



DevOps Methodology and Infrastructure as Code

PRESENTED BY:

Shain Singh, Security Architect [APCJ]

Agenda

- DevOps Methodology and Concepts
- Infrastructure as Code Concepts
- F5 Toolchain Components
- Composing Workflows
- Continuous Integration & Delivery Pipelines

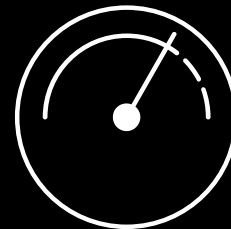
Establishing History



Henry Ford

Ford Motor Company

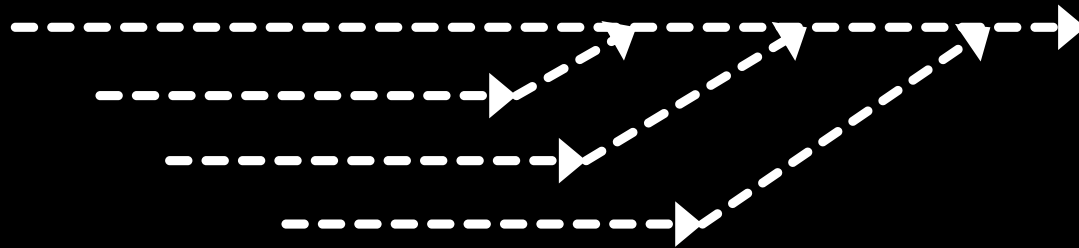
- **Assembly/Flow Line**
- Determine constraints
- Place them at the front of the process
- **Results**
- Predictable delivery time and margins
- Lower cost to consumer



Taiichi Ohno

Toyota Production System (TPS)

- **LEAN Manufacturing**
- Flow Line
- Kanban
- **Just in Time (JIT)**

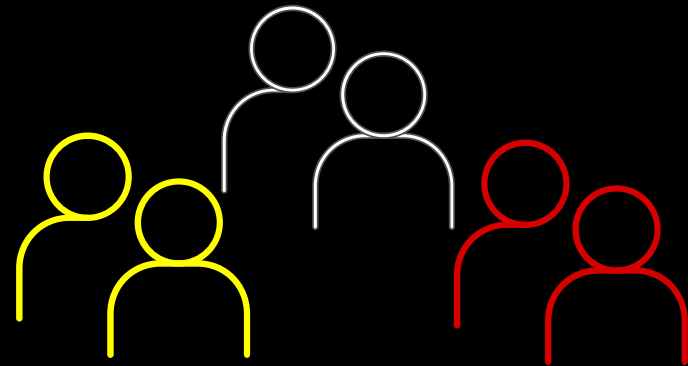


W. Edwards Deming

14 Points for Management

- **Business Effectiveness**

- Break down barriers between departments (9)
- Cease dependence on inspection to achieve quality (3)
- Improve constantly and forever the system of production and service (5)

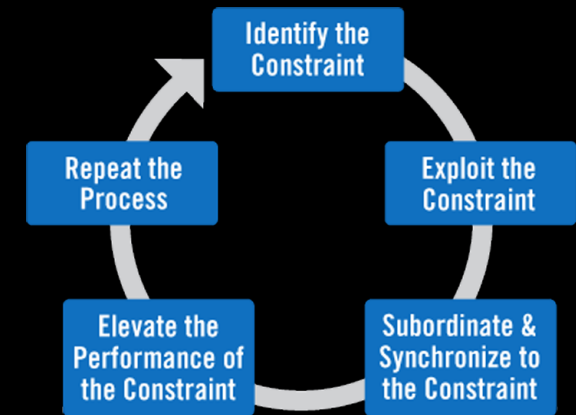


Eliyahu M. Goldratt

Theory of Constraints (TOC)

- **System Thinking**

- Identify the system's constraint(s).
- Decide how to exploit the system's constraint(s).
- Subordinate everything else to the above decision(s).
- Elevate the system's constraint(s).
- Warning! If in the previous steps a constraint has been broken, go back to step 1, but do not allow inertia to cause a system's constraint.[4]

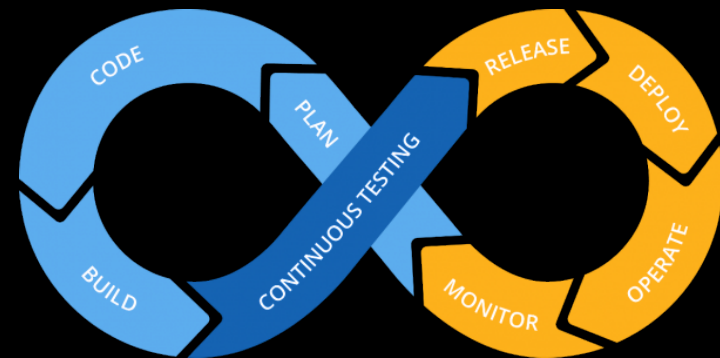


Gene Kim

The Phoenix Project

- **The Three Ways**
- System Thinking
- Amplify Feedback Loops
- Culture Of Continual Experimentation

Culture, Automation, Measurement, and Sharing



System Thinking Exercise



Infrastructure as Code (IaC) Concepts



Source-of-Truth

- Review from Class 1
- Source-of-Truth is defined as a system or object that contains the authoritative representation of a service.
- Changes for a service should propagate (push) from the source-of-truth to sub-ordinate systems.

- IaC can use Source Code Management concepts and tools
- Familiar to Application Development teams
- Provide ITIL-like capabilities (Access Control, Change Review, Audit, etc.)
- Enables visibility into production environment (**Production is not a mystical entity**)
- Super-NetOps training uses Git as a reference **Source-of-Truth**
- <https://git-scm.com>

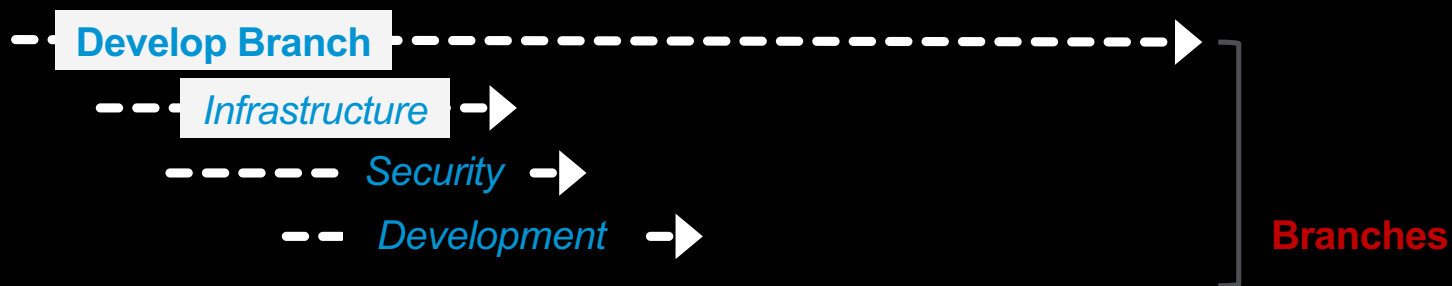
Pipelines

- Programmatic actions or processes that mutate an environment based on changes to the **Source-of-Truth**
- Deploy/mutate target environment to match Source-of-Truth
- Implement automated tests
- Super-NetOps training uses Jenkins as a reference Pipeline tool
- <https://jenkins.io>



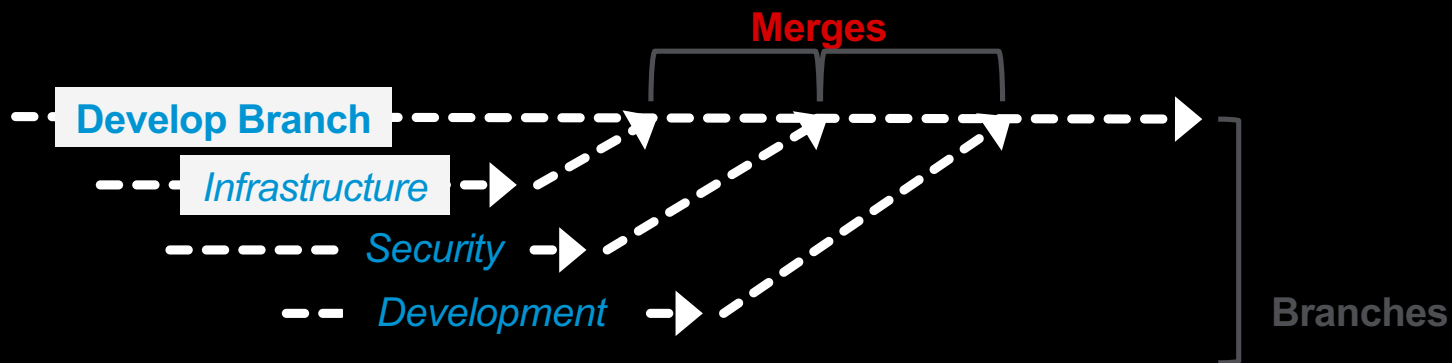
Branches

- **Branches** are used within a **Source-of-Truth** to represent a specific set of changes
- Individuals & Teams maintain their changes in their own branch
- Enables parallel **Contribution** and **Collaboration**
- Super-NetOps training uses the following **pattern**
- **Develop** branch represents pre-production state
- **Master** branch represents production state



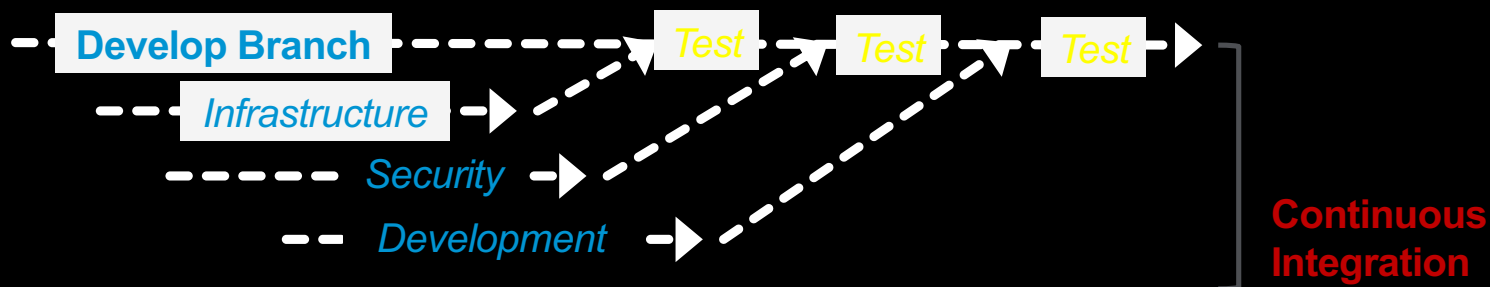
Merging

- **Merging** combines two branches together
- Conflicts can arise when the same object is modified in both branches
- Conflicts can be avoided by **Team Collaboration**
- Super-NetOps training uses the following pattern
- **Develop** branch should be merged to **Master** branch
- **Master** branch is pushed to Production



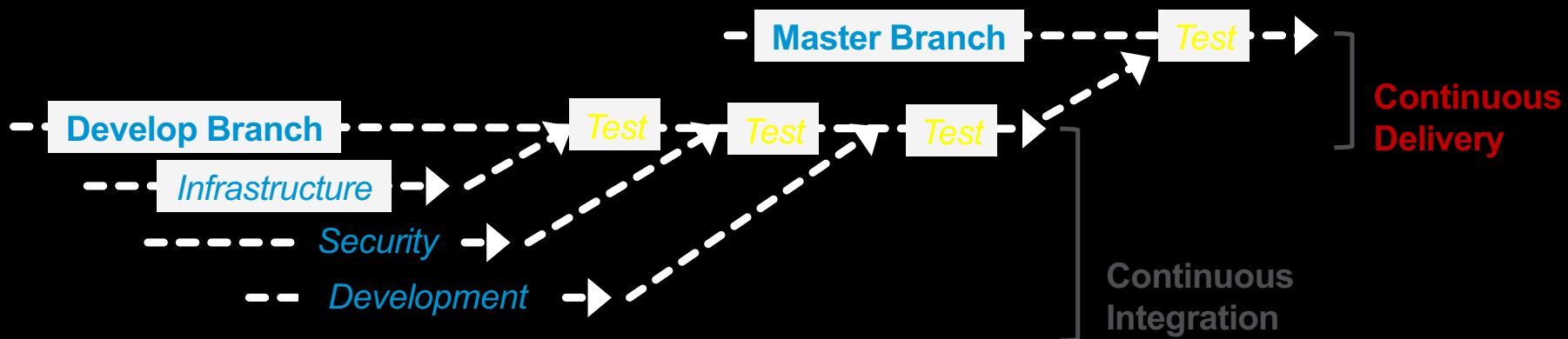
Continuous Integration

- Software Engineering concept
- Originated in the Booch method (1991), used in Extreme Programming
- Continuously merge individual **branches** to a shared **branch**
- Requires robust **unit** testing of changes
- Align with Agile Development Sprints
- Commit Early, Commit Often
- Prevents “Integration Hell”



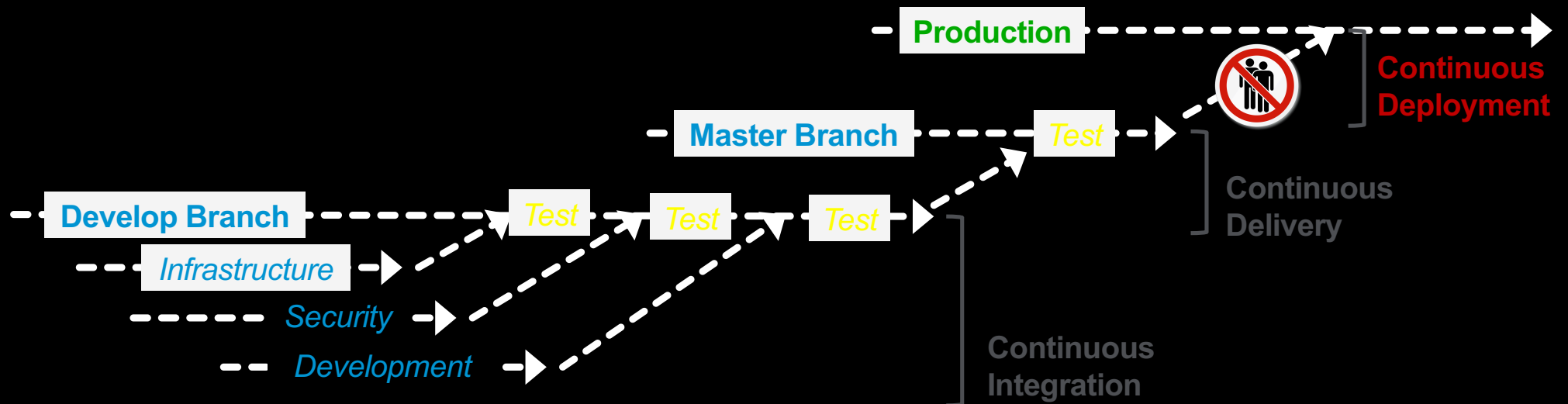
Continuous Delivery

- Ability to deliver change in an automated manner
- Requires robust **acceptance** testing of changes
- Does not require automated deployment to production



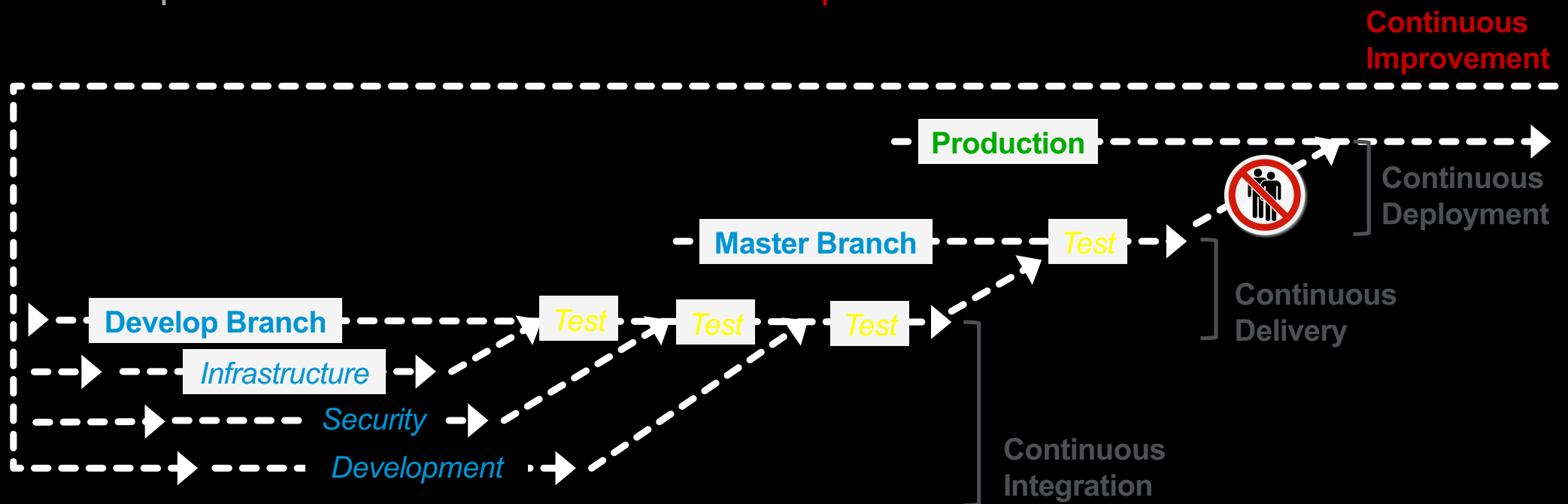
Continuous Deployment

- Automated deployment of all changes to a production environment
- Requires robust **user acceptance** testing of changes
- Must be weighed against business risk profile

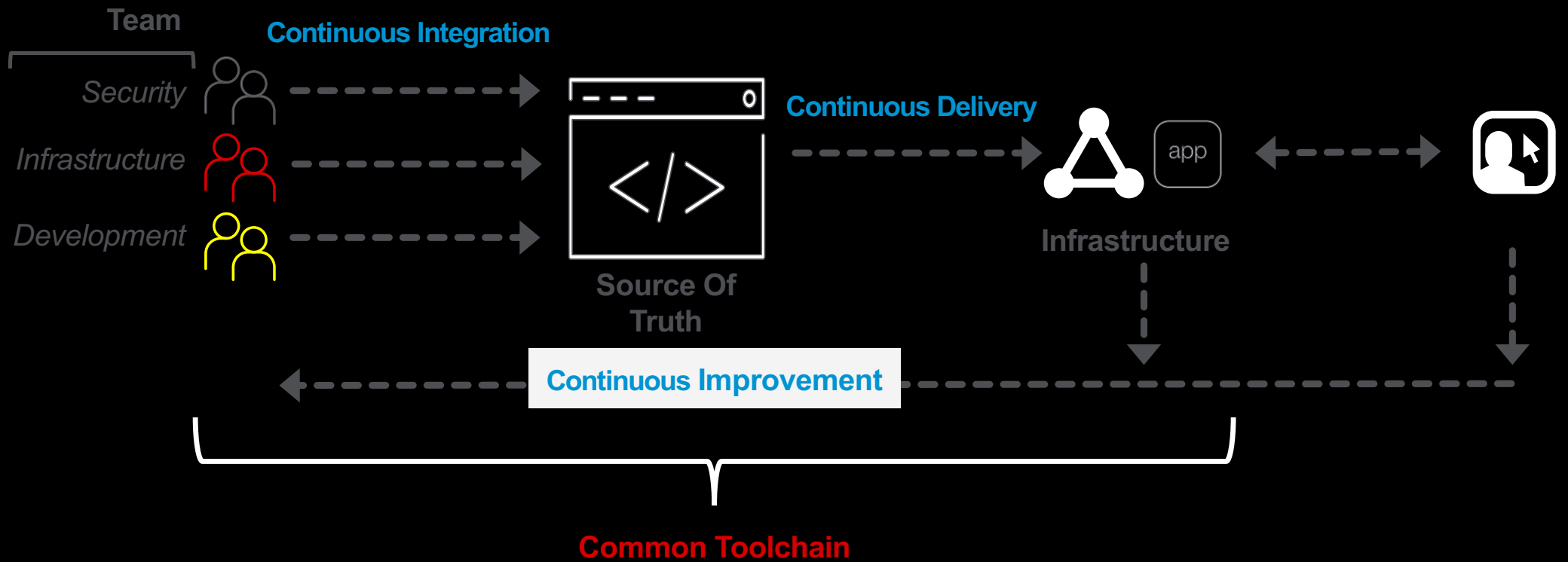


Continuous Improvement

- Feedback loop that enables improvement thru rapid iteration
- Focus on incremental change over time
- Requires robust **consumer feedback loop**



F5 IaC Delivery Pipeline

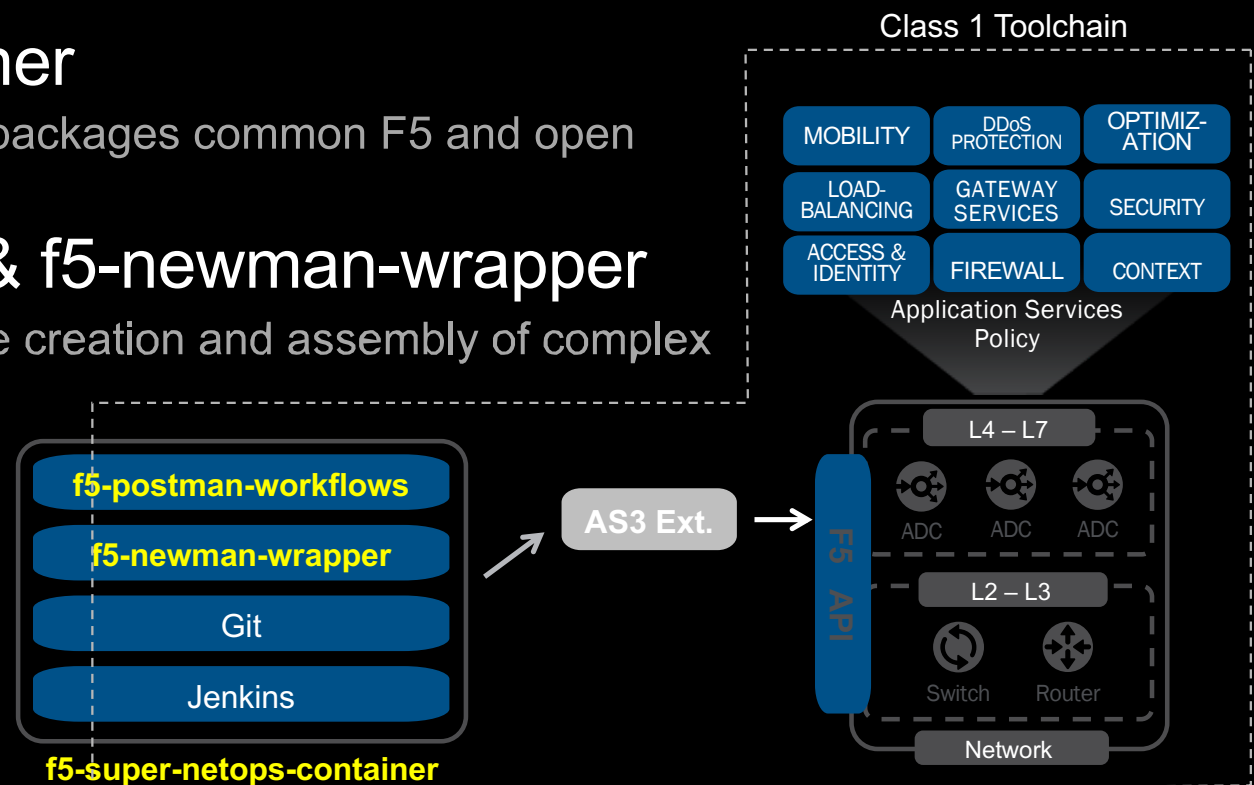


F5 Toolchain Components



F5 Toolchain Components

- Build on Class 1 Toolchain
- **f5-super-netops-container**
- Container based platform that pre-packages common F5 and open source tools (Git, Jenkins, etc)
- **f5-postman-workflows & f5-newman-wrapper**
- Extensions for Postman that enable creation and assembly of complex workflows

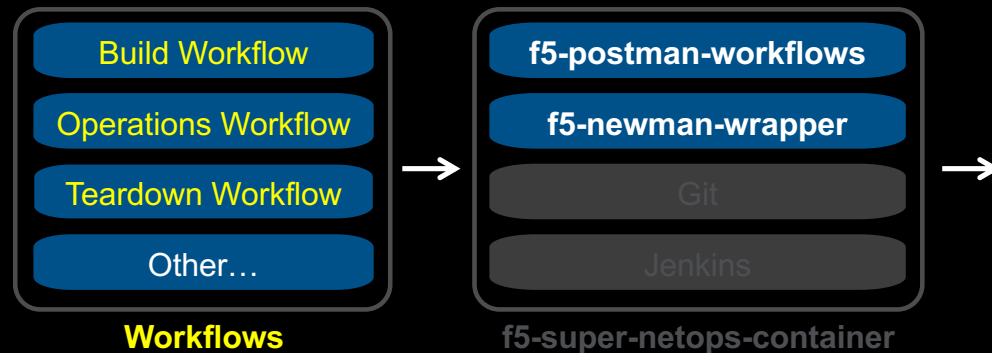


Composing Workflows



Composing Workflows

- Use f5-newman-wrapper to compose workflows
- Based on Postman collections that use the f5-postman-workflows framework

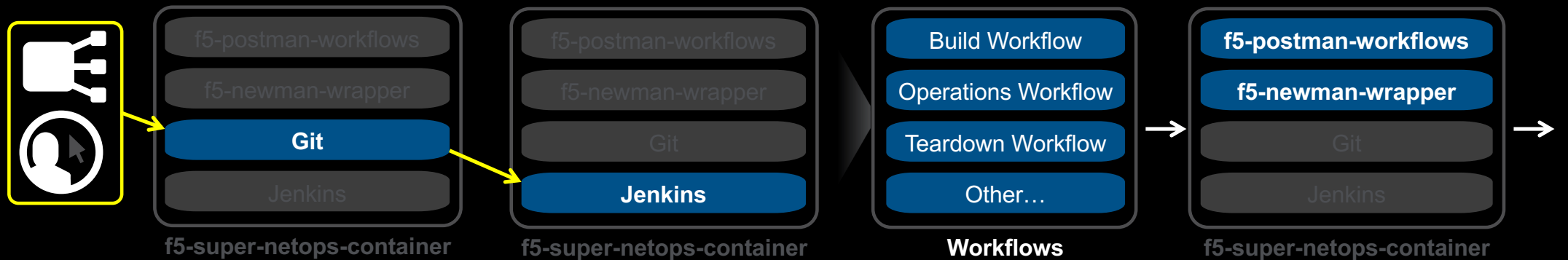


Continuous Integration & Delivery Pipelines



Continuous Integration & Delivery Pipelines

- Interact with Git SCM as our **Source-of-Truth**
- Use Jenkins to execute **Pipelines** based on changes to the **Source-of-Truth**



Thank You





SOLUTIONS FOR AN APPLICATION WORLD