



# QUALITY FOUNDRY - MAESTRO

MERGING QUALITY ASSURANCE DISCIPLINE  
WITH MODERN MACHINE ANALYTICS

[WAYNE.KIRK.SCHMIDT@GMAIL.COM](mailto:WAYNE.KIRK.SCHMIDT@GMAIL.COM)

FEBRUARY 16, 2025

# Overview – Testing Struggles, Strategy and Solution

- **Struggles**
  - Speed is needed to build modern applications
  - But we struggle with complexity
  - Distributed teams, goals, components add to coordination woes
- **Strategy**
  - Have teams focus on what they do best; developing
  - Have a highly automated means to coordinate testing
  - Capture and annotate everything possible to help with analysis
- **Solution**
  - Teams can focus on specific components
  - Teams can understand how their work fits into the whole
  - Teams can get immediate feedback on their work

# Struggles – Juggling and walking a tightrope at the same time!

## Multiple Teams

- Varied development styles
- Varied deployment schedules
- Varied development tools

## Multiple Criteria

- Deployment per event
- Deployment per customer
- Deployment per release

## Multiple Tests

- Scalability
- Security
- Stability
- Regression
- Compliance
- Business Logic

## Multiple Items

- Code
- Configuration
- Stack
- Environment
- Name-Space
- Data – Edge Cases
- Data – Volume
- Scale – Data
- Scale – Connections
- Security
- Stability
- Dependencies

## All can contribute to Testing Risks

- Confusion of Coordination
- Confusion of Communication
- Lack of Consistency



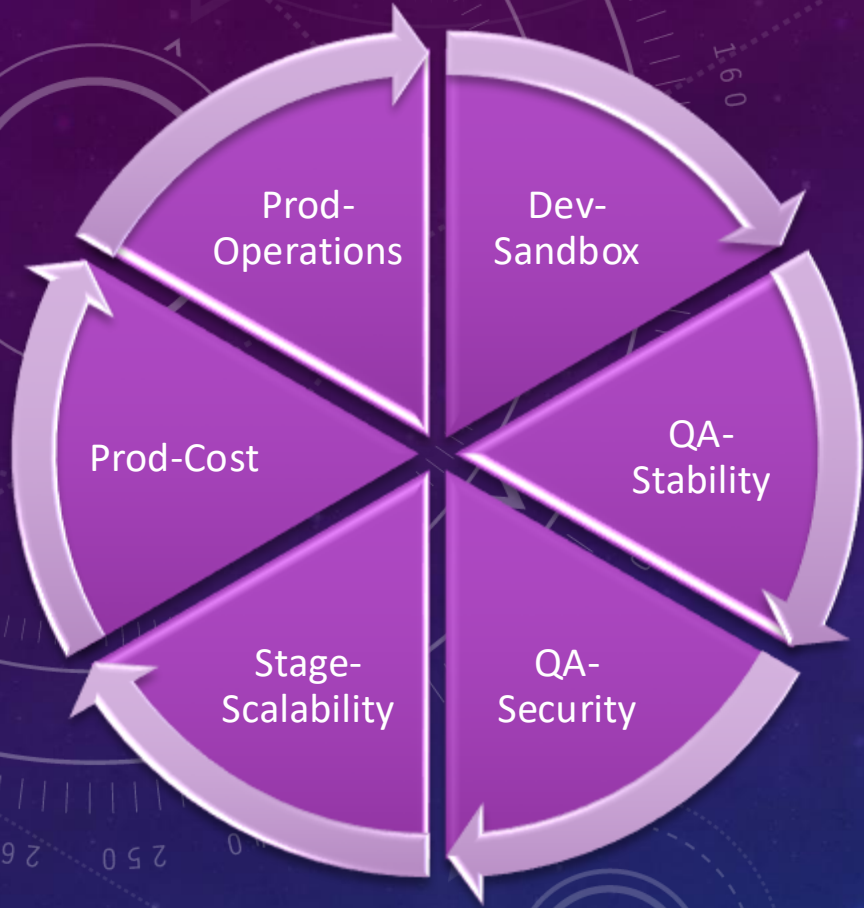
# Strategy – Build a Testing Foundry and Library

- Create a Testing Foundry.
  - Organized testing Approach – Build a cycle around Passed, Failed, Adjust
  - Automate the cycle – Make this triggered and continuous.
  - Instrument the output – date/time, data, and output
- Store history and context into a testing Library. Use this as training material.
  - Structure the output for easy reuse as inputs
  - Determine activity within Foundry and decisions for the Library
- Continuously Optimize!
  - Build on successes.
  - Retrace based on failures.
  - Adjust and Change.

# Strategy – Implement your very own QA Maestro.

- Configure a **Maestro**, who provides suggestions and conducts the testing
  - Based on playbooks
  - Continuously learning
  - Goal is compounding improvement
- **Maestro** uses machine analytics to read from the **Library**
- **Maestro** builds up knowledge how to direct the **Foundry**
- **Maestro** learns how to make code better through small “quick wins”
  - Organized Documentation
  - Basic Best Practices in code
  - Basic Code structure

# Solution – Factory Foundation – Testing Stages and Steps

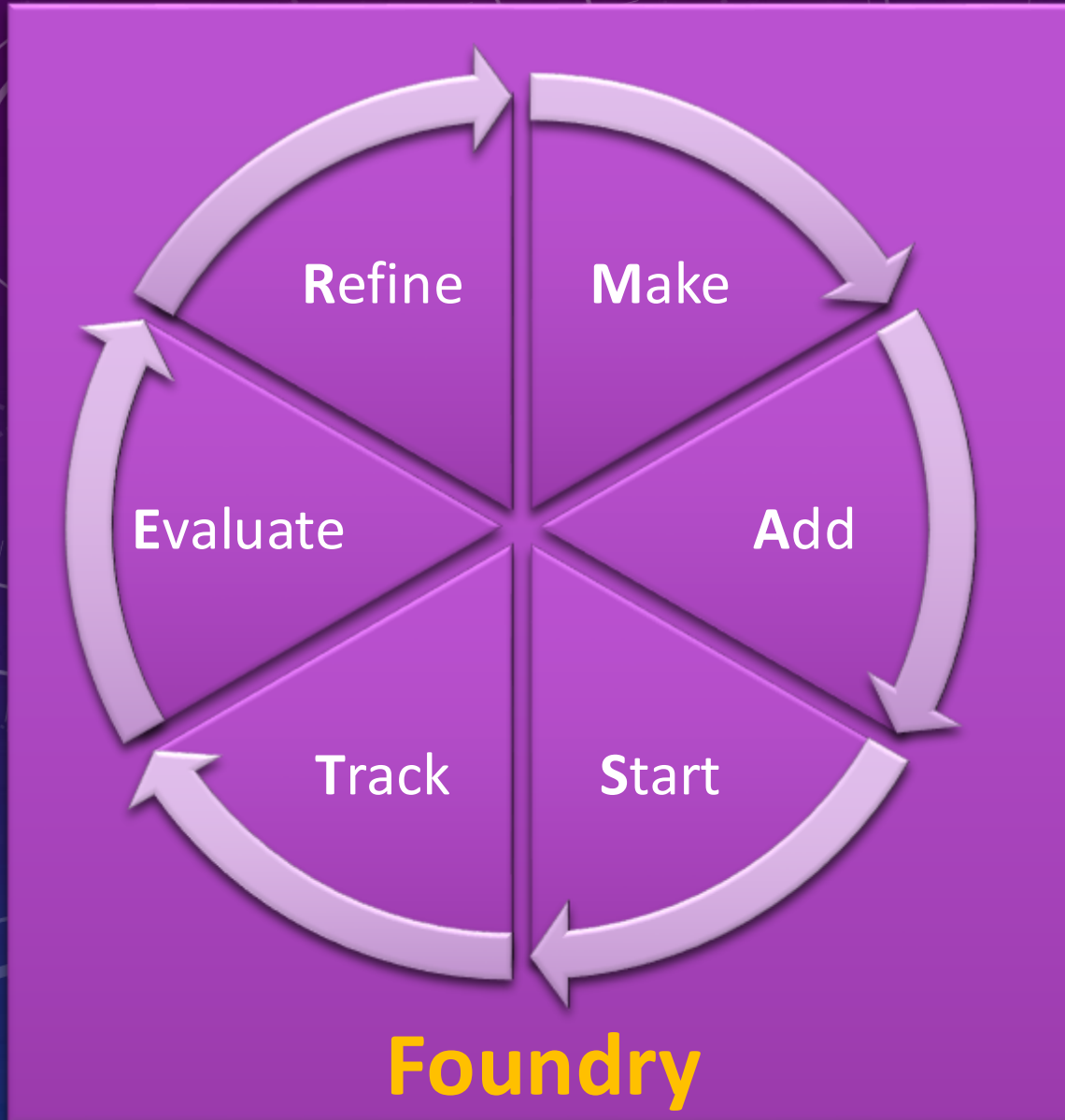


- Each Stage can be driven from a time/date or manual trigger
- Environment is cloned from a desired environment
- Changes to any component for development are tracked
- Follows the **MASTER** steps to testing and evaluation
- Release is based on (re)building environment
- Filters and proxies used as protection for data as well

M	A	S	T	E	R
Make the environment.	Add the appropriate data and configurations.	Start the structure testing.	Track the output of the testing.	Evaluate the data and output collected.	Refine the testing approach.



# Solution – Bringing it all together



# Solution – Sample Toolbox

Role	Free Software?	Commercial Products?	Description
API	yes	yes	Build API for controlling Maestro, Factory and Library
CLI Binding	yes	yes	Build clients out of API specification
Glue Code	yes	yes	Handling specific automation tasks
Configuration	yes	yes	People can use YML or XML as well
Template	yes	yes	Means to coordinate preparation and activity
Build Process	yes	yes	Tie source control to activities to kick off evaluation
Source Repository	yes	yes	Source control for data under test
Credentials	yes	yes	Ability to store data
Full Environment	yes	yes	Means to build virtual environment(s)
Messaging/Bus	yes	yes	Messaging Bus for communication
Certificates	yes	yes	Certificates for driving secure communication
Data Replay	yes	yes	Replay Data into the environment
Data Capture	yes	yes	Collecting the data from the environment



# About the Author:



Wayne Kirk Schmidt

Email: [wayne.kirk.schmidt@gmail.com](mailto:wayne.kirk.schmidt@gmail.com)

Phone: +81 090 9969 6111

LinkedIn: <https://www.linkedin.com/in/waynekirkschmidt>

Portal: <https://sites.google.com/view/waynekirkschmidt/>

Wayne is a seasoned technology professional with over 30 years of industry experience, with a background spanning operations, development, engineering, senior management, audit and compliance, across Financial, Insurance, Security, Technology, and Consulting firms.

Throughout his career, Wayne has focused on security, integration, automation, and testing, focusing on the tools to deliver change securely, with speed, and at scale.

By combining hands on approach with strategic thinking and a passion for education, he has consistently built high-performing teams, driven digital transformation, achieving sustainable results in business where Change Is.