Test and Training Enabling Architecture (TENA)

Gene Hudgins
TENA User Support Lead
Agenda and Objectives

- The TENA Architecture
  - Architecture Structure
  - Architecture Details
    - Meta-Model
    - Object Model
    - Middleware
    - Repository and Utilities
  - JMETC
    - JMETC Purpose
    - JMETC Accomplishments
    - JMETC Products
    - JMETC Plans

- Learning Objectives
  - The TENA approach to interoperability
  - The basic elements of TENA
  - The basic advantages of TENA
  - What to do to get more information on TENA
  - The new enterprise-wide approach to joint testing (JMETC)
  - JMETC’s commitment to using TENA for its basic technical approach

- Summary and Conclusions
Test Resource Management Center

“We are the stewards of the DoD test and evaluation (T&E) infrastructure”

VISION

“The Department of Defense T&E Ranges & Facilities will be fully capable of supporting the Department with quality products and services in a responsive and affordable manner”

GOAL TO ACHIEVE THE VISION:

Robust and Flexible T&E Capabilities to Support the Warfighter
Test Resource Management Center
Sec. 231, FY 2003 National Defense Authorization Act
DoD Directive 5105.71, March 8, 2004

- DoD Field Activity
- Direct Report to USD(AT&L)
  ⭐⭐⭐⭐ SES Director

- Oversee T&E Budgets
  - MRTFB
  - Other T&E Facilities Within & Outside DoD

- Biennial 10-Year Strategic Planning

- Administer T&E Investment Programs
  (CTEIP & T&E/S&T)

- Annual T&E Budget Certification
  Military Departments & Defense Agencies

- MRTFB Policy Oversight
Test Resource Management Center (TRMC) Organization

UNDER SECRETARY OF DEFENSE (ACQUISITION, TECHNOLOGY AND LOGISTICS)
Honorable Ashton B. Carter

Director
Dr. John B. Foulkes

Principal Deputy
Derrick Hinton

Deputy Director
Test Resources
Kevin Hannon

Deputy Director
Joint Investment Programs & Policy
Derrick Hinton

Deputy Director
Test Infrastructure
Fred Myers

Deputy Director
Strategic Planning
Suzanne Strohl

Executive Office
(Chief Fin Officer, Senior Mil Asst, Program Support, Executive Support, IT, etc)

Policy
Sheila Wright

T&E/S&T Program
George Rumford

CTEIP
Gerry Christeson

JMETC Program
Chip Ferguson
Test and Training Enabling Architecture

(TENA)
TENA Mission

Currently, range systems tend to be non-interoperable, “stove-pipe” systems

The purpose of TENA is to provide the architecture and the software implementation necessary to

- Enable Interoperability among range systems, facilities, simulations, C4ISR systems in a quick, cost-efficient manner, and
- Foster Reuse for range assets and for future developments

- Support the warfighter (Joint Vision 2010/2020)
- Enable simulation-based acquisition
- Foster test and training integration
- In the long term: SAVE MONEY!

Lay the Foundation for Future Test and Training Range Instrumentation
Where TENA SDA Fits in DoD
TENA Development Strategy

- TENA is revised based on user feedback and lessons learned from working software implementations.
- TENA will be revised in the future based on future implementations.

*TENA is based on real-world tests at real ranges.*
The Software Engineering Institute defines an Open System as “a collection of interacting software, hardware, and human components designed to satisfy stated needs with interface specifications of its components that are fully defined, available to the public, maintained according to group consensus, in which the implementations of the components conform to the interface specifications.”

TENA is maintained according to a consensus of its users assembled as the TENA Architecture Management Team (AMT)

TENA Middleware exists and is being used to support real events
- Government owned, without proprietary software

TENA is freely releasable (Distribution A) to non-US entities
- We have many non-US users in Britain, France, Sweden, Denmark, etc.

Currently there are no plans for standardizing TENA in the same way as DIS and HLA have been standardized (IEEE)
- However, we are looking into innovative mechanisms to get the same usability and confidence with TENA as we do with open standards
- TENA’s business model is not the same as the DIS and HLA business models
TENA Architecture Management Team (AMT)

- AMT: A technical forum providing an open dialogue between users and TENA developers to understand current issues and agree on solutions
  - Provide more insight to current capabilities and on emerging technical challenges to discuss a common approach to satisfying requirements

- AMT Process: Identify issues, vet concerns, debate potential solutions, and agree on a way forward with active participation from all stakeholders
  - TENA maintained according to consensus of its users assembled as the AMT
  - TENA design and improvements driven from the AMT exchange
  - Current improvements to the next release of TENA are a direct result from vote by AMT members at previous meeting
  - Industry heavily participates at AMT meetings

- AMT Schedule: Quarterly (Next AMT Meeting 29 July 2010 in Seattle, WA)

Registration Open on https://www.tena-sda.org
Architecture Management Team (TENA AMT)

- **AMT Members:**
  - 329 Armament Systems Group (329 ARSG)
  - Aberdeen Test Center (ATC), Aberdeen Proving Ground, MD
  - Air Armament Center (AAC), Eglin AFB, FL
  - Air Force Flight Test Center (AFFTC), Edwards AFB, CA
  - Army Operational Test Command (OTC), Fort Hood, TX
  - Common Training Instrumentation Architecture (CTIA)
  - Dugway Proving Ground (DPG)
  - Electronic Proving Ground (EPG)
  - integrated Network Enhanced Telemetry (iNET)
  - Interoperability Test and Evaluation Capability (InterTEC)
  - Joint Fires Integration & Interoperability Team (JFIIT)
  - Joint National Training Capability (JNTC)
  - Naval Air Warfare Center – Aircraft Division
  - NAWC – Weapons Division
  - Naval Aviation Training Systems Program Office (PMA-205)
  - Naval Undersea Warfare Center (NUWC)
  - NAVSEA Warfare Center - Keyport
  - P5 Combat Training System (P5CTS)
  - Pacific Missile Range Facility (PMRF)
  - Redstone Technical Test Center (RTTC)
  - T&E/S&T Non-Intrusive Instrumentation
  - White Sands Missile Range (WSMR)

- **Meetings every 3 months**

- **US Advising Members:**
  - BMH Associates, Inc.
  - Boeing
  - Cubic Defense
  - DRS
  - Embedded Planet
  - EMC
  - Kenetics
  - MAK Technologies
  - NetAcquire
  - Science Applications International Corporation (SAIC)
  - Scientific Research Corporation (SRC)
  - Scientific Solutions, Inc. (SSI)

- **International Participation**
  - Australia
  - Denmark
  - France
  - Singapore
  - Sweden
  - United Kingdom

What is an Architecture?

- An architecture is a segmentation of a system (or system of systems) such that the primary pieces are identified, as well as their purpose, function, interfaces, and inter-relatedness, along with guidelines for their evolution over time.

- Architectures put constraints on developers. These constraints make possible the achievement of higher level goals.

- These higher-level goals are called the system’s driving requirements.

- An architecture is a bridge from requirements to design.
TENA Architecture Overview
1. **Interoperability**
   - The characteristic of a suite of independently-developed components, applications, or systems that implies that they can work together, as part of some business process, to achieve the goals defined by a user or users.

2. **Reusability**
   - The characteristic of a given component, application, or system that implies that it can be used in arrangements, configurations, or in enterprises beyond those for which it was originally designed.

3. **Composability**
   - The ability to rapidly assemble, initialize, test, and execute a system from members of a pool of reusable, interoperable elements.
   - Composability can occur at any scale—reusable components can be combined to create an application, reusable applications can be combined to create a system, and reusable systems can be combined to create an enterprise.
Achieving Interoperability and Reuse

- **Interoperability requires**
  - A common architecture
  - An ability to meaningfully communicate
    - A common language
    - A common communication mechanism
  - A common context
    - A common understanding of the environment
  - A common understanding of time
  - A common technical process

- **Reuse and Composability require the above, plus**
  - Well defined interfaces and functionality for the application to be reused
  - Place to store reusable components
TENA Compliancy Levels

**TENA Level 1**
- Uses the TENA Middleware
- Defined as TENA Objects

**TENA Level 2**
- Standard use and definition of Time
- Only uses the TENA Middleware

**TENA Level 3**
- Data Archiving (when available)
- Uses Standard Objects (whenever possible)
- Standard Control

Each level describes the usage and definition of TENA objects, with increasing complexity and features.
TENA specifies an architecture for range resources participating in logical ranges.
Logical Range
Simple Example

- TENA specifies a peer-to-peer architecture for logical ranges:
  - Applications can be both clients and servers simultaneously
  - In their role as servers, applications serve TENA objects called “servants”
  - In their role as clients, applications obtain “proxies,” representing other applications' servants. Only servers can write to their servant objects’ publication state

- The TENA Middleware, the TENA objects, and the user’s application code are compiled and linked together
What is a Meta-Model, and Why is it Important?

**What is a Meta-Model?**

- A meta-model is “a model that defines an abstract language for expressing other models,” from Common Warehouse Metamodel specification by Dr. Daniel T. Chang.
- All computer languages have meta-models
- The TENA Meta-Model describes the features of objects defined in a logical range object model (LROM)

**Why is it important?**

- The TENA Meta-Model is the architectural construct that specifies both the rules for defining an LROM and the requirements for the middleware
Requirements for Defining the TENA Meta-Model

- Must support distributed computing
- Must be rich enough in features to support the object modeling needs of the entire test and training range community
  - Objects and Messages
- Must provide a semantic unification of information amenable to the creation of a simple, yet powerful, standard TENA Object Model
- Must be as easy to use and understand as possible given the above requirements

These requirements led to the invention of the **Stateful Distributed Object**, combining the best features of CORBA and the HLA in one easy-to-use concept.
Stateful Distributed Objects (SDOs)

- An SDO is a combination of two powerful concepts:
  - a distributed object paradigm (like the one used in CORBA)
  - a distributed publish and subscribe paradigm

- Benefits of this combination:
  - A conventional distributed object-oriented system offers no direct support to the user for disseminating data from a single source to multiple destinations
  - A conventional publish-subscribe system does not provide the abstraction of objects with a set of methods in their interface
  - Interface to SDOs is a lot simpler and more usable than the combination of interfaces to their underlying technologies
Clients and Proxies, Servers and Servants

- Remote Method Invocation

Client Application

Proxy Object on Client

- User Application
- Proxy for Object 27
- Remote Interface
- Publication State
- Cache
- Local Methods Interface
- Local Methods Implementation

Server Application

Servant Object on Server

- Object 27
- Remote Interface
- Remote Interface Implementation
- Publication State
- Local Methods Interface
- Local Methods Implementation

TENA Middleware

Network
Clients and Proxies, Servers and Servants

- Publication State Dissemination and Access
Clients and Proxies, Servers and Servants

- Local Methods used on both Client and Server
TENA Objects are Compiled In

● Why use compiled-in object definitions?
  ● Strong type-checking
    ● Don’t wait until runtime to find errors that a compiler could detect
  ● Performance
    ● Interpretation of methods/attributes has significant impact
  ● Ability to easily handle complex object relationships
  ● Conforms to current best software engineering practices

● How do you support compiled-in object definitions?
  ● Use a language like CORBA Interface Definition Language (IDL) to define object interface and object state structure
  ● Use code generation to implement the required functionality

● Thus the concept of the TENA Definition Language (TDL) was created
  ● Very similar to IDL and C++
TENA Object Models

- Enable semantic interoperability among range resource applications
- Provide the “common language” that all range resource applications use to communicate

Object Model Stages

- **User-Defined Objects** – objects defined solely for the purpose of a given logical range by TENA users
- **TENA Candidate Objects** – objects defined as potential standards, which are undergoing test and evaluation by the community prior to standardization
- **TENA Standard Objects** – objects developed and supported by the TENA SDA, which have been approved for standardization by the AMT
JMETC uses standard TENA Object Models to define a standardized interface between test platforms.

- **Current Object Models:**
  - TENA TSPI
  - TENA Platform
  - TENA AMO
  - Radar Object
  - GPS-Based System Object
  - TENA Engagement

- **Future Object Models:**
  - TENA Measurand
  - TENA Weather Server
  - Tactical Message Sets
  - Time Management
  - Middleware Management
**Case 1: Reading and writing in the same coordinate system**

- **Client Application**
  - User Application
  - Geocentric-Position
  - Geocentric SRF
  - get_geocentric Position() method

- **Server Application**
  - User Application
  - Geocentric-Position
  - Geocentric SRF
  - set_geocentric Position() method

**Diagram Description**
- TSPI with Coordinate Conversions
- Platform 27
- Local Methods Interface
- Position
- Private data
- Coordinate Conversions Local Methods
- TENA Middleware
- Network
Case 2: Reading and writing in different coordinate systems

- Write in Geocentric (ECEF), read in Geodetic (latitude/longitude/altitude)
TENA Common Infrastructure

**Components:**
- Repository
- Logical Range Data Archive
- Middleware

**Purpose:**
- Provide the common, standardized, software mechanism that makes communication about objects in the TENA Object Model as efficient and simple as possible throughout the entire range event lifecycle
**Purpose:** to contain all the information relevant to TENA that is not specific to a given logical range

**Current Repository Contents:**
- All TENA Object Models, both standard and user-designed
- All TENA software (middleware, helpdesk cases, tools, gateways, reusable applications, and reusable components)
- All TENA documentation
- Lessons learned from previous uses of TENA
- Provide an easy-to-use secure interface to all of this information

**The Repository is a collection of technologies based around a wiki-like front end**
TENA Web Portal  
http://www.tena-sda.org/

- Registered user account required
- Contains
  - News
  - Meeting Notices
  - Documentation
  - Middleware
  - Object Models
  - Training Materials
TENA Middleware
Purpose and Requirements

- **Purpose**: high-performance, real-time, low-latency communication infrastructure used by range resource applications and tools during execution

- **Requirements**:
  - Fully support TENA Meta-Model
  - Be easy to use and highly reliable
  - Many varied communication strategies and media
    - Including management of quality-of-service
    - Including object-level security services
  - Be high-performance, including
    - Support multiple information filtering strategies
    - Support user-defined filtering criteria
  - Support a wide variety of range-relevant platforms (hardware/operating system/compiler)
  - Be technology neutral

Middleware Release 6 available for Download
### Key Release 6 Improvements and New Capabilities

<table>
<thead>
<tr>
<th>New Middleware Capabilities</th>
<th>Metamodel and Model Improvements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced Filtering</td>
<td>Fundamental Sized Type Aliases</td>
</tr>
<tr>
<td>OM Subsetting Support</td>
<td>Const Qualifier</td>
</tr>
<tr>
<td>SDO State Processing Support</td>
<td>Optional Attributes</td>
</tr>
<tr>
<td>Self-Reflection Option</td>
<td>SDO Initializers</td>
</tr>
<tr>
<td>Object Reactivation</td>
<td>Middleware Metadata</td>
</tr>
<tr>
<td>Separate Inbound/Outbound ORBs</td>
<td>Middleware IDs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>New Event Management Capabilities</th>
<th>Usability Improvements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object Model Consistency Checking</td>
<td>Observer Pattern</td>
</tr>
<tr>
<td>Remote Object Termination</td>
<td>(with Callback Aggregation)</td>
</tr>
<tr>
<td>Execution Manager Fault Tolerance</td>
<td>Local Methods Factory Registration</td>
</tr>
<tr>
<td>Embedded Diagnostics</td>
<td>Code Installation Layout</td>
</tr>
<tr>
<td>TENA Console</td>
<td></td>
</tr>
</tbody>
</table>

37
## Key Release 6 Improvements and New Capabilities

### New Middleware Capabilities
- Advanced Filtering
- OM Subsetting
- Object Model Consistency Checking
- Remote Object Termination
- Separate Inbound/Outbound ORBs

### Metamodel and Model Improvements
- Fundamental Sized Type Aliases
- Const Qualifier
- Optional Attributes
- SDO Initializers
- Middleware Metadata
- Middleware IDs

### New Event Management Capabilities
- Observer Pattern (with Callback Aggregation)
- Local Methods Factory Registration
- Code Installation Layout

### Usability Improvements
- Observer Pattern
- Remote Object Termination
- Execution Manager Fault Tolerance
- Embedded Diagnostics
- TENA Console
- Enhanced data distribution
- Optimized network usage
- Better ways to define data
- Remove ambiguity
- Improved reliability
- Enhanced troubleshooting
- Easy to use
- Harder to use wrong

### Usability Improvements
- Enhanced data distribution
- Optimized network usage
- Better ways to define data
- Remove ambiguity
- Improved reliability
- Enhanced troubleshooting
- Easy to use
- Harder to use wrong
The capability to restore an SDO servant after a crash, such that subscribing applications consider updates to apply to the existing proxy they hold.

Notionally:

// Before crash
std::string filename = "C:Servant01.save"
pServant->save(filename)

// After crash
// Recreate the servant
std::auto_ptr<OMSample::Platform::ReactivationInitializer> rinit(
  pSF->createReactivationInitializer(filename) );
rinit->set_...  // reset the initial values
pServant = pSF->reactivateServant( pRemoteMethods, rinit );

This recreates the Servant with identical SDO ID as before.
Example - Advanced Filtering

- SDOs and Messages can now be assigned an integer “tag”
- Users can decide what these integer tags mean
- Subscribers can now subscribe to type/tag combinations
  - Subscribe to all Platforms –OR–
  - Subscribe to Platforms with tag 1

- Assign an initial tag:
  
  ```c++
  unsigned long tag = 1;
  pPlatformServant = pServantFactory->
      createServantUsingDefaultFactory( *initializer,
          DIME::Interest::Metadata(tag), communicationProperties );
  ```

- API for changing tags on servant not set yet
- Subscribe to SDO with a tag:
  
  ```c++
  unsigned long tag = 1;
  pSession->subscribeToSDO< OMsample::Platform::ProxyTraits >(pPlatformStrategy,
      DIME::Interest::Metadata(tag) );
  ```
Release 6
Platform Support

- Linux Fedora Core 6
- Linux Fedora Core 6, 64-bit
- Linux Fedora 12
- Linux Fedora 12, 64-bit
- Linux Red Hat Enterprise Workstation 4
- Linux Red Hat Enterprise Workstation 4, 64-bit
- Linux Red Hat Enterprise Linux 5.2
- Linux Red Hat Enterprise Linux 5.2, 64-bit
- Linux openSUSE 11.latest
- MAC OS X 10.6, Snow Leopard (Intel 64-bit)
- Windows XP, Visual Studio 2005
- Windows Server 2003, 64-bit, Visual Studio 2005
- Windows XP, Visual Studio 2008
- Windows Vista, Visual Studio 2008
TENA Application Architecture

- **Purpose:** Explains how applications should be built
- Emphasizes that the middleware and the LROM are linked into the application

**APPLICATION CODE:**
Specific to an individual application

**OBJECT MODEL CODE:**
Common across a given logical range

**TENA CODE:**
Common across all TENA applications
Creating a TENA Application

1. LROM object definitions

2. LROM object implementations

3. User Application code

Auto-generated

Created by the logical range developers

Created/modified by the range resource developers

TENA Middleware Library

Application Object Code

LROM Object Library

Compiler

Linker

Code Generator

Generated LROM Definition Source Code

TENA Application

User Application Code

Servant

Proxy

TENA Middleware
Auto-Code Generation With TENA

- Our desire is for the input to the TENA auto-code generator be standard XMI (generated from UML)
- Challenges: XMI not yet implemented in a standard way by tool vendors, and current auto-code generation capability is based on TDL
- Current Interim Solution – Use MagicDraw plug-in to create TDL from UML
- Next Steps
  - Implement TENA Metamodel in Eclipse Modeling Framework using ECore representation – define TENA Modeling Language (TML)
  - Create XMI ↔ TML, TDL ↔ TML translators
  - API and framework being developed to support various “code generation plugins” used to automatically create specialized code based on FreeMarker templates
TENA Integrated Development Environment (TIDE)

- **TIDE** is a tool designed to assist developers in the creation, development, testing and deployment of TENA applications.

- **Initial Capabilities**
  - Catalog installed object models on a user’s machine
  - Migrate user applications between object model versions
  - Migrate user applications between middleware versions
  - Browse and download object models available in the TENA Repository
  - Request object model distributions from the TENA Repository

- **TIDE 2.0 is the current version**
  - Available at [http://www.tena-sda.org/tide](http://www.tena-sda.org/tide) web site
Gateway Builder

- MSR Program is focused on integration of distributed live, virtual, and constructive (LVC) systems into a common synthetic battle space that comprises various simulation protocols, training ranges, live systems and platforms.
- Gateway Builder streamlines integration process and reduces time and effort of creating gateways.
- Gateway Builder is a flexible, extensible, graphically driven tool that automatically generates gateways to bridge simulation and live protocols.
- Gateway Builder supports mappings between TENA, DIS, and HLA and message-based protocols using any object model.
Gateway Builder (GWB) Fielding during JBD2

**Diagram Description**

- **Pax River**
  - **Aggregation Router**
    - **E₁**
      - J TEN Key
    - **E₂**
      - AF-ICE Key
    - **E₃**
      - SDREN Key

- **JTEN**
  - Test Bay 8
  - JFCOM
  - AFCA, Scott AFB
  - Network Monitoring of AF-ICE Enclave Hub

- **JMETC**
  - JSIC
  - ACCN
  - EPG
  - TTEC
  - Bldgs. 3147 and 3440
  - ACETEF
  - Charleston SSC
  - Pax River
  - WSMR
  - RTTC (DTCC, GMAN)
  - Eglín AFB
  - Eglín AFB

**Key Details**

- **E₁** = TacLane with JTEN Key
- **E₂** = TacLane with AF-ICE Key
- **E₃** = TacLane with SDREN Key
TENA Video Distribution System

Camera Object Model / Camera Control

<<TENA Class>>
System
(TENA Embedded)
+platformID : UniqueID
+embeddedSystemID : UniqueID
+timeStamp : Time
+azimuth : TENA:float
+elevation : TENA:float
+roll : TENA:float
+offsetFromPlatformCenterFront : TENA:float
+offsetFromPlatformCenterRight : TENA:float
+offsetFromPlatformCenterDown : TENA:float

<<TENA Class>>
Camera
(EventStream)
+description : TENA:string
+zoomFactor : TENA:float
+azimuthMinimum : TENA:float
+azimuthMaximum : TENA:float
+elevationMinimum : TENA:float
+elevationMaximum : TENA:float
+zoomMinimum : TENA:float
+zoomMaximum : TENA:float
+HomeCamera() : TENA:boolean
+PanAzimuthTiltElevation( azimuthOffset : TENA:float, elevationOffset : TENA:float ) : TENA:boolean
+PointToPosition( position : Position ) : TENA:boolean
+Zoom( zoomFactor : TENA:float ) : TENA:boolean
+ZoomInOut( zoomOffset : TENA:float ) : TENA:boolean

Controls Pan-Tilt-Zoom (PTZ) Camera or Camera Platform
Interface Verification Tool (IVT)

- Designed to support the integration testing of TENA applications
  - TENA Standard OM’s
  - JNTC and InterTEC LROM’s
- Provides real-time monitoring, logging and statistics gathering
- Operates in three different roles, either stand-alone or in combination:
  - Data Subscriber Role
  - Data Publisher Role
  - DIS to TENA Gateway Role
SIMDIS Use of TENA

- Duration testing using SCORE TSPI data feed
  - Four consecutive days
    - Win XP, Red Hat 9, Solaris 5.8
    - Processed 180,000+ entities
  - Two consecutive days
    - Win XP, Red Hat 9
    - Processed 53,000+ entities

- Results and observations
  - No issues with discovery latency
  - No issues with update latency
  - No issues with CPU usage
  - No issues with memory usage
Gradual Deployment of TENA

- **Now**
  - Existing Range Application
  - Existing Range Application
  - Existing Range Application
  - TENA-Range Gateway
  - Other sites

- **A Few Years**
  - Existing Range Application
  - Existing Range Application
  - TENA-Range Gateway
  - Re-architected TENA-compliant Application
  - New TENA Application
  - Other sites

- **Eventually**
  - Existing Range Application
  - TENA-Range Gateway
  - Re-architected TENA-compliant Application
  - New TENA Application
  - Other sites
The TENA team is available to offer advice and assist any organization looking to use TENA

- Advice on overall design approach and trade-offs to consider
- Recommended Object Models to reuse
- Recommendations on how to design new Object Models
- Implementation / Code Designs Reviews
- Awareness of similar systems and lessons learned
- Hands-on training classes on TENA capabilities
- Hands-on training classes on using “TIDE” (a TENA Development Tool)
  - Eases developing TENA interface
  - Assists incorporating different Object Models
  - Upgrade utility for HLA applications migrating to TENA

Opportunity to Get Assistance in Using TENA
E-mail request to: feedback@tena-sda.org
TENA in a Resource Constrained Environment (TRCE) S&T Background

- **Low Data Rate Networks**
  - TENA must be able to establish and maintain data connections on low data rate networks
  - Need to optimize use of low data rate networks to support relevant operational scenarios

- **Wireless Networks**
  - Current range environments use wireless links extensively for various systems under test

- **Variable Quality Networks**
  - T&E systems poorly tolerate high loss, link failure, or heterogeneous links
  - Need to provide data continuity for degraded or heterogeneous networks

- **Specification of Interests**
  - Subscribers must be able to specify data “interests” to more efficiently use available & limited network resources

TRCE Phase 1 will:
- Developed Use Cases and Requirements
- Developed Proof-of-Concept Applications to Investigate Candidate Technologies
- Quantified Benefits of Candidate Technologies
  - Representative Laboratory Environment
- Successful Phase 1 Technology Demonstration
- Recommended Technologies for Further Development and Inclusion in the TENA Middleware

TRCE is providing TENA for variable quality and low data rate network links including wireless networks
TRCE Use Case OV-1
Joint Mission Environment Test Capability (JMETC)
JMETC Overview

• JMETC provides the infrastructure for testing in a Joint environment
  • Developmental Test, Operational Test, Interoperability Certification, Net-Ready Key Performance Parameters compliance testing, Joint Mission Capability Portfolio assessments, etc.

• Time and cost savings
  • Readily-available, persistent connectivity with standing network security agreements

• Increased capability/connectivity
  • Enables more robust testing earlier in the acquisition process
  • Provides common, certified tools to streamline integration process
  • Establishes test capability aligned with Joint National Training Capability (test and training)

• Being built based on customer requirements

*Used whenever you need to link resources together to conduct a distributed test event*
What is JMETC?

• A corporate approach for linking distributed facilities
  • Enables customers to efficiently evaluate their warfighting capabilities in a Joint context
  • Provides compatibility between test and training

• A core, reusable, and easily reconfigurable infrastructure
  • Consists of the following products:
    • Persistent connectivity
    • Middleware
    • Standard interface definitions and software algorithms
    • Distributed test support tools
    • Data management solutions
    • Reuse repository

• Provides customer support team for JMETC products and distributed testing
JMETC Benefits

• Provides Department-wide capability for:
  • Evaluation of a weapon system in a joint context
  • DT, OT, Interoperability Certification, Net-Ready KPP compliance testing, Joint Mission Capability Portfolio testing, etc.

• Provides test capability aligned with JNTC
  • Both use TENA architecture
  • Both use Network Aggregator

• Reduces time and cost by providing
  • Readily available, persistent connectivity with standing network security agreements
  • Common integration software for linking sites
  • Distributed test planning support tools

• Provides distributed test expertise

The corporate solution to distributed LVC Testing
JMETC Enables Distributed Testing

Joint Operational Scenarios

Integrated Test Resources

Virtual Prototype
- TENA Standard Interface Definitions
- TENA Common Middleware

Hardware in the Loop
- TENA Standard Interface Definitions
- TENA Common Middleware

Installed Systems Test Facility
- TENA Standard Interface Definitions
- TENA Common Middleware

Range
- TENA Standard Interface Definitions
- TENA Common Middleware

Environment Generator
- TENA Standard Interface Definitions
- TENA Common Middleware

Threat Systems
- TENA Standard Interface Definitions
- TENA Common Middleware

JMETC Infrastructure

Reuse Repository

Customer Support

* TENA: Test and Training Enabling Architecture
• **Persistent Connectivity**: Establish and maintain dedicated VPN on SDREN with standing security agreements.

• **Middleware**: Adopt and sustain downloadable TENA universal data exchange software.

• **Standard Interface Definitions & Software Algorithms**: Adopt and sustain downloadable TENA-established interface definitions and software algorithms (e.g., Radar, TSPI, coordinate and unit conversions, etc.).

• **Distributed Test Support Tools**: Collaborate with CTEIP and the Services to adopt and sustain InterTEC software tools and other existing software tool capabilities that meet JMETC customer needs.

• **Data Management Solutions**: Collaborate with CTEIP to develop new capabilities that meet JMETC customer needs.

• **Reuse Repository**: Establish and maintain on-line web portal with relevant distributed event information (latest middleware, software components, documentation, lessons learned, etc.).

*Implementation dependent on funding and customer need*
JMETC Supports:

- Testing across full spectrum of acquisition process
  - Developmental Test, Operational Test
  - Interoperability Certification
  - Net-Ready KPP compliance
- Joint mission portfolio testing
- Evaluation of weapons systems in joint mission environment
- Conduct of live, virtual or constructive testing
- Conduct of joint testing and training

*Used whenever you need to link resources together to conduct a distributed test event*
JMETC VPN Site Selection Criteria

• Sites are added to the JMETC VPN based upon:
  • Customer Requirements
  • Potential for reuse in joint testing

• Sites not on JMETC VPN build plan may fund their own addition to JMETC infrastructure
JMETC VPN Services Provided

- JMETC SYSCON
  - JMETC Personnel available to test, monitor, and troubleshoot network connectivity
  - Online Help Desk

- Registered IP Address Space
  - IP Addresses provided to approved JMETC Sites as required
  - Request IP Addresses through the JMETC SYSCON

- JMETC Domain Name Service (DNS)
  - Primary DNS IP Address (S.47.251)
  - Domain: JMETC.SMIL.MIL

- CISCO Call Manager – (S.253.2)
  - JMETC sites currently leveraging VoIP service
  - Request VoIP service through the JMETC SYSCON
JMETC Connectivity

- Dedicated, trusted connectivity on SDREN (part of the GIG)
- Encrypted for Secret – System High
- DISA-registered IP address space
- Active monitoring of network performance
- Capable of supporting multiple simultaneous test events

As of 31 Mar 2010
Network Aggregation Bridging Networks

Key:
- Available
- Capable

Aggregation Router
- DISN - LES
- JTEN
- SIPRNet
- IO Range
- JMETC VPN
- 3CE

at Pax River
JMETC VPN: Reusable Persistent Connectivity

- Reuse enables the customer to avoid:
  - Acquiring network equipment
  - Processing the security agreements
    - Obtaining Authority to Connect (ATC)
    - Obtaining Authority to Operate (ATO)
  - Generating agreements to connect with test partners
  - Testing the equipment installation
  - Testing the network configuration

- Reuse enables the customer to:
  - Test capabilities early and often
  - Execute Unscheduled / Unplanned testing whenever needed
  - Focus on the test rather than the network

Customer time and dollars not spent on infrastructure by leveraging JMETC VPN sites
JMETC Tools Repository

• The Tools Repository is one of JMETC’s six Product Line responsibilities

• Responsibility to Define the Criteria and Process for Identifying ‘Best of Breed’ tools for distributed T&E events.
  • Define Requirements
  • Create Selection Process and Products
  • Identify Existing Tools & Capabilities
  • Establish JMETC web page

• You can access the Tools Repository either:
  • Directly at https://www.tenasda.org/display/TOOLSREPO/JMETC+Tools+Repository
  • Or, from the JMETC Home Page https://www.tenasda.org/display/JMETC/Home

The primary purpose of the JMETC Tools Information Repository is to provide a searchable database of tool information, capabilities and requirements.
InterTEC Operational View-1
TENA-Based Integrated Test Tool Applications

20 Integrated Apps in Spiral 2

C4ISR Instrumentation & Analysis
- Data Capture
- Stimulation
- Analysis
- Display

Test Control
- Planning
- Rehearsal
- Control
- Monitoring
- Reporting

Virtual Components
- HWIL Interfaces
- Message Generation

Live Components
- Range Interfaces
- Range Instrumentation

Constructive Components
- Simulation Interfaces

Joint C4ISR Test Environment

Synthetic Battlespace Environment

Distributed Test Suites
InterTEC Integration with JMETC
Inextricably Intertwined

- JMETC supports InterTEC during their spiral development
- InterTEC expands JMETC toolbox with certified C4ISR Test Tools
Benefits of Reusing JMETC Sites

The number 51 (on the bottom right) is the number of times the customer was able to avoid:

- Acquiring network equipment
- Processing the security agreements
  - Obtaining authority to connect
  - Obtaining authority to operate
- Testing the equipment installation
- Testing the network configuration

<table>
<thead>
<tr>
<th></th>
<th>JMETC VPN Sites</th>
<th># of Times Sites have been Reused</th>
</tr>
</thead>
<tbody>
<tr>
<td>JMETC VPN at End of FY07</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>InterTEC Spiral 2 Sys Integration Test</td>
<td>+ 15</td>
<td>+ 4</td>
</tr>
<tr>
<td>Joint Battlespace Dynamic Deconflict.</td>
<td>+ 6</td>
<td>+ 6</td>
</tr>
<tr>
<td>InterTEC Spiral 2 Sys Accept. Test</td>
<td>+ 4</td>
<td>+ 19</td>
</tr>
<tr>
<td>Single Integrated Air Picture</td>
<td>+ 1</td>
<td>+ 9</td>
</tr>
<tr>
<td>Datalink Studies</td>
<td>+ 2</td>
<td></td>
</tr>
<tr>
<td>JMETC VPN at End of FY08</td>
<td>33</td>
<td>51</td>
</tr>
</tbody>
</table>

Customer time and dollars not spent on infrastructure by leveraging JMETC VPN sites
Recent and Upcoming TENA and JMETC Events (Partial)

- **Test Events**
  - Digital Collection, Analysis and Review System, Jan 2009 (JMETC)
  - Joint Surface Warfare, Feb 09 (JMETC)
  - InterTEC Spiral 2, Jun-Jul 2009 (JMETC)
  - SIAP JDEP Combined Hardware-in-the-Loop Phase 5, Jan-May 2009 (JMETC)
  - MultiService System-of-Systems Testbed, Jul 09, (JMETC)
  - Strategic Integrated M&S Capability Event, May-Aug 09, (JMETC)
  - Unmanned Aerial Systems in National Air Space, Jul-Oct 09 (JMETC)
  - Joint Electronic Warfare Assessment for Test and Evaluation, Sep 09 (JMETC)
  - Tactical End-to-end Closed Loop Simulator, Late 09 (JMETC)

- **Experiments**
  - Joint Expeditionary Force Experiment (JEFX) 09-2/3, Feb-Apr 09 (JMETC)
  - JEFX-09-4 B-2 (Spirit ICE), Aug 09 (JMETC)

- **Training Events**
  - Daily Training, Eielson AFB, now
  - Red Flag Alaska (RFA) 09-1, October 2008, Pacific Alaska Range (PARC)
  - JDEWR Cope Tiger 09, Mar 2009, PARC
  - RFA 09-2, April-May 2009, PARC
  - Distant Frontier, May-June 2009, PARC
  - Northern Edge 09, June 2009, PARC
  - Talisman Sabre 2009 - Australian Army and US Army, July 2009, Shoalwater Bay, Queensland Australia
  - RFA 09-3, July-Aug 2009, PARC
  - JDEWR Talisman Sabre 09, July 2009, PARC
  - RFA 09-1, October 2009
  - RFA 09-2, April 2010
  - Northern Edge, June 2010
  - RFA 09-3, Aug 2010
  - Fallon AFB, Daily Training, 2010

**JMETC Events are distributed geographically and across services via the JMETC VPN**
Summary of JMETC

• Supports the full spectrum of Joint testing, supporting many customers in many different Joint mission threads
• Being built based on customer requirements
  • JMETC support can be tailored to customer needs
• Partnering with Service activities and leveraging existing capabilities
• Coordinating with Joint Forces Command to bridge test and training capabilities
• Users Group provides an open forum to present emerging requirements as well as new technologies & capabilities
Training Applications
Alaska Training Range Evolution Program (ATREP) use of TENA

ATREP’s intent is to enhance the existing Pacific Alaska Range Complex air and ground capabilities by providing a force-on-force (FOF) training capability that fully integrates and supports joint and coalition components for both air and ground training in live, virtual, and constructive (LVC) domains.

High Side
- TENA ICADS
- TENA ACMI
- TENA 9C2
- TENA DIADS
- TENA SimShield

Low Side
- TENA MOKKITS
- TENA MILES 2000
- TENA I-HITS
- TENA UMTE
• Enhance interoperability through use of DoD standard interfaces and middleware
• Take advantage of features of TENA Middleware for intra-range data processing (e.g., publish-subscribe mechanism for control of data flow)
• Create opportunity to reuse and leverage TENA applications and tools developed for other ranges
• Make PMRF developed TENA applications available to other ranges
• Leverage DoD joint test infrastructure provided by JMETC
PRITEC at PMRF

PMRF Sensor
Data sent using a TENA Object Model Interface

TELEMETRY DATA

PMRF TENA Client

tNet

iNet

PMRF

SDREN / JMETC
TENA Software Decommutation System (TSDS)

- Offers a native TENA interface to publish telemetry data directly into TENA object models
- Provides software based approach to telemetry stream decommutation
- Supports definition of new telemetry streams through the use of auto code generation technologies
  - Provides ease of use to Range personnel
  - Offers potential cost savings throughout the entire T&E community

TSDS brings TENA interface to PMRF Telemetry
Concluding Remarks
Summary of What We Have

An Architecture for Ranges, Facilities, and Simulations to Interoperate, to be Reused, to be Composed into greater capabilities

- A Working Implementation of the Architecture
  - TENA Middleware currently works on Windows, Linux, and Sun

- A Process to Develop and Expand the Architecture
  - AMT Meetings

- A Technical Strategy to Deploy the Architecture
  - Gateways provide interim solutions as TENA interfaces

- A Definition of Compliancy
  - Levels of compliancy to enhance communication among systems engineers and investment decision makers

- A Persistent Network to permanently connect test sites
  - JMETC network enabled with TENA allows new tests to be performed with much less lead time and expense compared to the past
Important Contact Information

- **TENA Website:** [http://www.tena-sda.org](http://www.tena-sda.org)
  - Download TENA Middleware
  - Submit Helpdesk Case ([http://www.tena-sda.org/helpdesk](http://www.tena-sda.org/helpdesk))
    - Use for all questions about the Middleware

- **JMETC Program Office Contact:**
  - E-mail: jmetc-feedback@jmetc.org
  - Telephone: (703) 604-0350 ext. 0
  - JMETC Website: [http://www.jmetc.org](http://www.jmetc.org) – under construction

- **TENA Feedback:** [feedback@tena-sda.org](mailto:feedback@tena-sda.org)
  - Provide technical feedback on TENA Architecture or Middleware
  - Ask technical questions regarding the TENA architecture or project
  - Provide responses to AMT action items
  - Request TENA training