Systems Integration: Effective DOD Test & Evaluation
Bob Koczat

- Senior Engineering Fellow: Raytheon (retired 2008) 35 years Systems, Software, Test, Project engineering, Program Management

- DOD PATRIOT Air Defense Systems: Technical director, Lead Engineer, Systems Integration
  
  ➢ *Lead Roles in systems/software development, integration, systems testing, search-track test events, missile firings, interoperability, DTE & OTE testing (4) at various test sites and missile ranges (i.e. WSMR, etc)*

- FAA STARS Air Traffic Control Systems: Systems Integration Lead

- The SPECTRUM Group, Wash DC: Engineering Member

- TESTPLANT Consulting: Introduction of Automated Testing with eggplant into DOD, Aviation, and Aerospace Sectors
Systems Integration: Agenda

- Test & Evaluation: Stating the Issues
- Effective Test & Evaluation: Objectives
- System Integration planning, Test Configurations, and Team definition
- Systems Integration execution
- System base-lining, regression testing, automated testing strategies
- T & E readiness evaluation and assessments
- Summary
Test & Evaluation: Goals and Objectives
DAS: Defense Systems Acquisition Cycle

The Materiel Development Decision precedes entry into any phase of the acquisition management system.

- Entrance criteria met before entering phase.
- Evolutionary Acquisition or Single Step to Full Capability.

Diagram:
- User Needs
- Technology Opportunities & Resources
- System Acquisition:
  - Pre-Systems Acquisition
  - Systems Acquisition
  - Sustainment

Decision Points:
- Material Decision
- Milestone B
- Milestone Review

Decision Points if PDR is not conducted before Milestone B:
- PDR A
- CDR A

Milestones:
- LFIP/IOT&E
- FRP Decision Review
Engineering Development: Test & Evaluation Conceptually & Historically

SYSTEM SPEC: CAPABILITIES

Requirements
Development
Integration
System Integration
V&V, Acceptance Testing
DTE
OTE

Fault Profiles
Open
Closed

Fixes, Integration
Requirements updates

Fixes, Integration
Requirements updates
Systems Integration: Effective Test & Evaluation Objectives

- Systems Integration is embedded in the critical path of the engineering development Life-Cycles (PLM), project planning, software integration, V & V testing, and system & operational testing.

**SYSTEM CAPABILITIES (CPPs)**

- **Capability Performance parameters**

**DIAGRAM:**
- **Requirements**
- **Development**
- **Integration**
  - **System Integration**
  - **V & V Testing**
  - **Fixes, Integration Requirements updates**
- **DT/OT**

**Effective Test & Evaluation**
Effective Test & Evaluation: Objectives

Systems Integration strategies provide for:

- Early visibility into the system, hardware, and software operational condition
- System capability situational awareness throughout engineering development and Program Life-Cycles (PLM), Phases
- A midgame-endgame mechanism to adapt to requirements & software agility, managing the chaotic phases
- Achieve T & E operational readiness
- Ability to conduct a cost effective, quality evaluation and assessment of system and operational performance during DTE and OTE Tests
Systems Integration Planning, Teams
Systems Integration Test Configurations

- **eggPlant System**
  - Generate script
  - Send scripts to app
  - Keyboard Mouse Screens
  - Scripts, Suites
  - Application Screens
  - Test Verification Reports

- **UserMenus**
  - C2 Command & Control
    - VNC
    - Applications
      1. TACTICAL
      2. INIT
      3. DIAG
      4. SIM
      5. TRAINING

- **Databases**
- **Data Collection**
- **Scenarios**
  - Sensor, Radar Launcher, GPS
  - C4I
  - SIM
  - Training

- **Access | Insight | Experience**
Engineering Development: Systems Integration Planning

**Incremental Development, Test:**

**Build 1**
- Infrastructure, executive, operating systems, interfaces, protocols
- Initialization, DBs
- Menus, Operator input
- Site adaptation
- Communications

**Build 2**
- C2/C3 Systems operation
- GPS auto emplacement
- Search/Track
- HW Links, Interfaces

**Build 3**
- System Monitoring
- System Modes

**Build 4**
- Engagements

**System Integration Planning**

*Capability Performance Parameters:* define and build upon a set of systems integration tests for continuous base-lining and regression tests by Incremental build.
CPPs (Capability Performance Parameters)

**CPP1**: Establish basic system operating system-applications scheduling functions, system execution, display menus/switches/mouse functionality, system initialization for site adaptation, initial communications links, data collection

**CPPs**: Build upon CPP1, to create full regression test suite for system base-lining, regression throughout the remaining development, integration, and system test phases, DTE readiness

**New Requirements**

**CPP1-CPP5**: Establish system, displays, initialization, search/track, engagement functions, ID changes, C3 link communications, Status monitor, full data collection
Systems Integration Planning

Development Team

Build 1
Build 2
Build 3
Build 4

Functional Integration Team

- Infrastructure
- Operating systems
- Low level Interfaces
- Protocols
- Display
- Status Monitor
- Communications
- Surveillance
- Track
- Radar Interface
- Engagements
- Guidance
- Contingencies
- C3 systems

Systems Integration Team

Systems Team Requirements
V & V Team
Test Site Teams

SYSTEM CPP INTEGRATION TEST PLAN,
SYSTEM TEST, TEST SITES
The CMMI Product Integration process area describes system integration strategies supporting effective Test and Evaluation execution, such as:

- **Setup a team, identify stakeholders, team roles & responsibilities:**
  - Systems Engineering
  - Software Engineering
  - Test Engineering
  - PMO

- Establish a System integration plan via “system capability” testing, tracking, and capability/limitation measurement techniques through base-lining, regression, & auto testing

- Coordination of integration, V & V, system Test, and operational Test “system shall” test coverage, breadth & depth

- Establish and utilize entrance and exit criteria disciplines to form the basis of readiness review meetings and critical release decisions

- Establishment of integrated lab facility and test site resources, Build/Test tools, and an instrumentation/analysis logistics plan

- Execute an incremental plan for achieving “full” system level integration by testing with “live” hardware, with a planned mix of simulators in the total system

A C C E S S  |  I N S I G H T  |  E X P E R I E N C E
Systems Integration Execution & Test Configurations
Systems Integration Execution

Build 1
CPPs: Capability Performance Test Execution via Systems Integration Test Plan

Test Design
Run Regr1

Build 2

Test Design
Run Regr2

Build 3

Test Design
Run Regr3

Build 4

Test Design
Run Regr4

System Integration

Performance Measurement,
Requirements Divergence

TESTPLANT
System base-lining, regression, automated testing

Fixes, Integration updates Requirements updates

CPP1
CPP2

CPP1
CPP2
CPP3

CPP1
CPP2
CPP3
CPP4

CPP1
CPP2
CPP3
CPP4
CPP5
System base-lining, Regression testing automated testing strategies
System Base-Lining, Regression test

U.S. patent 7,870,504. Other patents pending.
Use TESTPLANT automated testing tools to integrate and test C3-C2 configurations, detect errors, interface & link problems, and operational anomalies through system base-lining, regression and systems integration automated tests. Utilize operator panel scripts and scenarios for designing tests, executing automated tests, verifying system execution results through display image recognition capabilities, and test reporting and analysis tools.
TESTPLANT: Automated testing

Remote non-invasive testing
...anywhere

U.S. patent 7,870,504. Other patents pending.
## TESTPLANT Capabilities: CPP Testing

### SCRIPTING
- Auto Test Designs via Image, Mouse, Switch, Keyboard capturing, scripting

### Auto Execution
- Auto Test Execution via display scripts, scenario events, designed sequences

### TEST Verification
- Verify executed Test events via captured real time displays, image recognition capabilities

### Analysis
- Generate Test Reports from Tests run, events, results

<table>
<thead>
<tr>
<th>SCRIPITING</th>
<th>Auto Execution</th>
<th>TEST Verification</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Auto Test Designs</strong> via Image, Mouse, Switch, Keyboard capturing, scripting</td>
<td><strong>Auto Test Execution</strong> via display scripts, scenario events, designed sequences</td>
<td><strong>TEST Verification</strong> of executed Test events via captured real time displays, image recognition capabilities</td>
<td><strong>Analysis</strong> of generated Test Reports from Tests run, events, results</td>
</tr>
</tbody>
</table>

- **Automatic script creation**
- **Script DISPLAY commands, & Test event sequences**
- **Create events on captured images, contingencies**
- **Script elaboration, editing, tailoring**
- **Loops, delays, pause, continue**
- **Expected Results**
- **Build in Auto-Image Verification tests, Pass/Fail**
- **Saved, organize into suites, scenarios, REGR tests, functional libraries**

- **Perform Actions on images in scripts**
- **Mouse actions**
- **Keyboard actions**
- **Switch actions**

- **Script, suite, library, scenario selection**
- **Test scheduling**
- **Test management**
- **CM organization**
- **Execution storage of data**
- **Provide repeatability base-lining & regression testing for dynamically changing configurations of Software/Hardware updates/fixes**

- **Display screens sent back to eggplant**
- **Auto verify expected results built into scripts**
- **PASS/FAIL, bug detection**
- **Save/store results, screens**
- **STOP or Continue test execution**

- **Test displays, screens saved**
- **Resultant actions during test**
- **Test action timing data**
- **Test results vs. expected**
- **Pass/Fail data saved**
- **Script/Suites ID run, date/time**
- **Correlate Test Results, Reports to operational data**
Systems Integration with TESTPLANT Automated Testing

CPP Test Designs

- Script selection
- Test Repeatability
- Base-lining, Regression Tests

Scripted Operator inputs

TESTPLANT CPPs

Databases

Continuous and dynamic new Reqts, Software build updates, error fixes, enhancements

Continuous Base-lining and Regression testing via Test designs/scripts

Auto Test execution, Test verification, Test Results

Detect Software errors, link faults, systems anomalies

Data Collection

SIMULATORs, RADARs, SYSTEMs
TESTPLANT System Integration/Automated testing in Aviation Systems

Scripts, suites, libraries -> TESTPLANT CPPs -> Auto execution, test verification, test results

Display Screens Recognition

Pilot keyboard and menu inputs

Co-pilot keyboard and menu inputs

Databases Navigation performance user, terrain

Data collection

FMS
- VNAV
- perf
- speed
- LNAV
- flight plan
- position

Pull down menu scenarios

TestPlant System Integration/Automated testing in Aviation Systems
T & E readiness evaluation and assessments
T & E Readiness Evaluation & Assessment

BUILD Fixes, Integration Requirements updates

Build 5.1
Build 5.2
Build 5.3

System Integration

Performance & Capability Measurement
Requirements Divergence

CPP1  CPP5
CPP2  CPP6
CPP3  CPP7
CPP4  CPP8

PDRS
IPRS

Contractor, Government, DTE Team assessments

V & V Testing
System Testing at Sites

INTEROP tests

DTE/OTE

ACCESS  |  INSIGHT  |  EXPERIENCE
Summary
Recommendations/Solutions

**Effective Test & Evaluation**
- Contract requirements
- Operational requirements
- Define the Divergence
- Full System Capabilities (>KPP)

**Performance Measurement**
- System base-lining
- Regression testing
- Automated Testing, Repeatability
  -- TESTPLANT
- Early Visibility into operational capability
- Problems, limitations understood

**T & E Agility**
- Evolving requirements
- Adaptability, base-lining
- in Program Life Cycles

**Systems Integration**
- Full System Capabilities
- Measure capability
- Track, Report
- PDRs review
- Compliance Matrix

**CPPs**
- DODD 5000
- DODI 5000
- FAR

**Early Visibility Into Operational Capability**
Contact Information

Name: Robert Koczat

Phone: 603 - 560 - 1687

Company: The SPECTRUM Group

Email: Robert.Koczat@comcast.net
       Bkoczat@spectrumgrp.com