

# JTLS-GO

## Version Description Document

January 2020



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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS  
(JTLS-GO 5.1.5.0)

[Blank Page]

## ABSTRACT

The Joint Theater Level Simulation - Global Operations (JTLS-GO<sup>®</sup>) 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 5.1.5.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 5.1.5.0 is a Maintenance release of the JTLS-GO 5.1 series that includes an updated wespac51 demonstration database as well as updated repository data held in the repository51 database. There are no major Engineering Change Proposals (ECPs) included with this release, but there are a few minor ECPs that required no new data or data format changes, 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

WARNING – This document contains technical data whose export is restricted by the Arms Export Control Act (Title 22, U.S. C., Sec 2751, et seq.) or the Export Administration Act of 1979, as amended, Title 50, U.S.C., App. 2401 et seq. Violations of these export laws are subject to severe criminal penalties. Disseminate in accordance with provisions of DoD Directive 5230.25.

Distribution authorized to U.S. Government Agencies and private individuals or enterprises eligible to obtain export-controlled technical data in accordance with DoD Directive 5230.25 (date of determination). Controlling DoD office is Joint Staff, J7 - Joint Force Development.

Copyright 2020 - 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 Obsolete/Outdated Documents .....	1-1
1.2.2 Unchanged Documents .....	1-1
1.2.3 Updated Documents .....	1-2
1.2.4 New Documents .....	1-2
1.2.5 Delivered Software Components .....	1-2
1.2.6 Released Databases .....	1-4
1.3 INTERFACE COMPATIBILITY .....	1-5
1.3.1 Support Software .....	1-5
1.3.2 JTLS-GO Information Assurance Compliance .....	1-7
1.3.3 JTLS-GO High Level Architecture Compliance .....	1-8
1.4 DATABASE MODIFICATIONS .....	1-9
1.4.1 JTLS-GO Using Legacy Default Symbol Set .....	1-9
1.4.2 JTLS-GO Using New Default Symbol Set .....	1-9
1.4.3 Standard Repository 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 Special Installation Instructions .....	1-10
1.5.3.1 OTH-Gold Ship Types .....	1-10
1.5.3.2 Generation Of Combat System Summary Files .....	1-11
1.5.3.3 Aircraft Type Foreign Key Definitions .....	1-12
2.0 ENGINEERING CHANGE PROPOSALS.....	2-1
2.1 JTLS-2019-14602 Resupply Interactions For GlobalSim Federation .....	2-1
2.2 JTLS-2019-14615 Reformat Spreadsheet Program MGRS Conversion .....	2-1
2.3 JTLS-2020-14625 Quick Order To Change Mission Speed/Altitude .....	2-2
2.4 JTLS-2020-14627 IMT Column Size Adjustment .....	2-2
2.5 JTLS-2020-14628 Add Concept Of COMAO ID .....	2-2
2.6 JTLS-2020-14630 ATO Lite Game .....	2-3
3.0 SOFTWARE TROUBLE REPORTS.....	3-1
3.1 JTLS-2019-14537 JSAT Not Provided Primary WHIP By ICP .....	3-1
3.2 JTLS-2019-14539 IIR Report Does Not Always List Damaged JDPI .....	3-1
3.3 JTLS-2019-14540 JSAT Classification With Spaces .....	3-1
3.4 JTLS-2019-14541 Referenced Aircraft Class Can Be Deleted .....	3-2
3.5 JTLS-2019-14542 JODA Errors When Package TOT Changed .....	3-2
3.6 JTLS-2019-14543 IIR Detected Ships Show Improper Names .....	3-2
3.7 JTLS-2019-14544 Invalid Ship Category Sent Via OTHGold .....	3-2

3.8	JTLS-2019-14546 CEP Receives JSAT Order And Enters Debug .....	3-3
3.9	JTLS-2019-14547 Statistics File Takes Up Too Much Memory .....	3-3
3.10	JTLS-2019-14548 WHIP/DDSC/JOBE Map Find Result .....	3-3
3.11	JTLS-2019-14549 Naval OTHGold Category Value Invalid .....	3-4
3.12	JTLS-2019-14550 ELS Targets Not Moving With Owing Units .....	3-4
3.13	JTLS-2019-14551 Resurrected Entities Alive But On-Fire .....	3-4
3.14	JTLS-2019-14552 ELS Crash Inconsistent Assets In CEP .....	3-4
3.15	JTLS-2019-14553 DDS EEV Unit Report Migration .....	3-5
3.16	JTLS-2019-14554 Warning 1635 Options Nonfunctional .....	3-5
3.17	JTLS-2019-14555 Warning 1502 Did Not Open Table .....	3-5
3.18	JTLS-2019-14556 Warning 1255 Correction Does Not Work .....	3-5
3.19	JTLS-2019-14557 Wrong Table Name In Properties Key .....	3-5
3.20	JTLS-2019-14558 TBMCS And JTLS-GO ICAOs Will Not Map .....	3-5
3.21	JTLS-2019-14559 Warning 1636 Link 16 Feed Name .....	3-6
3.22	JTLS-2019-14560 OTHGold File Resource Not Closed On Failure .....	3-6
3.23	JTLS-2019-14561 HIP Missing Shipment Type For Supplies .....	3-6
3.24	JTLS-2019-14562 Consolidate Font Settings For Web Login Page .....	3-6
3.25	JTLS-2019-14563 TBMCS JTOI Fixes .....	3-6
3.26	JTLS-2019-14564 ATOT Foreign Squadrons Linked To Tasking Units .....	3-6
3.27	JTLS-2019-14565 JSAT Issues With Large Number Of Satellites .....	3-7
3.28	JTLS-2019-14566 SIP Assumes Ground Unit Owns Targets .....	3-7
3.29	JTLS-2019-14567 LOITERING_MUNITION Load Weapon Type .....	3-7
3.30	JTLS-2019-14568 ATOT Alert Package Empty Location Task .....	3-8
3.31	JTLS-2019-14569 ATOT Missing Error Report Details .....	3-8
3.32	JTLS-2019-14571 ATOT EC Mission Transit Point Locations .....	3-8
3.33	JTLS-2019-14572 ATOT Transit Taskings And Mission Start Time .....	3-8
3.34	JTLS-2019-14573 TACELINT Message Field Formats .....	3-8
3.35	JTLS-2019-14574 ELS Memory Leaks And Variable Inconsistencies .....	3-9
3.36	JTLS-2019-14575 Issues Creating New Units and HRUs in the ELS .....	3-9
3.37	JTLS-2019-14576 JOI Memory Leaks And Unreleased Resources .....	3-9
3.38	JTLS-2019-14577 JOI/JTOI Inappropriate Times Data Type .....	3-10
3.39	JTLS-2019-14578 Non-Configuration Managed File Will Not Save .....	3-10
3.40	JTLS-2019-14579 i18n Injection Vulnerability .....	3-10
3.41	JTLS-2019-14580 OPAREA Incorrect Side Visibility Array .....	3-10
3.42	JTLS-2019-14581 JSAT Memory Leaks And Type Mismatch Values .....	3-10
3.43	JTLS-2019-14582 HIP Crashes When HRU Destroyed .....	3-10
3.44	JTLS-2019-14583 Split Mission Jumps To Equator .....	3-11
3.45	JTLS-2019-14584 Federation Data SVP Corrections .....	3-11
3.46	JTLS-2019-14585 Crash Left Over Rendezvous Point .....	3-11
3.47	JTLS-2019-14586 Stationary Target Not Placed On Game Board .....	3-12
3.48	JTLS-2019-14587 Naval Unit Placed On Land Grid SVP Error .....	3-12
3.49	JTLS-2019-14588 Adding Link 16 Target To Game Crash .....	3-12
3.50	JTLS-2019-14590 ATOT Auto-Link Perceived Foreign ICAOs .....	3-13

3.51	JTLS-2019-14591 JOI OTHGold Reset Comments To Empty .....	3-13
3.52	JTLS-2019-14592 ATO Mission Type Information .....	3-13
3.53	JTLS-2019-14593 DSP Incorrect Output Directory .....	3-13
3.54	JTLS-2019-14595 ATOT Not Writing Orbit Path .....	3-13
3.55	JTLS-2019-14596 Link 16 Capable SAM Vs Controller Damage .....	3-14
3.56	JTLS-2019-14597 Associated Targets Not Usable In Federation .....	3-14
3.57	JTLS-2019-14598 ATO Parser Incorrect Date .....	3-14
3.58	JTLS-2019-14599 JOI Column Order Inconsistent .....	3-15
3.59	JTLS-2019-14600 ATOT Extra Orbit Task .....	3-15
3.60	JTLS-2019-14601 JOI Send Email Using Sendmail Utility .....	3-15
3.61	JTLS-2019-14603 ELS Naming Scheme For Target Elements .....	3-15
3.62	JTLS-2019-14604 OTHGold Fixes .....	3-16
3.63	JTLS-2019-14605 Link 16 Autoselect Network Structure .....	3-16
3.64	JTLS-2019-14606 ATOT Attack Package Mission Missing Location ..	3-16
3.65	JTLS-2019-14609 Crash Merging Formations .....	3-16
3.66	JTLS-2019-14610 ELS Convoys With Destroyed Trucks .....	3-17
3.67	JTLS-2019-14611 CEP Batch Run Without AAR .....	3-17
3.68	JTLS-2019-14612 Sinking Ship Station Location Crash .....	3-17
3.69	JTLS-2019-14613 Fortify TRIPP Report Browser Null Dereference ...	3-18
3.70	JTLS-2019-14614 IMT Package Mission Quick Button Inoperative ..	3-18
3.71	JTLS-2019-14616 Unit Moving While In Attached Posture .....	3-18
3.72	JTLS-2019-14617 Memory Leak Print OPM Foreign Units .....	3-18
3.73	JTLS-2019-14618 Refuel Spreadsheet Provide Fuel Magically .....	3-18
3.74	JTLS-2019-14619 Nodes At Same Location Crash .....	3-19
3.75	JTLS-2019-14620 Apache Port Support .....	3-19
3.76	JTLS-2019-14621 HIP JTLS-GO Owned Attributes .....	3-19
3.77	JTLS-2019-14622 Add Target Move To Warning 1606 .....	3-19
3.78	JTLS-2020-14623 Replenishing Unit Did Not Follow Moving Unit .....	3-20
3.79	JTLS-2020-14631 Package Mission Against JDPI Does Not Work .....	3-20
3.80	JTLS-2020-14634 Gather Grids Incorrect When Point On Line .....	3-20
3.81	JTLS-2020-14635 Unusable Location In SVP Correction .....	3-21
3.82	JTLS-2020-14636 Crash Ending Coalition Support In Dual Grid .....	3-21
3.83	JTLS-2020-14637 ELS Not Saving Customized Templates .....	3-21
3.84	JTLS-2020-14638 Cruise Missiles At Zero Altitude .....	3-21
3.85	JTLS-2020-14639 No ATO Mission Type For Package Mission .....	3-22
3.86	JTLS-2020-14640 Air Mission ELINT Detections Not Turned Off .....	3-22
3.87	JTLS-2020-14642 Missing End Record Character On MTF Message .....	3-22
3.88	JTLS-2020-14643 Submarine Contact Report Error .....	3-22
3.89	JTLS-2020-14644 Execute HRU Event Not Setup Properly .....	3-22
3.90	JTLS-2020-14645 DJINN Uploading Newly Created Ground Unit .....	3-23
3.91	JTLS-2020-14646 Great Circle Distance Under 1 Meter Error .....	3-23
3.92	JTLS-2020-14647 ATO Parser Error Report Should Be Warning .....	3-23
3.93	JTLS-2020-14648 ATOT Reports Missing Mission Tasks .....	3-23

3.94 JTLS-2020-14649 DSA Spreadsheet Program Issues .....3-24

3.95 JTLS-2020-14650 OPM Formatting Error .....3-24

3.96 JTLS-2020-14651 Crash Destroying Auto Move Task .....3-24

3.97 JTLS-2020-14652 Manage SMA Order .....3-25

3.98 JTLS-2020-14653 ELS Template Unassigned Echelon Crash .....3-25

3.99 JTLS-2020-14654 ATOT Converts ACO Circles To DSAs .....3-25

3.100 JTLS-2020-14655 Infinite Loop Ship Running Out Of Fuel .....3-25

3.101 JTLS-2020-14656 Magic Air Ops Order Crash .....3-26

3.102 JTLS-2020-14657 Ghost Updates Always Saved .....3-26

3.103 JTLS-2020-14658 DDS Slowdown Loading Large Networks .....3-26

3.104 JTLS-2020-14659 Order Group Editor Save Empty Groups .....3-27

3.105 JTLS-2020-14660 Aircraft Class Link 16 Capable Flag Mismatch ..3-27

3.106 JTLS-2020-14661 Link16 And OTHGold Unknown Relationship ....3-27

3.107 JTLS-2020-14662 WHIP View Mission MTF Text From Controller ...3-27

3.108 JTLS-2020-14663 TACELINT Service Compatibility Fixes .....3-28

3.109 JTLS-2020-14665 Air Defense Fires Zero Weapons At Missile .....3-28

3.110 JTLS-2020-14666 DDSC Edit Railroad/River Arcs .....3-28

3.111 JTLS-2020-14667 ATOT Package Mission Order Target Type List ...3-28

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

APPENDIX A. ABBREVIATIONS AND ACRONYMS ..... A-1

APPENDIX B. Version 5.1.5.0 DATABASE CHANGES ..... B-1

APPENDIX C. Version 5.1.5.0 REPOSITORY CHANGES ..... C-1

    4.6 Added TUPs: ..... C-1

    4.7 Updated Combat Systems for TUPs: ..... C-2

    4.8 Updated Combat Systems and Renamed TUPs: ..... C-2

    4.9 Deleted TUPs: ..... C-2

    4.10 Added Ground Unit: ..... C-3

    4.11 Added Combat Systems: ..... C-3

    4.12 Added SUPs: ..... C-4

    4.13 Added Naval Units: ..... C-4

    4.14 Changed Naval Unit SUPs: ..... C-5

    4.15 Changed Naval Unit Homebase: ..... C-5

    4.16 Deleted Naval Units: ..... C-5

    4.17 Added HUPs: ..... C-5

    4.18 Added HRUs ..... C-5



4.19 Added ADA Class: ..... C-6  
4.20 Added SSMs: ..... C-6  
4.21 Added Supply Categories: ..... C-6  
4.22 Added Target Type Group: ..... C-7  
4.23 Added Targetable Weapons: ..... C-7  
4.24 Added New SKL Table: ..... C-7  
4.25 Added New PK Table: ..... C-7  
4.26 Updated PK Tables: ..... C-7  
4.27 Added New PH Table: ..... C-8  
4.28 Updated PH Tables: ..... C-8  
4.29 Added AKL Table: ..... C-8  
4.30 Added UOMs: ..... C-8  
4.31 Added SAL: ..... C-8  
4.32 Changed TW Table (TIME PER ROUND): ..... C-8  
4.33 Changed TW Table (TIME TO PREPARE FIRE): ..... C-9  
4.34 Changed TW Table: ..... C-9  
4.35 Added Aircraft Target Class: ..... C-9  
4.36 Changed Aircraft Classes: ..... C-9  
4.37 Added Sensors: ..... C-10  
4.38 Changed Global Values: ..... C-10  
4.39 Changed Small Boat Table: ..... C-10  
4.40 Changed Jammer Table: ..... C-10  
4.41 Updated Russian Order of Battle: ..... C-10

## 1.0 INTRODUCTION

### 1.1 SCOPE

This *JTLS-GO Version Description Document (VDD)* describes Version 5.1.5.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO<sup>®</sup>) software suite. JTLS-GO 5.1.5.0 is a Maintenance release for the JTLS-GO 5.1 series.

JTLS-GO 5.1.5.0 includes the entire JTLS-GO suite of software, a repository of engineering level data, and a realistic demonstration scenario based on the Western Pacific theater of operations, called “wespac51”. No database format modifications have been made for this release, but a static data error for allowable OTH-Gold Ship Types was corrected in Version 5.1.1.0. This does require a user to execute a correction procedure for all of their Version 5.1.0.0 scenarios loaded in Oracle. Information on this procedure can be found on [Page 1-11](#).

Descriptions of minor Engineering Change Proposals (ECPs) implemented for this release are provided in [Chapter 2.0](#). Explanations of all Software Trouble Reports (STRs) corrected in this release are provided in [Chapter 3.0](#). Outstanding STRs are provided in [Chapter 4.0](#). Changes made to the JTLS-GO 5.1 engineering data repository are provided in [APPENDIX C](#).

JTLS-GO 5.1.5.0 executes on the Red Hat Enterprise Linux Version 7.6 64-bit operating systems. The Web-Hosted Interface Program (WHIP<sup>®</sup>) 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 Portable Document Format (PDF) within a documents subdirectory.

#### 1.2.1 Obsolete/Outdated Documents

No documents have been deleted or become outdated as a result of this release.

#### 1.2.2 Unchanged Documents

- *JTLS-GO Analyst Guide* (JTLS-GO Document 01, Version 5.1.3.0)
- *JTLS-GO Configuration Management Plan* (JTLS-GO Document 03, Version 5.1.2.0)
- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 5.1.3.0)
- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 5.1.2.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 5.1.3.0)

- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 5.1.3.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 5.1.3.0)
- *JTLS-GO Repository Description* (JTLS-GO Document 14, Version 5.1.2.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 5.1.2.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 5.1.2.0)
- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 5.1.2.0)
- *JTLS-GO Air Services User Guide* (JTLS-GO Document 24, Version 5.1.2.0)

### 1.2.3 Updated Documents

- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 5.1.5.0)
- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 5.1.5.0)
- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 5.1.5.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 5.1.5.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 5.1.5.0)
- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 5.1.5.0)

### 1.2.4 New Documents

No new documents are delivered with JTLS-GO 5.1.5.0.

### 1.2.5 Delivered Software Components

JTLS-GO 5.1.5.0 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)
- JTLS Symbols Application (JSYMS)

- Database Development System (DDS)
  - Database Configuration Program (DCP)
  - DDS Client User Interface (DDSC)
- 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-Hosted Interface Program (WHIP) and its component programs:
  - Apache Server (APACHE)
  - JTLS XML Serial Repository (JXSR)
  - Order Management Authority (OMA)
  - Synchronized Authentication and Preferences Service (SYNAPSE)
  - XML Message Service (XMS)
  - Total Recall Interactive Playback Program (TRIPP)

When operating the TRIPP capability in current JTLS-GO releases, users are not prevented from logging into an actively running TRIPP making a connection to the same Replay JXSR. A TRIPP, as documented, requires its own Replay JXSR to control and perform the replay of the recorded simulation events.

This situation as a consequence can have more than one user concurrently control the playback of the game, as the shared Replay JXSR will honor each of the user's playback requests and will then change what each connected user sees on their TRIPP instance. This issue will be addressed in a future JTLS-GO release by only permitting one login per TRIPP user instance, such as a WHIP login prevents from users logging into the same WHIP instance.

- Entity Level Server (ELS)
- JTLS Operational Interface (JOI) for both OTH-Gold and Link 16 generation
- Tactical Electronic Intelligence (TACELINT) Message Service
- KML Operational Interface (KOI)
- JTLS Transaction Interface Program (JTOI)
- JTLS Interface Network Navigator (JINN)
- JTLS Order of Battle Editor (JOBE)
- JTLS Geographic Information System (GIS) Terrain Building Program

Instructions for installing JTLS-GO 5.1.5.0 are provided in the *JTLS-GO Installation Manual*. Installing a previous version of JTLS-GO prior to installing JTLS-GO 5.1.5.0 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.

### 1.2.6 Released Databases

This release includes the following sample unclassified databases:

- The scenario "repository51" serves as a repository of engineering level data. Although not useful as a scenario, it does follow all of the database requirements for a scenario, and should be loaded into your Oracle scenario table-space. With JTLS-GO 5.1.5.0, it is possible to access and copy records from the repository51 database into your own developed scenarios.

- The scenario “wespac51”, which is based on the Western Pacific theater of operations and is suitable for training and demonstrations.

### 1.3 INTERFACE COMPATIBILITY

#### 1.3.1 Support Software

JTLS-GO 5.1.5.0 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 7.6 (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.

- JTLS-GO 5.1 has been tested with the following versions of Linux 7:

Red Hat Linux 7.6 - This operating system license must be purchased, but it has been approved by the Defense Information Systems Agency (DISA) for use by U.S. Government Agencies.

Oracle Linux 7.6 - This operating system is free to download, use, and distribute, and is provided in a variety of installation and deployment methods. It has been approved by DISA for use by U.S. Government Agencies.

CentOS Linux 7.6 - A free version of Linux 7 that has **not** been approved by DISA for use by U.S. Government Agencies.

- There are no restrictions on the operating system used for client workstations, except that the operating system must have a Java-enabled web browser. JTLS-GO 5.1.5.0 has been tested on the following operating systems:

Red Hat Linux Enterprise Edition Version 7.6.

CentOS Linux Version 7.6.

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

- The JTLS-GO 5.1 series no longer uses Oracle Java, and has moved to the latest version of OpenJDK 8, which is OpenJDK 8 Version 242. We no longer deliver the Java Runtime Environment (JRE) within the JTLS-GO delivered software TAR files. Each user organization must obtain the latest version of the OpenJDK Red Hat Package Manager (RPM) and install the RPM on the servers used by JTLS-GO.

- JTLS-GO uses IcedTea to provide the OpenJDK web start capability that implements the web-enabled JTLS-GO functionality. The current version of JTLS-GO supports IcedTea version 1.8.3.

IcedTea version 1.7.1 does not support use of unsecured HTTP to access web-enabled JTLS-GO components for the DDS Client application. Users must configure their Glassfish instance, using the DCP program, to use HTTPS, by enabling the "Use SSL" column (this is enabled by default) when running with IcedTea version 1.7.1.

Red Hat Linux version 7.7 continues to distribute with IcedTea version 1.7.1. There are available RPM packages for a later version of IcedTea, so users who wish to install a later version of IcedTea must explicitly install IcedTea, including on their client workstations. IcedTea version 1.8.3 is required to use unsecure HTTP with JTLS-GO.

- JTLS-GO database tools require use of a certified Oracle database server and the full Oracle Client installation for runtime requirements. Additional installation details can be found in [Section 1.5.2](#) of this chapter.
- 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 Agency that can obtain source code and plans 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 needed only if you are a U.S. Government Agency that can obtain source code and plans 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 needed only if you are a U.S. Government Agency that can obtain source code and plans 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.

- JTLS-GO 5.1 implements SSH Tunneling between Apache and the services, and among the services. Rigorous testing should be done prior to use in any exercise, and particular attention should be paid to network performance under load.

### 1.3.2 JTLS-GO Information Assurance Compliance

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-GO, as software that executes on DoD systems, must comply to the mandates of the IA program, as well as requirements of 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 keeps the software up-to-date and properly installs all security patches. In previous versions of JTLS-GO, Java 8, as delivered by Oracle, fulfilled this mandate by implementing an expiration date for its software. After a certain date, all Java-related programs



stopped working, whether connected to an open network or not. All JTLS-GO releases were closely linked to the Java expiration date.

Information available at the time of this release indicates that OpenJDK will not implement an expiration date. In order to fulfill the DoD IA requirement, Maintenance releases of JTLS-GO will be released as new versions of OpenJDK are released. In a future version of JTLS-GO, all delivered Java programs will check that the latest version of Java is being used. Each local installation will be able to override the warning generated when the program detects that the latest version is not properly installed.

JTLS-GO has completed the IA program mandates and the JTLS-GO 5.1 series of releases has been granted an Authority To Operate (ATO) on DoD systems. Contact the U.S. Government Program Manager, Mr. Don Weter ([donald.e.weter.civ@mail.mil](mailto:donald.e.weter.civ@mail.mil)), for additional information.

### 1.3.3 JTLS-GO High Level Architecture Compliance

The JTLS-GO 5.1.5.0 release is fully High Level Architecture (HLA) compliant, and includes all the programs required to run JTLS-GO in an HLA mode. JTLS-GO uses the 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 GESI wargaming system have been accomplished. Future plans include expanding the capabilities of the JTLS-GO-GESI federation, called "GlobalSim".

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. 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, aircraft, and ground forces.

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.

All JTLS-GO software needed to run GlobalSim is included in this delivery.

The HLA RTI (Run Time Infrastructure) executive program recommended for use with this release is Pitch pRTI Evolved 4.4.2.0. However, this program is not included in the JTLS-GO 5.1.5.0 delivery. Users may obtain a full installation package of the RTI executive program from Pitch Corporation ([www.pitchtechnologies.com](http://www.pitchtechnologies.com)). 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

Significant database structure differences exist between the JTLS-GO 5.1 series and the previous JTLS-GO 5.0 series database structure.

To upgrade your JTLS-GO 5.0 scenario to JTLS-GO 5.1 compatibility, see instructions listed in Chapter 3.1 of the *JTLS-GO DDS User Guide*.

### 1.4.1 JTLS-GO Using Legacy Default Symbol Set

If a user organization is still using the pre-JTLS-GO 5.0.0.0 legacy default symbol set, prior to unloading your JTLS-GO 5.1.0.0 formatted data from your Oracle database server into the JTLS-GO 5.1.0.0 scenario American Standard Code for Information Interchange (ASCII) text files, you must execute the JSYMS program using the procedure outlined in Appendix B.11 of the *JTLS-GO DDS User Guide*. This procedure will reorganize the structure of the .gs and .scf symbols-related files.

### 1.4.2 JTLS-GO Using New Default Symbol Set

You should not make any modifications to the Default Symbol Set delivered with JTLS-GO 5.1.5.0, but end user organizations are free to use the Default Symbol Set in their scenarios and alter the scenario symbol set to meet specific organizational needs.

### 1.4.3 Standard Repository Changes

The JTLS-GO 5.1 series of JTLS-GO is the first series in which R&A is delivering an unclassified data repository called “repository51”. In future Major releases of JTLS-GO, [APPENDIX B](#) will provide a summary of the data structure changes made to the data repository. No data structure changes have been made in this Maintenance release; therefore, [APPENDIX B](#) is empty. Refer to Appendix B in the *JTLS-GO 5.1.0.0 Version Description Document*, included with this release, for data structure changes made for the JTLS-GO 5.1 series.

## 1.5 INSTALLATION

### 1.5.1 Installation Instructions

The *JTLS-GO Installation Manual*, a PDF file available for direct download, is part of this JTLS-GO delivery. It provides detailed instructions for installing a new version of JTLS-GO.

### 1.5.2 Oracle Compatibility and Installation

A full Oracle Client (not Instant Client) installation 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 certain Oracle utilities, such as sqlldr, imp, exp, and tnsping, 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 installation of the Oracle Client to a different directory that mounts to JTLS-GO (a simple NFS mount will suffice). 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-GO .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.

Between the free Express Editions (XE) of Oracle, the 18c XE has a large footprint and much more complex database architecture compared to the 11gR2 XE version. For test environments and scenario building purposes, or for collecting AAR data for a short period of time, installation and setup of the 11gR2 XE version would be much simpler.

The DDS application utilizes the Oracle GlassFish J2EE server, which, like the JTLS-GO WHIP Apache server, is delivered with JTLS-GO 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 Special Installation Instructions

This section describes special instructions that should be followed because of errors corrected in this version and previous bug releases for the JTLS-GO 5.1 series.

#### 1.5.3.1 OTH-Gold Ship Types

JTLS-GO 5.1.0.0 was delivered with an old list of OTH-Gold ship types. This problem was corrected in JTLS-GO 5.1.1.0. To properly implement this solution, users must execute the following additional procedures for each of their JTLS-GO Version 5.1.0.0 scenarios loaded in Oracle.

Users who have already executed this procedure for their scenarios after installing JTLS-GO Version 5.1.1.0, 5.1.2.0, 5.1.3.0, or 5.1.4.0, do not need to execute the provided procedure again.

New OTH-Gold ship types were added to JTLS-GO, due to STR JTLS-2019-14238.

1. Execute the following command:

```
cd $JTLSHOME/script/dds/version5.1/scripts/
```

2. Execute the following command:

```
sqlplus yourScenario/OraclePassword @update_oth_gold_types.sql
```

3. Verify the related execution listing file under the \$JDATA/scenario/ directory for errors.
4. Unload your scenario using the JTLS-GO Menu, Options 1 -> 1 -> 4
5. Verify the .srw ascii file for your scenario.

To fix the problem renaming Force Sides, due to STR JTLS-2019-14267:

6. Reload the database.

### 1.5.3.2 Generation Of Combat System Summary Files

STR JTLS-2019-14518 Move Combat System Summary Files, delivered as part of JTLS-GO 5.1.4.0, solved a problem overwriting the summary Combat System Character Separated Value (.csv) files during Batch Runs. This STR was solved by moving the location of the .csv files from the game/<scenario\_name>/location directory to a sub-directory under each checkpoint.

For every active game scenario, the following procedure options must be executed to establish the directory structure needed by the STR solution. Users who have already executed this procedure for their scenarios after installing JTLS-GO 5.1.4.0, do not need to execute one of the selected options again:

- Option 1: Rerun the Setup Procedure for each active scenario.
- Option 2: Hand-create a cbtsys\_summary sub-directory in the game/<scenario\_name> directory. This can be accomplished using the following steps from a command terminal for each existing game that has already been set up and prepared for execution:
  - a. Enter the command: **game** - this puts the terminal in the \$JTLSHOME/game directory.
  - b. Enter the command: **cd <scenario\_name>** (for example, **cd wespac51**) - this puts the terminal in the game/<scenario\_name> directory.

- c. Enter the command: `mkdir cbtsys_summary`. This creates the necessary new directory.

### 1.5.3.3 Aircraft Type Foreign Key Definitions

STR JTLS-2019-14541 Referenced Aircraft Class Can Be Deleted, delivered as part of JTLS-GO 5.1.5.0, solved a problem that allowed deletion of aircraft types that were referenced by units. The STR was solved by removing a rule in the database that allowed the user to set the aircraft type-related foreign keys of the unit tables to NULL.

After loading JTLS-GO 5.1.5.0, the user must unload and then reload their JTLS-GO 5.1 scenarios to have the new foreign key definitions in their database schemas.

## 2.0 ENGINEERING CHANGE PROPOSALS

The following model capabilities were added to JTLS-GO 5.1.5.0 as a result of implementing authorized Engineering Change Proposals (ECPs).

ECPs JTLS-2020-14625, JTLS-2020-14627, JTLS-2020-14628, and JTLS-2020-14630 were built on an emergency basis to meet NATO exercise requirements. The documentation for these ECPs will be completed as part of the JTLS-GO 6.0 series, due to be released in the Summer of 2020. These ECP are technically not closed until all documentation requirements have been fulfilled. These ECPs will also be included in the JTLS-GO 6.0 Version Description Document.

### 2.1 JTLS-2019-14602 Resupply Interactions For GlobalSim Federation

#### Summary of Model Change Request

For the GlobalSim Federation, a number of HLA Interactions were implemented. These interactions allow the GESI simulation to request supplies from and to dropoff supplies to a JTLS-owned supply storage area (SSA).

#### Design Summary

The interactions used in the GlobalSim Federation are the same as those used by the NETN Federation. This consistency in the interaction details was done to allow external federates to easily join the GlobalSim Federation for purposes of resupply. Changes were made to the JTLS HLA Interface Program and to the CEP in support of this capability.

### 2.2 JTLS-2019-14615 Reformat Spreadsheet Program MGRS Conversion

#### Summary of Model Change Request

A user had a space-separated spreadsheet which contained Military Grid Reference System (MGRS) Coordinates. The Reformat Spreadsheet Program (RSP) did not have the capability to convert MGRS Coordinates to Latitude and Longitude Coordinates.

#### Design Summary

If a user has a space-separated spreadsheet with MGRS Coordinates, the RSP will output Latitude and Longitude coordinates in the processed order file. Existing SIMSCRIPT code from the MGRS directory was added to the RSP.

### 2.3 JTLS-2020-14625 Quick Order To Change Mission Speed/Altitude

#### Summary of Model Change Request

The Change Mission Parameter panel was too long, and typically a user only needed to be able to change the speed or altitude of an air mission. A quick order to change speed of an air mission and the altitude of an air mission should be provided.

#### Design Summary

Two new quick orders were created: Change Mission Speed and Change Mission Altitude. These quick orders are currently located on the Air menu. A user can alter their menu panels to make these new quick orders available on any desired menu panel.

### 2.4 JTLS-2020-14627 IMT Column Size Adjustment

#### Summary of Model Change Request

The IMT panel definition files contain information on the default size of each IMT column, based on the maximum size of the data contained in the column. For example, air mission names can be 13 characters long, so the column width is 13 characters long. Some users build their orders so that the mission name is no more than 9 characters long. There is no way to reduce the size of the column, so extra space causes IMT panels to take up too much room on the WHIP screen.

The user should be able to change the size of a field as desired.

#### Design Summary

The IMT panels have been changed to allow users to expand or reduce the size of each IMT column as desired.

### 2.5 JTLS-2020-14628 Add Concept Of COMAO ID

#### Summary of Model Change Request

The North Atlantic Treaty Organization (NATO) Air Component has the concept of Combined Air Operations (COMAO) groups. COMAOs are made up of one or more individual missions and one or more Air Mission Packages.

The Air Component's reporting requirements are based on the COMAO to which these missions and Air Mission Packages belong. Other than reporting requirements, a COMAO has no functional implications. To help the response cell with its necessary reporting requirements, NATO requested that the COMAO identifier to which a mission belongs can be shown on the IMT.

## Design Summary

Each air order was altered to include a field for the COMAO ID to which the mission belongs. This optional field is a text string and appears on each mission's ATO tab. If the COMAO ID is filled in on the order, the mission is assigned the assigned COMAO ID, which can be viewed on the IMT screen. Users can then sort on this field as desired.

The Air Tasking Order (ATO) Parser for NATO's Integrated Command and Control (ICC) system was altered to retrieve the mission's COMAO ID from the ICC-generated ATO file and automatically fill the new field in the ATO Translator-generated order files.

All existing user air orders are still valid.

### 2.6 JTLS-2020-14630 ATO Lite Game

## Summary of Model Change Request

Users had difficulty achieving enough speed to check the next day's ATO during a test game, before submitting the generated orders to the production game. The ATO checks watch to make sure that missions take off and fly expected flight paths. Any issues can then be fixed prior to entering the orders in the game that feed the real-world C4I systems.

The Controller should have the ability to turn off non-air-related events to speed the process of checking an ATO in a test game.

## Design Summary

A flag that can only be set by Technical Control personnel has been introduced into the system. If set, all non-air-related events are skipped, resulting in only the air-related events executing.

All users should understand that this capability is designed for testing purposes, and should **never** be used in a production game.





### 3.0 SOFTWARE TROUBLE REPORTS

This chapter summarizes Software Trouble Reports (STRs), which 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; however, because of the model enhancements included with JTLS-GO 5.1.0.0, no STRs identified in prior JTLS-GO series are considered valid. Uncorrected errors identified for JTLS-GO 5.1.5.0 are documented in this chapter and should be corrected in future maintenance releases.

#### 3.1 JTLS-2019-14537 JSAT Not Provided Primary WHIP By ICP

*One of the items that the JTLS-GO Satellite Service (JSAT) requires to send orders to the Combat Events Program (CEP) is the name of the Primary Controller WHIP. It is the responsibility of the Interface Configuration Program (ICP) to provide this information to the JSAT via an initial configuration file. The tag that should hold the Primary Controller WHIP name was empty, leaving the JSAT without required information to do its job.*

Previous versions of JTLS-GO had a toggle button within the ICP to indicate if a WHIP was a primary WHIP for its Force Side. This was replaced with a Command Authority field to indicate if the WHIP was a Force Side Primary, had Shared Authority, or had No Authority. The ICP was still attempting to find the Primary Controller WHIP via the toggle button setting, which consequently did not return a value. The code was modified to check the Command Authority field instead.

#### 3.2 JTLS-2019-14539 IIR Report Does Not Always List Damaged JDPI

*If a target with associated JDPIs have been detected and the target is part of a BE Facility, the target damage is reported as part of the BE Facility summary and the associated damaged JDPIs are not included in the report.*

The model determined that the damage target was already reported and so it did not need to check for damaged JDPIs associated with the target. The code was altered slightly. The check for damaged JDPIs was placed before the check used to determine if a target had already been included in the report.

#### 3.3 JTLS-2019-14540 JSAT Classification With Spaces

*The JSAT was never updated to handle the JTLS-GO 5.1 database format, which permits the scenario classification to contain whitespace characters. If the scenario classification contains whitespaces, the read of the data was thrown off and corrupted the start time for the scenario, impairing its ability to produce proper time-stamped orders.*

The code which reads the scenario classification was modified to handle whitespaces in accordance with the format downloaded from the DDS.

### 3.4 JTLS-2019-14541 Referenced Aircraft Class Can Be Deleted

*An aircraft class can be deleted even it was referenced by a ground unit. This was true for all unit types except squadrons.*

It was possible to delete an aircraft class by setting the related field of the unit record to NULL, because of the ON DELETE SET NULL rule for the aircraft type-related foreign keys of the unit tables (except squadrons). This rule has been removed, automatically blocking deletion of an aircraft class if it was referenced by the unit tables.

Users **must** unload and then reload their JTLS-GO 5.1 scenarios to have the new foreign key definitions in their database schemas.

### 3.5 JTLS-2019-14542 JODA Errors When Package TOT Changed

*If the user changes the Time On Target time for an Air Mission Package, the JODA throws errors because the mission's tasks are being recreated.*

The code to update the tasks had an incorrect parameter passed, called mandatory create. The programmer misinterpreted the meaning of the parameter. The mandatory create flag told the model it must create a new task in the JODA. This is not what was supposed to happen. The programmer wanted a mandatory update, which is done automatically when the task changes.

The mandatory create flag was changed from YES to NO and the tasks are now not being re-created.

### 3.6 JTLS-2019-14543 IIR Detected Ships Show Improper Names

*A ship detected while collecting intelligence information on a naval base was not properly reported in the IIR message.*

A simple format error in the data that was supposed to be displayed caused the problem. The format error was corrected.

### 3.7 JTLS-2019-14544 Invalid Ship Category Sent Via OTHGold

*An OTHGold message was generated for an opposing side naval unit that had the OTHGold Category field filled with an invalid value.*

The user did not have the OTHGold Ship Category filled in all of the Ship Real World data records. Once the data was filled, all work as designed, but JTLS-GO should not allow an invalid field to be sent to a C4I system.

The model was changed to default the Ship Real World OTHGold Category value to UNK or Unknown. If using a link to an OTHGold system, database personnel should ensure that good

OTHGold data is filled in the database, resulting in the maximum benefit of the C4I system to the exercise audience.

### 3.8 JTLS-2019-14546 CEP Receives JSAT Order And Enters Debug

*After receiving a National Asset Pass order from the JSAT, the CEP entered debug mode due to an error.*

There was a problem reading the satellite's route points in the JSAT code. There are two functions, GenerateNAP() and QuickFind(), that calculate the number of route points for the satellite, but each used a different formula for the calculations. The number calculated by GenerateNAP() went into the ORBIT\_ROUTE field of the order. This declared the number of route points in the order. The number calculated by QuickFind() was used to generate the route points that actually were included in the order.

Testing revealed that these numbers were equal most of the time, but there were cases where the numbers would differ by 1, and the number calculated by GenerateNAP() was always greater. This meant, for example, if GenerateNap() calculated 61, but QuickField() calculated 60 route points, the CEP read order routines expected to read 61 route points in the order while only 60 route points were actually entered to the order, causing the error.

Since the number of route points that were actually written out were based on QuickFind()'s formula, the formula used by GenerateNAP() was replaced with the formula used by QuickFind().

### 3.9 JTLS-2019-14547 Statistics File Takes Up Too Much Memory

*The statistics file was taking up 8 GB of memory, but most of the data in this file has values of zero.*

The data in the statistics arrays are printed only if they contain values which are greater than 0.

### 3.10 JTLS-2019-14548 WHIP/DDSC/JOBE Map Find Result

*When a find on map is performed in the WHIP, DDSC or JOBE, there is a maximum limit of 100 results that prevents cluttering the map with too many results. This limit exists because users are allowed to perform wildcard searches, and they could produce an excessive amount of results. The limit check was performed on the full find result set and not as the results were compiled. This implementation did not provide the best performance.*

The check was re-implemented to limit the results as they are being compiled, and when the limit is reached, the search stops and the user is immediately notified to refine their search.

### 3.11 JTLS-2019-14549 Naval OTHGold Category Value Invalid

*When an other-side ship is detected, its Ship Unit Prototype (SUP) is set for the detecting side to UNKNOWN SUP if the detection level is LOCALIZATION or CLASSIFICATION. This UNKNOWN SUP was being initialized with an INVALID value for OTHGold Category.*

The UNKNOWN SUP is now initialized with a valid value of "UNK" for OTHGold Category.

### 3.12 JTLS-2019-14550 ELS Targets Not Moving With Owning Units

*Target entities were not moving with their owning units. The parent unit moved toward a new location, but the entities associated with the owned target did not move.*

The hierarchy of target elements within the ELS was modified to facilitate usage by external models, such as MUSE, NWARS, and GESI. Aggregate targets may have had multiple elements and two levels of parent objects within the ELS. The extra level was not necessary and added to more complex interactions with external programs. A single level of parent objects was implemented. Additional changes were required to resolve the remaining issue regarding the movement of owned targets.

### 3.13 JTLS-2019-14551 Resurrected Entities Alive But On-Fire

*A player destroyed aircraft entities in a squadron, and the controller decided to restore unit strength using the Controller Change Unit order. The unit was restored to full strength and all entities were marked as undamaged in the ELS. The appearance attribute for the aircraft entities continued to show them as "On-Fire".*

In normal ELS processing, combat systems were always set to appear "on-fire" when the system was destroyed. This "on-fire" appearance lasts for a specified period of time after they are destroyed. The appearance attribute was never properly re-set to "Normal" when the entity was resurrected. Code changes were made to ensure that "On-Fire" entities were restored to a "Normal" appearance when the unit strength was restored to 100 percent.

### 3.14 JTLS-2019-14552 ELS Crash Inconsistent Assets In CEP

*A truck convoy was created in the aggregate model which had an undetermined number of assets. The convoy was then destroyed in the model and on the JODA. The ELS had previously created a disaggregated convoy with no entities. The ELS crashed when it attempted to access the first truck entity in the convoy.*

In the ELS code which destroys a convoy, protection code was added to prevent the crash. The true source of this problem was corrected in the CEP code.

### 3.15 JTLS-2019-14553 DDS EEV Unit Report Migration

*When the DDS Migration tool migrated an External Event Unit Report, the unit list in the child table was empty, even though the number of units value is greater than 0.*

There was a misspecification in the migration post-processing rule specified for the External Event Unit Report table. The mistake was corrected in the dmrules.xml.

### 3.16 JTLS-2019-14554 Warning 1635 Options Nonfunctional

*Corrective actions for Warning 1635 did not open the sensor table and failed to set the LINK16 flag to NO.*

In both cases, the parameter for the sensor subcategory field was incorrect. Both corrections were modified to use the correct parameter.

### 3.17 JTLS-2019-14555 Warning 1502 Did Not Open Table

*One corrective action for Warning 1502 did not open the unit table.*

The corrective action needed the text for the unit type in order to determine the proper table to open, which was not provided. The source code and SVP-Template file were both updated.

### 3.18 JTLS-2019-14556 Warning 1255 Correction Does Not Work

*The corrective action for Warning 1255 to open the combat system table to modify the combat system non-combat fuel modifier does not work.*

The correction action for Warning 1255 to modify the non-combat fuel usage parameter was removed as a corrective action. The routine which generates the warning is computing total fuel needed values. It is not feasible to determine which exact combat system would need the Non-combat Fuel Usage modified.

### 3.19 JTLS-2019-14557 Wrong Table Name In Properties Key

*A typo was found in the dds\_drm\_help.properties file.*

csp\_up.up\_value was written as ccp\_up.up\_value in the properties for the CSP\_UP table. The typo was corrected.

### 3.20 JTLS-2019-14558 TBMCS And JTLS-GO ICAOs Will Not Map

*TBMCS ICAOs could not map to JTLS-GO ICAOs.*

It was not possible to map TBMCS airbase ICAO codes to JTLS-GO ICAO codes, because the two systems did not use the same country code standard. TBMCS used FIPS standard, which is a

two-character format, while the JTLS-GO database was setup using the ISO-3166-1-TRIGRAPH standard, which is a three-character format. The ATOT was changed to convert the database Country Code standard to the FIPS standard when working with a TBMCS Air Tasking Order.

### 3.21 JTLS-2019-14559 Warning 1636 Link 16 Feed Name

*Warning 1636 should show the Link 16 feed name.*

The Link 16 feed name was added for both the owning and associated units referenced in the text.

### 3.22 JTLS-2019-14560 OTHGold File Resource Not Closed On Failure

*A file resource was not being closed if it failed to be created or written to.*

The file resource is now closed in all cases.

### 3.23 JTLS-2019-14561 HIP Missing Shipment Type For Supplies

*The JTLS-GO HLA Interface Program (HIP) needed to know the shipment type for supplies when running with the GlobalSim Federation. All of the supply names and their DIS enumerations were provided, but the shipment type was absent from the initialization data.*

The routines to provide initialization data for the HIP were modified to include the required data.

### 3.24 JTLS-2019-14562 Consolidate Font Settings For Web Login Page

*The table of the JTLS-GO web login page had inconsistent font formatting and styling, resulting in ad hoc links differing from the static links for the WHIP, TRIPP, and other non-dynamic functions.*

The font styling of the web login page has been consolidated at the table level instead of at the table cell level.

### 3.25 JTLS-2019-14563 TBMCS JTOI Fixes

*The initialization file produced by the JTLS-GO Transaction Operational Interface (JTOI) failed to print the record terminator for some operating locations without runways.*

The JTOI now provides a record terminator for operating locations with null runways.

### 3.26 JTLS-2019-14564 ATOT Foreign Squadrons Linked To Tasking Units

*When a multi-side ATO is translated by the ATO Translator (ATOT), it sometimes links foreign squadrons to the ATO Tasking units, overwriting the user linking.*

The ATOT automatically links known squadrons with the tasking units listed in the ATO. In the case of a multi-side ATO, the ATO may contain squadrons from two or more Sides. If a recognized

squadron is listed in the ATO and the ATOT already has the squadron's aircraft type linked, the ATOT would link the squadron to the tasking unit automatically, ignoring the fact that the user indicated the squadron belonged to a foreign Side. The user's proper side specification was being overwritten by the ATOT's automatic linking algorithm.

This has been fixed. The automatic linking will now exclude foreign squadrons. If a user links a squadron to a foreign Side, it will not be overwritten.

### 3.27 JTLS-2019-14565 JSAT Issues With Large Number Of Satellites

*The JSAT GUI required that the user click "Ok" after each satellite is processed. This interaction with the dialog box caused significant slowdowns when generating orders or database files for large numbers of satellites. The associated order processing in the CEP also produced errors because the orders created by JSAT did not include a Reference field. Finally, the names for satellites were inconsistent in length. The order expected a 15 character name, the TLE files allowed for 24 character names, and the external events in the DDS were limited to 15 characters.*

Code changes were made in the JSAT and the JSAT GUI to fix these issues. The required dialog box and "Ok" button were removed. A box appeared when an error occurred, but in normal operations an entry is created in the console and no other interaction is required from the user. The orders were changed to include a reference field.

For the inconsistent length of satellite names, the order field was changed to allow for 24 characters, but the JTLS-GO 5.1 database is locked, so the external event name was not changed.

### 3.28 JTLS-2019-14566 SIP Assumes Ground Unit Owns Targets

*While performing checks on a variety of targets, the SIP was not printing the prototype name of the target-owning unit whenever there is an error found with the target's range and the owner is a Naval unit.*

The SIP routine for checking the consistency of target ranges assumed the target's owner was a ground type unit and never a Naval unit. The routine logic has been changed to first determine what type of unit owns the target, then the name of the unit's prototype is obtained for later error reporting.

### 3.29 JTLS-2019-14567 LOITERING\_MUNITION Load Weapon Type

*The model crashed after sending a Set Load Assignment order to change the weapon load for an aircraft labeled as "LOITERING\_MUNITION".*

The weapon load that was to be assigned to the aircraft did not contain an air-to-ground weapon type. A check has been added to make sure the weapon load to be assigned to the aircraft



contains an air-to-ground weapon type, in order to satisfy the rules for a legal aircraft load for a "LOITERING\_MUNITION."

### 3.30 JTLS-2019-14568 ATOT Alert Package Empty Location Task

*When the ATOT is presented with an set of ATO missions in a package on alert, it assumes the missions have a task location. It generates the orders for the package and the missions to attack an empty location. This causes an error when checked by the Order Group Editor in the WHIP.*

Two routines that generate the orders were modified to check the existence of a non-empty location in each mission of a package.

### 3.31 JTLS-2019-14569 ATOT Missing Error Report Details

*The ATOT does not include the squadron name for a mission found to be departing a location that is not consistent with the squadron location in the Error 455 report. Also, the ATOT does not include the type of aircraft for a mission having an improper load for its tasking in the Error 506 report.*

Both pieces of mission data are now included in the respective reports.

### 3.32 JTLS-2019-14571 ATOT EC Mission Transit Point Locations

*Whenever an Electronic Combat mission is included in the ATO and has transit points, the ATOT does not fill in the transit point locations of the order utility.*

The ATOT code used the incorrect index number in the order utility that is used to identify transit data. Although the point locations were included in the ATOT-produced order utility, the utility was being rendered by the WHIP without the point locations. This has been fixed.

### 3.33 JTLS-2019-14572 ATOT Transit Taskings And Mission Start Time

*The ATOT ignores the start time of transit points whenever the transit points come first in the list of tasking for ATO missions. So, a mission that starts its tasking with a transit point does not get a mission start time that considers the start at this first point.*

The ATOT code searches for the first tasking in the list of tasks for the mission. The start time for the mission is obtained from the start of this first tasking. However, the code ignored transit type tasks. This has been fixed.

### 3.34 JTLS-2019-14573 TACELINT Message Field Formats

*TACELINT message fields for radio frequency, pulse rate, prefire pulse rate, and pulse duration required formatting changes.*

The TACELINT Message Service has been modified to format the radio frequency value to a precision of one decimal place. This value is calculated at run-time by the CEP. The pulse rate, prefire pulse rate, and pulse duration fields are text fields in the database, which previously were stated to require a unit of measure at the end of the value, such as "MS" for milliseconds. The new system no longer requires a unit of measure, so these text fields must be updated manually in the database.

### 3.35 JTLS-2019-14574 ELS Memory Leaks And Variable Inconsistencies

*The ELS was found to have some memory leaks and places where an integer was being passed into a routine, which expected an unsigned integer.*

Code corrections were made to eliminate all instances where a potential memory leak was found. Similarly, improvements were made to use consistent data types in places where an integer was used and an unsigned integer was expected.

### 3.36 JTLS-2019-14575 Issues Creating New Units and HRUs in the ELS

*When new units are created in the CEP, the ELS must also properly represent the associated entities. When customized HRU were created, the ELS was not assigning the correct numbers of combat systems to the disaggregated objects. The same issue occurred when new units were created through detachments.*

When entities were assigned to new units (both HRU and Detachments), the ELS created the disaggregated objects using templates. Most templates were based on the combat system counts in the unit prototypes (TUPs, SUPs, and HUPs). However, for customized HRU and for many detachments, the prototype-based templates were not suitable for the assigned combat systems. New templates needed to be built on-demand to fulfill the requirements of these unique units.

The ELS was modified to check if a customized template was needed, and then utilize those templates to create new units (HRU and Detachments).

### 3.37 JTLS-2019-14576 JOI Memory Leaks And Unreleased Resources

*The OTHGold Message Service, TACELINT Message Service, and TBMCS JTOI have a portion of shared code that caused a memory leak. The OTHGold Message Service also failed to release a file resource if the resource failed to be opened.*

The OTHGold Message Service, TACELINT Message Service, and TBMCS JTOI have been fixed to properly free allocated memory in the shared code, and close file resources in the event of an error.

### 3.38 JTLS-2019-14577 JOI/JTOI Inappropriate Times Data Type

*The TACELINT Message Service, Link16 Message Service, OTHGold Message Service, ICC JTOI, and NECCCIS JTOI were storing times in unsigned data types. However, the operating system and C++ use a signed data type for time storage.*

The TACELINT Message Service, Link16 Message Service, OTHGold Message Service, ICC JTOI, and NECCCIS JTOI have been updated to use a data type that corresponds directly to that used and provided by the operating system and C++.

### 3.39 JTLS-2019-14578 Non-Configuration Managed File Will Not Save

*The non-configuration managed file does not save when doing a DDS download.*

The file was being saved, but it was being moved to the backup directory. The script was changed to move a copy of it into the backup directory, thus maintaining the original in the primary scenario directory.

### 3.40 JTLS-2019-14579 i18n Injection Vulnerability

*Fortify scans revealed that the i18n tool is vulnerable to XML External Entity (XXE) attack.*

The javax.xml.parsers.DocumentBuilderFactory and the javax.xml.transform.TransformerFactory were not properly configured to protect against XXE. To correct this situation, the "secure-processing" feature of the TransformerFactory was enabled, and the "external-general-entities" and "external-parameter-entities" features of the DocumentBuilderFactory were disabled.

### 3.41 JTLS-2019-14580 OPAREA Incorrect Side Visibility Array

*The OPAREA side visibility array was different that what was specified in the order.*

The array was incorrectly reserved resulting in the error. The problem was solved.

### 3.42 JTLS-2019-14581 JSAT Memory Leaks And Type Mismatch Values

*Fortify scans identified two JSAT functions that allocated memory but failed to release it. The scans also identified a function that assigned the value of a signed character to an unsigned character. Finally, there was another function that called the C-Library function fgetc(). Function fgetc() returns an integer, but the returned value was assigned to a character type.*

Each of the programming issues, which had no visible effect to the user, was corrected.

### 3.43 JTLS-2019-14582 HIP Crashes When HRU Destroyed

*The HIP crashes following bombardment of an HRU and the HRU is WIPED OUT.*

The HIP was not properly handling the deletes of the destroyed HRU supplies. The HIP was corrected to properly delete the supplies that formerly belonged to the destroyed HRU.

#### 3.44 JTLS-2019-14583 Split Mission Jumps To Equator

*A Recce mission with two aircraft was running away from an enemy interceptor. A Split Air Mission order was sent to the Recce mission, which created a second Recce mission. The new Recce mission jumped to the equator and began heading home.*

The logic that split the mission failed to handle the situation where the parent mission was running away from an interceptor. The posture of the parent mission (RUNNING.AWAY) and the parent's pending task list were copied to the new mission correctly. After the split, only the parent mission was actually being intercepted because an interceptor cannot intercept two missions simultaneously (by design).

When the next Move Along Route Air event occurred for the new mission, the logic entered the Running Away routine to compute the new mission's next location. However, because the new mission was not actually being intercepted (which is correct), the computation was bypassed and the new mission jumped to the equator.

To fix this problem, new logic was added to stop the new mission from running away and execute the next pending task that was copied from the parent mission.

#### 3.45 JTLS-2019-14584 Federation Data SVP Corrections

*The scenario verification process includes the capability to immediately correct problems in the scenario data. No such corrections were available for the federation-related errors and warnings.*

No code existed to correct issues found in the data used in federations. The structure existed for this purpose, but no instructions were written to support the process. Several corrections were added the existing errors and warnings to facilitate the building of scenario databases to be used in federations.

#### 3.46 JTLS-2019-14585 Crash Left Over Rendezvous Point

*If a package is delayed, it still has a Leave Rendezvous Point Event scheduled. If the package is canceled prior to the event executing, the event is not also canceled. This eventually leads to a crash when the Leave Rendezvous Point event is executed.*

The Destroy Air Mission Package logic did not cancel the outstanding Leave Rendezvous Point event under the condition that the package was in a delay state. The logic now does so.

### 3.47 JTLS-2019-14586 Stationary Target Not Placed On Game Board

*A stationary target was owned by a ground unit in the game. The unit and its owned targets were the only objects in the grid. The unit was magic moved to a new grid location. As expected, the stationary target remained at its fixed location as seen on the map on the WHIP. A user fired upon this target with weapons, but it resulted in no damage. A terrain report for the location of this target listed zero objects in the grid.*

When the unit was magic moved to a new grid, the stationary target was incorrectly removed from its previous location on the game board. The unit was correctly relocated to a new position, but the stationary target was never re-associated with its grid location. The latitude and longitude for the target were correct, as shown on the map in the WHIP. However, internally in the CEP, the target was not properly placed in a grid. A code change was made to correct this issue.

### 3.48 JTLS-2019-14587 Naval Unit Placed On Land Grid SVP Error

*Using the DDS, a user moved a naval unit to a land grid that was not dual capable. After running the SVP from the SVPR, two errors appear: Error 406 reports that a naval unit has not been placed in a grid that is not a small island, ocean, or dual capable grid, hence it was placed in a land grid with no depth. Error 725 reports the terrain grid is not deep enough for the naval unit.*

Only Error 406 is generated when a naval unit is placed on a land grid. Error 725 should only be generated if a naval unit is in a legal grid and the depth needs to be changed.

### 3.49 JTLS-2019-14588 Adding Link 16 Target To Game Crash

*The user attempted to add a new Link 16-capable Sensor Target to the game. The model crashed in the middle of this multi-step procedure.*

To add the Link 16 target to the game, several steps are needed.

1. The target must be created.
2. The target must be added to the desired Link 16 network.
3. If desired, the target must be given its own JU Number.

The crash occurred because the user accomplished Step 3 prior to Step 2. The target detected an air mission and the model saw that it was both Link 16-capable and had a JU Number ,so the code assumed it must be on a Link 16 Network.

Two things were done to ensure this crash would not happen again:

1. The code no longer has the assumption that just because a target is Link 16-capable and has a JU Number that it is on a network. If it is not on a network, the target is not allowed to make a Link 16 detection.

2. The code was changed so if the Controller attempts to give a target a JU Number and the target is not on a Link 16 network, the order is refused.

At the same time, the code was improved to make sure that when a target is given a new JU Number by a Controller, the new number does not cause an overlap with other JU numbers on the same network.

### 3.50 JTLS-2019-14590 ATOT Auto-Link Perceived Foreign ICAOs

*The ATOT does not include foreign bases for the automatic linking of ICAOs during a translation. Any of the ICAOs discovered by the ATOT from the ATO that are foreign are left unlinked.*

The ATOT logic was excluding any ICAO from a perceived foreign side during the automatic linking phase of the translation. This has been changed to allow foreign bases, as long as the translation side knows about the foreign base's ICAO.

### 3.51 JTLS-2019-14591 JOI OTHGold Reset Comments To Empty

*The OTHGold Message Service allows users to set a Comment in the Remarks ("RMKS") field of each message header. By default, the Comments field is empty and is set to "NONE." However, when setting the Comments field back to an empty value, an error occurred.*

The OTHGold Message Service has been modified to accept a value of "NONE" in the Comments field when set by the user.

### 3.52 JTLS-2019-14592 ATO Mission Type Information

*Mission types BCAP (Boat Cap) and DSUPR (Defensive Suppression) are not defined as mission types recognized by JTLS-GO. Additionally, the primary target type for mission type ASW is SHIP, but should be SUB.*

Mission types BCAP and DSUPR were added to the ato\_constants.xml file.

The primary target type for an ASW mission was changed to SUB.

### 3.53 JTLS-2019-14593 DSP Incorrect Output Directory

*The DSP program is writing the DSA list file in the primary game directory, when it should be writing to the orders directory.*

The routine gendsaord.sim was modified to output the DSA list file to the proper directory.

### 3.54 JTLS-2019-14595 ATOT Not Writing Orbit Path

*Whenever the ATOT creates a mission order for an ATO mission that includes an orbiting path the ATOT writes the order containing an orbit polygon.*

The ATOT code assumed that the list of points associated with the orbit task for the mission was a polygon if the number of points was greater than two. This has been changed. The code now checks to see if the list of points possibly corresponds with an Air Space defined in the ACO data. If so, the ATOT then checks to see if the list of points represent a path or a polygon while writing the tasking portion of the mission order.

### 3.55 JTLS-2019-14596 Link 16 Capable SAM Vs Controller Damage

*The Controller created an Assess Weapon Damage event specifying an air-delivered cruise missile against a naval unit that had Link 16 capable SAM/AAA point defense. Since the Controller action invokes an immediate point defense and engagement of the ship, a full-blown cruise missile was not created in the model. However, when the SAM/AAA point defense targets performed the point defense calculation and reported the engagement to the HLA Interface Program (HIP), which is done even when the HIP is not connected, it expected a full cruise missile object and attempted to access the perception array of the missile, resulting in a crash.*

If a Controller Assess Weapon damage involves a missile we now create the full missile object, but we do not initialize it with the JODA thus preventing it from appearing on the WHIPs. This initialization creates all the necessary attributes, including the perception array, for point defense interactions.

### 3.56 JTLS-2019-14597 Associated Targets Not Usable In Federation

*In order to interact with a Unit or Target in a Federation, the object must have been owned by JTLS-GO or another federate. If a unit or target was owned by JTLS\_ONLY, then that object could only be played in JTLS-GO because it was never seen by an external model. The ELS never disaggregated objects which are marked as JTLS\_ONLY, so no entities were published for these objects.*

*For targets owned by a unit, the owning model was determined by that of the owning unit. However, in the case of targets with only an associated unit, the owning model was never determined from the unit. Targets with just an associated unit were always defaulted to JTLS\_ONLY. This meant that these targets were not usable in a Federation.*

The CEP code was changed to such that targets inherit the ownership attribute of (1) their owning unit, or (2) their associated unit. If the associated unit for a target had an owning federate of JTLS-GO, then the target would also get marked as JTLS-GO. This change allowed the targets associated with units to also be participants in a Federation.

### 3.57 JTLS-2019-14598 ATO Parser Incorrect Date

*Whenever the ATO includes a REPOSTN record having an arrival date that does not include a two-digit year, the Parser inserts the current year. This is a problem when the date of the ATO is for a previous year.*

The ATO Parser reads each field of the REPOSTN record and discovers the arrival date for the associated destination. However, it is possible that the date may not include the year. This is acceptable, but in this case the Parser would derive the year from the DTG of the ATO. The variable holding the date value for the mission was not getting the derived year, however. So it was defaulting to the current year, which may not be the year of the current ATO period.

This has been fixed. The proper logic for deriving the year is now in effect for this variable.

### 3.58 JTLS-2019-14599 JLOI Column Order Inconsistent

*The ordering of columns in the ICP for the JLOI was not consistent with the column ordering for the JOI configuration.*

The column ordering of the ICP JLOI configuration has been made consistent with that of the JOI.

### 3.59 JTLS-2019-14600 ATOT Extra Orbit Task

*A mission in the ATO that was to return to an airbase that was different from the departure airbase wound up having an extra empty orbit tasking in its order utility.*

The ATO Parser writes XML route data into a file that is read by the ATOT. Normally the Parser withholds the departure and arrival locations from the beginning and ending of a route because they are understood from the parent squadron. When the arrival (return) location is different from the departure location the Parser will include the arrival location in the route XML data. The ATOT creates order fields for the mission based on the locations provided by the Parser in the route data.

The location of the arrival location in the route was causing the ATOT to inject a false orbit tasking in the order. This has been changed to exclude the orbit task when the location of the potential orbit is actually the return (arrival) location.

### 3.60 JTLS-2019-14601 JLOI Send Email Using Sendmail Utility

*The JLOI was programmed to send email in a manual fashion, which made debugging, troubleshooting, and setup difficult.*

The JLOI has been modified to send email reports using the Linux command "sendmail". This make setup and troubleshooting easier to manage.

### 3.61 JTLS-2019-14603 ELS Naming Scheme For Target Elements

*In the aggregate model, targets sometimes had multiple elements associated with a single object. When the ELS disaggregated these objects into individual elements, each object was given a name which had leading number for the element. However, when a target was made up of just a single element, the ELS continued to add a zero to the name. This leading zero caused some confusion for external federate programs.*



The ELS code was modified to change the method in which target elements were named. For multi-element targets, the names continued to use a leading number to differentiate the elements. For single-element targets, the element name in the ELS was now the same as for the aggregate target. This change helped to clarify the entity-level target elements in federations.

### 3.62 JTLS-2019-14604 OTHGold Fixes

*OTHGold was reporting an incorrect Force Code for UNKNOWN Naval objects.*

OTHGold has been modified to report the correct Force Code for UNKNOWN Naval objects. A change was also made to report the Long Name of Naval objects in the RMKS record when these objects have been fully identified.

### 3.63 JTLS-2019-14605 Link 16 Autoselect Network Structure

*The Link 16 Message Service GUI module allows users to assign a Link 16 Network Structure to a running service. When only one option is available, most users would interpret that option as being selected and proceed through the dialog with nothing selected.*

The Link 16 Message Service module has been modified to automatically select the first item in the Network Structure list. This gives a visual indication to the user that something is selected. In the instances where only one item exists, the user can proceed without needing to select the only option.

### 3.64 JTLS-2019-14606 ATOT Attack Package Mission Missing Location

*The ATOT produces mission orders for attack package missions, but, if the mission is attacking an unknown location or the location is not provided, the ATOT writes 0.0, 0.0 for the location. This location is valid but does not notify the user about the location problem. When the order is sent, the mission attempts to fly to this location and disrupts the other package missions.*

The ATOT was modified to include logic for checking the inclusion of an unknown location for a mission whenever the mission is attacking at a location. Error 216 is produced for the mission when this happens.

### 3.65 JTLS-2019-14609 Crash Merging Formations

*Two formations were ordered to merge into a single formation, causing a game crash. When finding a good station location for each ship, the bearing and range of the ship from its old formation was passed to a routine that expected the new formation's location instead. This resulted in the argument for a ship with a bearing and range of 178 degrees at 18.5 km being treated as a formation center of 178 North 18.5 East, resulting in a crash due to an invalid latitude.*

Instead of passing the bearing and range of the ship from the formation center, the two arguments were modified to correctly pass the latitude and longitude location of the accepting formation.

### 3.66 JTLS-2019-14610 ELS Convoys With Destroyed Trucks

*When a truck convoy was damaged and some of the trucks were destroyed, the ELS continued to move the truck along its route. The trucks appeared to move as if they were not damaged.*

The ELS was not properly processing the destruction of assets in convoys. Code changes were made to immobilize the destroyed truck entities and mark them as destroyed.

### 3.67 JTLS-2019-14611 CEP Batch Run Without AAR

*The CEP is able to run in Batch mode, which executes multiple runs of the CEP. At the end of each run, the AAR creates a dump file for that run. The script running the Batch process waits for that AAR dump file to finish writing. However, if the AAR is not running during this Batch, the script would wait and never move on to the next run of the Batch.*

A flag has been added to the Batch processes, allowing a user to specify if the AAR will be creating dump files during the Batch execution.

### 3.68 JTLS-2019-14612 Sinking Ship Station Location Crash

*A ship that was part of a moving formation, and was moving internally within the formation to its station location, was severely damaged and started sinking. This set the ship's speed to zero and its posture to incapable. The formation continued to move away from the sinking ship, but each time the formation moved the ship would check if it had a route to its station location.*

*The formation eventually moved to a location where the sinking ship did not have a direct path, nor a sea route, to its station location, and the logic to find a grid based solution was invoked. This logic also attempts to determine the time the ship will require to get to the station location based on the ship's speed. This time calculation resulted in a divide by zero causing the model to crash.*

The code in the Move Formation event was modified to prevent a ship with zero that should be moving to its station location from doing the station location check. Since the ship cannot move, there is no need to calculate the path from the ship to its station location.

If the Controller increased the ship's strength to stop the sinking process, the ship did not start moving again because the speed was not being adjusted. This was also fixed.

A minor memory leak when a ship is calculating a proper station location within the formation was discovered and fixed.

### 3.69 JTLS-2019-14613 Fortify TRIPP Report Browser Null Dereference

*The Report Browser component of the TRIPP was flagged as vulnerable by the Fortify scan.*

The flagged instance was attempting to print a URL object on a caught exception on itself, creating the possibility that on an exception occurrence the print will also fail. The code was changed to print the more informative URL string on an exception occurrence instead of the object in question.

### 3.70 JTLS-2019-14614 IMT Package Mission Quick Button Inoperative

*The IMT definition for the Package Quick button on the IMT Air Mission screen did not properly retrieve all missions in the package.*

The method used to retrieve the proper missions was to retrieve all missions that have the same package ID. However, an Air Mission Package Object is not a displayable object and the retrieval was only properly working for displayable objects. The IMT Quick button retrieval was expanded to properly retrieve based on an ID for any type of object.

### 3.71 JTLS-2019-14616 Unit Moving While In Attached Posture

*An order for a future group move was sent, and some of the units in the move's Group Unit List were attached to the Lead Unit. The process of performing the attach removed these units from the game board. These units still had a scheduled task to move, and when the task time arrived they started their movement logic. When the first move event occurred the units attempted to remove themselves from the game board again, resulting in a game crash.*

The processing of the task was modified to perform a number of checks that are performed for an immediate movement. These include ensuring the unit is not being transported by a convoy, is not embarked on a formation, is not involved in an airlift, and is not attached. If any of these conditions apply the task is not started.

### 3.72 JTLS-2019-14617 Memory Leak Print OPM Foreign Units

*The OPM generation procedure crashed during the JTLS-GO 6.0 Alpha Test. The problem was that it was possible to not release a set of arrays if a blank side is included in the database. Using Simscript III this causes a crash, but when using Simscript II.5 the problem is ignored. JTLS-GO 5.1 uses the old compiler and although the problem ignored it should not be.*

The code was corrected to match the JTLS-GO 6.0 version of the code, in which the arrays are released at all times.

### 3.73 JTLS-2019-14618 Refuel Spreadsheet Provide Fuel Magically

*The Magic Air Ops Order has several associated order spreadsheets. Two of the most used spreadsheets are the Top Off Fuel Spreadsheet and the Add Fuel Spreadsheet. The Top Off Fuel*

*Spreadsheet results in an order indicating that the fuel added to the mission should be provided magically. The Add Fuel Spreadsheet results in an order indicating that the fuel should be taken from the mission's normal providing unit, which is usually its home base. There is no way to tell on the spreadsheet which option is used for the generated orders. For this reason, it is important that the two spreadsheets work in the same manner.*

The end user has indicated that the magically provided fuel option meets their needs better. The change was made and now both the Top Off Fuel and Add Fuel spreadsheets results in orders that tell the model to magically provide fuel to the mission.

### 3.74 JTLS-2019-14619 Nodes At Same Location Crash

*The user added a branch to the sealane network using the Manage Move Order. The first new node was placed at the exact same location as the starting node for the new branch, resulting in a zero-length arc and the crash.*

The model was improved to throw away any nodes in the user-specified branch which would result in a zero-length arc.

### 3.75 JTLS-2019-14620 Apache Port Support

*Apache must support the standard ports for http (80) and https (443), to satisfy client security requirements.*

The ICP now allows ports 80 and 443 to be assigned as the http and https ports. The WSM can be used to start and stop the Apache server on the standard http and https ports, provided that the JTLS-GO user is also a sudoer.

### 3.76 JTLS-2019-14621 HIP JTLS-GO Owned Attributes

*Whenever control of a game object is passed to an external model, the HIP stops updating any of the JTLS-GO owned attributes of the object.*

The HIP attempts to sends updates to the federation for all objects. This is typically all of the objects because they all originate in JTLS-GO. If ownership of an object is passed to an external model in the federation, the HIP still tries to send updates for the passed object. Even though the object has been passed, some attributes of the object may still be owned by JTLS-GO (shared ownership). However, the HIP code was checking the overall ownership of the object and exiting of the update process for externally owned objects.

This code in the HIP has been changed. The HIP now updates any attributes specifically owned by JTLS-GO regardless of overall object ownership.

### 3.77 JTLS-2019-14622 Add Target Move To Warning 1606

*An auto correction for Warning 1606, to move a owned target to the owning unit, was missing.*

Owned targets must be within the owning unit radius. A SVPR auto correction option was added to move the target to the owning unit location when selected.

### 3.78 JTLS-2020-14623 Replenishing Unit Did Not Follow Moving Unit

*A naval vessel was given a Replenish Unit order for another naval vessel that was moving. The replenishing vessel plotted a route to the current location of the receiving vessel. By the time the replenishing vessel arrived at that location, the receiving vessel had moved away, causing the replenishing vessel to move again. This behavior repeated several times.*

*Eventually, the replenishing vessel caught up with the receiving vessel because the receiving vessel was ordered to reduce its speed considerably. The replenishing vessel stopped moving and the Offload Supplies task commenced as expected. The receiving vessel continued to move away, leaving the replenishing vessel behind during the replenishment process.*

No logic existed to properly take into account the movement of a receiving unit. New logic was implemented to cause the replenishing unit to adjust its arrival route toward the receiving unit each time the replenishing unit moves. If the replenishing unit is following an ingress route specified in the Replenish Unit order, the arrival route is not adjusted until the ingress is completed (By definition, ingress routes must be followed).

New logic was also added to cause the replenishing unit to follow the receiving unit until the Offload Supplies task is complete (SLP REPLENISHMENT TIME). The new logic applies to both naval and ground units.

### 3.79 JTLS-2020-14631 Package Mission Against JDPI Does Not Work

*If the user submits an order for an attack mission in an air mission package and specifies a JDPI, the order is always rejected by the model.*

The problem was that both the name of the package and the name of the JDPI were being placed on the same word on the order structure. The mission either could not find the package or the specified JDPI. The name of the package was placed on a different word.

The checkorder added a new check that catches two field\_refs using the same word on the main panel or an order tab. However, within a given group, the same word can overlap within options in a given group.

### 3.80 JTLS-2020-14634 Gather Grids Incorrect When Point On Line

*If a rail or road node is exactly on a left or right grid line, and the opposite arc node is north of the first node, the model does not properly compute the grids lines through which the arc travels. This applies to both the CEP and SIP.*

A typo in the code resulted the model assigning a direction of west when the direction should have been directly north.

### 3.81 JTLS-2020-14635 Unusable Location In SVP Correction

*SVP Error 956 reported that a network node was improperly located in an ocean grid. The first correction for this error was meant to find the location of this network problem on the map. This find resulted in the position of the center of the grid where the network node was located. This was not useful if it was a large grid, or there was more than one node in the grid.*

The correction was modified to located the actual position of the network node. This facilitated a proper correction for the error in a more efficient manner.

### 3.82 JTLS-2020-14636 Crash Ending Coalition Support In Dual Grid

*If the user tells an HRU that does not own a small boat to stop supporting a naval unit while the unit is in a dual-capable grid, the model will crash when a new task is given the HRU.*

The model assumed that the HRU that left the ship was in the water portion of the dual-capable grid, and when it tried to move to its next task, the model crashed because HRU was not in a small boat. To correct the problem, two improvements were made:

- a. If the HRU is placed in a dual-capable or small island grid, and it does not have a small boat, the model assumes that the ship dropped the HRU on land.
- b. If the HRU is placed in the ocean, the model magically provides the HRU with a life raft similar to the life raft given to a downed aircrew.

### 3.83 JTLS-2020-14637 ELS Not Saving Customized Templates

*The aggregate model allowed the player to create customized HRUs, which have a different set of combat systems that was defined in the High Resolution Unit Prototype (HUP). The ELS successfully created these customized HRUs by also creating customized templates. These customized templates were not saved to disk when they were created. This prevented the ELS from restarting from a checkpoint.*

Code changes were made to save the customized templates at the time they were used by a new HRU. After this correction, the ELS was able to restart from a checkpoint which contained customized HRUs.

### 3.84 JTLS-2020-14638 Cruise Missiles At Zero Altitude

*Cruise Missiles starting flying at their database-assigned altitude, and then along their flight path would go to an altitude of zero.*

The issue occurred when creating the route for the missile. The missile's altitude was not placed on the route points and when the missile traveled through the route point, its altitude was set to zero.

### 3.85 JTLS-2020-14639 No ATO Mission Type For Package Mission

*When translating an ATO, the ATOT creates an appropriate mission type, but also has the ability to tell the model the name of the mission type that existed in the ATO. This capability does not exist for package missions.*

The problem was that the Package Mission Order did not have a field for this information. The order was corrected and the ATOT changed to properly fill the formerly missing order field.

### 3.86 JTLS-2020-14640 Air Mission ELINT Detections Not Turned Off

*When an air mission moves away from an enemy emitter, the detected emitter objects are not being properly deleted.*

When picking up the Electronic Intelligence (ELINT) coverage tags, the flag indicating that the tag represented an ELINT collector was not being properly set. Because it was not set, the model did not know that it should look for detected emitters and destroy the emitter detection object.

While testing this fix, it was noted that if the user sends an air mission with an ELINT sensor to a single point orbit location, the tags may not get moved. For speed purposes, the model only updates the coverage tags every MAX MISSION MOVEMENT TIME and then only if the location does not change. The mission at a high rate of speed moved in to the orbit location and never moved the tags around its orbit location. Because of the speed, the tags were several hundred kilometers behind the mission's orbit location. This very unusual circumstance was solved by making sure that the mission moves the tags when it arrives and starts to orbit.

### 3.87 JTLS-2020-14642 Missing End Record Character On MTF Message

*Two USMTF messages were found without an end of record character for the last record.*

The message file was changed to output the end of record characters "//".

### 3.88 JTLS-2020-14643 Submarine Contact Report Error

*Message 6600 was not being properly generated. It was missing a space between two data elements.*

A space was added to correct the problem.

### 3.89 JTLS-2020-14644 Execute HRU Event Not Setup Properly

*A call to the SETUP EVENT routine occurs at the beginning of every JTLS-GO event. This routine writes to the CEP screen that the event is executing, and starts a statistics timer to help determine where the model is spending processing time. This routine is not called from the event EXECUTE HRU TASK.*

The proper call to SETUP EVENT was added to the Event EXECUTE HRU TASK.

### 3.90 JTLS-2020-14645 DJINN Uploading Newly Created Ground Unit

*After the JINN component in the DDS (DJINN) imported data returned from a JOBE user, the JINN could not upload the data to the database if there were any newly created ground units.*

This problem was fixed in the DJINN source code. A similar problem was corrected for FARP units.

Because the classic JINN application, which was launched separately from the DDS using the jtlsmenu or Javamenu, may cause data upload problems, and the DJINN component has been verified to function correctly, the option to start the classic JINN has been removed from the jtlsmenu and Javamenu.

### 3.91 JTLS-2020-14646 Great Circle Distance Under 1 Meter Error

*The great circle distance algorithm sometimes results in a very large number, when in fact the two locations were not exactly the same, but less than one meter apart.*

It was a very unusual circumstance, but for any two locations that are within one meter of each other, the Great Circle Distance algorithm should return a distance of zero.

### 3.92 JTLS-2020-14647 ATO Parser Error Report Should Be Warning

*Paired with the output from the TBMCS Adapter, the ATO Parser often discovers ICAO location strings in the ATO message that cannot be properly identified as ICAO codes, and therefore reports them as errors. The errors are reported as PE-221, but should be reported as a warning.*

The ATO Parser error message PE-221 was changed to warning PW-242. The Parser has no ability for determining proper ICAO codes. This should only be reported to the user as a warning so the user can make adjustments from the ATOT. Also, The Parsers used with the other ATO generating Adapters also were changed because they function similarly.

### 3.93 JTLS-2020-14648 ATOT Reports Missing Mission Tasks

*The ATOT is giving an error that an alert mission from the ATO is missing any tasking. This occurs when the alert site for the mission is not the home base. Missions planned to be on alert do not have tasking. They simply wait on the strip at their home base or at a remote base.*

The ATOT may create two sets of order utilities for missions that are suppose to be on alert. This is a feature that allows users to quickly select a pre-built tasking utility whenever the plan for the alert mission rapidly changes, typically after the ATO message has been processed. For the tasking utility to be created the mission must have tasking in the ATO along with being coded as an alert mission. This logic functions properly when an alert mission uses its home base ICAO as an alert site.



However, when an alert mission with no additional tasking used a remote ICAO as an alert site the ATOT executes code for creating the tasking utility. This ATOT code includes logic for checking for at least one task, which this alert mission did not have.

The ATOT logic has been changed so it will not execute the utility for alert missions when they do not have additional tasking.

### 3.94 JTLS-2020-14649 DSA Spreadsheet Program Issues

*The Directed Search Area (DSA) Spreadsheet Program (DSP) allowed the user to easily build large numbers of DSAs from a spreadsheet. The input spreadsheet listed the locations and types of areas used for collection. In JTLS-GO, these spreadsheets are meant to be stored in the \$JGAME/scenario/orders directory.*

*There are two ways to run the DSP: from the Javamenu or from the text-based jtismenu. When the DSP was run using the jtismenu, the DSP failed to access the input spreadsheet. As a result, the DSP did not run correctly using the jtismenu.*

The DSP did run successfully from the Javamenu, but not from the jtismenu. The script which executed the DSP from the jtismenu was incorrectly looking for the input spreadsheet files from the wrong directory. Changes were made to this script to fix this problem.

### 3.95 JTLS-2020-14650 OPM Formatting Error

*The OPM page which described the Faction descriptions had a formatting error. The line which listed the Combat System Prototype had a misplaced parentheses and an extra space in the text.*

The code to generate the OPM was modified to correct these formatting errors.

### 3.96 JTLS-2020-14651 Crash Destroying Auto Move Task

*When an object attempts to start a task, the CEP asks if the object is in the proper location to execute the task. If the object is not in the proper location, a automatic move task is added to the object's task list. When the user submitted a magic move for the unit, the model crashed.*

As part of the order, the user indicated that the unit's pending tasks should be canceled as part of the magic move. The unit has three tasks on its task list. The first task was a Ground Wait task, because the unit ran into a terrain problem attempting to execute the second task, which was an automatic move to get the unit to its patrol area. The third task was the patrol task.

When the tasks were being canceled, the Ground Wait canceled, followed by the Automatic Move. The model determined that it was an automatic move and then also canceled the task associated with the automatic move. When the logic looped to destroy the third task, the model crashed because the third task had already been destroyed when the automatic move was destroyed.

The problem was solved by clearing the automatic move flag on the second task, because all tasks were going to be deleted anyway. A thorough check of the code was accomplished and four other loops destroying tasks were found that the same problem could potentially happen.

### 3.97 JTLS-2020-14652 Manage SMA Order

*After running the order checker, it was discovered that the Controller Manage SMA order had several errors. When creating a new barge or rail asset, the model would use default data if the user failed to input data. The message a player received after creating a new rail asset was not complete.*

The model no longer use default data, because the data fields on the order are now mandatory.

When creating a new barge asset, the Player would receive a message indicating that the Player created a new barge asset and list all the current barge assets. The model would not do the same when creating a new rail asset. The Player would receive a message indicating the new rail asset was created, but the message did not list the current rail assets. The message now lists all the current rail assets.

### 3.98 JTLS-2020-14653 ELS Template Unassigned Echelon Crash

*The ELS crashed while attempting to save a customized template. A Controller changed the combat system counts for a Tactical Unit Prototype. This triggered the ELS to generate a unit-specific template while the game was running. Such customized templates were automatically saved to disk during the creation process. The crash indicated that the echelon level for this particular template was zero. This caused the crash because zero is not a valid echelon level.*

The ELS code was modified to ensure that a template never has an echelon level of zero. If there was no valid echelon to use for a new or modified unit, then the ELS assigned the echelon to be the value reserved for an UNNAMED or UNKNOWN echelon. This change resolved the crash situation.

### 3.99 JTLS-2020-14654 ATOT Converts ACO Circles To DSAs

*The ATOT creates DSA orders and graphic slides for airspace features that are defined in the ACO. However, when a circular airspace is translated by the ATOT, its radius is always 1 km, regardless of the radius of the corresponding airspace.*

The logic used by the ATOT was incomplete. The ATOT should have converted the radius to 1 km, but only when the radius was not provided from the ACO airspace. This has been fixed.

### 3.100 JTLS-2020-14655 Infinite Loop Ship Running Out Of Fuel

*The model went into an apparent infinite loop when a ship was attempting to move and did not have enough fuel to accomplish the full move.*

The model computed how far the ship could travel with its available fuel, and then performed that move, and because the ship had very little fuel (but more than what the model calls ALMOST.ZERO), the model repeated this computation with no activity visible to the user. If the amount of available fuel was less than ALMOST.ZERO, the ship would have reported being out of fuel and would have stopped, but instead it continued to perform the computation repeatedly.

The new code now asks if the amount of fuel is less than ALMOST ZERO, if the computed move distance is less than one METER, or the computed move time is less than ONE SECOND. Under any of these circumstances, the ship reports being out of fuel.

### 3.101 JTLS-2020-14656 Magic Air Ops Order Crash

*The model crashed attempting to top off fuel for an air mission that was in Aircraft Delay.*

The user did not closely look at the IMT and first attempted to top-off the fuel for the aircraft delay mission. The order was successfully executed even though there were no aircraft in the mission. Because the fuel for the mission did not increase, the user decided to give the mission 50000 gallons of fuel, even though the aircraft could only hold 600 gallons of fuel.

Because the mission has no aircraft, and the amount of fuel was so large compared to what the aircraft could hold, the model's computations could not handle the situation. The fuel that could be given ended up being computed as an extremely small negative number. When attempting to write out this value, the model crashed.

The solution was to reject the Magic Air Ops order if the aircraft was not in the air. The order will now be rejected when the air mission is on the ground.

### 3.102 JTLS-2020-14657 Ghost Updates Always Saved

*A ghost detection in JTLS-GO is the observation of a location where the side thought it should see something, but nothing was there. Each time a Recce mission moves, it collects and save newly obtained information. The model always thought ghost detections were new and thus saved each detection even though the empty area had been observed before.*

*The end result of this situation was that a long deployed air mission could potentially have tens of thousands of ghost updates in its information set. This causes a checkpoint problem because the maximum number of updates that a single air mission can hold is 65535.*

The Intel staff has never been able to explain how to properly report the lack of seeing an object. Until specifics can be worked out, the decision was to never save ghost updates. They are currently not reported and they should not be saved.

### 3.103 JTLS-2020-14658 DDS Slowdown Loading Large Networks

*The DDS attempts to retrieve a target associated with a road or railroad arc when it paints the arc. With a very large road or railroad network combined with a large number of targets, this*

*consumes large amounts of CPU time. The DDS was failing to check if there was a non-empty value for the arc's associated target before searching through the target list.*

The DDS now checks if an arc's associated target is a non-empty value before searching through the target list.

### 3.104 JTLS-2020-14659 Order Group Editor Save Empty Groups

*Users are able to create Order Groups to organize orders in the Order Group Editor. Some users may choose to create Order Groups before they have orders to assign to the group. Empty Order Groups were successfully saved and visible to the user upon creation, but failed to be displayed to the user if the WHIP was restarted.*

The Order Group Editor now displays empty Order Groups to users when they restart their WHIP.

### 3.105 JTLS-2020-14660 Aircraft Class Link 16 Capable Flag Mismatch

*The Link 16 Message Service decides which air missions send messages based on the Link 16 Capable flag of the mission's Aircraft Class. This Capable flag has the values of NO, YES, and SEND\_AIR\_INFO. The enumerations of these values were mismatched between the JTLS-GO Data System Protocol (JDSP) and the Link 16 Message Service. This caused air missions that should have been sending J13.2 messages to only send J2.2 messages, and air missions that should have been sending J2.2 messages to fail to report.*

The Link 16 Message Service has been changed to match the enumerated Link 16 Capable flag values of the JDSP.

### 3.106 JTLS-2020-14661 Link16 And OTHGold Unknown Relationship

*The Link 16 and OTHGold message services report the relationships of tracks in the game. When reporting tracks from side 11, the relationship for the tracks was always assigned to "UNKNOWN". This is an artifact from JTLS-GO moving from a maximum of 10 sides to 20 sides.*

The Link 16 and OTHGold message services have been updated to account for the current maximum number of sides available in a game.

### 3.107 JTLS-2020-14662 WHIP View Mission MTF Text From Controller

*The WHIP can display air mission MTF text from a processed ATO. The WHIP failed to find the associated mission MTF text when querying from a Controller WHIP.*

The WHIP has been modified to query for MTF text from a Controller WHIP.

### 3.108 JTLS-2020-14663 TACELINT Service Compatibility Fixes

*The TACELINT receiving system had several strict parsing requirements. The first line of the SOI record failed to end on a record break, and the AMPN record content was incompatible with the receiving system.*

Changes were made to make the TACELINT Message Service (TEMS) output compatible with this system. The first line of the SOI record now ends in a record break, and the AMPN record has been removed until JTLS-GO can provide the expected data. In addition, a logging statement was stated that the TEMS had closed the JODA socket connection removed that, due to the high frequency that it was logged.

### 3.109 JTLS-2020-14665 Air Defense Fires Zero Weapons At Missile

*An Air Defense Report indicated that zero weapons were fired on a missile. The report should only be generated when an Air Defense Site actually fires.*

There was a logic error when trying to fire on a missile. If the site had its preferred weapon available to fire, but there was no PK associated with firing the weapon against the incoming missile, and the site had an alternate weapon specified, but none of those alternate weapons available, the logic resulted in generating a damage message which indicated that zero weapons were fired. This has been corrected.

### 3.110 JTLS-2020-14666 DDSC Edit Railroad/River Arcs

*Users could not change the railroad or river to which an arc belonged in the DDSC railroad arcs and river arcs tables.*

The ability to edit this attribute was set to "false" instead of "true." This has been corrected.

### 3.111 JTLS-2020-14667 ATOT Package Mission Order Target Type List

*The ATOT creates orders for attack missions as part of a package of missions. However, the order created by the ATOT for the attack mission never includes the associated target type list.*

The code logic of the ATOT incorrectly assumed that the target type list was only included when no other target options were being used by the mission (such as a victim unit or attack location). This has been changed to properly include the target type list when available, and for the unit and location targeting options.

## 4.0 REMAINING ERRORS

Every effort has been made to correct known model errors. All reproducible errors that resulted in Combat Events Program (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 5.1.0.0 represents a major release of new functionality, remaining outstanding errors from the JTLS-GO 4.1 series and earlier 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.



## 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
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
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
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
EEI	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
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	Japan Self-Defense Forces
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
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-GO 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
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
XML	Extensible Markup Language
XMS	XML Message Service

## **APPENDIX B Version 5.1.5.0 DATABASE CHANGES**

No database structure changes were made for JTLS-GO 5.1.5.0.





## APPENDIX C Version 5.1.5.0 REPOSITORY CHANGES

The following changes were made to the JTLS-GO 5.1 repository.

### C.1 Added TUPs:

- 3PSYOP.BN.CO\_US
- 3PSYOPS.BN.\_US
- 3PSYOPS.TM\_US
- 410CA.BN.CO\_US
- 410CA.BN.TM\_US
- 410CA.BN\_US
- 6PSYOP.BN.TM\_US
- 6PSYOPBN.DET\_US
- 6PSYOPS.BN.\_US
- 82.DIV.HHB\_US
- 82.MAIN.CP\_US
- 82DIV.TAC.CP\_US
- AVN.AHB\_US
- AVN.ARB\_US
- AVN.GSB.C3CO\_US
- AVN.GSB.CH47\_US
- AVN.GSB.HQ\_US
- AVN.GSB.MED\_US
- CAB.E.CO\_US
- CHEM.CO\_US
- CHEM.RESP.CO\_US

- E.CO.160AVN\_US
- GUAM.NB\_US
- HARPY.BTRY\_KR
- HHC.321CABDE\_US
- HHC.CAB\_US
- HHC.CHEM.BN\_US
- ORD.CO\_US
- SPT.BN.FABDE\_US

#### C.2 Updated Combat Systems for TUPs:

- AVN.SPT.BN\_US
- E1-160.MNTCO\_US
- TF1-160BNHQ\_US

#### C.3 Updated Combat Systems and Renamed TUPs:

- A1-160.AVN\_MH6\_US to A1-160.AVN.MH6M\_US
- B1-160.AVN\_AH6\_US to B1-160.AVN.MH6M\_US
- C1-160\_MH60K\_US to C1-160.MH60M\_US
- D1-160\_MH60K\_US to D1-160.MH60M\_US

#### C.4 Deleted TUPs:

- CAB.BDE.HQ\_US
- CHEM.BN.HHC\_US
- CMD.CO.UH60\_US
- GSA.BN.HQ\_US
- GSABHQ\_US
- HVY.CO.CH47\_US

- SPT.BN.FB\_US
- UH60.ASLT.BN\_US
- UH60ASLTBN\_US
- UH60ASLTBNLF\_US

C.5 Added Ground Unit:

- HARPY.BTRY\_KR

C.6 Added Combat Systems:

- M1A2C.120MM-TANK.AFHS
- APILAS\_AT.MAW.SR-HL
- BMM4A.AMBULANCE-VEH-LA
- BMM4B.BN.AID-VEH-LA
- BMM4C.MED.STATION-VEH-LA
- BTR4.30MM-IFV.HA
- BTR4.KSH-APC.LA.TW
- CM11.105MM-TANK.AFLS
- CM12.105MM-TANK.AFLS
- CM22.120MM-MTRSPHVY
- CM24A1.AMMO-VEH-LA
- CM27A1.RECOVERY-VEH-LA
- CM32.12.7MM-APC.LA.TW
- CM32.25MM-APC.LA.TW
- CM34.30MM-APC.LA.TW
- FLEET.USV\_AMPHIB
- HARBOR.USV\_AMPHIB

- M1278.JLTV.M240-APC.LA.OW
- M1278.JLTV.M2-APC.LA.OW
- M1278.JLTV.MK19-APC.LA.OW
- M1278A1.M240-APC.LA.OW
- M1278A1.M240-APC.LA.TW
- M1278A1.M2-APC.LA.OW
- M1278A1.M2-APC.LA.TW
- M1278A1.MK19-APC.LA.OW
- M1278A1.MK19-APC.LA.TW
- M1279.UTIL.JLTV-VEH-LA
- M1280.GP.JLTV-VEH-LA
- M1281.JLTV.M2-APC.HA.TW
- M1281.JLTV.MK19-APC.HA.TW
- M1281.JLTV.TOW-AT.LH.LO
- ORCA.XLUUV\_AMPHIB
- SEA.HUNTER.USV\_AMPHIB
- SNORKELER.UUV\_AMPHIB
- UNMANNED.LAND.VEH-BOMB

#### C.7 Added SUPs:

- GERMANTOWN\_US
- INDE.LCS.NSM\_US
- TORTUGA\_US

#### C.8 Added Naval Units:

- DELAWARE\_US (SUP - VIRGINIA\_US)

- OREGON\_US (SUP - VIRGINIA\_US)
- INOUYE\_US (SUP - BURKE.2A\_US)
- KEITH.ESB.5\_US (SUP - PULLER.ESB\_US)
- PUERTO.RICO\_US (SUP - SPEARHEAD\_US)
- STONE\_US (SUP - LEGEND\_US)

#### C.9 Changed Naval Unit SUPs:

- Changed SUP for GIFFORDS\_US to INDE.LCS.NSM\_US
- Changed SUP for MONTGOMERY\_US to INDE.LCS.NSM\_US

#### C.10 Changed Naval Unit Homebase:

- SPRINGFIELD\_US to New Homebase - PEARL.HARBOR.NB\_US

#### C.11 Deleted Naval Units:

- LOUISVILLE\_US (due to inactivation)
- OLYMPIA\_US (due to inactivation)
- PITTSBURGH\_US (due to inactivation)

#### C.12 Added HUPs:

- ORCA.EW.XLUUV
- ORCA.MK46.XLUUV
- ORCA.MK48.XLUUV

#### C.13 Added HRUs

- ENG.BN.3ID.ULV.BOMB1
- ENG.BN.3ID.ULV.BOMB2
- ENG.BN.3ID.ULV.BOMB3
- FLEET.EW.USV
- FLEET.M2.USV

- FLEET.MK19.USV
- FLEET.SEN.USV
- HARBOR.EW.USV
- HARBOR.M2.USV
- HARBOR.MK19.USV
- HARBOR.SEN.USV
- ORCA.EW.XLUUV
- ORCA.MK46.XLUUV
- ORCA.MK48.XLUUV
- SEA.HUNTER.USV
- SNORKELER.BOMB.USV
- SNORKELER.EW.USV
- SNORKELER.SEN.USV
- SNORKELER.WPN.USV

C.14 Added ADA Class:

- SKYSWORD2(SP)VE

C.15 Added SSMs:

- 3S-14.UVLS(8)
- DF-ZF.HGV.TEL

C.16 Added Supply Categories:

- CL.V.SA-SR-SKYSWORD1.SAM
- CL.V.SA-SR-SKYSWORD2.SAM
- CL.V.SS-RDR-LR-DF-ZF.HGV
- CL.V.SS-RDR-MR-SSN33

## C.17 Added Target Type Group:

- BRIDGE

## C.18 Added Targetable Weapons:

- DF-ZF.HGV
- HRU.ULV.BOMB
- SKYSWORD2.SAM
- SSN33(ZIRCON)

## C.19 Added New SKL Table:

- DF-ZF.HGV\_SKL
- HRU\_ULV\_BOMB
- RIM156.SUR\_SKL
- RIM7M.SUR\_SKL
- SSN33\_SKL

## C.20 Added New PK Table:

- DF-ZF.HGV\_PK
- HRU\_ULV\_TANK\_PK
- RIM156.SUR\_PK
- RIM7M.SUR\_PK
- SSN33\_PK

## C.21 Updated PK Tables:

- MK46M12ALTP\_PK to only attack Submarines
- MK46M12SLTP\_PK to only attack Submarines
- MK46M5.ALTP\_PK to only attack Submarines
- MK46M5.SLTP\_PK to only attack Submarines



## C.22 Added New PH Table:

- DF-ZF.HGV\_PH
- HRU\_ULV\_TANK\_PH
- RIM156.SUR\_PH
- RIM7M.SUR\_PH
- SSN33\_PH

## C.23 Updated PH Tables:

- MK46M12ALTP\_PH to only attack Submarines
- MK46M12SLTP\_PH to only attack Submarines
- MK46M5.ALTP\_PH to only attack Submarines
- MK46M5.SLTP\_PH to only attack Submarines

## C.24 Added AKL Table:

- RIM156.SUR\_AK
- RIM7M.SUR\_AK

## C.25 Added UOMs:

- DF-ZF
- SKYWORD2.SAM
- SSN33(ZIRCON)

## C.26 Added SAL:

- SKYWORD2\_SAL

## C.27 Changed TW Table (TIME PER ROUND):

- Changed TIME PER ROUND from 1D to 15S for RIM156.SM2.BK2
- Changed TIME PER ROUND from 1D to 15S for RIM156.SM2.BK3A
- Changed TIME PER ROUND from 1D to 15S for RIM156.SM2.BK3B

- Changed TIME PER ROUND from 1D to 15S for RIM156.SM2.BK4
- Changed TIME PER ROUND from 1D to .09S for RIM174.SM6
- Changed TIME PER ROUND from 1D to 15S for RIM7M\_P.NSSMS

C.28 Changed TW Table (TIME TO PREPARE FIRE):

- Changed TIME TO PREPARE FIRE from 0M to 15S for RIM156.SM2.BK2
- Changed TIME TO PREPARE FIRE from 0M to 15S for RIM156.SM2.BK3A
- Changed TIME TO PREPARE FIRE from 0M to 15S for RIM156.SM2.BK3B
- Changed TIME TO PREPARE FIRE from 0M to 15S for RIM156.SM2.BK4
- Changed TIME TO PREPARE FIRE from 0M to 15S for RIM174.SM6
- Changed TIME TO PREPARE FIRE from 0M to 15S for RIM7M\_P.NSSMS

C.29 Changed TW Table:

- For RIM174.SM6 checked MISSILE CABAPBLE and PRECISION GUIDED Columns from No to YES
- Added SRBM to the AIRCRAFT TARGET CLASS NAME Column for TW RIM174.SM6

C.30 Added Aircraft Target Class:

- HYPERSONIC

C.31 Changed Aircraft Classes:

- Changed AIRCRAFT CLASS HARPY from FIXED\_WING to ROTARY
- Changed C-17.GLOBEMASTER AIR\_REFUEL\_FLAG to YES
- Changed HARPY TAKEOFF LENGTH and LAND LENGTH from 1000ft to 0
- Changed AIRCRAFT CLASS - PROBABILITY DETECTION to .0008 for F35A
- Changed AIRCRAFT CLASS - PROBABILITY DETECTION to .0008 for F35B
- Changed AIRCRAFT CLASS - PROBABILITY DETECTION to .0008 for F35C
- Changed AIRCRAFT CLASS - PROBABILITY DETECTION to .0008 for J20

- Changed AIRCRAFT CLASS - PROBABILITY DETECTION to .0008 for SU57

C.32 Added Sensors:

- MPQ78\_LAA
- MPQ90\_LAA

C.33 Changed Global Values:

- Changed Classification in Global Values from UNCLASSIFIED to UNCLASSIFIED FOUO

C.34 Changed Small Boat Table:

- Set Small Boat SDV Operational Depth from 0 to 80 feet

C.35 Changed Jammer Table:

- Updated Jammer Names and WEAPON\_DEGRADE\_FACTOR - USE Columns

C.36 Updated Russian Order of Battle:

- Added over 600 Units to Russian Side