

# An Integrated Test Environment for Systems Engineering

Ryan Brooks  
Roberto Escobar

Boeing  
Mesa, AZ

# What is OSEE's OTE Framework?

- The **Open System Engineering Test Environment (OTE)** is a framework for requirements-based testing in the context of an overall systems engineering approach
- OTE has been used commercially to qualify mission software for Boeing's next generation Apache Attack Helicopter
- Being contributed as a component of the Open System Engineering Environment (OSEE) Eclipse project
- Facilitates seamless flow between test development, debugging, execution, and result analysis

# Integrated Requirements-Based Testing

- Bidirectional traceability between software requirements, application code, and tests is provided through OSEE Application Framework
- Coverage holes detection via traceability
- Action Tracking System identifies test impacts driven by requirement changes
- Test development status tracking
- Detailed, integrated test status and coverage reporting

# Architecture

Dynamic Service Discovery

JINI Lookup

Relational Database

Remote File System

OSEE Datastore

OSEE Application Server

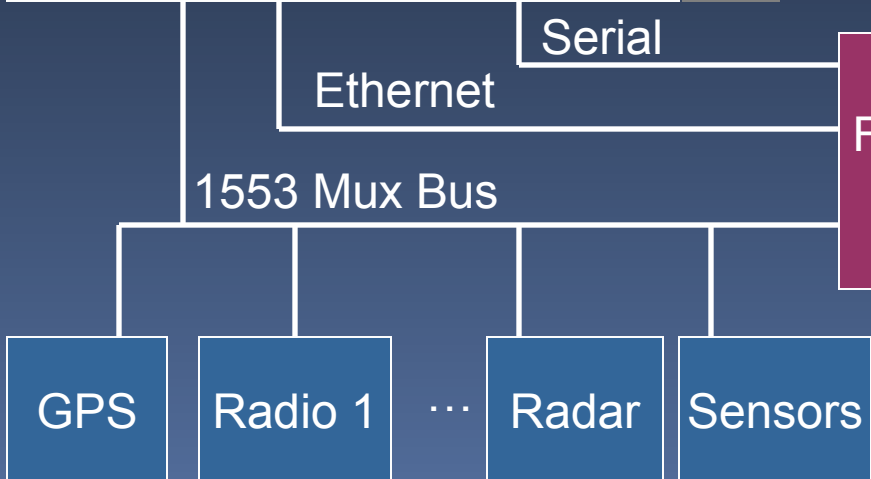


OSEE Test Server

RMI



OSEE Test Client



Unit under test containing embedded flight code

# Test Environment Service

- Provides dynamic lookup of resources
- Provides both soft real-time and simulated capabilities
- Schedules periodic execution of models (simulation components)
  - API for easy creation of simulated components
- Supports a user configurable number of simultaneous client connections
- Manages I/O and testing resources

# Extensible Real-time Messaging System

- Supports communication with real hardware via
  - MIL-STD-1553 MUX
  - Serial
  - Wire
  - Analog and Digital discretetes
  - Publish/Subscribe Data Distribution Service (DDS) through Ethernet

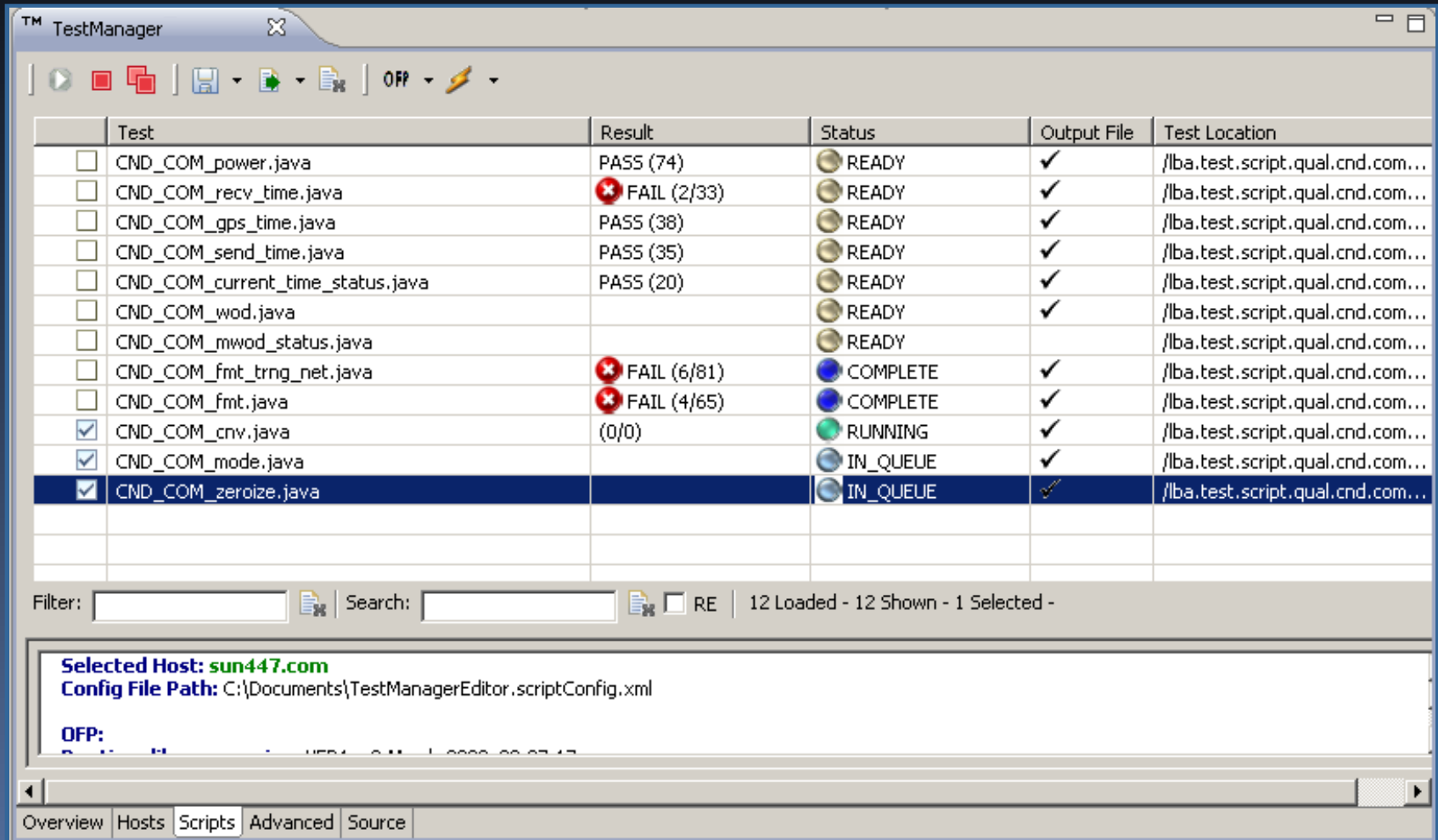
# Real-time & Simulated Functional Testing

- Simulated Environment (eases demand on limited test station hardware resources)
- Tests (without modification of any kind) can be run in both soft real-time and simulated environments (simultaneously, if desired)
- Simulated components can be used with both environments
- Streaming automated test point tally and rollup of pass/fail determination
- Interactive Testing (automated tests with user input)

# Test Manager

- Provides a common interface for functional tests in real-time and simulated environments across all levels of testing fidelity
- Test results streamed in real-time from test service to test manager
- Create, save, and load run lists
- Integrated with JDT/CDT debugger
- Lists available test services that can be used to run test files
- Real-time display of test service usage information
- Supports the execution of multiple simultaneous batches within a single workspace

# Test Manager



The screenshot shows the Test Manager application window with a table of test results. The table has columns for Test, Result, Status, Output File, and Test Location. The tests listed include CND\_COM\_power.java, CND\_COM\_recv\_time.java, CND\_COM\_gps\_time.java, CND\_COM\_send\_time.java, CND\_COM\_current\_time\_status.java, CND\_COM\_wod.java, CND\_COM\_mwod\_status.java, CND\_COM\_fmt\_trng\_net.java, CND\_COM\_fmt.java, CND\_COM\_cnv.java, CND\_COM\_mode.java, and CND\_COM\_zeroize.java. The results show various statuses such as PASS, FAIL, READY, COMPLETE, RUNNING, and IN\_QUEUE.

	Test	Result	Status	Output File	Test Location
<input type="checkbox"/>	CND_COM_power.java	PASS (74)	READY	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_recv_time.java	✗ FAIL (2/33)	READY	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_gps_time.java	PASS (38)	READY	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_send_time.java	PASS (35)	READY	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_current_time_status.java	PASS (20)	READY	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_wod.java		READY	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_mwod_status.java		READY		/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_fmt_trng_net.java	✗ FAIL (6/81)	COMPLETE	✓	/lba.test.script.qual.cnd.com...
<input type="checkbox"/>	CND_COM_fmt.java	✗ FAIL (4/65)	COMPLETE	✓	/lba.test.script.qual.cnd.com...
<input checked="" type="checkbox"/>	CND_COM_cnv.java	(0/0)	RUNNING	✓	/lba.test.script.qual.cnd.com...
<input checked="" type="checkbox"/>	CND_COM_mode.java		IN_QUEUE	✓	/lba.test.script.qual.cnd.com...
<input checked="" type="checkbox"/>	CND_COM_zeroize.java		IN_QUEUE	✓	/lba.test.script.qual.cnd.com...

Filter:  Search:  RE  12 Loaded - 12 Shown - 1 Selected -

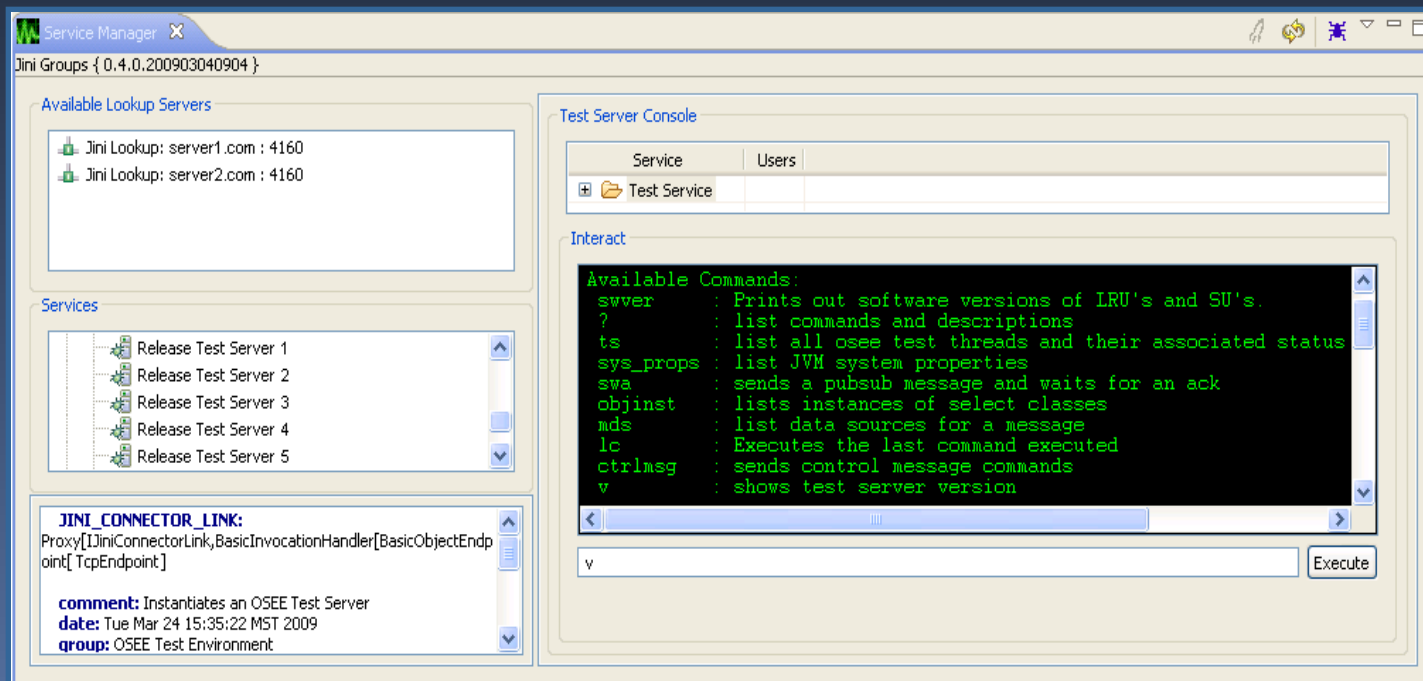
**Selected Host:** sun447.com  
**Config File Path:** C:\Documents\TestManagerEditor.scriptConfig.xml

**OFF:**

Overview Hosts **Scripts** Advanced Source

# Service Manager

- Integrated test server resource management
- Remote viewing and manipulation of test service



The screenshot displays the Service Manager application window. The title bar reads "Service Manager" and the main window title is "Jini Groups { 0.4.0.200903040904 }".

**Available Lookup Servers:**

- Jini Lookup: server1.com : 4160
- Jini Lookup: server2.com : 4160

**Services:**

- Release Test Server 1
- Release Test Server 2
- Release Test Server 3
- Release Test Server 4
- Release Test Server 5

**JINI\_CONNECTOR\_LINK:**

Proxy[IJiniConnectorLink, BasicInvocationHandler[BasicObjectEndpoint[TcpEndpoint]]

**comment:** Instantiates an OSEE Test Server  
**date:** Tue Mar 24 15:35:22 MST 2009  
**group:** OSEE Test Environment

**Test Server Console:**

Service	Users
Test Service	

**Interact**

**Available Commands:**

```
swver : Prints out software versions of LRU's and SU's.  
? : list commands and descriptions  
ts : list all osee test threads and their associated status  
sys_props : list JVM system properties  
swa : sends a pubsub message and waits for an ack  
objinst : lists instances of select classes  
nds : list data sources for a message  
lc : Executes the last command executed  
ctrlmsg : sends control message commands  
v : shows test server version
```

Input field: v      Execute

# Test Output

- Test output file format is XML
- Interactive out files (user can navigate the out file)
- User selectable views of out files
- Automatic correlation of a run-time test point to the test source line that generated it (integrated into JDT using problem view and markers)
- Full test output can be stored in the OSEE object-oriented persistence layer for later analysis and results summary and reporting

# Test Output Editor

✓ lba.test.script.qual.acs.asm.ACS\_ASM\_air\_data\_display.d699ceaa947201efbbbdfdf89df03b1.tmo

## OTE Test Output

▶ Test Overview

▼ Results Summary - FAILED - Pass[5] Fail[13]

Title	Expected	Actual	Elapsed Time
● [124] - TP 1 { ASM_ACSYS_MODES.FAT_OUT_OF_TOL }	IN_TOLERANCE [0]	IN_TOLERANCE [0]	0
● [125] - TP 2 { ASM_ACSYS_MODES.FAT }	10(0xA)	128(0x80)	0
● [126] - TP 3 { ASM_ACSYS_MODES.FAT_SIGN }	MINUS [0]	PLUS [1]	0
● [133] - TP 4 { ASM_ACSYS_MODES.FAT_OUT_OF_TOL }	OUT_OF_TOLERANCE [1]	IN_TOLERANCE [0]	1002
● [135] - TP 5 { ASM_ACSYS_MODES.FAT }	10(0xA)	128(0x80)	1000
● [136] - TP 6 { ASM_ACSYS_MODES.FAT_SIGN }	PLUS [1]	PLUS [1]	0
● [145] - TP 7 { ASM_ACSYS_MODES.PA_OUT_OF_TOL }	IN_TOLERANCE [0]	IN_TOLERANCE [0]	0
● [146] - TP 8 { ASM_ACSYS_MODES.PA_10000_S }	ZERO [0]	ZERO [0]	0
● [147] - TP 9 { ASM_ACSYS_MODES.PA_1000_S }	2(0x2)	0(0x0)	1001
● [148] - TP 10 { ASM_ACSYS_MODES.PA_100_S }	6(0x6)	0(0x0)	1001
● [149] - TP 11 { ASM_ACSYS_MODES.PA_10_S }	7(0x7)	0(0x0)	1000
● [150] - TP 12 { ASM_ACSYS_MODES.PA_SIGN }	MINUS [0]	MINUS [0]	0
● [159] - TP 13 { ASM_ACSYS_MODES.PA_OUT_OF_TOL }	OUT_OF_TOLERANCE [1]	IN_TOLERANCE [0]	1001
● [161] - TP 14 { ASM_ACSYS_MODES.PA_10000_S }	TWO [2]	ZERO [0]	1001
● [162] - TP 15 { ASM_ACSYS_MODES.PA_1000_S }	6(0x6)	0(0x0)	1001
● [163] - TP 16 { ASM_ACSYS_MODES.PA_100_S }	5(0x5)	0(0x0)	1001
● [164] - TP 17 { ASM_ACSYS_MODES.PA_10_S }	4(0x4)	0(0x0)	1001
● [165] - TP 18 { ASM_ACSYS_MODES.PA_SIGN }	PLUS [1]	MINUS [0]	1001

Filter:  | Search:  | RE | 18 Loaded - 18 Shown - 0 Selected -

▶ Ofp Log

▶ Ofp Version

▶ Ote Log

Overview | Details

Test points linked to test code that generated

# Messaging and Playback GUIs

- Monitor, manipulate and record real-time messaging data
- Advanced regular-expression searching for messages
- Import/Export view lists
- Playback environment
  - Supports advanced debugging of the test environment using repeatable conditions
  - Investigate real flight recordings

The screenshot shows the MessageWatch application window. The status bar indicates "Status: Connected" and "Message Jar: message JAR is loaded". There are "REC" and "CSV" buttons. The main area contains a table with the following data:

Name	Updates	Value	Offset	MSB	LSB	Bit Size	Type
ACFT_SYSTEM_STATUS	2						
ENG1_TORQUE		0	0	0	7	8	IntegerElement
ENG2_TORQUE		5	4	0	7	8	IntegerElement
ACFT_SYSTEMS_STAT	0						
LATITUDE		0.0	26	0	31	32	FixedPointElement
LONGITUDE		0.0	22	0	31	32	FixedPointElement
ACFT_POSN_SEL	0						

Below the table, it says "3 Loaded - 7 Shown - 1 Selected". A "Filter:" input field is present. The bottom panel shows the details for the selected element:

```
Element Name: ENG1_TORQUE
Value: 0
Byte Offset: 0
```

# Questions?