

A satellite view of Earth is shown on the left side of the slide, featuring the Gulf of Mexico, the United States, and parts of the Atlantic Ocean. A red horizontal bar is overlaid on the middle of the image.

Building Automated Testing Competence Across an Organization

Integrated Defense System (IDS)

Matthew Thomann

Senior Systems Engineer II

March 2, 2016

Building Automated Testing Competence Across an Organization

Raytheon
Integrated Defense Systems

NON-EXPORT-CONTROLLED TECHNICAL INFORMATION

This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

Agenda

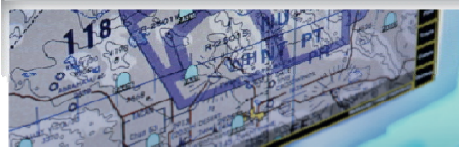
- Who we are
- *TestForward* Approach
- Common Execution Framework
- Methods to Build Automated Testing Competence
- Conclusion

Raytheon Company Overview

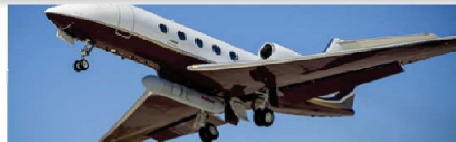
Raytheon
Integrated Defense Systems

A technology and innovation leader specializing in defense, civil government and cybersecurity markets throughout the world.

- 2014 NET SALES: \$23 BILLION
- 61,000 EMPLOYEES WORLDWIDE
- HEADQUARTERS: WALTHAM, MASSACHUSETTS



C5ISR



ELECTRONIC WARFARE



MISSILE DEFENSE



PRECISION WEAPONS



TRAINING & SERVICES



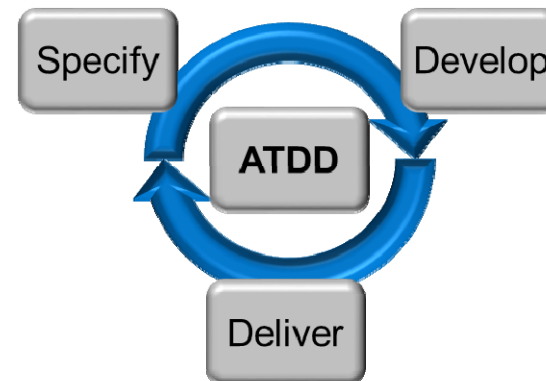
CYBER

Introduction to IDS - SVTAD

- **Integrated Defense Systems:**
 - Specializes in air and missile defense, large land- and sea-based radars and systems for managing command, control, communications, computers, cyber, intelligence, surveillance and reconnaissance (C5ISR)
 - Produces air traffic management systems, sonars, torpedoes and electronic systems for ships
- **System Verification Test & Analysis Directorate (SVTAD):**
 - Responsible for the Integration, Verification, and Validation (V&V) of all IDS products
 - Develops the processes, standards, and expertise to ensure that our customers products are taken from subsystems to integrated deployed systems

TestForward

- In an effort to boost system quality and speed delivery, SVTAD is applying the Acceptance Test-Driven Development (ATDD) approach to system integration and test, including
 - In-sprint collaboration of integration and test activities with development flow
 - Automated system verification testing at the mission thread level
- This initiative, *TestForward*, is driven by the confluence of Raytheon's
 - Development of Agile practices
 - Shift to mission thread-based testing
 - The push to SI&T test automation



Building Automated Testing Competence Across an Organization

Raytheon
Integrated Defense Systems

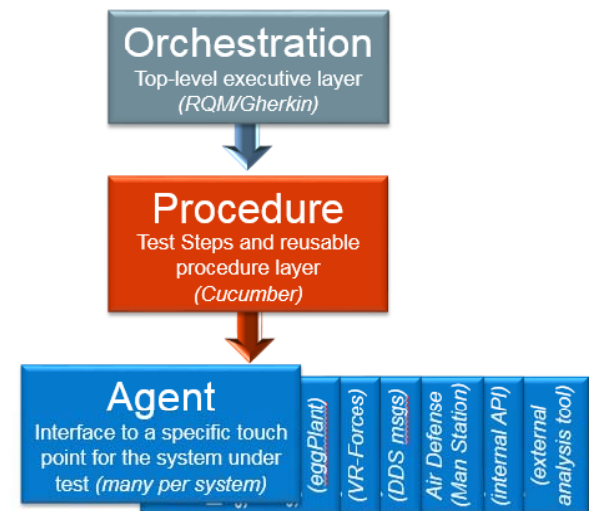
- Challenges
 - Hundreds of engineers across dozens of programs
 - Building competencies (tools, technologies and processes) to quickly enable engineers to start contributing to a “better” way of doing SI&T
- Apply unified *TestForward* Approach
 - Establish Common Execution Framework
 - Use a standard test scripting language and implementation strategy
 - Apply modular interface components to system interfaces
 - Develop Cadre to provide on program technical leadership
 - Rapidly capture and propagate detailed engineering techniques
 - Establish informal community based sharing



***The TestForward Approach Rapidly Builds Organizational Competency
in Test Engineering***

TestForward Approach

- Define a standard framework: multi-layered, federated
 - One instance: Rational Quality Manager for overall test management, Cucumber as Test Framework and Jenkins for Automated Regression
- Specify tests via a high level test language
 - Using natural ATDD language (ex. Gherkin)
- Bind to easy to use scripting language
- Use a modular interface approach with agents
- Make test procedures reusable
- Adaptable to various programs needs



Benefits of *TestForward* Approach

- A well-defined test engineering process provides
 - **Faster employee ramp up:** a known process fixes a learning target
 - **Higher delivered system quality:** SI&T capability is driven by methodology effectiveness
 - **Technological Currency:** brings the engineering processes inline with current industry state of practice, aligning with a wide array of online and third-party resources
 - **Consistency:** lessons-learned are shared across teams and programs
 - **Autonomy:** defined processes give engineers a clear view of what is expected and lets them creatively contribute
 - **Adaptability:** process improvements are rapidly deployed and elevate all practitioners

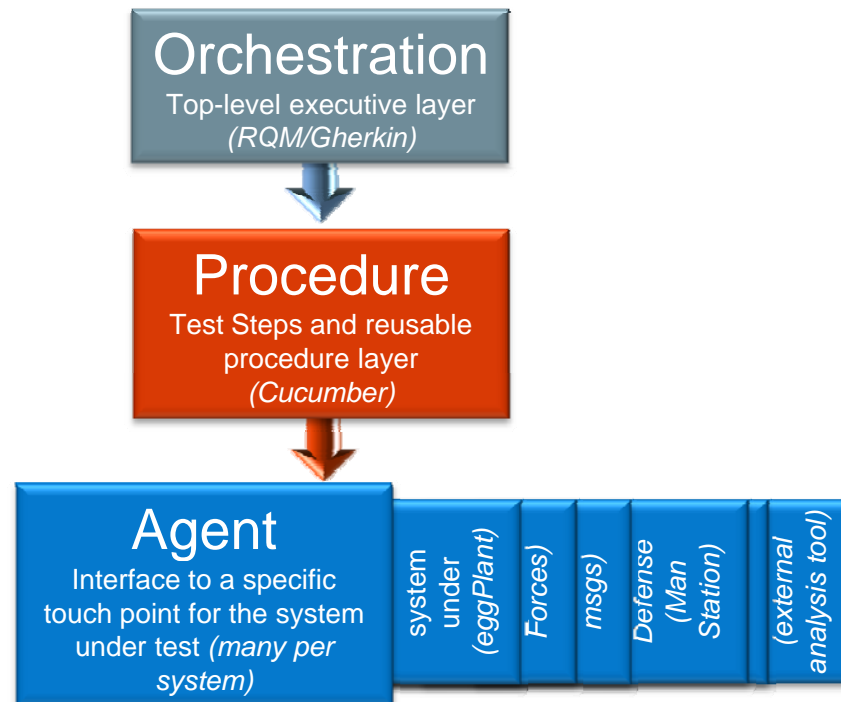


Common Execution Framework

- Effective Test Automation requires a common execution framework with a multi-layered modular architecture
 - Architect with an open and flexible interface approach between components and layers for adaptability and to enhance modularity
 - Incorporate current industry techniques and best practices
 - Utilize a reusable platform which is adaptable to varying program needs
 - Focus on ease-of-use
 - Continually evolve to incorporate new techniques and underlying test automation technologies
- Challenges of implementing a common execution framework
 - Cultural Impact: Need to help programs switch from legacy SI&T Processes
 - Effort to identify where an automation framework could provide benefit to a legacy program – when will ROI be realized?

TestForward Execution Framework

- Raytheon *TestForward* method is a set of SI&T test development and execution techniques explicitly developed to take advantage of a multi-layer test execution framework:



3/7/2016

Standard Test Specification Language

- A key technique in ATDD is the use of a Test-Specific Language to specify executable tests in natural language
- Gherkin is a capable and popular language
 - Supports a simple test-oriented structure
 - A few key terms mixed with natural English
 - Testers use terms unique to their system
 - Test procedures are then filled in based on the user's test execution framework

```
1 @SingleABTScenarios
2 Feature: ABT Air Surveillance
3   As an ADOC team member
4   I want to maintain situation awa
5   So that I can respond to threats
6
7 Background:
8   Given Eggplant session is active
9   And an ABT operator is logged on
10  And VRForces is started on http:
11  And VRForces scenario 17 is runn
12
13 Scenario: Single ABT appears on disp
14   When 1 ABT enters our airspace
15   Then the operator should observe
16
17 Scenario: ABT Marked Hostile Changes
18   Given 3 ABT enters our airspace
```

Orchestration

Top-level executive layer
(RQM/Gherkin)

Standard Test Implementation Approach

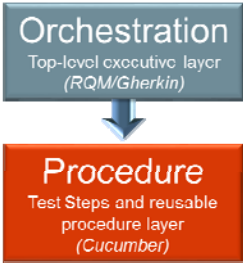
```
Given Eggplant is connected to $workstation1 using $testSuite1  
And $operator1 is logged in  
And the VRForces agent is listening on $vrforcesUrl  
And VRForces scenario $scenario1 is running
```

- Generate Step Definitions
- Go To Step Definition
- Run SpecFlow Scenarios
- Debug SpecFlow Scenarios
- Source Control



```
[Given(@"Eggplant is connected to (.*) using (.*)")]  
public void GivenEggplantConnectedToWorkstationUsingTestSuite(string workstation, string suite)  
{  
    string _workstation = Common.ProcessParameter(workstation);  
    string _suite = Common.ProcessParameter(suite);  
  
    this.result = this.eggplant.StartSession(_suite);  
    this.CheckStatus();  
    this.result = this.eggplant.Execute("Connect (name:\"" + _workstation + "\"");  
    this.CheckStatus();  
}
```

Test and SW Engineers Work Together to Create Test Step Procedures Calling Agent Services



Standard Test Implementation Approach **Raytheon** Integrated Defense Systems

- Benefits of utilizing a standard test scripting language
 - Build consistent skill set across common languages
 - Ability to readily grow our knowledge base from industry standard resources
 - Easily share techniques and libraries across teams and programs
- Enables consistent system interface strategy – Agents!



Agents - Modularity and Platform Independence

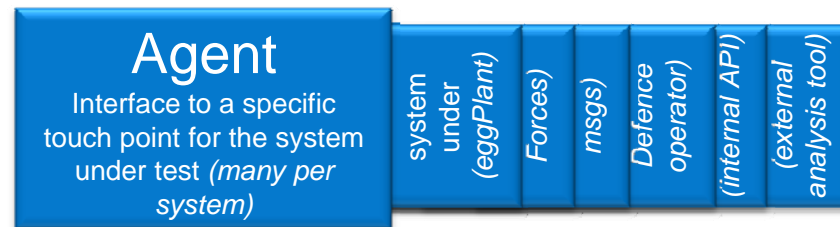
- An **Agent** is a software component that serves as an interface to one aspect of the System Under Test (SUT)




- This federated architecture - Cucumber procedures driving distributed Agents - is a key enabler to achieving
 - Flexible test and deployment topologies via Agent communications
 - Platform Independence, decoupling test procedures from Agent implementation
 - Decoupled interfaces - add/modify/reuse individual Agents independently

Agents - Modularity and Platform Independence

- The Agent solves the hard problem of connecting to the SUT
- Agents to be developed by SW Engineers following proven software engineering disciplines
 - Highly Robust
 - Easy and convenient to use for SI&T Engineers
 - Highly configurable and adaptable
- Most agents can be re-used across programs



Expertise Transfer via Cadre Model

ca·dre  (kā'drā, -drə, kād'rē, kā'dər)

n.

1. A nucleus of trained personnel around which a larger organization can be built and trained: *a cadre of corporals who train recruits.*

- Explicitly tasked with rapidly learning *TestForward* techniques and technologies and guiding new practitioners on client programs
- On program expert guidance
- Medium to long term engagement
- Propagate best practices throughout the organization
- Muddy boots style of technical leadership



Test Automation Wiki

- Wiki is a lightly structured high accessible potentially two-way exchange of information
- The front line deployment definition of the *TestForward* methodology is in each program's SI&T Wiki
 - Development of this document should be a collaborative process which involves the input of more than one individual
 - Adaptable to the ever changing test automation principles and best practices
 - Easily shared not only within the program but cross company
 - Easily updated by test engineers

| Test Automation |
|---|
| <ul style="list-style-type: none">• Automated Testing 1.0 - Tools Setup• Automated Testing 2.0 - Creating Gherkin and Cucumber• Automated Testing 2.1 - EggPlant Programming• Automated Testing 3.0 - SUT Connections• Automated Testing 4.0 - ClearCase Procedures for Test Development• Automated Testing 4.1 - dev and test Baselines• Setup Test Environment using eggPlant• VR-Forces Class supplemental data• Community of Practice - Test Automation |

Wiki Challenges

- Some contributors are better writers than others
- Important process documentation must be consistent and concise (verbally these are typical of committee output)
- Access is controlled on a program basis and not easily shared
- As a Wiki grows organization and ease-of-use can be impacted

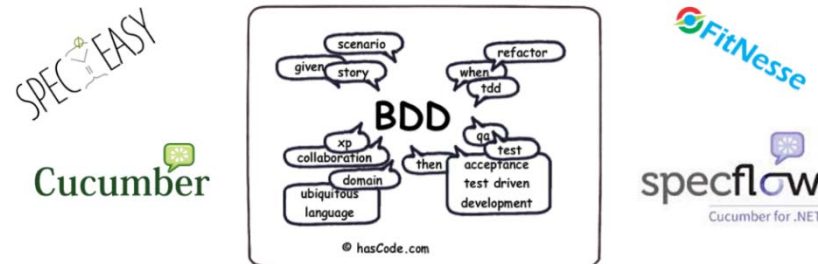


Test Automation Community of Practice (COP)

Raytheon
Integrated Defense Systems

- We have established an environment for sharing information and experiences with Test Automation
 - Bring together people interested in all things related to test automation
 - Provide a forum for discussion about topics of interest
 - Surface common problems and work toward solutions
 - Advance the state of automated testing on the program driven by actual need
 - Share knowledge and experiences to grow as individuals
 - Generate enthusiasm

Test Automation Community of Practice



Conclusion

- *TestForward* is shaping up to be an effective approach for engaging and training our Integration & Test Community
- Tools, Technologies and Processes are in place and are being adopted and adapted by multiple programs
- Community of Practice and the Wiki have been very successful in exchanging knowledge and ideas and building best practices
- *TestForward* has leadership and engineering support and enables our Test Organization to deliver higher quality systems while saving costs.



Speaker Info

- Matthew Thomann
- Matthew_q_Thomann@Raytheon.com
- 978.858.9094