

JTLS-GO

Version Description Document

October 2017



DEPARTMENT OF DEFENSE
JOINT STAFF J7
116 LAKE VIEW PARKWAY
SUFFOLK, VA 23435-2697

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 5.0.9.0)

[Blank Page]

ABSTRACT

The Joint Theater Level Simulation - Global Operations (JTLS-GO[®]) is an interactive, computer-based, multi-sided wargaming system that models combined joint and coalition resource air, land, naval, and Non-Governmental Organization (NGO) environments.

This *JTLS-GO Version Description Document (VDD)* describes the new features of the Version 5.0.9.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 5.0.9.0 is a Maintenance release of the JTLS-GO 5.0 series that includes an updated SBDKOR50 Standard Database, as well as minor model functionality changes implemented as Engineering Change Proposals (ECPs), which are summarized in Chapter 2. Code modifications that represent corrections to known Software Trouble Reports (STRs) are described in Chapter 3. Remaining and outstanding STRs are described in Chapter 4.

This publication is updated and revised as required for each Major or Maintenance version release of the JTLS-GO model. Corrections, additions, or recommendations for improvement must reference specific sections, pages, and paragraphs with appropriate justification and be forwarded to:

JTLS-GO Development Team Leader
ROLANDS & ASSOCIATES Corporation
120 Del Rey Gardens Drive
Del Rey Oaks, California 93940 USA
jtlsdev@rolands.com

Copyright 2017 - ROLANDS & ASSOCIATES Corporation - All Rights Reserved

[Blank Page]

TABLE OF CONTENTS

ABSTRACT	iii
1.0 INTRODUCTION.....	1-1
1.1 SCOPE	1-1
1.2 INVENTORY OF MATERIALS	1-1
1.2.1 Unchanged Documents	1-1
1.2.2 Updated Documents	1-2
1.2.3 Delivered Software Components	1-2
1.2.4 Released Databases	1-4
1.3 INTERFACE COMPATIBILITY	1-4
1.3.1 Support Software	1-4
1.3.2 JTLS-GO High Level Architecture Compliance	1-7
1.4 DATABASE MODIFICATIONS	1-8
1.4.1 JTLS-GO Using Legacy Default Symbol Set	1-8
1.4.2 Upgrading JTLS-GO Default Symbol Set	1-8
1.4.3 Standard Database Changes	1-9
1.5 INSTALLATION	1-9
1.5.1 Installation Instructions	1-9
1.5.2 Oracle Compatibility and Installation	1-9
1.5.3 Disabling Certificate Authority	1-10
1.5.4 Map Vector File Format Update	1-11
2.0 ENGINEERING CHANGE PROPOSALS.....	2-1
2.1 JTLS-2017-13330 ELS To Receive Snap Report From MUSE	2-1
3.0 SOFTWARE TROUBLE REPORTS	3-1
3.1 JTLS-2017-13299 Lifted Unit Destroyed During Airlift Issues	3-1
3.2 JTLS-2017-13300 Airlift Unit Canceled While Loading Issues	3-1
3.3 JTLS-2017-13301 HRU Battle Damage Incorrectly Reported	3-2
3.4 JTLS-2017-13302 Logic Error Canceling Complete Formation Task	3-2
3.5 JTLS-2017-13303 Airlift Estimate Report Inaccurate	3-2
3.6 JTLS-2017-13304 Order Spreadsheet Handling Error	3-3
3.7 JTLS-2017-13305 Entered Order Spreadsheet UOM Values Incorrect ...	3-3
3.8 JTLS-2017-13306 ATO Parser Allows Bad Characters To Pass	3-3
3.9 JTLS-2017-13307 SVP Error/Warnings Improved	3-4
3.10 JTLS-2017-13308 Fixes From Findbugs Scans	3-4
3.11 JTLS-2017-13309 DDS Combat System Rollup Report Issues	3-4
3.12 JTLS-2017-13310 Unable To Close WHIP	3-4
3.13 JTLS-2017-13311 Unable To Find Formations On DDS Map	3-4
3.14 JTLS-2017-13312 Sealane Editor Used Wrong Speed	3-4
3.15 JTLS-2017-13313 Scenario And Game Name Identical	3-5
3.16 JTLS-2017-13314 Exception Editing JOBE Prior To Map Creation	3-5
3.17 JTLS-2017-13315 SIP Crash Database Testing	3-5

3.18	JTLS-2017-13316 Link16 - Feed OFF By Default	3-5
3.19	JTLS-2017-13317 SVP Errors After Running ASC	3-5
3.20	JTLS-2017-13318 Controller Damage Report Missing Data	3-5
3.21	JTLS-2017-13319 Translating ATO Change Message	3-6
3.22	JTLS-2017-13320 OMA Crash Land/Water Location	3-6
3.23	JTLS-2017-13321 ATO Parser Stop Time For RECCE Missions	3-6
3.24	JTLS-2017-13322 Enemy ARU Fails To Engage HRU	3-7
3.25	JTLS-2017-13323 Replay JXSR Crash	3-7
3.26	JTLS-2017-13324 Disappearing Naval Routes	3-8
3.27	JTLS-2017-13325 Replay JXSR Useless Reports	3-8
3.28	JTLS-2017-13326 Great Circle Distance For Close Location Errors	3-8
3.29	JTLS-2017-13327 Manage Formation Tasks Crash	3-9
3.30	JTLS-2017-13328 Postponing Current Mission Task Crash	3-9
3.31	JTLS-2017-13329 Error In Message From Debug Object Order	3-9
3.32	JTLS-2017-13331 ELS Crash While Writing Checkpoint	3-10
3.33	JTLS-2017-13332 ELS Crash For Transfer Aircraft Missions	3-10
3.34	JTLS-2017-13333 Black Runways Not Gathered	3-10
3.35	JTLS-2017-13334 Crash When Sinking Ship Creates Lifeboats	3-10
3.36	JTLS-2017-13335 Error Allowed Units To Cross Uncrossable Rivers	3-10
3.37	JTLS-2017-13336 GIS Tool River/Road Preprocessor Spurious Arcs	3-11
3.38	JTLS-2017-13337 Crash When Naval Unit Attempts to Kill Convoy	3-12
3.39	JTLS-2017-13338 Game Start Supply Shortages	3-12
3.40	JTLS-2017-13339 Special System Libraries For Newer Compiler	3-12
3.41	JTLS-2017-13340 Crash When No Supplies For Tanker Mission	3-12
3.42	JTLS-2017-13341 Mission Ignores Altitude Change	3-12
3.43	JTLS-2017-13342 ELS Lifeboats Missing Rescued Passengers	3-13
3.44	JTLS-2017-13343 Manage Formation Tasks Order Panel Incorrect	3-13
3.45	JTLS-2017-13344 Fire Mission Task Cancel Message Blank Fields	3-13
3.46	JTLS-2017-13345 OTHGold Time Late, Duplicate Track Info	3-14
3.47	JTLS-2017-13346 OTHGold Failed Update Status Label	3-14
3.48	JTLS-2017-13347 Minor OPM Issues	3-14
3.49	JTLS-2017-13348 Empty Cell Designators In AAR Report Scripts	3-14
3.50	JTLS-2017-13349 Change HQ Order Crash	3-15
3.51	JTLS-2017-13350 Crash While Finding Squadron Divert Location	3-15
3.52	JTLS-2017-13351 Package Names From ATOT Conflict	3-15
3.53	JTLS-2017-13352 JTOI ICC Radar/SAM Targets Reference Name	3-15
3.54	JTLS-2017-13353 Wrong Warning Number Used In SVP	3-15
3.55	JTLS-2017-13354 Tech Tool Copy Confirmation Window Too Large	3-16
3.56	JTLS-2017-13355 Civilian Air Traffic Mission Resource Event Crash	3-16
3.57	JTLS-2017-13356 Civilian Air Mission Unit Delay Crash	3-16
3.58	JTLS-2017-13357 SVPR Warning 1462 Corrections	3-16
3.59	JTLS-2017-13358 Modify Attack Package Debug	3-17
3.60	JTLS-2017-13359 SUP Consumption Messages Missing	3-17

- 3.61 JTLS-2017-13360 Checkpoint Restart Crash3-17
- 3.62 JTLS-2017-13361 SIP Errors Counts And Totals Do Not Match3-18
- 3.63 JTLS-2017-13362 Apache Module Crash During TCP Connection3-18
- 3.64 JTLS-2017-13363 CEP Crash Convoy Crossing National Boundary ...3-18
- 3.65 JTLS-2017-13364 GIS Tool: Misclassified Terrain Grids3-18
- 3.66 JTLS-2017-13365 DCP Glassfish Default Database Name3-19
- 3.67 JTLS-2017-13366 DDS Schema Script Removed From Directory3-19
- 3.68 JTLS-2017-13367 DCP Saving Domain Error3-20
- 3.69 JTLS-2017-13368 ATOT Missing Airbase Creates Error3-20
- 3.70 JTLS-2017-13369 Equipment Not Sheltered On Unit Arrive3-20
- 3.71 JTLS-2017-13370 Crash Changing Station While Repositioning3-20
- 3.72 JTLS-2017-13371 Air Mission Not Properly Reporting Fuel Delay3-20
- 3.73 JTLS-2017-13372 Layer Ordering In DDS And WHIP3-20
- 3.74 JTLS-2017-13373 TBM Stair-Step On TDACS3-21
- 3.75 JTLS-2017-13375 Link-16 Message Service Leaks Memory3-21
- 3.76 JTLS-2017-13376 Crash When Mission Runway Not Long Enough ...3-21
- 3.77 JTLS-2017-13384 Infinite Loop - Ships Auto Firing3-21
- 3.78 JTLS-2017-13385 Checkpoint Start With Large Terrain File3-22

- 4.0 REMAINING ERRORS..... 4-1
 - 4.1 DDSC – TMU Line Mode Changes Multiple Grids 4-1
 - 4.2 DDSC – Multiple Types In DDS History Table 4-1
 - 4.3 WHIP - Pipeline Not Shown On IMT 4-1
 - 4.4 DDSC/WHIP/JOBE - CADRG Map Zoom 4-1
 - 4.5 WSM - Many Messages Cause Lockup 4-1
 - 4.6 Strategic Airlift Does Not Work 4-1
 - 4.7 Tactical Ground Formation Attacks Do Not Work 4-1
 - 4.8 Materiel Handling Equipment (MHE) Does Not Work 4-2
 - 4.9 Port Amphibious Operations Do Not Work 4-2
 - 4.10 Multiple Point Ground Group Move Results Not Consistent 4-2

- ABBREVIATIONS AND ACRONYMS A-1

- Version 5.0.9.0 STANDARD DATABASE CHANGES B-1

- Version 5.0.9.0 DATABASE CHANGES C-1

1.0 INTRODUCTION

1.1 SCOPE

This *JTLS-GO Version Description Document (VDD)* describes Version 5.0.9.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO[®]) software suite. JTLS-GO 5.0.9.0 is a Maintenance delivery for the JTLS-GO 5.0 series of releases.

JTLS-GO 5.0.9.0 includes the entire JTLS-GO suite of software and the SDBKOR50 Standard Database that supports a realistic scenario based on the current Korean Peninsula theater of operations. Database modifications that were accomplished to upgrade the previous JTLS-GO Standard Database to this current version are summarized in this chapter, as well as [APPENDIX B](#). Detailed descriptions of minor Engineering Change Proposals (ECPs) implemented for this release are provided in [Chapter 2.0](#).

JTLS-GO 5.0.9.0 executes on the Red Hat Enterprise Linux Version 6 64-bit operating systems. The Web-Hosted Interface Program (WHIP[®]) user workstation interface can be executed on any operating system from any Java-compatible Web browser.

1.2 INVENTORY OF MATERIALS

This section lists documents and software that are relevant to JTLS-GO. All JTLS-GO documents included in this delivery are provided in PDF format within a documents subdirectory.

1.2.1 Unchanged Documents

JTLS-GO 5.0.9.0 is a bug release of the JTLS-GO system. Other than date changes, the following documents have had no significant changes since the last release of JTLS-GO.

- *JTLS-GO Analyst Guide* (JTLS-GO Document 01, Version 5.0.9.0)
- *JTLS-GO Air Services User Guide* (JTLS-GO Document 03, Version 5.0.9.0)
- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 5.0.9.0)
- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 5.0.9.0)
- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 5.0.9.0)
- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 5.0.9.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 5.0.9.0)
- *JTLS-GO Standard Database Description* (JTLS-GO Document 14, Version 5.0.9.0)

- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 5.0.9.0)

1.2.2 Updated Documents

The following documents have been changed or improved since the last release of JTLS-GO.

- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 5.0.9.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 5.0.9.0)
- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 5.0.9.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 5.0.9.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 5.0.9.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 5.0.9.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 5.0.9.0)
- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 5.0.9.0)

1.2.3 Delivered Software Components

JTLS-GO 5.0.9.0 may be delivered either on a CD or as a set of compressed TAR files to be downloaded. Either method includes the complete suite of software executable code and command procedures. The following software components are included with this release:

- Combat Events Program (CEP)
- Scenario Initialization Program (SIP)
- Interface Configuration Program (ICP)
- Reformat Spreadsheet Program (RSP)
- Database Development System (DDS)
- JTLS Symbols Application (JSYMS)
- ATO Translator Service (ATOT)
- ATO Generator Service (ATOG)
- ATO Retrieval Program (ATORET)
- JTLS Convert Location Program (JCONVERT)

- Count Critical Order Program (CCO)
- JTLS HLA Interface Program (JHIP)
- After Action Review Client (AARC)
- Scenario Data Client (SDC)
- Order Entry Client (OEC)
- Order Verification Tool (OVT)
- JTLS Object Distribution Authority (JODA)
- Web Services Manager (WSM)
- Web Services
 - Apache Server (APACHE), version 2.4.26
 - JTLS XML Serial Repository (JXSR)
 - Order Management Authority (OMA)
 - Synchronized Authentication and Preferences Service (SYNAPSE)
 - XML Message Service (XMS)
- Web-Hosted Interface Program (WHIP)
- Total Recall Interactive Playback Program (TRIPP)
- Entity Level Server (ELS)
- JTLS Operational Interface (JOI) OTH-Gold, Link-16, and TACELINT
- KML Operational Interface (KOI)
- JTLS Transaction Interface Program (JTOI) supporting
 - ICC 2.8.2 and 3.0.2
 - NEC-CCIS
 - TBMCS
- JTLS Interface Network Navigator (JINN)
- JTLS Order of Battle Editor (JOBE)

- Database Configuration Program (DCP)
- DDS User Interface (DDS)
- JTLS Terrain Building Program

Instructions for installing JTLS-GO 5.0.9.0 are provided in the *JTLS-GO Installation Manual*. Installing a previous version of JTLS prior to installing JTLS-GO 5.0.9.0 is not necessary. No other upgrade beyond installation of the compressed TAR files (or CD) is required. The software provided with this delivery is a complete release that includes all files and code required to execute JTLS-GO.

1.2.4 Released Databases

This release includes the following sample unclassified databases:

- The scenario developed as the Korea Standard Database and named SDBKOR50 is a large-scale, seven-sided scenario database reflecting the approximate starting positions of units involved in the Korea Peninsula theater of operations. This example scenario was developed using unclassified data sources and is consequently not completely accurate. Discrepancies among actual units and their locations are not detrimental to the intended purpose of this database, which is to provide a recognizable and realistic scenario that demonstrates the simulation capabilities and supports JTLS training.
- The scenario DEMSDBKOR50, which is a reduced version of SDBKOR50, has also been updated.
- The scenario blank50 is the SDBKOR50 database with all force structure data removed, which can be used as a framework for building your customized database.

1.3 INTERFACE COMPATIBILITY

1.3.1 Support Software

JTLS-GO 5.0.9.0 requires the following versions of support software, including operating systems, compilers, scripting utilities, database tools, transfer protocols, and display managers.

- Operating system for the model: Red Hat Linux Enterprise Edition Version 6.9 (ES), 64-bit architecture.

Security Enabled (SE) Linux must not be enabled on systems used to execute JTLS-GO or its components. Tests reveal that SE Linux-enabled systems cause frequent and random JXSR crashes and errors. Use of SE Linux to support JTLS-GO is currently not authorized.

A RedHat Linux documented loop-back interface problem continues to cause degraded performance issues on Red Hat/CentOS 6 systems. This issue negatively impacts JTLS services when executed on the same server as Apache.

While we expect Red Hat to eventually correct this issue, significant effort has been placed on JTLS-GO optimization to minimize impact of the problem. Still we recommend users with more than 25 concurrent WHIP/TRIPP users to avoid this Red Hat/CentOS loopback issue by running Apache and the SYNAPSE on different physical or virtual servers. For users with significantly more than 25 concurrent WHIP/TRIPP users, consider not running other JTLS services, such as the JXSR and XMS, on the same server used by Apache.

See the *JTLS-GO Installation Manual*, Appendix D for some recommended system tuning changes that can mitigate these performance issues.

STR JTLS-2017-13229 implements SSH Tunneling between Apache and the services and among the services. While testing has been conducted in the development environment, rigorous testing should be done prior to use in any exercise. Particular attention should be paid to network performance under load.

When using a Linux system as a client, if a dialog saying “Too many open files” appears, see the *JTLS-GO Installation Manual*, Section 5.5 for a work around. If this occurs on a server, see the *JTLS-GO Installation Manual*, Appendix D.

JTLS-GO 5.0 has been tested with the following versions of Linux 6:

RedHat Linux 6 - this operating system license must be purchased.

Oracle Linux 6 - this is a free and distributable version of Linux which has been approved for use by U.S. Government agencies by the Defense Information Systems Agency (DISA).

CentOS 6 - a free version of Linux 6 that has not been approved by DISA for use by U.S. Government agencies.

- There are no restrictions on the operating system for client workstations, except that the operating system must have a Java-enabled web browser. JTLS-GO 5.0.9.0 has been tested on the following operating systems:

Red Hat Linux Enterprise Edition Version 5 and 6.

CentOS Linux Version 5 and 6.

Windows Vista, Windows 7, and Windows 10 can be used only if the workstation is an external HTTP client of the simulation network.

- Java 8 Update 152 is required for all platforms and must be used to support all workstations.

Because of recent incidents of intrusions into software systems, the United States Department of Defense (DoD) has implemented a strong and strictly enforced Information Assurance (IA) program. JTLS, as software that executes on DoD systems, must comply to the mandates of the IA program, along with all of the third party software used by JTLS-GO, such as Oracle and Java.

One of the DoD IA requirements is that the software must implement a methodology that ensures that the end user keep the software up-to-date and all security patches are properly installed. Java 8 has fulfilled this mandate by implementing an expiration date for its software. After a certain date, all Java related programs will stop working, whether you are connected to an open network or not.

JTLS-GO software releases are linked to the Java 8 security updates to meet this requirement. This version of JTLS-GO is released using Java 8 Update 152, which expires on 16 January 2018. A new version of JTLS-GO will be released prior to that expiration date.

- JTLS-GO database tools require use of a certified Oracle database server and the full Oracle Client installation for runtime requirements. Refer to [Section 1.5.2](#) of this chapter for additional installation details.
- Windows software, X11R5 server, Motif 1.2 Library, Motif Window Manager: These items are included as part of the supported versions of Red Hat Linux ES.
- TCP/IP is required for inter-process communication between the JODA data server and all user interface programs. The version of TCP/IP included with the supported versions of Red Hat Linux ES is sufficient.
- The Perl script language is used by the JTLS-GO system and game setup scripts. The version of Perl included with the supported versions of Red Hat Linux ES is sufficient. The Perl program is typically located in the /usr/bin directory. If Perl is installed in a another location, a link should be created from the /usr/bin directory to this program.
- SIMSCRIPT II.5 (SIMSCRIPT to C) translator/compiler: SIMSCRIPT is required for recompiling JTLS-GO code. It is not necessary to have a SIMSCRIPT compiler to execute JTLS-GO, because all JTLS-GO software executables are statically linked with the SIMSCRIPT libraries. The compiler is needed only if you are a U.S. Government organization that can obtain source code and plan to re-compile JTLS-GO SIMSCRIPT code. To obtain a SIMSCRIPT compiler, contact CACI Inc.

- ANSI C Compiler: It is not necessary to use a C compiler to execute JTLS-GO. This compiler is used only by U.S. Government organizations that can obtain source code and intend to re-compile any of the JTLS-GO component programs. The C Compiler version delivered with the supported versions of Red Hat Linux ES is sufficient.
- C++ Compiler: It is not necessary to use a C++ compiler to execute JTLS-GO. This compiler is used only by U.S. Government organizations that can obtain source code and intend to re-compile any of the JTLS-GO HLA component programs. The C++ Compiler version delivered with the supported versions of Red Hat Linux ES is sufficient.
- The JTLS-GO DDS (Database Development System) application uses these open source libraries:

JFreeChart, licensed under LGPL (GNU LESSER GENERAL PUBLIC LICENSE) by Object Refinery Limited, <http://www.object-refinery.com>

JCommon, licensed under LGPL2.1 (GNU LESSER GENERAL PUBLIC LICENSE version 2.1 or later) by Object Refinery Limited, <http://www.object-refinery.com>

Commons-math3-3.0.jar, licensed under Apache Software Foundation (Apache License, Version 2.0) <http://www.apache.org/licenses/LICENSE-2.0>HLA Compliance

- KML Operational Interface (KOI)

The Keyhole Markup Language (KML) Operational Interface (KOI) server utility enables the model to feed operational simulation data to any version of Google Earth™. The display capabilities and data transfer features of this terrain viewer are sufficiently robust to be used as a base-level operational interface. Operational Players who may be restricted from using the COP, C2PC, or other C4I systems may be able to install and use Google Earth and configure the KOI to provide a capability that resembles C4I for observing perception Force Side data.

Chapter 3 of the *JTLS-GO C4I Interface Manual* describes requirements and procedures for using the KOI capabilities.

1.3.2 JTLS-GO High Level Architecture Compliance

The JTLS-GO 5.0.9.0 release is fully High Level Architecture (HLA) compliant, and includes all the programs required to run JTLS-GO in an HLA mode. JTLS-GO has moved to a new Federation Object Model (FOM) located in the \$JGAME/data/hla directory. Federation testing of JTLS-GO is not complete, but initial tests with CAE's Gefechts Simulation (GESI) have been accomplished. Future plans include expanding the capabilities of the JTLS-GO-GESI federation, called GlobalSim continue.

GlobalSim is a comprehensive constructive simulation solution for joint training and wargaming that helps commanders and all levels of staff prepare for a range of operational scenarios. The

solution combines JTLS-GO with CAE's GESI constructive tactical entity-level simulation system as a federation called GlobalSim. CAE's GESI constructive simulation system is designed to run complex and comprehensive exercises from the company level up to division level. The CAE GESI system is used to represent a virtual battlefield, including weapons, vehicles, aircrafts, ground forces and more. Combining JTLS-GO and GESI brings together operational and tactical level constructive simulations to prepare commanders and staff to make timely, informed and intelligent decisions across the full spectrum of operations, including conventional combat, disaster relief, and operations other than war.

From the JTLS-GO perspective, all software needed to run GlobalSim is included in this delivery.

The HLA RTI (Run Time Infrastructure) executive program (rtiexec) recommended for use with this release is RTI-NG-Pro-v7.0 or Pitch pRTI Evolved 4.4.2.0. However, these programs are not included in the JTLS-GO 5.0.9.0 delivery. Users may obtain a full installation package of the RTI software from either vendor: Raytheon Company (<http://www.raytheon.com>) or Pitch Corporation (www.pitch.se). For information about executing the HLA RTI Executive and other HLA-related software, refer to the appropriate HLA documentation and user guides.

1.4 DATABASE MODIFICATIONS

No database structure differences exist between JTLS-GO 5.0.9.0 and JTLS-GO 5.0.0.0.

Significant database structure differences exist between JTLS-GO 5.0.9.0 and any previous 4.1 series database.

To upgrade your JTLS 4.1 scenario to JTLS-GO 5.0 compatibility, see instructions listed in the *JTLS-GO DDS User Guide*, Chapter 3.1.

1.4.1 JTLS-GO Using Legacy Default Symbol Set

Prior to unloading your newly upgraded JTLS-GO 5.0.0.0 formatted data from your Oracle database server into a JTLS-GO 5.0.0.0 scenario ASCII file, you must execute the JSYMS program, which will reorganize the structure of the .gs and .scf symbols-related files. See the *JTLS-GO DDS User Guide*, Appendix B.11 for instructions.

1.4.2 Upgrading JTLS-GO Default Symbol Set

The JTLS-GO Default Symbol Set delivered with this release has been modified to use the US 2525C symbol standard and well as the NATO Joint Military Symbol [APP-6(c)] standard. If you choose to propagate this new Default Symbol Set to any of your scenarios that use the 4.1 Default Symbol Set, see the *JTLS-GO DDS User Guide*, Appendix B.12 and B.13 for instructions.

You should not make any modifications to the Default Symbol Set, but you are free to copy the Default Symbol Set and create your own set to meet your needs.

1.4.3 Standard Database Changes

Additional Standard Database changes that are applicable to JTLS-GO 5.0.9.0 are described in [APPENDIX B](#).

1.5 INSTALLATION

1.5.1 Installation Instructions

The *JTLS-GO Installation Manual*, included in the compressed TAR file that is part of this JTLS-GO delivery, provides detailed instructions for installing a new version of JTLS-GO. The *Installation Manual* can be found in the /documents subdirectory of the JTLS-GO TAR file.

With the JTLS-GO 5.0 series, there have been significant changes to the directory structure which holds the files for the Online Player Manual (OPM). Users must remove all existing files from their JTLS 4.1 or earlier version \$JGAME/{scenario} directory before setting up their scenario for execution with the JTLS 5.0 series. Prior to running Option 3 - "Setup System for a Specific Scenario" of the JTLS-GO menu, all users must remove the existing files in their scenario game directory. Failure to clean-up the \$JGAME/scenario directory will result in a program crash while generating the OPM (Option 4).

The JDS Protocol has changed with this release. Any users that hook to the JODA or EODA to get information on the existence and/or status of objects must recompile their code with the new JDSP header files, and link with the new JDSP libraries. Within the protocol, a force side has been added to all AAR objects, The BE Facility object has been assigned a damage_percent attribute, and BE Objects have been added to link targets and units to a BE Facility.

1.5.2 Oracle Compatibility and Installation

A full Oracle Client installation (not Instant Client) that matches your database server version is currently a requirement for running some JTLS-GO applications. The Oracle Instant Client is not sufficient for JTLS-GO applications because Oracle utilities, such as sqldr, imp, exp, tnsping, etc., are missing. If you have applied a patchset to your database server, the same patchset should be applied to the Oracle Client installation. A 64-bit Oracle Client installation must be used. The JTLS-GO scenario/database modification process also expects Oracle 11.2.0.1 or higher full Oracle Client installation.

Some sites NFS mount their database server as Oracle Client; other sites prefer a full install of the Oracle Client to a different directory that mounts (simple NFS will suffice) to JTLS-GO. Your system administrator can choose the appropriate installation. Assigning the full Oracle Client installation location (or mount point) as the ORACLE_HOME in the JTLS .cshrc file allows connecting to an Oracle database server (11.2.0.1 or higher - including 11gR2 XE) running on any Oracle-certified database server platform.

The new DDS application utilizes the Oracle GlassFish J2EE server, which, like the JTLS-GO WHIP Apache server, is delivered with JTLS and requires no separate installation.

Refer to Chapter 6 of the *JTLS-GO Installation Manual* for additional details pertaining to the Oracle installation.

1.5.3 Disabling Certificate Authority

Industry standard programming practices strongly recommended that Web Start technologies be signed by a Certificate Authority (CA) to validate that an application being run comes from a trusted and registered source under the PKI. Keeping with high security standards, R&A has opted to sign its web enabled applications with a certificate from the COMODO, which uses the Public Key Infrastructure (PKI) to validate the digital signature. Local area networks, intra-nets and standalone systems that have no outside connection to the Internet must configure their installation to disable certificate revocation checks. If this step is not taken under a closed environment, the JTLS-GO web-enabled applications will experience a significant delay in start up while the PKI attempts to validate the certification on COMODO's servers via the Internet.

The following outlines the method that should be used to turn off certificate revocation checks. Again, this step should only be taken if the WHIP, TRIPP and/or DDS are going to be run in a closed environment.

Bring up the Control Panel. On Linux, typing 'jcontrol' in a terminal window will bring up the configuration panel. On Windows, go to the Windows 'Control Panel', select 'Programs' then click on 'Java'. Both Linux and Windows will display the same configuration panel.

Navigate to the 'Advanced' tab and in the list of configuration options, select the 'Do not check' radio button and click 'OK'. The image below illustrates the corresponding selection in the Control Panel.

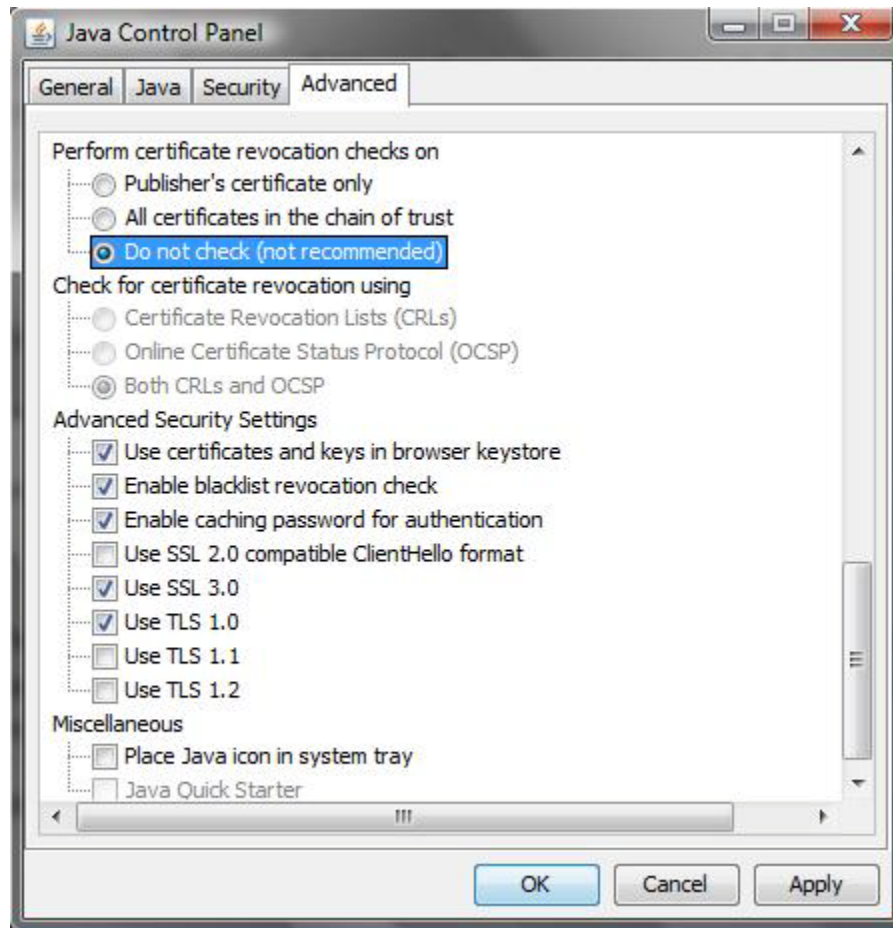


Figure 1.1 Control Panel

Completing this step will allow you to run the JTLS-GO web enabled applications without any external checks and launch the application without the validation delay.

1.5.4 Map Vector File Format Update

The format of the files used to display outline maps on the WHIP and DDSC has changed. Files in the new format are delivered with JTLS-GO. The files are located in `$JTLSHOME/data/maps/vector` and are in XML format.

For users who desire to use any custom files that are in the JTLS 4.1 format, a conversion script called "outlineConverter" is provided. The script can either convert individual files or a directory of files. The user can also choose to convert all types of outlines or only the shoreline outline (coded 9999 in the 4.1 format).

The script has 2 usages:

- To convert a single file:

```
outlineConverter -a <map_file> [-c shoreline]
```

- To convert all files in a directory. The file extension is assumed to be .map.

```
outlineConverter -d <directory> [-c shoreline]
```

The optional “-c shoreline” argument causes only the shoreline vectors to be converted. Rivers and national boundaries no longer follow hex-edges in JTLS-GO 5.0.9.0, but are represented as distinct lines with latitudes and longitudes. These terrain features can be individually turned on and off using the WHIP filter capability. Thus, the need to represent rivers and national boundaries in the vector maps is eliminated. The ability to represent these features, as part of the vector map, has not been removed from the WHIP, but the user should consider whether these old vector features should or should not be displayed by the vector maps.

The output files are placed in the same directory as the input files.

2.0 ENGINEERING CHANGE PROPOSALS

This chapter summarizes model capabilities added to JTLS-GO 5.0.9.0 as a result of implementing minor Engineering Change Proposals (ECPs).

2.1 JTLS-2017-13330 ELS To Receive Snap Report From MUSE

Summary of Model Change Request

A new capability was added to the connection between JTLS and MUSE to accept snap reports. This allows MUSE to share detections with JTLS for a specific rectangular area.

Design Summary

The ELS was improved to receive orders from external models which support the sharing of perception between the models. When MUSE attempted to send this order to the ELS, it needed to send it at the current game time (NOW) rather than at a specific time and date. Code changes were made to support this time specification.

3.0 SOFTWARE TROUBLE REPORTS

Software Trouble Reports (STRs) describe software code errors that have been discovered by JTLS-GO Users or developers and have been corrected. [Chapter 4.0](#) describes STRs that remain outstanding from previous versions.

3.1 JTLS-2017-13299 Lifted Unit Destroyed During Airlift Issues

A Mobility Mission order was submitted to airlift a ground Unit. While the mission was loading the ground Unit, a Controller Change Unit order was submitted to reduce the Unit to 0% strength. All the operational combat systems and supplies not currently being loaded were destroyed. Combat systems in maintenance were not affected. The combat systems and supplies currently being loaded were allowed to complete. The mission flew to the offload location and delivered the assets, then returned to the pickup location to load two more combat systems that were in maintenance. The Air Movement Report failed to show the maintenance systems being carried en route to the offload location. Furthermore, the Combat System IMT did not update.

The number of combat systems in maintenance being carried by an airlift mission were added to the Air Movement Report. The number of operational systems being carried was already present.

The logic that updates the lifted Unit's combat systems to the JDS (and the IMT) did not execute when the only remaining combat systems at the pickup location happened to be in maintenance. New code was added to correct the omission and cause the JDS update in that situation.

3.2 JTLS-2017-13300 Airlift Unit Canceled While Loading Issues

A mobility mission was ordered to airlift a ground Unit. While executing its first Load task at the embarkation point, the mission was canceled using the Cancel Air Mission order. The Load task stopped executing and the mission flew home as expected. However, the Unit's Combat System IMT showed fewer systems available than before the Load task began.

Because the airlift mission had not yet left the embarkation point for the first time, the combat systems that were being loaded were returned to the ground Unit as expected. The logic, however, failed to update the JDS to show the assets added back to the Available column in the IMT. The Logistics Report showed the correct amounts had been added back. To correct this problem, a subroutine call was added to update the JDS.

Subsequent testing with another mission that contained enough aircraft to lift the entire ground Unit in a single load had a similar issue. The IMT in this case contained all zeroes. To correct this problem, a subroutine call was added to re-issue all the combat systems to the Unit before updating the JDS. Additionally, the MISREP incorrectly reported the ground Unit was already at the destination when the airlift was unexpectedly halted. The subroutine that generates that submessage was corrected to report the ground Unit was returned to its embarkation.

3.3 JTLS-2017-13301 HRU Battle Damage Incorrectly Reported

An HRU was ordered to raid/attack a ground Unit. The Player damage reports and Controller damage reports erroneously attributed the losses to artillery firing at an amphibious operation.

The problem was found in the routines that evaluate high-resolution combat. An argument passed into the assess damage routine indicated the battle was ground combat, instead of high-res combat, for the purpose of AAR reporting. However, further down the calling chain the Controller and HRU damage routines interpreted ground combat as counter amphib because .GROUND.COMBAT and .COUNTER.AMPHIB are identically defined as integer 4 in the preamble. Consequently the report logic printed the counter amphib report version. New logic was added to change the strike type to .HIGHRES.BATTLE if at least one of the battle participants is an HRU. This change causes the correct high resolution battle damage information to be generated in the damage reports.

While testing this problem, other issues were found and corrected:

- When the Controller damage report was generated for the ground Unit's return fire, the ground Unit's coordinates were printed as the HRU's coordinates.
- The firing object's coordinates were not initialized before being used to access the weather conditions. Consequently, the weather at the equator (0,0) was always accessed when determining the probability of hit.
- The determine hit or miss routine passed the latitude coordinate in place of the longitude coordinate to the calculate PH routine multiple times.

3.4 JTLS-2017-13302 Logic Error Canceling Complete Formation Task

A logic error was generated when the model attempted to cancel a completed formation task event. The formation did not have one.

This was a perfectly normal situation. The formation patrol order was entered to indicate that it should continue until otherwise tasked. In this situation there would not be a Complete Formation Task event. The logic error was removed.

3.5 JTLS-2017-13303 Airlift Estimate Report Inaccurate

The Airlift Estimation order, which generates a report showing the estimated duration of an airlift/airdrop Unit operation, was accurate only if one squadron (i.e. one mission) was specified in the order. If more than one squadron/mission was specified to participate, the report indicated zero duration for some of the squadrons' missions.

A logic flaw in the algorithm prevented all the specified squadrons from participating fully in the operation. In fact, only one of the squadrons was included for every event (fly to pickup, load Unit, fly to destination, unload Unit, return to pickup, and so on until the operation was complete),

while the other squadrons waited. The flaw was corrected. At the same time, the report was enhanced to show the timeline associated with each squadron (mission) as it completes each event in the airlift/airdrop Unit process.

In addition, the logic treated the distance between the squadron location and the pickup location as zero if within COLOCATION.DISTANCE. In the standard database, the colocation distance is 20 kilometers. The code was changed to use MAX.OFFSET.DISTANCE, which is 1 kilometer in the standard database, to ensure flight time is included in the estimate when separation distance is over 1 kilometer.

3.6 JTLS-2017-13304 Order Spreadsheet Handling Error

Order spreadsheets fill a value in for optional fields. For instance in Naval Move (NM), the optional speed is defaulted to 0.0 KNTS. This value gets sent to the model and appears to be ignored. In Manage Naval Unit Tasks (MNUT), there are a number of optional fields. There are fields that are mutually exclusive so this leads to the order failing verification.

Optional fields in order spreadsheets had initial values hard-coded to be set such as 0.0 KNTS for the optional Naval Speed in the Naval Move order. Optional fields are now empty until the User enters a value. This makes the order spreadsheets more consistent with the standard order panels.

3.7 JTLS-2017-13305 Entered Order Spreadsheet UOM Values Incorrect

Order Spreadsheet Unit-of-measure (UOM) fields incorrectly converted entered values, such as in the Naval Speed UOM field for the Naval Move order.

Spreadsheet UOM values entered were being incorrectly converted in the table field causing the displayed value to be inconsistent with what had been entered for the UOM. This was corrected to perform a proper conversion, and to convert only when the order spreadsheet is being verified or sent to the model for processing as is done in standard order panels.

3.8 JTLS-2017-13306 ATO Parser Allows Bad Characters To Pass

The ATO Translator can not read XML files from the ATO Parser that contain double quotes or ampersand symbols. The problem begins when the ATO Parser reads an ACO or ATO MTF message, which contains these bad characters. The Parser does not filter out the bad characters before generating the ACO and ATO XML document for the Translator. When the Translator attempts to read the XML document containing the bad characters it fails and reports that the XML file is unreadable.

The XML documents produced by the Parser have a group of USMTFRecords tags containing text from the original MTF message. In this section the Parser writes raw text from the MTF message that is being parsed. If the text from the original message contains double quoted strings or ampersand symbols the text in the USMTFRecords section will also contain these symbols. This

causes a poorly formed XML document and the Translator fails during the read. The Parser was modified to transform the bad characters while generating the ACO and ATO XML message.

3.9 JTLS-2017-13307 SVP Error/Warnings Improved

Numerous improvements were required to various Scenario Verification Program Repository (SVPR) errors and warnings to improve the User experience when using the SVPR in DDS.

Capabilities were added to the SVP and Documentation was updated.

3.10 JTLS-2017-13308 Fixes From Findbugs Scans

The Java code is being scanned by the Findbugs static code analyzer and discovered various issues.

The issues found by Findbugs scans were corrected.

3.11 JTLS-2017-13309 DDS Combat System Rollup Report Issues

During database development, a side can have multiple top Units. The Comparison Combat System Rollup Report failed to render when a side had multiple top Units. Additionally, the Individual version was inaccurate.

The report logic was modified so that it correctly handled the data for a side with multiple top Units.

3.12 JTLS-2017-13310 Unable To Close WHIP

When the Exit menu option or the 'x' in the window was selected, nothing happened and the WHIP had to be killed.

An Exception was thrown while building the exit dialog. Added a workaround to ignore the Exception. If the Exception occurs, the User will be unable to dismiss the dialog with the 'x'. The "Cancel" button will still work.

3.13 JTLS-2017-13311 Unable To Find Formations On DDS Map

The DDS Map find function was not operable for formations.

The capability of using the Map find function in the DDS to search for functions was added.

3.14 JTLS-2017-13312 Sealane Editor Used Wrong Speed

The dialog to change a Sealane arc in the DDS used ground speed instead of naval speed.

The UOM type for the Sealane editor was changed from ground speed to naval speed.

3.15 JTLS-2017-13313 Scenario And Game Name Identical

Several scripts assume the scenario and database names are the same, so the database name should not normally be editable in the DCP.

The database name was made read only in the DCP, unless a special command line argument is passed to the DCP.

3.16 JTLS-2017-13314 Exception Editing JOBE Prior To Map Creation

If the User renamed a Unit in the Command Hierarchy prior to the map being created, an exception would occur.

The JOBE code was modified to prevent editing until the map has completed initialization and is visible.

3.17 JTLS-2017-13315 SIP Crash Database Testing

The SIP Crashed during database testing. A CSP CS record had been removed to create a desired Error 230. Although this error was properly identified, the SIP crashed before completing.

The SIP was changed to permit a graceful completion, while still identifying the CEP/SIP critical Error 230.

3.18 JTLS-2017-13316 Link16 - Feed OFF By Default

The Link 16 service feed was on by default. This caused messages to be delivered before filters could be put in place by Users.

The Link 16 service feed is now off by default.

3.19 JTLS-2017-13317 SVP Errors After Running ASC

The Automatic Supply Calculation (ASC) sets the wet and dry carry and storage capacities to be at least the total of the basic load. In some cases, due to numeric stability errors introduced during download, the SIP will find a number of errors or warnings related to the prototype not having sufficient capacity to store its basic load.

The ASC was modified to ensure that the carry capacities are at least 1.0 more than the sum of the basic load.

3.20 JTLS-2017-13318 Controller Damage Report Missing Data

An HRU was ordered to attack/raid a ground Unit. During the exchange of fire, the Controller Damage Report did not show any combat system losses inflicted by the ground Unit on the HRU. However, the HRU IMT and Player HRU Damage Report showed losses. Note that the Controller

Damage Report did show the losses inflicted by the HRU on the ground Unit, as did the ground Unit IMT and the Player Unit Damage Report.

When an HRU took losses, the routine that generated the Controller Damage Report was missing the necessary logic to obtain the HRU damage records and write out the results. The missing code was added to the routine and to the message template to display the HRU's losses in the report.

3.21 JTLS-2017-13319 Translating ATO Change Message

Whenever the ATOT is used to translate an ATO change, the new missions and the cancel missions for the period are processed properly, but all other missions are not. Orders for these other missions are withheld but the order utilities for the missions and the associated order groups are still getting created. This is confusing, because the User believes there are orders to be sent from the groups when there are only unnecessary utilities available from the groups.

The logic used by the ATOT for withholding the mission orders during translation of a changed ATO was not properly applied to the code that produces the order utilities.

3.22 JTLS-2017-13320 OMA Crash Land/Water Location

An order was being sent that attempted to move a naval vessel to a location within a 15 second grid. This resulted in the OMA crashing. This same situation would have occurred with 30 second grids. The crash was due to the fact that the grid data for that location did not exist.

The terrain level/grid data only contains those grids which are not covered by higher resolution grids. This avoids the CEP and other programs reading in data which will never be used. Thus it is not unusual for some grids to not exist. The real problem was not a missing grid, but that the calculation of which grid the location was in was wrong.

When reading the terrain level and grid data the OVT library, which is used by the OMA, determines the resolution of each layer, its starting lat/lon, and its ending lat/lon. This is then used to determine the location of each individual grid. However, for 30 second and 15 second grids the resolution was determined using integer math, $30/3600$ and $15/3600$ respectively. In C, which the OVT Library is written in, integer math truncates the real portion of the result and returns an integer resulting in a resolution of 0 in these cases. This resulted in the starting lat/lon and ending lat/lon of the layer being the same and caused the OVT Library to miss the layer when determining if a location was in it. This was easily solved by switching to floating point math in the equations, $30/3600.0$ and $15/3600.0$ respectively.

3.23 JTLS-2017-13321 ATO Parser Stop Time For RECCE Missions

The ICC uses a GTGTLOC MTF record for attack and Reconnaissance (RECCE) missions. For attack missions, the Not Later Than (NLT) time of the GTGTLOC record is not required whenever the Time On Target (TOT) time is filled. However, for RECCE missions, the ICC fills both the TOT

and NLT times intending to indicate the duration for orbiting at the location, which is also specified in the GTGTLOC record.

The ATO Parser does not extract the NLT time and apply it to the task completion on RECCE missions as the ICC intends. For RECCE missions in the MTF message generated by ICC the ATO Parser uses the TOT time from the GTGTLOC record as both start and end times for the orbit task.

The ICC F058, F068 and NEC CCIS ATO Parsers were modified to extract the NLT time when it is available and use the time as the completion time for the orbit.

3.24 JTLS-2017-13322 Enemy ARU Fails To Engage HRU

A non-covert HRU was detected by a nearby enemy Unit with weapons free rules of engagement (ROE). The enemy Unit did not attack the detected HRU. Note that in the reverse situation (i.e. when the HRU was the aggressor on Patrol/Ambush/Raid) the HRU did attack the enemy Unit or HRU.

The CEP routine that is invoked to start the battle between the enemy ground Unit (or an enemy HRU) and the detected HRU was unfinished. In fact, the routine was merely a stub. There was no logic present to initiate combat. The missing logic was added to the routine to cause the enemy ARU/HRU to attack the HRU. Note a different routine already allowed the engagement to occur in the reverse situation, i.e. when the HRU conducts a Raid, Patrol, or Ambush against the enemy ARU/HRU.

While testing this solution, the Controller message generated by the Set HRU Parameters order did not display the Detection Information correctly when the display All Data option was selected. Various conversion errors were present in the message. The problem was corrected in the message template which was missing the appropriate sub messages.

3.25 JTLS-2017-13323 Replay JXSR Crash

The Replay JXSR crashed while accepting a console connection at approximately 48 days of game run.

It was discovered that a system call to accept a new connection, or open a file would in certain circumstances provide a file descriptor of very large value. This file descriptor exceeded the number of values the JXSR was able to store, so the JXSR would crash. When investigating the number of file descriptors used by the Replay JXSR during very long run games it was discovered that the JXSR was holding to many of them for its store of delta files. In fact multiple descriptors were held for the same delta files. Further investigating this uncovered a JDSP library used by the replay JXSR which does not close the file descriptor and return it to the system before re-opening and accepting a new descriptor. Over time the descriptors provided by the system would grow in this way.

The JDSP library has been fixed so it closes the descriptor before re-opening and getting a new one.

3.26 JTLS-2017-13324 Disappearing Naval Routes

WHIP Map naval routes were disappearing, with no error or indication to why it occurred.

During the progression of a game, when necessary, the model will write out a new set of map filters that the WHIP loads to add or remove object prototypes that are available to filter their display features on or off. On demand routes, naval routes in this case, were not being temporarily saved while loading the new map filters causing any previously turned on individual routes to be cleared after the load completed. In addition, the naval routes filter panel SUP filter settings were being cleared and were also corrected by removing an unnecessary call that attempted to reset the previously set filters. These changes corrected the disappearing naval routes.

3.27 JTLS-2017-13325 Replay JXSR Useless Reports

As the Replay JXSR is running for a scenario it frequently writes reports in the Replay JXSR log file regarding the replay files. These files exist at the time the error is logged but the error states "file does not exist".

During normal execution, the Replay JXSR accesses a set of files located in the TRIPP directory for the associated scenario. The Replay JXSR uses a JDSP library function, called `file_list_filtered_directory()`, which returns the list of files located at the specified directory path. This function was not appending the file name of each file in the directory to the path before making a system call to determine if the file was a directory. Without the path prefixed to the file name, the system call would fail and log an error. The failure on each file in the TRIPP directory did not effect the Replay JXSR, because the complete list of files was always returned.

This path and file name problem was fixed. The function now combines the path and file name before making the system call.

3.28 JTLS-2017-13326 Great Circle Distance For Close Location Errors

Error 403, which checked to see if the runways assigned to an air base were located within the Unit radius, occurred unexpectedly when running the Scenario Verification Program (SVP). This error was reported even though the runways were at nearly the same location as the air base.

The code to check for Error 403 used a computation of the great circle distance between the air base and its runways. When these locations are nearly the same, but not exactly the same, errors occur from the formulae used to compute the distance between the objects. The equations were not suitable for these small distances. An additional check was added to the code to prevent these errors. When the separation between the objects was very close to zero, then the distance

between them was set to a value of almost zero. This code change resolved the issue from the SVP.

3.29 JTLS-2017-13327 Manage Formation Tasks Crash

The CEP crashed while a User was trying to manage tasks for a naval formation.

This crash occurred during the processing of the order to manage formation tasks. The code was preparing to send out a message to the Player, but the data required for the message was not found. The routine required a value for a data type representing the object type formation, but the routine which translates object types to data types did not have code for those types. It was added to the routine and the message from the order was processed correctly.

3.30 JTLS-2017-13328 Postponing Current Mission Task Crash

A Player attempted to add an air mission to an existing air mission package. When this occurred, one of the existing package missions was forced to postpone its current task. The model crashed during the routine processing the postponement. In this case, the mission was executing a rendezvous task.

There is an existing routine in the CEP which handles the postponement of tasks for air missions. It contains code to handle all of the different types of tasks which may be postponed. For the case of the rendezvous task, there was no set of code for that specific type of task. A basic entry was added for this case.

3.31 JTLS-2017-13329 Error In Message From Debug Object Order

A User sent the Debug Object CEP Order, but the resulting message was inconsistent with the settings in the order.

This order allowed the User to specify one or more objects to be flagged for debugging purposes. The User had turned on debugging for an Air Mission, a Unit, a Ship, a JDPI, and a Target. The fields for debugging a Formation, an HRU, a Supply Run, and a Cruise Missile were left blank. When this order was processed, the resulting message was correct for the Air Mission, Unit, and Ship. However, the message said that a Formation was being debugged with the name being that of the JDPI. Similarly, the message said an HRU with the name of the specified Target was being debugged.

The actual object settings for debugging were assigned correctly. The problem occurred when printing out the data for the message. Since some of the fields contained (unassigned) blank names, the message was improperly associating these fields with the wrong object types in the text. Changes were made to ensure that no text fields were left blank. If one or more of the object types are unassigned, then the message has a dash assigned to that object. This eliminated the problem in the resulting message.

3.32 JTLS-2017-13331 ELS Crash While Writing Checkpoint

The ELS crashed while attempting to write the checkpoint file associated with air missions.

This crash occurred because the ELS encountered an air mission which did not have data in a required array. Every mission in the ELS holds an array which defines the offset locations for the individual aircraft in the mission. When the ELS started to write these data into a checkpoint, it crashed because the array was empty. This array was incorrectly cleared during a change of the mission type. A User had ordered the mission to change from a reconnaissance mission to a patrol mission. During this process, the array for the mission offset locations was incorrectly removed. Code changes were made to correct this problem.

3.33 JTLS-2017-13332 ELS Crash For Transfer Aircraft Missions

When a Transfer mission was flown, the ELS crashed while assigning attributes to the newly created squadron. This was the new home squadron for the aircraft.

This crash occurred because the ELS template for the new squadron had an illegal value for the unit's command level. The ELS uses such templates to lay-out the unit's combat systems on the ground. Every template must have a legal command level. In this case, it was zero - which was not allowed. Modifications were made to ensure that a legal command level was always assigned to the template.

3.34 JTLS-2017-13333 Black Runways Not Gathered

When an airbase is magic moved on top of an unowned runway, it does not find the runway and take control of the available runway.

Magic moved airbases now take control of unowned runways they move on top of.

3.35 JTLS-2017-13334 Crash When Sinking Ship Creates Lifeboats

The model crashed when a sinking ship abandons ship and creates lifeboats. The crash only occurs when the ship's prototype number is greater than the total number of Tactical Unit Prototypes.

The problem was solved by using the proper routine to access the Combat System Prototype used by the sinking ship.

3.36 JTLS-2017-13335 Error Allowed Units To Cross Uncrossable Rivers

Direct ground unit movement was unimpeded by river barriers. Based on its agility characteristics, the Unit should not have been able to cross the river.

In the CEP, the capability to cross barriers was represented in an attribute of the Mobility Counter-Mobility (MCP) prototype. This attribute held the time delay that is needed for the unit to cross a barrier. This time delay is specified for each type of barrier and each agility type.

When a unit was ordered to move to a new location, the model evaluated the capability for that unit to cross any barriers along the route. This was done by comparing the time delay in the unit's MCP with a fixed value. An error occurred because the code was not always evaluating the time delay in the same manner when it determined if a barrier was uncrossable. This code inconsistency resulted in units being able to cross rivers despite their intended capabilities. Modifications were made to consistently use the correct logic to represent the effects of uncrossable barriers. This is as it is defined in the *JTLS-GO Data Requirements Manual* for the MCP BT AGT TIME DELAY attribute.

3.37 JTLS-2017-13336 GIS Tool River/Road Preprocessor Spurious Arcs

River and road preprocessors do not generate networks that follow the pattern of the input polylines.

The road, river and rail preprocessors take a polyline and break it up to form individual arcs and nodes. JTLS-GO attributes are also added to each arc and node. The network is built by identifying polylines that meet where nodes overlap to form junctions. Ideally, the latitude and longitude coordinates of the overlapping nodes will be identical.

In this case, the nodes are "merged": one node is arbitrarily retained by the network, and the other overlapping node(s) are removed from the network, since they become duplicate nodes. In some cases, however, the coordinates aren't identical, but are very close. In this case, the distance between the nodes must meet a threshold held by the SEARCH_RADIUS global variable. If the distance is within this constraint, the nodes are merged.

Given a test node, a search for nodes that are within the SEARCH_RADIUS is performed by using OpenMap's QuadTree class. The QuadTree provides a quick capability to find objects (in this case nodes) within a specified search region. Using the SEARCH_RADIUS to define the search region, the QuadTree returns a collection of nodes within the region, but it sometimes returns an additional node outside of the region. This outlier becomes merged with the test node, and as a result one can easily see the pattern of the arcs rendered on screen to deviate from the pattern of the original polylines. To resolve this problem the distance between the test node and the nodes returned by the QuadTree search is calculated. If the distance is within the SEARCH_RADIUS, the node is merged with the test node; otherwise, the node is ignored.

The problem was originally confirmed by running the river preprocessor on a sample of river polylines. Running the preprocessor on the same polylines after the fix was implemented, it was visually verified that the nodes and arcs generated retained the pattern of the original polylines.

3.38 JTLS-2017-13337 Crash When Naval Unit Attempts to Kill Convoy

A naval unit attempted to kill a convoy and the model crashed, with a reason of "subscript out of range". The prototype of the unit was being accessed to determine the overlap between the Unit and the convoy. The overlap algorithm assumed that the unit attempting to kill the convoy was a Tactical Unit Prototype (TUP) Unit.

The problem was solved. If the killing unit is a naval unit, then the overlap between the ship and the convoy is assumed to be 1.0.

3.39 JTLS-2017-13338 Game Start Supply Shortages

At game start, there are many units identified through the Message Browser as being short in supplies. Most are related to the ADA resupply categories. The DDS scenario verification does not report any shortages nor does a manual review of those categories in the DDS and OPM.

A detailed examination of the CEP code found the problem to be related to the storage of the data in the computer memory. The display values of real numbers may in some cases represent numbers that have been rounded. This appears to be the case with the supply shortage values. This situation has been detected in other parts of the JTLS-GO software and is directly attributed to the storage of the values internally to the machine. A code change was made to adjust for this particular occurrence.

3.40 JTLS-2017-13339 Special System Libraries For Newer Compiler

The source code for the JXSR, KOI, OTHGMS, TEMS and libjds_cpp would not compile under the newer C++ compiler (gcc version 4.8.5).

The source code was modified to include system libraries that allowed the newer compiler to build the object code. The object code is then linked to create the listed binaries.

3.41 JTLS-2017-13340 Crash When No Supplies For Tanker Mission

The CEP crashed when a tanker mission was not launch capable, because it could find absolutely no fuel.

The CEP logic now properly considers the situation in which absolutely no fuel is available for a tanker mission.

3.42 JTLS-2017-13341 Mission Ignores Altitude Change

A Manage Air Mission Task order was sent to reduce the altitude of an orbiting mission. The new altitude was intended to place the mission under the threat ring of a nearby SAM site. The Air Mission Tasking Orders IMT showed the new altitude for the Orbit task, but the SITREP and Air Mission IMT did not. Furthermore, the mission was shot down by the SAM site, even though the altitude specified in the order was below the SAM site's minimum engagement altitude.

The routine that processes the Manage Air Mission Tasks order assigned the new altitude to the aircraft's AM ALTITUDE attribute, but failed to assign the new altitude zone to the AM ALTITUDE ZONE attribute. The AM ALTITUDE ZONE is used by the model to determine if a SAM site covers the mission and can be engaged. Because the attribute still held the higher zone, the model correctly engaged the mission, even though the mission was actually flying below SAM coverage. The missing AM ALTITUDE ZONE assignment statement was added to the routine to correct the problem.

Also, the new altitude was not sent to the JDS to update the SITREP and Air Mission IMT. The missing subroutine call was added.

Finally, a minor clarification was made in the Player message that is received when the Change Air Mission Parameter order is sent to change a mission's altitude. The message text incorrectly stated that the orbit altitude had been changed which was not true for any future Orbit tasks that were pending. The Orbit task altitude is never changed by the Change Air Mission Parameter order.

3.43 JTLS-2017-13342 ELS Lifeboats Missing Rescued Passengers

When the ELS created lifeboats, there were no passengers assigned to the resulting HRUs.

When an aggregate ship sank, the ELS was properly generating a lifeboat. Each lifeboat was a newly created HRU with an assigned boat. The ELS was not correctly transferring the personnel from the ship to each new HRU. ELS code was changed to correct this problem. When a ship is sunk, the appropriate number of personnel are now correctly assigned to the lifeboats.

3.44 JTLS-2017-13343 Manage Formation Tasks Order Panel Incorrect

The sequence number of an existing task was unable to be changed without making any other changes, because the OMA rejected the order if only the "New Task Number" field is filled in.

The MFT.xml order file lacked a constraint that allowed only the "New Task Number" field to be filled in. The order file was changed to add the proper constraint.

3.45 JTLS-2017-13344 Fire Mission Task Cancel Message Blank Fields

An OAS mission did not execute its Fire Weapon task. A message informed the Player that the Fire Weapon task was canceled because the mission did not detect the ground object. The ground object type and name fields were blank in the message. Consequently, the Player could not determine which ground object was not detected.

The cancel reason was incorrectly placed in the routine that determines which task cancellation Player sub-message to send. The routine was corrected.

3.46 JTLS-2017-13345 OTHGold Time Late, Duplicate Track Info

It was not possible to determine what tracks were considered timelate by the service from the WHIP module. It was also difficult to determine how many critical track name duplicates were present.

Time Late tracks (when time late cut off settings are enabled) now have their "Status" table cell displayed in gray. A column has also been added on the summary table to display the number of timelate tracks per object type. The timelate cut off refresh interval has been changed from seconds to minutes.

Active tracks that share the same track name are considered to be "critical duplicates" and the total number of critical duplicate tracks per object type is now listed in the summary table.

3.47 JTLS-2017-13346 OTHGold Failed Update Status Label

When already connected to an OTHGold service, the service status label and title frame of the WHIP module failed to update when connecting to a different service.

The WHIP module now no longer checks to see if it is already connected to a service when processing connection updates.

3.48 JTLS-2017-13347 Minor OPM Issues

Two minor issues were noted in the Online Player Manuals. First, on the page for a specific Tactical Unit Prototype, the label in front of the time between organic intelligence reports is marked as a link that extends across a large amount of white space. Second, on the page listing a specific Ship Unit Prototype, the data describing the shelter priority is listed underneath the submarine data.

The link for the Tactical Unit Prototype was shortened and the shelter data was relocated.

3.49 JTLS-2017-13348 Empty Cell Designators In AAR Report Scripts

AAR reports scripts are designed to allow the formatting of AAR reports. The scripts query Oracle with a request and Oracle returns an HTML response providing the data in a tabular format. Since this response does not contain <thead> and <tbody> tags, which allow the end-user to perform column sorting, the script runs the response through a stylesheet to add these tags. In order to do that, the script must first convert the file in a proper XHTML format. In addition to this stripping, Oracle returned an escape sequence for an empty cell, " ", which the stylesheet could not handle and was therefore failing.

Besides converting non-XML line break and paragraph tags into properly terminated tags, the script was modified to also strip out all escape sequences.

3.50 JTLS-2017-13349 Change HQ Order Crash

A crash occurred when a Change HQ order, which would have caused a circular command chain, was rejected. Specifically the crash happened attempting to set the associated Pending Execution Task (PET) with a rejection reason, when there was no associated PET.

A Change HQ can happen as both a PET, which represents part of a larger plan such as a move order with a New HQ at the end, or may be an instantaneous action, such as with a Change HQ order. Since the absence of a PET is perfectly normal, the code was modified to check for the PET before attempting to assign a cancel reason to it.

3.51 JTLS-2017-13350 Crash While Finding Squadron Divert Location

The model crashed while attempting to find a divert location for a squadron.

This crash occurred because the model encountered an airbase which had no assigned runways. The search for a good divert location was checking to see if there were any nearby units which provided a suitable place to land. Since the airbase had no runways, the model crashed when it tried to check to see if there was a enough length in the available runways. The SVP already flagged such airbases with a Warning. They were not suitable divert locations and the code was modified to skip them in this search.

3.52 JTLS-2017-13351 Package Names From ATOT Conflict

Whenever an ATO uses the same name for an air mission package on consecutive days the ATO Translator allows the name exactly as provided by the ATO message. This causes the new package order to be rejected by the simulation.

The ATOT was modified to append to the package name the ATO day for the start of the ATO period. This will prevent following package orders for a subsequent day from conflicting with the previous package order of the same name..0

3.53 JTLS-2017-13352 JTOI ICC Radar/SAM Targets Reference Name

Radar and SAM targets were using the JTLS JODA attribute of "name_prefix" as their ICC ID.

Radar and SAM targets now use the JTLS JODA attribute of "reference_name" as their ICC ID.

3.54 JTLS-2017-13353 Wrong Warning Number Used In SVP

An SVP Warning used the wrong number to describe the Warning.

In the code, Warning 1268 was incorrectly used to check the supplies used by a unit prototype (TUP or SUP). The code should have used Warning 1269 for this purpose instead of Warning 1268. The Warning number was changed in the code to correct this problem.

3.55 JTLS-2017-13354 Tech Tool Copy Confirmation Window Too Large

The copy confirmation dialog for existing slides was listing all the selected existing slide names in one line, making the dialog window excessively large and causing it to obscure the confirmation buttons.

Line wrapping and a scroll bar was added to the dialog window where the slide names or group names are displayed, to ensure proper size of the dialog window.

3.56 JTLS-2017-13355 Civilian Air Traffic Mission Resource Event Crash

A CEP crash occurred during the resource event for a civilian air traffic mission. There were no available resources, so the mission was canceled and the model crashed.

The model crashed because the task to complete the mission was scheduled for a time in the past. If an event is scheduled for a time in the past, then the program reports an error that an attempt was made to decrease the simulation time. A check was added to the code to prevent the model from using a time in the past when it was scheduling this event to end the mission.

3.57 JTLS-2017-13356 Civilian Air Mission Unit Delay Crash

A user inadvertently submitted the same air tasking orders twice to create Civilian Air Traffic. This resulted in duplicate air missions. When the user attempted to cancel one of the missions, which was on the ground in a Unit Delay posture, the model generated some unexpected tasks and a message reporting a failure to cancel the mission. The user was unable to cancel this mission, and the model crashed while trying to check the runway status at the destination.

This problem was reproduced with a mission in Unit Delay posture. The logic used in the model was trying to add a Take Off task and a Land task to the mission before it could be canceled. Since the mission was already on the ground at the proper location, there was no need for it to take off and land, and it should have just canceled the current task in place.

The failure message was correctly reporting the problem because the mission was currently in a Unit Delay. The contents of the message were misleading because the mission should cancel at its current location, rather than fly to its destination before it completed the mission.

Changes were made to the code to correct both of these problems. If the mission is on the ground at an appropriate location for the mission to be canceled, then it simply completes the mission without any attempt to take off and land.

3.58 JTLS-2017-13357 SVPR Warning 1462 Corrections

Warning 1462 does not provide enough information to identify and correct the problem. The solution to open up the proper TUP and Combat System record does not function correctly.

The message displayed for Warning 1462 is supposed to tell the user how many and what Combat Systems are short, but only tells the user what Combat Systems are short. An error in the positioning of the parameters was corrected.

The solution to bring up the correct TUP and Combat System record also had an error in the parameters, which was caused all the Combat Systems to be displayed, instead of the incorrect entry. The parameters were corrected. The user will now have the correct record displayed, unless the record does not exist.

3.59 JTLS-2017-13358 Modify Attack Package Debug

The CEP entered debug mode while trying to change the timing method for a package while it was loading.

The code in the routine that checks the current posture of a package to determine which changes are allowable did not contain all the possible package posture types. The code was modified to include all possible package postures. In this case, a package that is already loading is not considered eligible for modification. The user needs to delete and resend the package.

3.60 JTLS-2017-13359 SUP Consumption Messages Missing

No message is returned from the CEP when the Controller sends a order to set one of the special SUP consumption factors.

The code to set the new values was determined to be working correctly. The code to report back to the user what changes were implemented contained several errors, which caused the program to be unable to identify the changes for reporting.

3.61 JTLS-2017-13360 Checkpoint Restart Crash

The CEP crashed while restarting from a checkpoint. It was reading the extra ASCII data for that checkpoint.

This crash occurred while the model was reading the value for the current number of air missions. The checkpoint file contained a non-integer value for this number. The bad data was caused during the saving of the data. The value for this data field was larger than four characters, so the program wrote out four asterisks instead of a legal integer. This problem was due to the fact that there were more than 9999 air missions flying. The code only allowed for a four digit number.

The code was changed to allow for larger numbers of air missions, but a logic error was added to record that there may be a problem later with duplicate mission names.

3.62 JTLS-2017-13361 SIP Errors Counts And Totals Do Not Match

Unit Error 414 was not included in the Unit errors count. This caused a mismatch between the individual count types and the total.

The unit errors were printed out before all units checks were complete. Error 414 was discovered by the sip after the unit counts were printed. This means that Error 414 was included in the total and not in the unit count causing the mismatch. The issue was fixed.

3.63 JTLS-2017-13362 Apache Module Crash During TCP Connection

Whenever an Apache service connection is required for communicating with one of the JTLS services, the associated service module sometimes crashes.

The JDSP library function that allows service modules of the Apache to make connections to their associated service had a problem while returning memory after attempting to establish a TCP connection. When a system call is made for name resolution, the call can fail. When this happened, the memory returned from the call was left uninitialized and caused the crash when it was being returned to the system. This JDSP library function was modified to check for name resolution failure before attempting to return the memory.

3.64 JTLS-2017-13363 CEP Crash Convoy Crossing National Boundary

A convoy was attempting to cross a national boundary when the CEP crashed. The error reported that the model attempted to decrease the simulation time.

When the convoy moved to a new location, there was a national boundary located between its current and next destination. The CEP code checked to see if the convoy was allowed to cross this national boundary. In this case the delay time for making the crossing was set to a negative one. The check to determine if the boundary was cross-able was only for delay times of zero. Since the value was not zero, it added the value of negative one to the total delay time for the convoy to make its next move. This resulted in a time for the next move which occurred before the current game time. Code changes were made to ensure that the ability for objects to cross national boundaries is consistent throughout the CEP.

3.65 JTLS-2017-13364 GIS Tool: Misclassified Terrain Grids

There were inconsistent results when the GIS tool classified a terrain grid's type. For example, a grid that is predominantly covered by ocean polygons is classified correctly as OCEAN, but a second grid that is also predominantly covered by ocean polygons is classified as OPEN instead of OCEAN.

To determine the terrain type to assign a grid, the tool finds polygons that cover, wholly or partially, the grid. These polygons represent one of the ten JTLS-GO terrain types: OPEN, ROLLING_HILLS, FOREST, DESERT, SWAMP, MOUNTAIN, ICE, CITY, OCEAN, and SMALL_ISLAND. The areas of these polygons are calculated. If the polygon is contained entirely within the grid,

the area of the entire polygon is calculated; if only part of the polygon lies inside the grid, the polygon is clipped along the grid's side(s) and the area is calculated only for the portion lying in the grid. The areas from polygons of the same terrain type are summed. Finally, the grid is assigned the terrain type with the greatest sum.

The polygons are stored in ESRI shapefiles. The ESRI standard states that no two sides of a polygon cross over each other, the vertex coordinates are listed in clockwise order and the first and last vertex coordinates are identical. If there are holes in the polygon, the coordinates outlining these holes are listed in counterclockwise order. The tool makes the assumption that any polygon it reads follows these rules and does not perform any error checking. As it turned out, some of the source polygons did not satisfy the identical first and last vertex coordinates requirement. This had a major impact when the area of the polygon was calculated: up to an order of magnitude smaller than it should be. A check has been implemented to insure that the identical first and last vertex coordinates requirement is met. If the coordinates are not identical, another vertex is added to the polygon with coordinates that are identical to the first vertex.

There are two situations the user must consider with this fix:

1. If the user created a GIS Tool project based on the "default" master database, and has previously executed Edit->Terrain Layer->Calculate Grid Parameters, it is recommended that the user re-execute Calculate Grid Parameters.
2. If the user created a new master database, it is recommended that the user preprocess the database, again: From the JavaMenu select Options 1->7->2 and from the "World Databases" combo box, select the appropriate database and click the Ok button. When the preprocessing finishes, the user should start the GIS Tool with this master database, and for each project that uses this database, the user should open the project and re-execute Edit->Terrain Layer->Calculate Grid Parameters.

3.66 JTLS-2017-13365 DCP Glassfish Default Database Name

When the DCP is run for the first time on systems where the user name is not "jtlsdev" or "jtlsrel", the Glassfish configuration's database name is set to "unset", but the field can no longer be edited.

DCP code was changed to set the default database name always to be the same as the scenario name on all systems, regardless of the user name.

3.67 JTLS-2017-13366 DDS Schema Script Removed From Directory

The create_dds_schema script pointed to a directory that was not \$JHOME/jtlsdev, because the third-party Java code used by the script was not in the proper directory, as it has not been decided where that code will eventually reside.

The create_dds_schema script was removed from the \$JHOME/jtlsdev/script directory.

3.68 JTLS-2017-13367 DCP Saving Domain Error

The DCP would throw an exception when a domain was saved that did not have any data migration destinations configured.

A check was added to check that the destination setup frame exists before attempting to set up the destination domains.

3.69 JTLS-2017-13368 ATOT Missing Airbase Creates Error

When an airbase no longer existed in the game, the ATOT would encounter an error when loading a file containing that airbase.

The ATOT now checks for non-existent airbases when attempting to display them.

3.70 JTLS-2017-13369 Equipment Not Sheltered On Unit Arrive

Aircraft were not being sheltered when they arrived in theater.

The issue was caused because a new flag indicating that a submarine could be placed in a shelter (i.e. was in port) was being accessed for all units. The flag was made active for all units and was properly initialized when a unit arrives in the theater.

3.71 JTLS-2017-13370 Crash Changing Station While Repositioning

The model crashed when a user told a ship that was currently repositioning to a new station to change station again.

The problem was caused by a copy and paste error in the code. The problem was corrected.

3.72 JTLS-2017-13371 Air Mission Not Properly Reporting Fuel Delay

The model always determined that the reason a mission could not launch was because of a weapon delay (not enough weapons).

The issue was solved.

3.73 JTLS-2017-13372 Layer Ordering In DDS And WHIP

In the WHIP, the shape layer was covered by the world map layer. In the DDS the shape layer was covering the CADRG layer.

The order of the layers that have solid shapes was changed, so that Terrain is on top, followed by CADRG, Shape, then World Map.

3.74 JTLS-2017-13373 TBM Stair-Step On TDACS

Theater Ballistic Missile (TBM) flights were showing up on TDACS as stair-step movement.

This problem was reported by a user, but the model team has not specifically seen the problem. During testing with TDACS, the development team did notice that when working with larger grids, the TBM locations were not being updated, and a code change was made to update the TBM location even in larger grids. After implementing this change, TBMs appears on TDACS with smooth parabolic movement. The development team has attributed this to the fix, but due to the pressure of the test schedule, we did not try to recreate the undesirable stair-step movement.

In addition, TDACS experts reviewed all TDACS traffic and indicated that we should only update the missile track every 12 seconds, rather than the normal 3 seconds that has been used in the past.

Finally, it was noted that the TBM's source JU number was not being properly assigned following the source number rules when a target detected the TBM. This problem was also corrected.

3.75 JTLS-2017-13375 Link-16 Message Service Leaks Memory

System memory used for holding two Link-16 list structures is being lost, especially during subsequent downloads from the JODA.

The object structure that hold both a list of target objects that own this object and a list of loaded weapons for air mission objects were not handled properly by the code. When updates were received, which cause an update to each of these lists, the previous list items were sometimes being lost. The code was changed to handle cases when the updates are applied, without losing the memory for the replaced data.

3.76 JTLS-2017-13376 Crash When Mission Runway Not Long Enough

If a mission arrives back at home and determines that the runway is not long enough to land, it attempts to find a divert base. If no such base exists, the mission attempts to land on the short runway. If all aircraft in the mission are killed attempting to land on the short runway, the model crashes along with the aircraft attempting to land.

The logic error was found and fixed. Now only the aircraft crash, and not the model.

3.77 JTLS-2017-13384 Infinite Loop - Ships Auto Firing

The model entered an infinite loop trying to determine if any enemy ship could auto-fire on a ship that just moved.

The problem was that the linked list was being re-arranged as part of the logic while the model was looping through the same list checking all enemy ships. The logic error was corrected.

3.78 JTLS-2017-13385 Checkpoint Start With Large Terrain File

The CEP does not start from a checkpoint that was taken using a very large terrain.

The routine used by the CEP to save terrain data into a terrain data file corresponding to the associated checkpoint was not allowing sufficient digits for the number of terrain layers. This was increased from three to five digits. When a checkpoint is taken the number of levels can now be greater than one thousand.

4.0 REMAINING ERRORS

Every effort has been made to correct known model errors. All reproducible errors that resulted in CEP catastrophic software failures (crashes) have been corrected. Other corrections were prioritized and completed according to their resource cost-to-benefit relationship.

As JTLS-GO represents a major release of new functionality, remaining outstanding errors have been considered to be obsolete and no longer relevant to JTLS-GO and have been removed from consideration for correction at this time. In future maintenance releases, outstanding errors related to JTLS-GO will be listed in this chapter, with information provided regarding the extent of the error, as well as suggestions to avoid or minimize the effects of the problem.

4.1 DDSC - TMU Line Mode Changes Multiple Grids

When using the line mode in the TMU, more grids than the ones the line passes through are changed. This can also cause a warning about trying to change multiple layers to appear.

4.2 DDSC - Multiple Types In DDS History Table

If records for more than one table type are selected in the DDS History table, "History Details" will display details for only one type.

4.3 WHIP - Pipeline Not Shown On IMT

A pipeline being operated by a non-detected unit is not shown in the pipeline IMT.

4.4 DDSC/WHIP/JOBE - CADRG Map Zoom

When using the CADRG map projection, if the width of the map is less than the height the zoom tool does not work correctly.

4.5 WSM - Many Messages Cause Lockup

If a service produces a large number of log or error messages in a short period of time, it can cause the WSM to lockup.

4.6 Strategic Airlift Does Not Work

Users should not submit Strategic Airlift Orders

4.7 Tactical Ground Formation Attacks Do Not Work

Users should not submit Tactical Ground Formation Attack orders. Tactical Ground Formation moves work fine.

4.8 Materiel Handling Equipment (MHE) Does Not Work

The use of Materiel Handling Equipment (MHE) has been temporarily disabled in the model.

4.9 Port Amphibious Operations Do Not Work

Users should not submit an order to do an amphibious assault or an amphibious pickup with the Port Option. The user should simply indicate that “Beach” operations are desired or Magic Move the unit off of the formation at an appropriate time.

4.10 Multiple Point Ground Group Move Results Not Consistent

A Move order was submitted to a ground unit with a specified route consisting of three points. Four other ground units were specified as part of the group move (not a TGF move). After the lead unit completed its third Move task and the other units finished their movement, the position of the other units varied significantly from their starting locations relative to the lead unit before the first move began. The reason this happened is that when the group leader starts its move, it makes note of the relative position to each of the units in the group. Not all units were complete with their first move; therefore, the relative positions did not match the starting relative positions. To solve this problem, insure that all units have completed a previous group move before giving a second group move order.

APPENDIX A. ABBREVIATIONS AND ACRONYMS

Terms are included in this Appendix to define their usage in JTLS-GO design, functionality, and documentation.

AAA	Anti-Aircraft Artillery
AADC	Area Air Defense Commander
AAL	Air-to-Air Lethality
A/C	Aircraft
ACP	Air Control Prototype
ADA	Air Defense Artillery
AEW	Airborne Early Warning
AFB	Air Force Base
AG	Air-Ground (Air-to-Ground)
AI	Air Interdiction
AIM	Air Intercept Missile
AIREF	Air Refueling
AKL	Area Kill Lethality
AMMO	Ammunition
AO	Area of Operations
AOC	Air Operations Center
APC	Armored Personnel Carrier
ARECCE	Armed Reconnaissance
ARTE	Air Route
ARTY	Artillery
ASC	Automatic Supply Calculation
ASCII	American Standard Code for Information Interchange
ASW	Anti-Submarine Warfare
ATC	Aircraft Target Category
ATGM	Anti-Tank Guided Missile
ATK	Attack
ATO	Air Tasking Order
ATORET	Air Tasking Order Retrieve Program
ATOT	Air Tasking Order Translator
AWACS	Airborne Warning And Control System
AZ	Altitude Zone

BADGE	Bilateral Air Defense Ground Environment (used by Japan Defense Agency)
BAI	Battlefield Air Interdiction
BDA	Battle Damage Assessment
BDE	Brigade
BN	Battalion
C3	Command, Control, and Communications
C3I	Command, Control, Communications, and Intelligence
C4I	Command, Control, Communications, Computers, and Intelligence
CA	Civil Affairs
CADRG	Compressed ARC Digitized Raster Graphics
CAP	Combat Air Patrol
CAS	Close Air Support
CAT	Category
CCF	Central Control Facility
CCP	Command Control Prototype
CCU	Controller Change Unit
CEP	Combat Events Program
CMDR	Commander
COP	Common Operational Picture
CP	Combat Power
CS	Combat System
CSP	Combat System Prototype
CTAPS	Contingency Tactical Air Planning System
CTG	Commander Task Group
CTRL	Control keyboard command
DCA	Defense Counter Air
DCL	Digital Command Language
DDS	Database Development System
DEMSDB	Demonstration Standard Database
DISA	Defense Information Systems Agency
DIV	Division
DMA	Defense Mapping Agency
DoD	Department of Defense
DOS	Days of Supply

DPICM	Dual Purpose Improved Conventional Munitions
DS	Direct Support
DSA	Directed Search Area
DTG	Date Time Group
EC	Electronic Combat
ECM	Electronic Counter Measure
ECP	Engineering Change Proposal
EI	Essential Elements of Information
ELINT	Electronic Intelligence
ELS	Entity Level Server
EODA	Entity Level JTLS Object Data Authority
ETA	Estimated Time of Arrival
FARP	Forward Arming and Refueling Point
FLP	Fire Lethality Prototype
FLOT	Forward Location of Troops
FOL	Forward Operating Location
FWL	Frederick W. Lanchester (originated a differential equation model of attrition)
GAL	Gallon
GCCS	Global Command and Control System
GRTE	Ground Route
GS	General Support
GSR	General Support Reinforcing
GUI	Graphical User Interface
HARM	High-speed Anti-radiation Missile
HE	High Explosive
HELO	Helicopter
HMMWV	High Mobility Multipurpose Wheeled Vehicle
HQ	Headquarters
HRU	High Resolution Unit
HTML	Hypertext Markup Language
HTT	High resolution unit Target Type
HUP	High resolution Unit Prototype
ICM	Improved Conventional Munitions
ICP	Interface Configuration Program

ICPLogin	Interface Login Program
ID	Identifier
IFF	Identification Friend or Foe
IIP	Intelligence Information Prototype
IMT	Information Management Tool
INFO	Information
INTEL	Intelligence
JCATS	Joint Conflict And Tactical Simulation
JDA	Japan Defense Agency
JDPI	Joint Desired Point of Impact (formerly DMPI: Desired Mean Point of Impact)
JDS	JTLS Data System
JDSP	JTLS Data System Protocol
JEDI	JODA Entity Data Identifier
JMCIS	Joint Maritime Combat Information System
JMEM	Joint Munitions Effectiveness Manuals
JODA	JTLS Object Distribution Authority
JOI	JTLS Operational Interface
JPL	Jet Propulsion Laboratory
JRSG	Joint Rapid Scenario Generation (formerly JIDPS: Joint Integrated Database Preparation System)
JSDF	Japanese Self-Defense Force
JTLS	Joint Theater Level Simulation
JTLS-GO	Joint Theater Level Simulation - Global Operations
JTOI	JTLS Transaction Operational Interface
JXSR	JTLS XML Serial Repository
KIA	Killed In Action
KM	Kilometer
KNOTS	Nautical miles per hour
LA	Lethal Area
LAN	Local Area Network
LAT	Latitude
LB	Login Build (JTLS order type)
LDAP	Lightweight Directory Access Protocol
LDT	Lanchester coefficient Development Tool
LOG	Logistics

LOGIN	Logistics Input
LOGREP	Logistics Report
LONG	Longitude
LOTS	Logistics Over The Shore
LR	Long Range
M&S	Modeling and Simulation
MAPP	Modern Aids to Planning Program
MB	Megabyte
MCP	Mobility Counter-mobility Prototype
MCR	Model Change Request
MG	Machine Gun
MHE	Material Handling Equipment
MIP	Model Interface Program
MOGAS	Motor Gasoline
MOPP	Mission-Oriented Protective Posture
MOSAIC	NCSA user interface software
MOTIF	X Window System graphical interface
MP	Maneuver Prototype
MPP	Message Processor Program
MSC	Major Subordinate Command
MSG	Message
MTF	Message Text Formats
MUREP	Munitions Report
MUSE	Multiple Unified Simulation Environment
NCSA	National Center for Supercomputing Applications (University of Illinois)
NEO	Noncombatant Evacuation Operations
NFS	Network File Server
NGO	Non-Governmental Organization
NIS	Network Information Service or Network Information System
NM	Nautical Mile
NTSC	Naval Telecommunications System Center
OAS	Offensive Air Support
OBS	Order of Battle Service (formerly UGU: Unit Generation Utility)
OCA	Offensive Counter-Air

OJCS	Organization of the Joint Chiefs of Staff
OMA	Order Management Authority
ONC	Operational Navigation Chart
OPM	Online Player Manual
OPP	Order Preprocessing Program
OTH	Over The Horizon
OTH Gold	Over The Horizon message specification
OTH-T	Over The Horizon-Targeting
pD	Probability of Detection
pE	Probability of Engage
pH	Probability of Hit
pK	Probability of Kill
PKL	Point Kill Lethality
POL	Petroleum, Oil, and Lubricants
POSIX	International operating system standard based on System V and BSD
PPS	Postprocessor System
PSYOPS	Psychological Operations
RAM	Random Access Memory
RDMS	Relational Database Management System
RECCE	Reconnaissance (air missions)
RECON	Reconnaissance (ground missions)
REGT	Regiment
RNS	Random Number Seed
ROE	Rules Of Engagement
RPT	Report
RSP	Reformat Spreadsheet Program
SAL	Surface-to-Air Lethality
SAM	Surface-to-Air Missile
SAM/AAA	Surface-to-Air Missile/Anti-Aircraft Artillery
SC	Supply Category
SCP	Simulation Control Plan
SDB	Standard Database
SEAD	Suppression of Enemy Air Defense
SIMSCRIPT	Simulation programming language (product of CACI, Inc.)

SIP	Scenario Initialization Program
SITREP	Situation Report
SLP	Sustainment Log Prototype
SOF	Special Operations Forces
SP	Survivability Prototype
SQL	Structured Query Language
SR	Short Range
SRP	Start/Restart Program (a JTLS component)
SRTE	Sea Route
SSM	Surface-to-Surface Missile
STR	Software Trouble Report
SUP	Ship Unit Prototype
SVP	Scenario Verification Program
SYNAPSE	Synchronized Authentication and Preferences Service
TACELINT	Tactical Electronic Intelligence
TADIL	Tactical Digital Interface Link
TCP/IP	Transmission Control Protocol/Internet Protocol
TEL	Transporter Erector Launcher
TG	Target entity attribute prefix
TGS	Terrain Generation Service (formerly TPS:Terrain Preparation System)
TGT	Target
TMU	Terrain Modification Utility
TOE	Table of Organization and Equipment
TOT	Time Over Target
TOW	Tube-launched Optically-tracked Wire-guided missile
TPFDD	Time-Phased Force Deployment Data
TTG	Target Type Group
TTL	Target Types List
TUP	Tactical Unit Prototype
TW	Targetable Weapon
UBL	Unit Basic Load
UIM/X	GUI builder tool
UNIX	POSIX-compliant operating system
UNK	Unknown

UOM	Unit Of Measure
USA	United States Army (U.S. and U.S.A. refer to United States and United States of America)
USAF	United States Air Force
USCG	United States Coast Guard
USMC	United States Marine Corps
USMTF	United States Message Text Format
USN	United States Navy
UT	Unit entity attribute prefix
UTM	Universal Transverse Mercator
VIFRED	Visual Forms Editor
VMS	Virtual Memory System
VTOL	Vertical Take-Off and Landing aircraft
WAN	Wide Area Network
WDRAW	Withdraw
WEJ	Web Enabled JTLS
WHIP	Web Hosted Interface Program
WIA	Wounded In Action
WPC	Warrior Preparation Center
WPN	Weapon
WT	Weight
WW	Wild Weasel
XMS	XML Message Service

APPENDIX B. Version 5.0.9.0 STANDARD DATABASE CHANGES

No new changes were made to the Standard Database (SDB) for this maintenance release.

APPENDIX C. Version 5.0.9.0 DATABASE CHANGES

No changes were made to the JTLS-GO 5.0.9.0 database for this maintenance release.