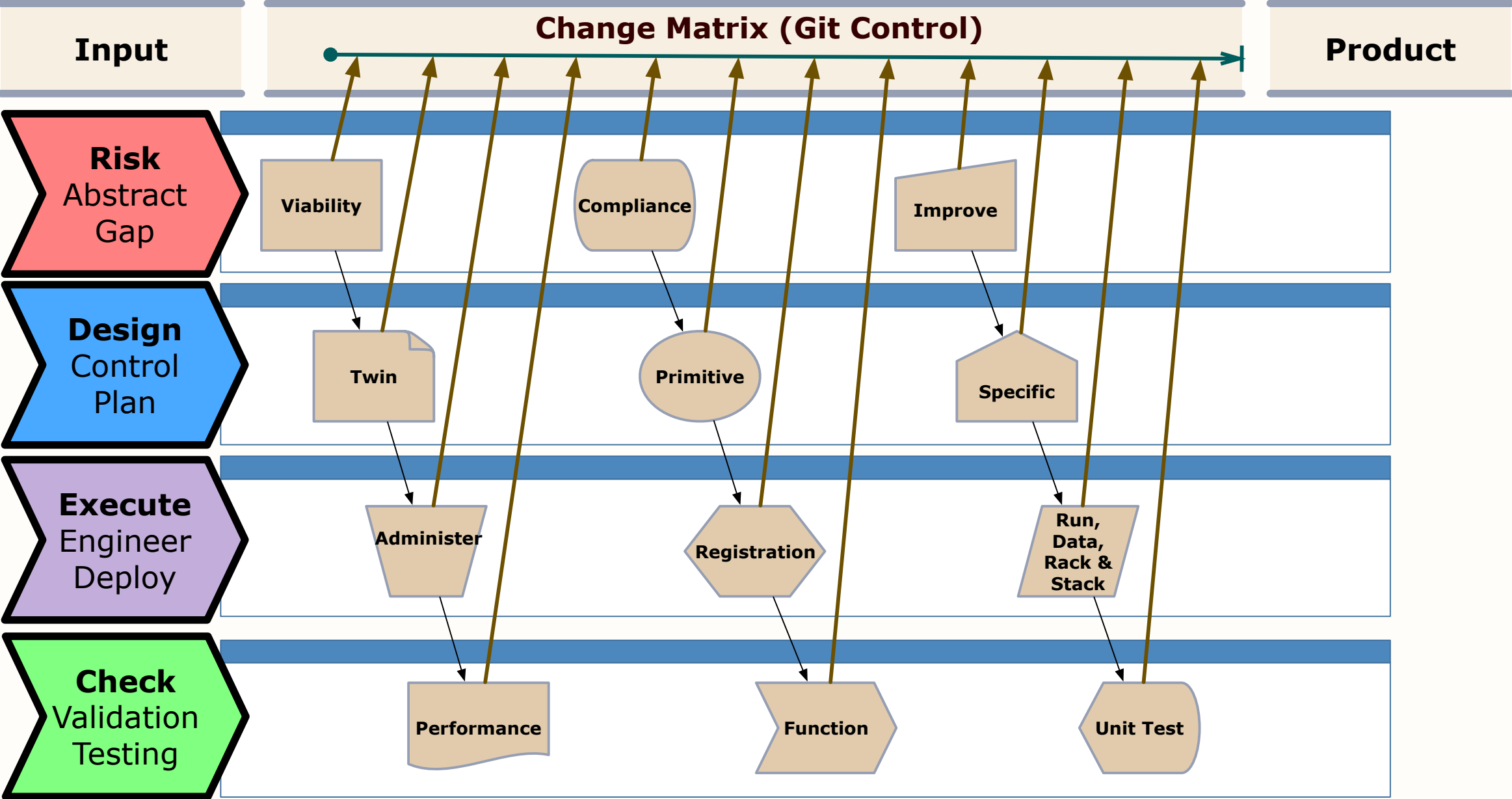
Four Phase Process



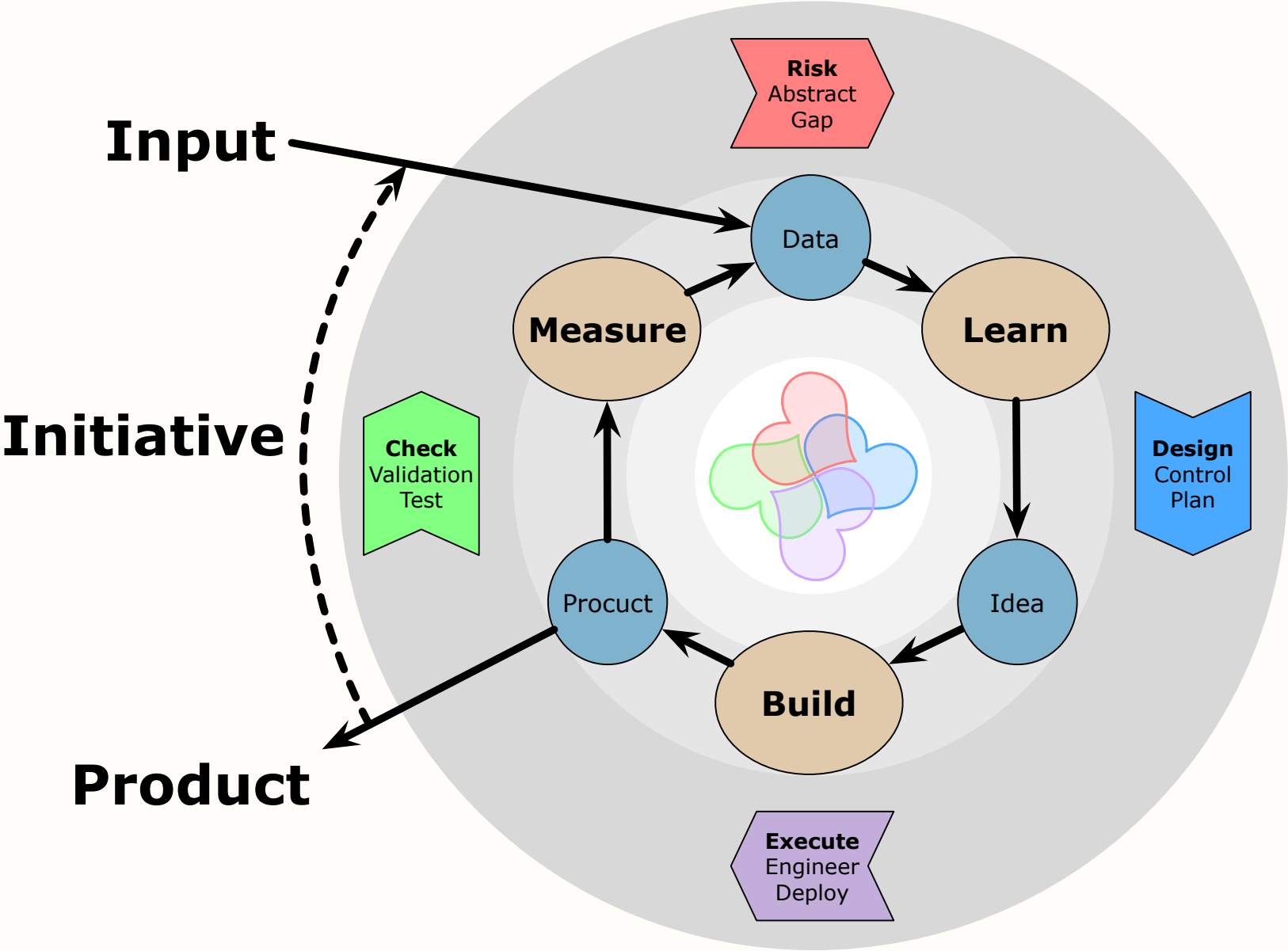
Continuous Delivery



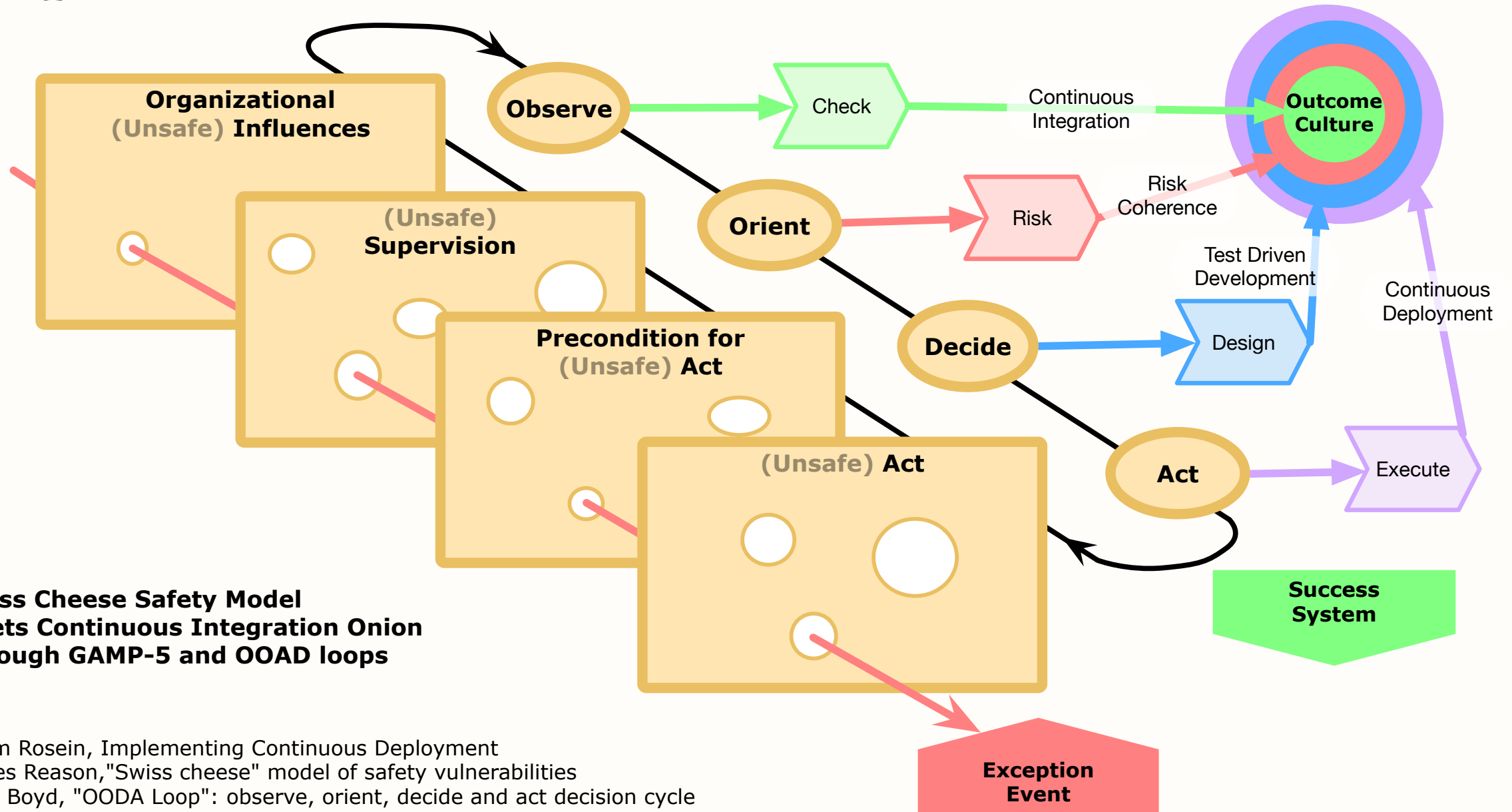
Objective Improvement



Improvement Loop



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