

JTLS-GO

Version Description Document

November 2021



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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 6.1.1.0)

[Blank Page]

ABSTRACT

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

This *JTLS-GO Version Description Document (VDD)* describes the new features of the Version 6.1.1.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 6.1.1.0 is a Maintenance release of the JTLS-GO 6.1 series that includes an updated repository of standard data, a demonstration scenario based in the western Pacific, as well as major model functionality improvements implemented as Engineering Change Proposals (ECPs). These ECPs 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
jtlsgo@rolands.com

Copyright 2021 - 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1.2.1 Obsolete/Outdated Documents.....	1-1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	1-4
1.3.1 Support Software	1-4
1.3.2 JTLS-GO Cybersecurity Compliance	1-7
1.3.3 JTLS-GO High Level Architecture Compliance.....	1-7
1.4 DATABASE MODIFICATIONS	1-8
1.4.1 JTLS-GO Using Legacy Default Symbol Set	1-8
1.4.2 JTLS-GO Using New Default Symbol Set.....	1-8
1.4.3 Standard Repository Changes	1-9
1.5 INSTALLATION	1-9
2.0 ENGINEERING CHANGE PROPOSALS.....	2-1
2.1 JTLS-2021-15323 NEW SVP CHECKS FOR FEDERATING WITH VBS4	2-1
2.2 JTLS-2021-15354 COUNT ORDERS BY UNIT	2-1
2.3 JTLS-2021-15360 MAINTAINING LINK 16 PICTURE	2-1
3.0 SOFTWARE TROUBLE REPORTS	3-1
3.1 JTLS-2021-15322 AUTO-GENERATION OF ORDER NAME FILE	3-1
3.2 JTLS-2021-15325 CRASH WHEN TWO MINES HIT SHIP	3-1
3.3 JTLS-2021-15328 WHIP SLIDE IMAGE GRAPHICS RE- SAVED ON EACH EDIT	3-1
3.4 JTLS-2021-15329 UNIT NOT ARRIVING WITH ZERO STARTING ON HAND SUPPLIES	3-1
3.5 JTLS-2021-15330 MORE COLOR CHOICES FOR TEXT GRAPHIC EDITOR PANEL	3-2
3.6 JTLS-2021-15333 FIX GRAPHIC ATTRIBUTES BEHAVIOR	3-2
3.7 JTLS-2021-15334 UNIT ATTEMPT TO INTERDICT CONVOY CRASH	3-2
3.8 JTLS-2021-15336 SSM TARGETS MISSING FROM UNIT WEAPON ARRAY	3-3
3.9 JTLS-2021-15337 CRASH WHEN MOBILITY MISSION PICKS UP UNIT	3-3
3.10 JTLS-2021-15338 INVALID COMMAND AUTHORITY OBJECT IN JODA	3-3
3.11 JTLS-2021-15339 STOP REPLENISH SUPPLY TASK BEFORE COMPLETE CRASH	3-4
3.12 JTLS-2021-15341 GIS TOOL IMPROVEMENTS	3-4
3.13 JTLS-2021-15343 AIR MISSION WEAPON FIRE AGAINST LOCATION	3-4
3.14 JTLS-2021-15344 ELS CRASH DURING NEW JODA DOWNLOAD	3-5
3.15 JTLS-2021-15345 CANNOT SELECT MISSILE IN CONTROLLER KILL ORDER	3-5

- 3.16 JTLS-2021-15346 CRASH CHECKING HRU BOAT RESOURCES AT GROUND PARENT 3-5
- 3.17 JTLS-2021-15348 SHIP UNIT PROTOTYPE OPM ERROR 3-5
- 3.18 JTLS-2021-15349 NO COMMAND AUTHORITY FOR TUNNEL SHELTERING ORDER 3-6
- 3.19 JTLS-2021-15350 LC2IS COUNTRY CODES IN NATO-DIGRAPH 3-6
- 3.20 JTLS-2021-15352 MDP E-MAIL DELIVERY FAILURE 3-6
- 3.21 JTLS-2021-15356 CRASH TURNING OFF NATIONAL ELINT COVERAGE 3-7
- 3.22 JTLS-2021-15358 DROPPED TRACKS ON SHIP DETECTIONS 3-7
- 3.23 JTLS-2021-15362 OPM REGENERATION ERROR READING CHECKPOINTS 3-8
- 3.24 JTLS-2021-15364 ALLOW FIRE MISSILE RANGE WORLDWIDE 3-8
- 3.25 JTLS-2021-15366 LINK 16 SOURCE UNIT NOT ON NETWORK CRASH 3-8
- 3.26 JTLS-2021-15368 XMS CRASH ACCESSING MESSAGE NOTIFICATION 3-8
- 3.27 JTLS-2021-15371 MAP NETWORKS/ROUTES DISPLAYED IN SAME COLOR 3-9
- 3.28 JTLS-2021-15372 JSAT GUI TWENTY-FOUR SATELLITE LIMIT 3-9
- 3.29 JTLS-2021-15377 IIR GENERATION CANNOT BE STOPPED 3-9
- 3.30 JTLS-2021-15378 TARGETS WITH MULTIPLE ELEMENTS 3-9
- 3.31 JTLS-2021-15379 PERSONNEL MEAN TIME TO REPAIR LARGE 3-10
- 3.32 JTLS-2021-15380 ATO PARSER NOT ADDING ROUTE LOCATIONS 3-10
- 3.33 JTLS-2021-15383 SVP CRASH AFTER 5.1 TO 6.0 DATABASE CONVERSION 3-10
- 3.34 JTLS-2021-15385 ATO PARSER INCLUDES MISSION LOCATION TASKS 3-10
- 3.35 JTLS-2021-15386 ATO TRANSLATOR MOBILITY MISSION ORDERS 3-11
- 3.36 JTLS-2021-15389 WHIP COMMAND AUTHORITY CHECK FAILURE 3-11
- 3.37 JTLS-2021-15393 OTHGOLD MODULE MULTIPLE ERROR POPUPS 3-11
- 3.38 JTLS-2021-15395 WHIP UNIT COMBAT MAP FILTER DOES NOT SET 3-11
- 3.39 JTLS-2021-15396 ICP CLASSIFICATION COLOR TEXT MISLEADING 3-12
- 3.40 JTLS-2021-15397 NO MESSAGE FOR GROUND UNIT OUT OF FUEL 3-12
- 3.41 JTLS-2021-15400 LINK 16 REDUNDANT FEED TOGGLE CONFIRMATION 3-12
- 3.42 JTLS-2021-15401 PLAYER PERIODIC REPORTS REFORMATTED 3-12

- 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

- APPENDIX A. ABBREVIATIONS AND ACRONYMSA-1

- APPENDIX B. VERSION 6.1.0.0 DATABASE CHANGES B-1

- APPENDIX C. VERSION 6.1.0.0 REPOSITORY CHANGES C-1

1.0 INTRODUCTION

1.1 SCOPE

This *JTLS-GO Version Description Document (VDD)* describes Version 6.1.1.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO[®]) software suite. JTLS-GO 6.1.1.0 is a Maintenance delivery for the JTLS-GO 6.1 series of releases.

JTLS-GO 6.1.1.0 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 “wespac61”. Database modifications that were accomplished to upgrade the previous JTLS-GO database format to this current version are summarized in this chapter, as well as [APPENDIX B](#). Detailed descriptions of the Engineering Change Proposals (ECPs) implemented for this release are provided in [Chapter 2.0](#).

JTLS-GO 6.1.1.0 executes on the Red Hat Enterprise Linux Version 8.4 64-bit operating systems. The Web-Hosted Interface Program (WHIP[®]) user workstation interface can be executed on any operating system from any Java-compatible Web browser.

1.2 INVENTORY OF MATERIALS

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

1.2.1 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 6.1.0.0)
- *JTLS-GO Air Services User Guide* (JTLS-GO Document 03, Version 6.1.0.0)
- *JTLS-GO Configuration Management Plan* (JTLS-GO Document 03, Version 6.1.0.0)
- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 6.1.0.0)
- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 6.1.0.0)
- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 6.1.0.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 6.1.0.0)
- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 6.1.0.0)

- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 6.1.0.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 6.1.0.0)
- *JTLS-GO Standard Database Description* (JTLS-GO Document 14, Version 6.1.0.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 6.1.0.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 6.1.0.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 6.1.0.0)
- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 6.1.0.0)
- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 6.1.0.0)
- *JTLS-GO DoD Architecture Framework* (JTLS-GO Document 22, Version 6.1.0.0)

1.2.3 Updated Documents

- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 6.1.1.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 6.1.1.0)

1.2.4 New Documents

No new documents are required for this version of the software.

1.2.5 Delivered Software Components

JTLS-GO 6.1.1.0 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)

The current JODA build number is 172.

- 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)

- Entity Level Server (ELS)
- JTLS Operational Interface (JOI) for both OTH-Gold and Link-16 generation
- Tactical Electronic Intelligence (TACELINT) Message Service
- Keyhole Markup Language (KML) Operational Interface (KOI)

- JTLS Transaction Interface Program (JTOI)
- JTLS Interface Network Navigator (JINN)
- JTLS Order of Battle Editor (JOBED)
- JTLS Geographic Information System (GIS) Terrain Building Program
- JTLS Master Integrated Database (MIDB) Tool
- JTLS Version Conversion Program (VCP)

VCP60 - Converts a JTLS-GO 5.1 database to a JTLS-GO 6.0 formatted database.

VCP61 - Converts a JTLS-GO 6.0 database to a JTLS-GO 6.1 formatted database.

Instructions for installing JTLS-GO 6.1.1.0 are provided in the *JTLS-GO Installation Manual*. Installing a previous version of JTLS prior to installing JTLS-GO 6.1.1.0 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 that serves as a repository of engineering level data called “repository61”. Although not useful as a scenario, it does follow all of the database requirements for a scenario, and should be loaded into your PostgreSQL scenario table-space.
- The scenario “wespac61”, which is suitable for training and demonstrations.

1.3 INTERFACE COMPATIBILITY

1.3.1 Support Software

JTLS-GO 6.1.1.0 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 Server (ES) Edition Version 8.4, 64-bit architecture.

JTLS-GO 6.1 has been tested with the following versions of Linux 8:

RedHat Linux 8.4 - this operating system license must be purchased.

Oracle Linux 8.4 - 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 8.4 - a free version of Linux 8 that has not been approved by DISA for use by U.S. Government agencies.

- There are no restrictions on the operating system for client workstations, except that the operating system must have a Java-enabled web browser. JTLS-GO 6.1.1.0 has been tested on the following operating systems:

Red Hat Linux Enterprise Edition Version 7.9 and 8.4

CentOS Linux Version 7.9 and 8.4

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

- JTLS-GO 6.1.1.0 is delivered with the AdoptOpenJDK 1.8 Update 312 package, which is equivalent to the current version of OpenJDK. Using AdoptOpenJDK provides two benefits:
 - a. Only the JTLS-GO account on the system servers access this version of Java. An installation site can use the JTLS-GO servers for programs other than JTLS-GO without impacting the version of Java used by other programs.
 - b. Security releases of AdoptOpenJDK software are produced on the same schedule as the Oracle OpenJDK security release procedure. An organization can expect to receive a bug release version of JTLS-GO within two-weeks of a new Java 1.8 security release. As long as a user organization installs all of the JTLS-GO bug releases, JTLS-GO can guarantee that the latest Java security release is being used on the servers. JTLS-GO no longer depends on system administration for implementing proper Java security update procedures.
- JTLS-GO uses IcedTea to provide the Java Web Start capability that implements the web-enabled JTLS-GO functionality. JTLS-GO supports IcedTea version 1.8.4.
- JTLS-GO database tools require a certified PostgreSQL 11.13 database server and the full PostgreSQL installation. A containerized solution, that fulfills this specification, is provided as part of the JTLS-GO download. It is not necessary to use the delivered containerized solution, but it is the easiest method to meet the requirements of JTLS-GO 6.1.1.0. There are several alternative methods available for obtaining the PostgreSQL 11.13 software. Refer to Chapter 6 of the *JTLS-GO Installation Manual* for additional installation details.
- Windows software, X11R5 server, Motif 1.2 Library, Motif Window Manager: These items are included as part of the supported versions of Red Hat Linux ES.

- TCP/IP is required for inter-process communication between the JODA data server and all user interface programs. The version of TCP/IP included with the supported versions of Red Hat Linux ES is sufficient.
- The Perl script language is used by the JTLS-GO system and game setup scripts. The version of Perl included with the supported versions of Red Hat Linux ES is sufficient. The Perl program is typically located in the /usr/bin directory. If Perl is installed in a another location, a link should be created from the /usr/bin directory to this program.
- SIMSCRIPT III (SIMSCRIPT to C) translator/compiler: SIMSCRIPT is required for recompiling JTLS-GO code. It is not necessary to have a SIMSCRIPT compiler to execute JTLS-GO, because all JTLS-GO software executables are statically linked with the SIMSCRIPT libraries. The compiler is needed only if you are a U.S. Government organization that can obtain source code and plan to re-compile JTLS-GO SIMSCRIPT code. To obtain a SIMSCRIPT compiler, contact CACI Inc.
- ANSI C Compiler: It is not necessary to use a C compiler to execute JTLS-GO. This compiler is used only by U.S. Government organizations that can obtain source code and intend to re-compile any of the JTLS-GO component programs. The C Compiler version delivered with the supported versions of Red Hat Linux ES is sufficient.
- C++ Compiler: It is not necessary to use a C++ compiler to execute JTLS-GO. This compiler is used only by U.S. Government organizations that can obtain source code and intend to re-compile any of the JTLS-GO HLA component programs. The C++ Compiler version delivered with the supported versions of Red Hat Linux ES is sufficient.
- The JTLS-GO DDS application uses these open source libraries:

JFreeChart, licensed under a GNU Lesser General Public License (LGPL) 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 an operational Command, Control, Communication, Computer Information (C4I) systems may be able to install and use Google Earth and configure the KOI to provide a capability that resembles C4I for observing perception Force Side data.

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

1.3.2 JTLS-GO Cybersecurity Compliance

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

One of the DoD requirements is that the software must implement a methodology that ensures that the end user keep the software up-to-date and all security patches are properly installed. 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.

OpenJDK™ has not implement an expiration date. In order to fulfill this DoD Cybersecurity requirement, JTLS-GO has moved to AdoptOpenJDK, a full OpenJDK Java environment with licensing alternations allowing an application to deliver the software. The following procedure has been established and approved by the JS/J7 Cybersecurity branch:

- Within days of an Oracle Java security release, AdoptOpenJDK produces an equivalent version using infrastructure, build and test scripts to produce pre-built binaries of the OpenJDK class libraries. All AdoptOpenJDK binaries and scripts are open source licensed and available for free.
- Within two-weeks of the AdoptOpenJDK release, JTLS-GO provides a bug release version (JTLS-GO 6.1.n.0) including a full Version Description Document (VDD) for download to all authorized agencies. All DoD agencies using JTLS-GO will be in full compliance with this specific Cybersecurity mandate as long as they download and use the bug released versions when distributed.

Contact the U.S. Government Program Manager, Mr. James Blank (james.e.blank.civ@mail.mil) to obtain the completed Cybersecurity paperwork and a current Gate completion certificate.

1.3.3 JTLS-GO High Level Architecture Compliance

The JTLS-GO 6.1.1.0 release is fully High Level Architecture (HLA) compliant, and includes all the programs required to run JTLS-GO in an HLA mode. JTLS-GO currently belongs to one federation known as 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. CAE's GESI constructive simulation system is designed to run complex and comprehensive exercises from the company level up to division level. The GESI system is used to represent a virtual battlefield, including weapons, vehicles, aircrafts, ground forces and more.

Combining JTLS-GO and GESI brings together operational and tactical level constructive simulations to prepare commanders and staff to make timely, informed and intelligent decisions across the full spectrum of operations, including conventional combat, disaster relief, and operations other than war.

From the JTLS-GO perspective, all software needed to run GlobalSim is included in this delivery. JTLS-GO uses the Federation Object Model (FOM) located in the \$JGAME/data/hla directory, Federation testing of JTLS-GO with CAE's GESI model have been accomplished. Future plans include expanding the capabilities of the GlobalSim federation.

The HLA RTI (Run Time Infrastructure) executive program (rtiexec) recommended for use with this release is Pitch pRTI Evolved 4.4.2.0. However, this program is not included in the JTLS-GO 6.1.1.0 delivery. Users may obtain a full installation package of the RTI software from Pitch Corporation (www.pitch.se). For information about executing the HLA RTI Executive and other HLA-related software, refer to the appropriate HLA documentation and user guides.

1.4 DATABASE MODIFICATIONS

Significant database structure differences exist between the JTLS-GO 6.1 series of the software and the previous JTLS-GO 6.0 series database structure. There are no database format changes between JTLS-GO 6.1.1.0 and the initial JTLS-GO 6.1.0.0 version of the software.

To upgrade your JTLS-GO 6.0 scenario to JTLS-GO 6.1 compatibility, see instructions listed in the *JTLS-GO DDS User Guide*, Chapter 3.1.

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 6.1.0.0 formatted data from your PostgreSQL database server into the JTLS-GO 6.1.0.0 scenario American Standard Code for Information Interchange (ASCII) text files, you must execute the JSYMS program using the procedure outlined in the *JTLS-GO DDS User Guide*, Appendix B.11. This procedure will reorganize the structure of the <scenario_name>.gs and databases symbol.scf file.

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 6.1.1.0, 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. Some new symbols have been

created to meet end-user requirements. No previously existing symbols were deleted nor were any of the preexisting symbol names changed.

This means that the user can easily move in this new symbol set. Please follow the steps outlined in the *JTLS-GO DDS Users Guide*, Section B.13, Updating Scenario Symbol Set.

1.4.3 Standard Repository Changes

R&A has continued to improve and expand the unclassified data repository, which has been renamed to “repository61.” The DDS comparison and synchronization function can be used to determine if any of the changes delivered are of use to a JTLS-GO user organization.

1.5 INSTALLATION

The *JTLS-GO Installation Manual*, a Portable Document Format (pdf) file available for direct download, is part of this JTLS-GO delivery. It provides detailed instructions for installing the new version of JTLS-GO and the installation of PostgreSQL 11.11 required to operate JTLS-GO 6.1.1.0.

2.0 ENGINEERING CHANGE PROPOSALS

This chapter summarizes new model capabilities added to JTLS-GO 6.1.1.0 as a result of implementing authorized Engineering Change Proposals (ECPs).

2.1 JTLS-2021-15323 New SVP Checks For Federating With VBS4

Summary of Model Change Request

Verifying JTLS-GO scenario data required additional checks in the SVP when linked with the external model VBS4.

Design Summary

Several Errors and one Warning were added to the SVP. The new Errors (5014 and 5015) and Warning (6020) involved the verification of DIS Codes used by combat systems in units controlled by VBS4. They are only evaluated when the option to “Check Federation Data” is selected.

2.2 JTLS-2021-15354 Count Orders By Unit

Summary of Model Change Request

An exercise audience required a count of orders sent to individual units in the exercise. The Count Critical Order (CCO) program was modified to handle this function.

Design Summary

The Count Critical Order file was provided another capability to check for orders with the UNIT attribute set and provide a tally of orders sent to all units in the game. This capability is expected to be rarely used, so it was not added to the Javamenu. The capability can be accessed from the command line and a robust usage help was provided to aid in this. If this capability becomes routinely used, the decision to exclude it from the Javamenu will be revisited.

While integrating this capability, it was noted that when searching for a count of a specific order, which was an existing capability, that the order by player terminal summary listed all orders sent rather than the specific type of order sent. That error was fixed.

2.3 JTLS-2021-15360 Maintaining Link 16 Picture

Summary of Model Change Request

Several shortcomings in the Link 16 tracking environment were upgraded to create a more realistic tracking environment. A dropped mission track number would be reused fairly quickly by

an unrelated newly-acquired mission, causing confusion, and if a track was lost and quickly re-acquired, it was often assigned a different track number.

These shortcomings provided an unrealistic COP picture to the Training Audience, because in real world operations a Link 16 detector would cycle through all its track numbers before reusing them, and operators within an AOC or its sub-nodes would resolve temporarily lost tracks.

Design Summary

Objects now retain knowledge of their most recent Link 16-assigned track number and network for each side. If a track is reacquired by the same network, it can be assigned the same track number.

Track numbers from dropped tracks are now placed into an unassigned queue instead of releasing them entirely. If a detector runs out of track numbers, it will release the unassigned track numbers and start using them again. This is a capability that already existed for TBM tracks, but was not being used for Air Mission tracks.

If a detector ran out of track numbers, it would release all unassigned track numbers. The detector now only releases unassigned track numbers associated with its track blocks.

3.0 SOFTWARE TROUBLE REPORTS

Software Trouble Reports (STRs) describe software code errors that have been discovered by JTLS-GO users or developers and have been corrected.

3.1 JTLS-2021-15322 Auto-Generation of Order Name File

When running the Count Critical Order (CCO) program from the Javamenu, the user may request statistics on a single type of order. The Javamenu provides a list of all order names in this situation, which it has acquired from the \$JGAME/data/ordernameslist file. This file is auto-generated whenever an order is successfully verified.

The generation of this file should not be done when an order is verified, but rather when other order files, such as \$JGAME/data/orders.def, are generated. The generation of these other files also checks to ensure all orders have been verified.

The generation of the file ordernameslist was removed from its own script and merged into the create_ordersdef script, which already generated the orders.def and orders.rsp files. The output file name was changed from ordernameslist to orders.name to be consistent with the other generated files. The specific stylesheet that generates the file was also renamed to reflect this shortened output file name. The javamenu program was updated to read this new file name.

3.2 JTLS-2021-15325 Crash When Two Mines Hit Ship

The model crashed when two mines from the same minefield, but in different depth zones, hit and damaged a ship.

The model was attempting to keep track of the mines that hit the ship and attempted to file the same weapon into the set used to save this information. This caused the crash. The model now accounts for this possibility.

3.3 JTLS-2021-15328 WHIP Slide Image Graphics Re-Saved On Each Edit

The new image graphics in the WHIP's drawing tool were saved to the server whenever the graphic was activated for editing, making the editing of images a slow process.

Saving an image graphic to the server is now only performed when the image has not been previously added. This allows the user to quickly edit image renderings on a slide after the save. The user is allowed to overwrite the image whenever adding the same or similarly named image files.

3.4 JTLS-2021-15329 Unit Not Arriving With Zero Starting On Hand Supplies

If a unit has data in the Initial Supply Table, then it should enter the theater with the UT SC Starting On Hand amount. Units that did not have an initial support unit and had a value of zero

for the Starting On Hand amount for a specified supply category were still receiving the basic load amount for the supply category when arriving in theater.

When assigning supplies to an arriving Unit without an initial support unit, the model checked if the Starting On Hand amount was greater than zero, rather than greater than *or equal* to zero. If the Starting On Hand amount is zero, this is now reflected when the Unit enters the theater.

3.5 JTLS-2021-15330 More Color Choices For Text Graphic Editor Panel

There were only 12 colors that the user could select from when changing the text color or background colors of a text graphic.

The number of color options for graphics, including text graphics, has been increased.

3.6 JTLS-2021-15333 Fix Graphic Attributes Behavior

When saving a Font, Font Style, Font Size or Rotation Angle to the Text Default Attributes, the saved attributes did not display correctly on the map.

When attempting to revert a Font, Font Style, Font Size or Rotation Angle in a text graphic to their original attributes, the new attributes were applied, instead of being reverted.

When attempting to revert arrowhead attributes for Line Attributes or Line Arrow Attributes, the new attributes were applied, instead of being reverted.

The behavior of the graphics attributes was corrected.

3.7 JTLS-2021-15334 Unit Attempt To Interdict Convoy Crash

A Ground Combat Unit had a zero distance move. This can happen easily under a variety of circumstances, but this time, the unit saw an enemy convoy and was trying to determine if the convoy was close enough to interdict.

To accomplish this algorithm, the model determines if the current unit move, as defined using a great-circle line, crosses the convoy's next move. The algorithm to determine if two great-circle lines cross crashed because one of the "lines" was not a line, but was in fact a single point.

This situation has happened before under different circumstances, and in each circumstance the model was changed to check if the move was a zero distance move before calling the routine to determine if the two great-circle lines cross. The problem was fixed by implementing two different code updates:

1. The model now checks if the unit has moved any distance before calling the routine to determine if the unit movement path intercepts the convoy's movement path. This is the equivalent fix that we have made in other parts of the code.

2. The routine that determines if two great-circle lines cross was also improved. If it determines that one of the two lines is in fact not a line, but a point, the algorithm passes back that there is no intersection before attempting the computation that crashes the game.

3.8 JTLS-2021-15336 SSM Targets Missing From Unit Weapon Array

When a unit instance is read into the game as part of the data initialization process, it is automatically given its Prototype Owned targets. Once that is done, the Unit's Weapons array is created and filled indicating the type of weapons that it can fire by its owned SSM targets.

Later in the data initialization procedure, the non-prototype owned targets are read and, if the target is owned by a unit, the target is given to the unit. If this target is an SSM target, the Unit's Weapons Array was not updated to include the types of weapons that could be fired by the unit's non-prototype SSM targets.

The Unit's Weapons Array is now updated after a real SSM target is read from the initialization database and given to the unit.

3.9 JTLS-2021-15337 Crash When Mobility Mission Picks Up Unit

Some new changes to properly save the supplies processed through a port, or through a unit's inherent load/offload capability in the After Action Review database, caused a problem when a Mobility Air Mission was used to lift a unit or supplies. The code assumed that Air Missions were handled in exactly the same way as convoys. This was not true. Air Missions only have the data structure that holds current supplies when they are actually holding supplies. A convoy always has this data structure.

Code was added to consider this difference when saving this data to the AAR.

3.10 JTLS-2021-15338 Invalid Command Authority Object In JODA

The JODA was discovered to contain a Command Authority Tag that did not reference the unit for which command authority was being granted. The WHIP was throwing exceptions when these tags were encountered, causing the WHIP to slow down.

The problem was fixed in two ways:

1. The WHIP code was improved to catch this exception so it would simply ignore the bad Command Authority Tag when encountered.
2. The issue was caused by the model not deleting the Command Authority Tags for an HRU when it was destroyed by either kinetic warfare or by rejoining its parent unit. The model was changed to properly delete all the Command Authority Tags for the HRU prior to deleting the HRU from the JODA.

3.11 JTLS-2021-15339 Stop Replenish Supply Task Before Complete Crash

The user stopped an ongoing ship-to-ship replenishment task. The crash occurred when writing out the list of supplies that were transferred prior to postponing the replenishment task.

The error occurred because a variable for the Supply Categories that were transferred was referred to by two different names. The wrong version of the variable was being accessed.

3.12 JTLS-2021-15341 GIS Tool Improvements

User feedback has identified the following improvements for the GIS Tool:

- 1. Allow automatic saving of a newly created terrain layer.*
- 2. Do not save terrain layers automatically after deleting a terrain layer.*
- 3. Allow the user to select only the unprocessed terrain layers from the Calculate Grid Parameters dialog.*
- 4. Do not allow a terrain layer to be created if it crosses the International Date Line.*

The changes made to the GIS Tool are described below:

1. An auto-save check box has been added to the New Terrain Layer dialog. If the check box is selected, the tool will automatically save the new layer after its grids have been generated. If the check box is clear, the layer will not be saved until the user selects "Save..." from the File menu.
2. Automatic saving is turned off permanently. This will allow the user to recover the deleted layer in case it was inadvertently deleted.
3. A "Select Only Unprocessed Layers" check box has been added to the dialog. When this check box is selected, a check mark will appear alongside each red terrain layer (those that have not had their grid parameters calculated). Deselecting the check box will remove the check marks alongside the red terrain layers.
4. The user creates a terrain layer by specifying the lower left and upper right corners of the layer. If the specified corners are such that the resulting layer will cross the date line, an error message will pop up in a message dialog, and the user will have to input a new pair of corners.

3.13 JTLS-2021-15343 Air Mission Weapon Fire Against Location

An air mission was ordered to fire weapons against a location. The mission flew to the targeted location but no weapons were expended. A alert was generated which stated that the mission did not have ROE to fire.

In the model, the process to fire weapons performed a check of the ROE. This check always resulted in a Hold Fire state when firing against a location. New code was added for the case of weapon fire against a location. To fire at a location, the mission must have permission to fire against at least one other Force Side.

3.14 JTLS-2021-15344 ELS Crash During New JODA Download

The ELS crashed when a new JODA download was requested by the controller.

When a new download was received, the ELS attempted to clean up the current data. This clean up is needed in order to recreate all objects from the download. The crash occurred because one unit had attached to another unit, but the ELS did not properly assign the hierarchy of units. A change was made to maintain a consistent hierarchy of units.

3.15 JTLS-2021-15345 Cannot Select Missile In Controller Kill Order

The Controller Kill Aircraft order has an option to kill missiles. When a Controller selected this option on the order panel, no missiles were listed in the drop down menu. This prevented the Controller from killing any missiles.

The order panel was using the wrong object type and name attribute to fill the possible missiles. A change was made to the order definition file to fix this problem.

3.16 JTLS-2021-15346 Crash Checking HRU Boat Resources At Ground Parent

If an HRU requires a Small Boat and that Small Boat was represented as one of the HRU's Combat Systems, but the Parent Unit did not have any of those combat systems, the model crashed.

The crash was due to accessing an incorrect variable. The problem was that the model attempted to allow for the Parent Unit to provide a different combat system to the HRU if the different combat system represented the same type of Small Boat. This was determined to be incorrect and only the CS desired by the HRU should be given to the HRU. This underlying problem was also fixed.

3.17 JTLS-2021-15348 Ship Unit Prototype OPM Error

The OPM page for Ship Unit Prototypes had an error in the table of combat system TOEs. The name of the combat system included bad data for the CS index.

In JTLS-GO 6.1.0.0, combat systems were converted from permanent entities to temporary entities. This meant that CS were represented as pointers instead of integers in the model. There was no longer an index associated with each CS, and these data were removed from the OPM page for SUPs.

3.18 JTLS-2021-15349 No Command Authority For Tunnel Sheltering Order

A Player did not have to have command authority over the Unit or HRU to be sheltered.

The WHIP will not allow the Player to select a unit for which it has no command authority. This has been corrected.

3.19 JTLS-2021-15350 LC2IS Country Codes In NATO-DiGraph

The JTLS-GO database was built with NATO-DiGraph Country Codes. When a LC2IS SIF, an importable XML file, is produced by the LC2IS Message Service (LC2MS), it includes the NATO-DiGraphs instead of the NATO-TriGraphs expected by the SIF.

A secondary error was discovered in the LC2MS interface when a few units became Wiped Out, which sets their Higher HQ unit to None. The interface is a command hierarchy tree by Force Side. The Force Side commander is identified as a unit on the Force Side without a Higher HQ, and the tree is constructed downwards from them. One of the newly Wiped Out units was picked up as the Force Side commander, and because it did not have any subordinates that was the only node in the tree.

The LC2MS already had code to translate the database Country Code format to NATO-TriGraphs, but there was an error in the function and it was always returning the input value - in this case NATO-DiGraph. The code was fixed to return the translated value rather than the input value.

Code was also added when searching for the Force Side commander to disregard those units that are Wiped Out. Within JTLS-GO, if the Force Side commander is Wiped Out command will transfer to another unit, so there is no issue with that situation.

3.20 JTLS-2021-15352 MDP E-Mail Delivery Failure

Messages that were designated to be forwarded to an e-mail account were not being delivered. The configuration specifies a header file to be used when the messages are sent as attachments. This header file may reside in either the \$JGAME/data/mdp directory or the \$JGAME/data_site/mdp directory.

When attempting to find the file a stringstream is used to construct the file name and check for it in the first directory. If that fails the stringstream was cleared and a filename for the second directory was constructed and checked. If that failed the stringstream was cleared again and the default filename in the first directory was constructed and sent to the Sendmail module to use as the body of the message. However, when sent with the Sendmail module, messages were being rejected because the module insisted there was no body to the message.

The error was using the clear() method to reset the stringstream between constructing file names. The clear() method does not actually reset the text but clears any I/O error bits. As a result a file that resides in the \$JGAME/data_site/mdp directory would first construct the

filename and path with \$JGAME/data/mdp, and when the file was not found there would append on \$JGAME/data_site/mdp to that.

This, of course, was an invalid directory structure so the file was not found. Then the default path and file were added onto the stringstream and passed to the Sendmail module as the body of the message. Since this path now had 3 filenames all appended together the file was not found and no body was added to the message. The correct method to reset the stringstream is now used and the problem is solved.

3.21 JTLS-2021-15356 Crash Turning Off National ELINT Coverage

The Controller switched the National Asset for ELINT coverage to NONE while an air mission with an emitting sensor was airborne. The action of switching to NONE caused all emitter objects owned by targets and air missions to be destroyed on the JODA. However, the code was attempting to turn off the Air Mission, not its sensor, as a JODA emitter object. This resulted in a crash for accessing the wrong type of entity.

The code was modified to pass the proper entity, the emitting object rather than the air mission, to the routine that deleted the object.

3.22 JTLS-2021-15358 Dropped Tracks On Ship Detections

JTLS-GO Naval tracks were being dropped on the OTH-Gold Message Service (OTHGMS) periodically for incomplete information. A new track would then be recreated when complete information was again available in the OTHGMS. This was happening even though the information on the track was still valid within the game. The dropping and recreation of the tracks was causing historical information about the tracks to be lost, and was causing the NORCCIS operator extra work to clean the system.

The incomplete status seen in the OTHGMS interface was tracked down to the updating of perceived unit information. As an update is processed each part of the update, such as vector, location, posture, or strength, are checked against the unit to determine if it is the most recent information. This check is performed because the collection and processing of the intelligence occur at different times, and for some sensors happen more quickly than for other sensors.

If the update location was more recent than what was already known within the update process, the location data was stored on local variables and these were used to update the JODA location. This was followed by checking if the COP location was more recent and, if so, the local variables were again used in a JODA update for the COP location. This was done without assigning the local variables. Thus, if a COP sensor made a detection, but before that detection was processed a non-COP capable sensor made and processed a detection that involved location, the processing of the COP capable sensor would set the COP location to ON,OE.

For the OTHGMS the ON,OE location is considered to be a lack of a location, resulting in incomplete track information and the dropping of the track. This was solved by assigning the

local variables before they are used in the JODA update. This same situation was found in the processing of HRU and Air Mission detections and was fixed.

3.23 JTLS-2021-15362 OPM Regeneration Error Reading Checkpoints

A checkpoint existed in the .tim file, which tracks the time of all checkpoints, for which the reason exceeded 82 characters on the line. The reason code was likely the result of a hand edit of the file. But the SIP crashed when it encountered the line because it had opened the file with a recordsize of 80 characters.

The open statement was modified to specify a recordsize of 6000, which is a standard recordsize used in JTLS-GO when dealing with checkpoints. The CEP code, which did not have a recordsize specified and would have defaulted to 132 characters, was also modified to use a recordsize of 6000.

3.24 JTLS-2021-15364 Allow Fire Missile Range Worldwide

A player attempted to use the Fire Missile order to shoot a long distance on a specific range and bearing but was limited to 1500 KM by the WHIP. This is an insufficient distance for many missiles, particularly TBMs. Whether or not a TBM would be shot on a range/bearing is considered immaterial to whether the model should support the capability.

The restriction resides in the XML file that defines the order and was switched from 1500 KM to 40,075 KM, allowing a range/bearing to hit anywhere on the world. It was also noted that the Manage Pending Fire order, which allows the attributes of a fire mission to be changed before firing, was also limiting the range to 1500 KM, and was changed to 40,075 KM.

3.25 JTLS-2021-15366 Link 16 Source Unit Not On Network Crash

The model crashed when a Link 16-capable sensor owned by a unit with a JU number attempted to access the Link 16 network of the unit, which was not assigned to a network. The situation happened as a Controller was assigning JU numbers, track blocks, and network assignments to units and the detection occurred before the final step.

The code was modified to ignore units that are not on a Link 16 network when determining a JU number to assign to the detection.

3.26 JTLS-2021-15368 XMS Crash Accessing Message Notification

The XMS accessed a message notification with a message number greater than the maximum (9999). The XMS attempted to determine whether a bit mask was set for the message, but the bit number exceeded the number of possible messages. This caused an abort and the XMS crashed.

The XMS was modified to protect against message numbers beyond the maximum, and report an appropriate error in the log whenever a notification is processed having a number that is too large.

The cause of receiving a message with a large message type number is still under investigation.

3.27 JTLS-2021-15371 Map Networks/Routes Displayed In Same Color

Unit ground routes and rail networks were both set to be displayed in brown on the map . It made it difficult to differentiate them on the map when a rail car unit's route that went over a railroad were both set to be on.

The rail networks' map display color was changed to be "darkgray" in the colors file to make them visually distinguishable from ground routes.

3.28 JTLS-2021-15372 JSAT GUI Twenty-Four Satellite Limit

The twenty-four satellite limit on JSAT .tle files no longer needs to be applied.

The satellite count check was removed from the code.

3.29 JTLS-2021-15377 IIR Generation Cannot Be Stopped

A Recce air mission was equipped with two nearly identical real time imagery sensors, which caused two Imagery Interpretation Reports (IIRs) to be generated for the same Directed Search Area (DSA) at approximately the same time. An Emission Control order was sent to turn off one of the sensors. The Mission Emitter Load Screen IMT and Player message confirmed the sensor was turned off. However, two IIRs continued to be generated from the mission.

In this particular situation, the scenario database incorrectly included a day and night version of the same sensor in the aircraft's default load, with the only difference being the night degradation factor. Only one of the sensors were necessary. The model was correct to generate a separate IIR for each sensor, albeit with slightly different detection results. The Repository database was reviewed and found to be correct: The aircraft type included only one sensor in the default load.

However, a model error did not exclude sensors that were turned off when more than one sensor was present on the mission. To correct this problem, new code was added to skip the intelligence Existing Update if the sensor associated with the update data event is turned off.

3.30 JTLS-2021-15378 Targets With Multiple Elements

A Target designed to consist of a single element has multiple elements.

The Target types Bridge, Facility, Jammer, Runway, Sensor Site and Communications Target with type COMMAND should only have a single target Element. Due to the design of the TUP/SUP POT target tables, it is possible for the user to enter values greater than 1 for these targets.

The new SVP Error 239 has been added to identify the situation for correction.

3.31 JTLS-2021-15379 Personnel Mean Time To Repair Large

The mean time to repair for some personnel combat systems is larger than the faction maximum treatment time.

The time it takes for personnel to recover should be less than the unit maximum treatment time at the unit as defined in the faction data. This will cause personnel to be considered WIA and then evacuated before they have used their full maintenance period.

The new SVP Warning 1162 has been added to identify the situation for correction.

3.32 JTLS-2021-15380 ATO Parser Not Adding Route Locations

When a mission includes polygon orbits at the beginning or the ending of its complete list of tasks, the ATO Parser should include point locations for the mission to fly just prior to and just after the polygon, which are the center point of the polygons. This is necessary for timing to correspond more closely with the ICC system.

The Parser was modified to include an additional task point at the start of its tasking whenever the first task is a polygon orbit. The Parser was also modified to include an additional egress point just after its tasking and whenever the last task is a polygon orbit. This allows for better tracking of missions generated by the ICC.

3.33 JTLS-2021-15383 SVP Crash After 5.1 To 6.0 Database Conversion

The customer had a JTLS 4.1 scenario database that needed to be converted to a JTLS-GO 6.0 database. When the conversion process generated the intermediate 5.1 database an SVP check was run. All SVP errors were fixed before converting from 5.1 to 6.0 (although numerous warnings were not fixed). The SVP crashed when the 6.0 database was checked.

This problem was resolved in JTLS-GO 6.0.5.0 SIP (STR JTLS-2021-15382). A check of the 6.1 SIP was conducted to see if similar coding issues also existed in the routine CHECK DISCODE USAGE. The only issue found was the direct comparison made between TG OWNING FEDERATE and the define-to-means .JTLS.ONLY. The code was corrected.

3.34 JTLS-2021-15385 ATO Parser Includes Mission Location Tasks

The ICC System often includes potential locations for a mission tasking in the ATO message. These locations are in addition to the planned tasking for the mission but are not included in the routing record for the mission. The Parser is creating the WHIP slides for each of the potential

locations but is also including them in the tasking in the order for the mission. The Parser should only be including the task locations listed in the routing for the mission.

The Parser was modified to withhold any locations listed in the mission description that are not included in the mission's routing record. The Parser will continue to create the WHIP slides however. In this way, just prior to sending the order, the operator may decide to plot the slide on the WHIP and use the slide geometry to select new points and modify the tasking for the mission.

3.35 JTLS-2021-15386 ATO Translator Mobility Mission Orders

The ATO Translator creates Mobility mission orders for ATO missions which represent movement of supplies for the exercise. However, the Mobility orders created by the ATOT have air drops of supplies at each air transit point along the tasking. This causes the mission to run out of supplies before completing the mission.

The ATOT was modified so it will not generate transit points with supply drops.

3.36 JTLS-2021-15389 WHIP Command Authority Check Failure

On specific WHIPs, the context order menu was failing to appear for all units owned by the WHIP's force side. The WHIP's unique ID was being stored in a data type on the WHIP that did not match the data type in the rest of JTLS-GO. This caused the value to be misread, which caused command authority checks to fail.

The WHIP now holds its unique ID in a matching data type so that the value is used consistently with the rest of JTLS-GO.

3.37 JTLS-2021-15393 OTHGold Module Multiple Error Popups

The OTHGold module queries the service regularly for updates. If the update times out, the module warned the user but the update timer was not canceled. This caused multiple warnings to be generated for the same error.

The OTHGold module now stops the refresh timer and disconnects from the service, as well as warns the user, when a server request times out.

3.38 JTLS-2021-15395 WHIP Unit Combat Map Filter Does Not Set

The WHIP's map filter settings includes the ability to filter units based on their combat status. This filter failed to be set for specific sides and would only work when changing the setting for all sides.

The unit combat filter has been fixed so that individual side settings can be applied.

3.39 JTLS-2021-15396 ICP Classification Color Text Misleading

The ICP allows users to change the color of the WHIP's classification banner. The demonstration text for this setting read as UNCLASSIFIED, which may be confusing because it may not match the current classification of the scenario.

The WHIP classification banner color setting's demonstration text has been changed to read as CLASSIFICATION.

3.40 JTLS-2021-15397 No Message For Ground Unit Out Of Fuel

When a ground unit conducting a move ran out of fuel, there was no message to inform the Player if the unit failed to submit a requisition for fuel.

A message is now generated informing the Player of the ground unit's current activity trouble.

3.41 JTLS-2021-15400 Link 16 Redundant Feed Toggle Confirmation

When a user toggles the Link 16 Feed, a dialog appears to confirm the user's selection. After confirmation, a second dialog appears notifying the user that the feed toggle was successful. This second dialog is redundant and bothersome to some users as it displays information that is already available to the user.

The Link 16 Module's feed toggle notification dialog has been removed.

3.42 JTLS-2021-15401 Player Periodic Reports Reformatted

After adding in additional Fuel Types, the Player's Airbase and Squadron periodic report used an index when referring to an airbase. In the aviation fuel data section, there was a line printed out for every fuel type the airbase owned containing the airbase's index number and the fuel type. It was not documented clearly why the index was being used and the second issue was that we did not need to print multiple lines using the same airbase.

The report has been condensed. For each section of the report, the airbase name is printed and then any data associated with that airbase is listed underneath.

4.0 REMAINING ERRORS

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

As JTLS-GO 6.1.0.0 represents a major release of new functionality, all outstanding errors have been reviewed. If the error could not be reproduced, it was considered obsolete and no longer relevant to JTLS-GO. These errors have been removed from consideration for correction at this time. This chapter contains the remaining errors that could be reproduced. As development team assets become available, these remaining known issues will be corrected.

In future maintenance releases, newly uncovered outstanding errors related to JTLS-GO will be listed in this chapter, along with information 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.

APPENDIX A. ABBREVIATIONS AND ACRONYMS

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

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

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

DPICM	Dual Purpose Improved Conventional Munitions
DS	Direct Support
DSA	Directed Search Area
DTG	Date Time Group
EC	Electronic Combat
ECM	Electronic Counter Measure
ECP	Engineering Change Proposal
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
JDA	Japan Defense Agency
JDPI	Joint Desired Point of Impact (formerly DMPI: Desired Mean Point of Impact)
JDS	JTLS Data System
JDSP	JTLS Data System Protocol
JEDI	JODA Entity Data Identifier
JMCIS	Joint Maritime Combat Information System
JMEM	Joint Munitions Effectiveness Manuals
JODA	JTLS Object Distribution Authority
JOI	JTLS Operational Interface
JPL	Jet Propulsion Laboratory
JRSG	Joint Rapid Scenario Generation (formerly JIDPS: Joint Integrated Database Preparation System)
JSDF	Japanese Self-Defense Force
JTLS	Joint Theater Level Simulation
JTLS-GO	Joint Theater Level Simulation - Global Operations
JTOI	JTLS Transaction Operational Interface
JXSR	JTLS XML Serial Repository
KIA	Killed In Action
KM	Kilometer
KNOTS	Nautical miles per hour
LA	Lethal Area
LAN	Local Area Network
LAT	Latitude
LB	Login Build (JTLS order type)
LDAP	Lightweight Directory Access Protocol
LDT	Lanchester coefficient Development Tool
LOG	Logistics

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

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

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

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

APPENDIX B. VERSION 6.1.0.0 DATABASE CHANGES

No changes were made to the JTLS-GO 6.1 database.

APPENDIX C. VERSION 6.1.0.0 REPOSITORY CHANGES

The following changes were made to the JTLS-GO 6.1 repository.