JTLS-GO

Version Description Document

March 2026

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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 6.3.9.0)

[Blank Page]

ABSTRACT

The Joint Theater Level Simulation - Global Operations (JTLS-GO®) is an interactive, computer-based, multi-sided wargaming system that models air, land, naval, Special Forces, 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 6.3.9.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 6.3.9.0 is a Maintenance release of the JTLS-GO 6.3 series that includes fixes to uncovered software issues, an updated repository of standard data, and a demonstration scenario based in the western Pacific. This release includes one new model enhancement, described 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 Director of Development
Valkyrie Enterprises LLC
1200 Piedmont Ave, Suite A
Pacific Grove, California 93950-5157
United States
jtlsgo@valkyrie.com

[Blank Page]

TABLE of CONTENTS

1.0 INTRODUCTION

1.1 SCOPE

This *JTLS-GO Version Description Document* (VDD) describes Version 6.3.9.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO®) software suite. JTLS-GO 6.3.9.0 is a Maintenance delivery for the JTLS-GO 6.3 series of releases.

JTLS-GO 6.3.9.0 includes the entire JTLS-GO suite of software, a repository of engineering level data, and a realistic demonstration scenario, called “wespac63”, based on the Western Pacific theater of operations. There were no database format modifications between this Maintenance release and the original JTLS-GO 6.3.0.0 version. Appendix B of the original *JTLS-GO 6.3.0.0 Version Description Document* summarized the database format changes made between the JTLS-GO 6.2 series and this JTLS-GO 6.3 series of the software system.

Detailed description of the new Engineering Change Proposals (ECPs) is provided in Chapter 2.0. Chapter 3.0 summarizes the Software Trouble Reports (STRs) that have been corrected and are delivered with this version of JTLS-GO 6.3.

JTLS-GO 6.3.9.0 executes on the Red Hat Enterprise Linux Version 9.4 and Oracle Linux 9.4 64-bit operating systems. The Web-Hosted Interface Program (WHIP®) user workstation interface can be executed on any 64-bit 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 6.3.8.0)
- *JTLS-GO Air Services User Guide (*JTLS-GO Document 02, Version 6.3.4.0)
- *JTLS-GO Configuration Management Plan* (JTLS-GO Document 03, Version 6.3.0.0)
- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 6.3.8.0)
- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 6.3.8.0)
- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 6.3.6.0)

- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 6.3.7.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 6.3.8.0)
- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 6.3.8.0)
- *JTLS-GO Repository Description* (JTLS-GO Document 14, Version 6.3.4.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 6.3.7.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 6.3.4.0)
- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 6.3.0.0)
- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 6.3.5.0)
- *JTLS-GO DoD Architecture Framework* (JTLS-GO Document 22, Version 6.3.0.0)

1.2.3 Updated Documents

- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 6.3.9.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 6.3.9.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 6.3.9.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 6.3.9.0)

1.2.4 New Documents

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

1.2.5 Delivered Software Components

JTLS-GO 6.3.9.0 may be delivered either on a CD or as a set of compressed TAR files to be downloaded. Either method includes the complete suite of software executable code and command procedures. The following software components are included with this release:

- Combat Events Program (CEP)
- Geo-Spatial Service (GSS)
- 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)apache –
- JTLS Object Distribution Authority (JODA)

  The current JODA build number is 215.
- Web Services Manager (WSM)
- Web-Hosted Interface Program (WHIP) and its component programs:

  Apache Server (APACHE) version 2.4.62

  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)

  JTOI_ICC302 - Used to feed NATO Integrated Command Control (ICC) Version 3.0.2 system.

  JTOI_ICC320 - Used to feed NATO ICC Version 3.2.0 system.

  JTOI_ICC340 - Used to feed NATO ICC Version 3.2.0 system.

  JTOI_ICC350 - Used to feed NATO ICC Version 3.2.0 system.

  JTOI_NECCCIS - Used to feed NATO Northern European Command, Command Control Information System (NECCCIS).

  JTOI_TBMCS - Used to feed US Theater Battle Management Core System (TBMCS).

- JTLS Interface Network Navigator (JINN)
- JTLS Order of Battle Editor (JOBE)
- 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.

  VCP62 - Converts a JTLS-GO 6.1 database to a JTLS-GO 6.2 formatted database.

  VCP63 - Converts a JTLS-GO 6.2 database to a JTLS-GO 6.3 formatted database.

Instructions for installing JTLS-GO 6.3.9.0 are provided in the *JTLS-GO Installation Manual*. Compared to the JTLS-GO 6.2 series, the JTLS-GO 6.3 series uses a significantly different version of PostgreSQL and the Linux operating system. If an organization has not already upgraded to the JTLS-GO 6.3 version, ensure special attention is given to following the documented operating system and PostgreSQL installation procedures. 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 “repository63”. 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 “wespac63”, which is suitable for training and demonstrations. The scenario has been updated to use the newest version of engineering level data.

1.3 INTERFACE COMPATIBILITY

1.3.1 Support Software

JTLS-GO 6.3.9.0 requires the following versions of support software, including operating systems, compilers, scripting utilities, database tools, transfer protocols, and display managers.

- Operating system for the model: Red Hat Linux Enterprise Server (ES) Edition Version 9.4, 64-bit architecture.

  JTLS-GO 6.3 has been tested with the following versions of Linux 9:

  RedHat Linux 9.4 - this operating system license must be purchased.

  Oracle Linux 9.4 - 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 Defense Information System Agency (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 be a 64-bit architecture with a Java-enabled web browser. JTLS-GO 6.3.9.0 has been tested on the following operating systems:

  Red Hat Linux Enterprise Edition Version 9.4

  Oracle Linux 9.4

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

- JTLS-GO 6.3.9.0 is delivered with the Adoptium project Temurin Java Development Kit (JDK) 1.8 Update 482 package. Both the ICP and DCP have the option for an organization to increase the maximum memory heap for the WHIP and DDSC. For large scenarios and databases, an organization should consider increasing the maximum heap size.
- 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 6.3.9.0 is being delivered PostgreSQL 15.16 that has been compiled under Linux 9.4 and is bundled with the tar files for this release. This version of PostgreSQL is the latest security patch release of PostgreSQL and is being delivered in accordance with US Department of Defense Cybersecurity requirements. It is not necessary to use the delivered solution, but it is the easiest method to meet the requirements of JTLS-GO 6.3.9.0. There are several alternative methods available for obtaining the PostgreSQL 15.16 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.

- 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.

- 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-2.0HLA 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.

- JTLS-GO 6.3.9.0, using the JODA service, allows connections and data exchange with customer client programs. The customer client programs are linked with a set of JTLS-GO-provided API libraries that permit a TCP/IP connection between the JODA and the client program. These API libraries, called JDSP libraries, are built for Linux and Windows and allow customers to built client applications on either of these operating systems. Below are the development environments under which each of the JDSP libraries are built:

  RedHat Linux 9.4 using gcc (GCC) 11.4.1 20231218 (Red Hat 11.4.1-3.0.1)

  Windows 10 using Visual Studio 2017 version 15.9.60 and Visual C++ 00369.60000.00001-AA807

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 keeps 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. The concept of an expiration date has been removed from the DoD requirement, but the concept of always using the latest version of third-party software remains a strong component of DoD Cybersecurity requirements.

The following procedure has been established and approved by the JS/J7 Cybersecurity branch to meet the software update requirement:

- 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 6.3.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.

The JTLS-GO 6.3 series has been issued an Exit Gate letter and certification from the JS/J7 Cybersecurity branch. Please contact the U.S. Government Program Manager, Mr. Douglas Failor (douglas.l.failor.civ@mail.mil) to obtain the completed Cybersecurity paperwork.

1.3.3 JTLS-GO High Level Architecture Compliance

The JTLS-GO 6.3.9.0 release is fully High Level Architecture (HLA) compliant, and includes all the programs required to run JTLS-GO in an HLA mode. JTLS-GO 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. Previous federation testing of JTLS-GO with CAE's GESI model has been accomplished using this FOM. The reader should note that the JTLS-GO Development Team, to date, has not been able to test this federation using this maintenance release of JTLS-GO. If there is interest in running this federation, please contact the JTLS-GO Help desk at jtlsgo@valkyrie.com.

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 6.3.9.0 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 6.3 series and the previous JTLS-GO 6.2 series database structure. Appendix B of the *JTLS-GO 6.3.0.0 Version Description*

Document has a summary of all database changes. To upgrade your JTLS-GO 6.2 scenario to JTLS-GO 6.3 compatibility, see instructions listed in the *JTLS-GO DDS User Guide*, Chapter 3.1.

Due to changes made in support of STR JTLS-2025-17088 Missile Capability Does Not Have Torpedo, the record has been added to the JTLS-GO 6.3.5.0 Configuration Managed tw_missile_capability lookup table. This is not a database format change, but to make this new record available for the database building effort, users should **unload** their JTLS-GO 6.3 series scenarios prior to installing JTLS-GO 6.3.5.0, and then **load** them following installation. Please refer to the *JTLS-GO 6.3.5.0 Version Description Document* for more details.

Upgrading earlier JTLS-GO versions to the JTLS-GO 6.3 format will automatically correct the problem. No additional action is required.

Due to the mandated Cyber-Security improvements made to the Database Configuration Program (DCP) and the Database Development System Client (DDSC) for JTLS-GO 6.3.4.0, each PostgreSQL database scenario needs to be reconfigured. If you skipped moving in JTLS-GO 6.3.4.0, please execute the procedure in the following note for each JTLS-GO 6.3 scenario:

- Go to the $JTLSHOME/data/scenario/<scenario_name>/config directory.
- Delete the entire directory and its subdirectories.
- Start the new DCP and create the needed DDSCs.
- Each organization can decide whether they want to use the new encrypted password capability and the new DDSC privilege capability. Refer the *JTLS-GO Database Development User Guide* for information concerning the new DCP interface.

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 6.3.0.0 formatted data from your PostgreSQL database server into the JTLS-GO 6.3.0.0 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

Users are reminded that organizations should not make any modifications to the Default Symbol Set delivered with JTLS-GO. End-user organizations are free to use the Default Symbol Set in their

scenarios and alter the scenario symbol set to their scenarios to meet specific organizational needs.

The JTLS-GO Default Symbol Set has not been changed since August 2020. At that time, some new symbols were created to meet end-user requirements. No previously existing symbols were deleted nor were any of the preexisting symbol names changed. If any existing scenarios have not moved to this new Default Symbol Set, 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

The JTLS-GO Database Team has continued to improve and expand the unclassified data repository, which has been renamed to “repository63”. 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. Specifically, significant effort has been started to represent additional Combat Systems to more closely match the Combat Systems recognized by the Joint Live Virtual Constructive (JLVC) federation of models. This effort is expected to be an ongoing effort for the next three to five months.

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 15.13.

Reminder: Ensure existing JTLS-GO Version 6.3 databases held in the PostgreSQL table-space are unloaded prior to installation and reloaded after installation, as explained in Section 1.4.

2.0 ENGINEERING CHANGE PROPOSALS

No Engineering Change Proposals (ECP) were delivered with this release.

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-2025-17502 Improve Personnel Portion Of Logistics Rollup Report

There are two sections in the Logistics Rollup Report that address the current personnel status of the specified Unit and its subordinates. The information in the two sections are not consistent, and it is difficult to provide information to support exercise audience requests.

The two sections of the report are:

- The total Personnel supplies that exist in the specified Unit and its subordinates. This portion of the report is needed to indicate what Wounded In Action (WIA) Personnel and Killed in Action (KIA) remains still need to be evacuated from the Unit.
- The total personnel Combat Systems that exist in the specified Unit and its subordinates. This portion of the report is needed to provide the exercise audience with a summary of the total Personnel still within the specified Unit and its subordinates. This includes the cumulative WIA and KIA for each type of Personnel Combat System.

These two sections do not match, and it is not easy to provide the exercise audience with a consistent explanation of the personnel status of the specified Unit and its subordinates.

Several changes were made to help with this issue.

Personnel Supply Section changes:

- One error was found in this section of the report - WIA and KIA supplies were passed using an insufficiently-detailed format. Thus, when only three Personnel Combat Systems became WIA supplies, the model indicated that there were 0 WIA supplies. The format was changed to output a format so the value 0.3 was output to the message, and is now displayed properly.
- Users did not know how to properly interpret this section of the report. There are two important concepts a user needs to keep in mind when reviewing the personnel supply portion of the report:

  The Personnel Combat System section of the report contains the cumulative number of WIA and KIA for the various Personnel Combat Systems. If some of those WIAs have been evacuated or repatriated, then the Personnel Supply Section of the report will not match. This is an intended and the primary reason that the Logistics report includes a Personnel Combat System summary and a Personnel Supply Summary Section.

When the model determines a person is a KIA, the remains for the KIA may not be recovered. This is a database parameter CSP PROB REMAINS RECOVERED. Thus, the Personnel Combat System section of the report may show 6 KIAs but the Supply section of the report only indicates that 5 Personnel remains are waiting to be repatriated.

Personnel Combat System Section:

Users reported that they need more information in this section to easily answer exercise audience questions: This portion of the table now includes the following information for each Personnel Combat System that exists in the specified Unit and its subordinates:

- Total TOE - this is new information.
- Total Operational - this existed in the previous version of the report.
- Maint / Hospital - this existed in the previous version of the report, but the exercise audience did not understand the label. The label has been changed to indicate the personnel are either Sick or Injured.
- WIA - this existed in the previous version of the report.
- KIA - this existed in the previous version of the report.
- Total - this is new information. The report now provides the total number of Personnel that are Operational, Injured, WIA and KIA for each Personnel Combat System.
- A footnote indicator (*) if the Total column is greater than the TOE. The footnote in the report indicates that this could happen if replacement Personnel had been sent to the unit.

Finally, column totals for this Personnel Combat System table were added to the report. Thus, the user now has immediate access to the Total Personnel TOE, Operational, Injured, WIA and KIAs for the specified Unit and its subordinates.

3.2 JTLS-2025-17504 Reformat Spreadsheet Program Double Default Fields

When running the Reformat Spreadsheet Program (RSP) with a control file that specifies default fields, if an error exists in a piece of data, the user is given the opportunity to not generate an order file. If the user chooses not to, fixes the bad data, and then processes the file again in the same RSP session, this results in duplicate default fields on the orders. The resulting order file will fail the OMA checks and will need to be generated again with a fresh RSP session.

Orders that passed the first generation attempt, but were not written out, were left in the set of orders. On the second pass, after reading in the corrected orders and adding them to the set of orders, the default fields were added to all orders in the set of orders. This resulted in those

orders that passed verification on the first attempt would appear twice, once with duplicate default fields, and once correctly.

The code was changed so that when the user chooses not to generate the order file all orders are removed from the set of orders. Thus, on the second pass through the RSP, the program is starting with a clean slate of orders.

3.3 JTLS-2026-17511 DDS/WHIP Obsolete Email Address

The DDS and WHIP still displayed the old JTLS-GO help desk email address (jtlsdev@rolands.com) in the stack traces and various parts of the application.

All of the help desk email addresses were updated to the new help desk email address (jtlsgo@valkyrie.com).

3.4 JTLS-2026-17514 OVT Crashes When Scenario Argument Provided

The Order Verification Tool (OVT) allows a number of optional arguments to be specified on the command line. One of these allows the specification of a scenario (–scenario <scenario>), which allows the user to bypass the scenario selection dialog. However, using the argument would crash the OVT.

scenario could also be obtained by specifying a configuration file withe the -cf argument. When this argument was used, the OVT failed to come up without producing an error message.

Both problems were identified and fixed in the source code.

Additionally, a shortcut was added to the script that starts the OVT such that if a single argument is specified it should be treated as a scenario. If the scenario is valid the argument is passed to the program as the proper "–scenario <scenario>" argument. This aligns the starting of the OVT with other JTLS-GO programs, such as the Web Services Manager (WSM).

While fixing this issue, a problem was found where the Apache tab was not providing information about the Apache server. This was traced to the OVT making HTTP requests (instead of HTTPS) for the required information. This issue was also fixed.

3.5 JTLS-2026-17516 Facility Target Partial Damage Inconsistency

When a JDPI Target is attacked and damaged, any linked Facility Target is also damaged to the same amount, which can vary anywhere from 0% to 100%.

However, when a Controller Change Target order is submitted to damage a Facility Target, the model allows only 0% (damaged) or 100% (undamaged) as the percent capable, with no variation between those values. It was impossible to damage a Facility Target to 50%, for example.

To remove this inconsistency, the logic was changed to permit Facility Targets to accept any percent capable value between 0% and 100%, as specified in the Controller Change Target order.

3.6 JTLS-2026-17519 Map Shape File Customization UI Improvements

Selecting a custom map shape file from the Interface Configuration Program (ICP) or Database Configuration Program (DCP) displays a dialog that allows the user to choose a new map shape file or use the default one. However, the user can only choose files from the $JGAME/data/maps/alternate directory, and the UI for selecting the new file is confusing.

The following changes have been implemented:

- Allow the user to choose files from both the $JGAME/data/maps/alternate and $JGAME/data_site/maps/alternate directories.
- Change the "Add", "Edit", and "Delete" buttons to "Alter" and "Default". "Alter" will prompt the user to select a file. "Default" will use the default map file.
- Change the title of the dialog frame to "Customize Map Shape File", to be consistent with the dropdown menu button from the ICP/DCP.
- In the file chooser menu, change the "Add" button text to "Select", and change the title of the file chooser menu to "Choose Customized Shape File".
- Add tooltip text popups when hovering over the "Alter" and "Default" buttons, explaining what each button's functionalities will accomplish.

3.7 JTLS-2026-17527 Fire Mission Not Updated When Victim Altered

When a missile from a fire mission reaches its impact location, the assigned victim may not be there, or may not be detected. When this happens, the missile is assigned another victim in the vicinity or is changed to a coordinate impact area. The attributes of the fire mission are not altered to reflect this change. This can cause a crash when creating the damage reports for the mission.

If the Assess Weapon Damage routine needs to change the victim of the missile, the fire mission data structures are not updated to reflect the automated model change.

3.8 JTLS-2026-17528 Emission Control Order Transmission Jammers

The Emission Control order did not consider Transmission Jammers.

The model was fixed to properly turn on and turn off Transmission Jammers that affect Satellite communications.

3.9 JTLS-2026-17532 Destroyed MUSE Mission Crash

A MUSE mission was passed back to the model by the ELS. When determining whether fuel was needed the model crashed. The mission should never have been passed back to the CEP, because it was destroyed, but the CEP needs to protect against such incidents.

The external mission pass-back procedure was improved so that the fuel consumption of the mission is always calculated, whether the mission will be accepted by the CEP or not.

3.10 JTLS-2026-17534 Reduce Ship Passive Sonar Detection Calculations

Ships that are in port should not be subject to passive sonar detections. Additionally, if a ship's location is already known, an additional passive detection is not necessary. Removing these passive detection situations will improve model performance.

The code was modified to implement these two processing-saving steps.

3.11 JTLS-2026-17536 HRUs Being Detected Too Quickly

A code error was found in the calculation for determining the time to detect an HRU. The number of foreign HRUs and ARUs within detection range of the covert HRU is central to the algorithm, with more objects covering the area decreasing the time until detection of the HRU. The error was that all assets covering the HRU, regardless of Force Side, contributed to any individual Side's detection time. This meant that even units and HRUs on the same Side as the covert HRU would help a foreign Side detect the covert HRU faster.

The code was modified so that only objects on the same Side as the Side attempting to detect the covert HRU are included in the detection algorithm.

3.12 JTLS-2026-17538 Unit With No Combat Systems Crash

A fully functioning detachment unit with all Combat System TOE values set to zero was introduced into the game when an airlift of a unit was aborted. As the unit was processed into the game, it went through logic to issue its combat systems from supplies. As part of this process an array is reserved for the number of combat systems in the unit. This was zero resulting in a crash for reserving a zero-length array.

A logic error was introduced to check if the unit has any Combat Systems. If it does not, the issue of Combat Systems is completed and exited without any Combat Systems being issued.

3.13 JTLS-2026-17540 BE History Should Hold SUP Instead Of Class

The model keeps a history of units, targets, air missions, and convoys detected within a Basic Encyclopedia (BE) Facility. For ship objects, the history was recording the Ship Class Name instead of the Ship Unit Prototype (SUP). The SUP is a more descriptive item and should be what is saved in the BE history.

The code was modified to save the SUP name instead of the Ship Class for detections of naval vessels.

3.14 JTLS-2026-17542 OTH Radar Missile Detection Terrain Masking

The algorithm to detect missiles was applying terrain masking and earth curvature to detections by Over the Horizon (OTH)-capable sensors. This restriction should not be applied to these sensors.

The missile detection code was modified to remove the restriction of terrain masking and earth curvature for sensors that are marked as being OTH-capable.

While debugging the situation, it was recognized that for non-OTH sensors, the terrain masking algorithm did not consider the antenna height of the sensor. It assumed that the sensor was at the elevation of the grid in which it was located. This problem was also solved.

Finally, for satellite sensors, JTLS-GO 6.4 has the correct computation for satellite altitude, but this was not being used. This error has little to no impact on the jamming of the satellite sensors, but for consistency the correct sensor altitudes were used.

3.15 JTLS-2026-17545 Changing DCA Mission To Protect ACM Crash

The model crashed when the user submitted a Change Mission Parameter order to use an ACM as its protection area.

JTLS-GO only allows DCA missions to protect a circular area, an OPAREA polygon, or a list of specific air missions. Those are the only three protection types modeled within JTLS-GO. A four protection alternative was available on the Change Mission Parameter order panel. This was an error.

The Protect ACM option was removed from the Change Mission Parameter order. There are no plans to add this capability to the model.

3.16 JTLS-2026-17547 Convoy Assets Not Removed From JODA

When a convoy is removed from the game, the corresponding JODA entities are not removed from the JODA.

The JODA objects are now destroyed and removed from the JODA database.

3.17 JTLS-2026-17549 Update Call Sign From External Model

The call sign for a mission was not updated within the CEP when it was assigned by an external model.

The call sign provided by the external model is now represented in JTLS-GO.

3.18 JTLS-2026-17553 Intercepting Aircraft Immediate Head For Fuel

An intercepting aircraft immediately stopped intercepting and headed for fuel.

The check for whether the DCA mission was suitable to intercept the selected foreign mission assumed that the interceptor would conduct the intercept at its normal cruise speed. Due to a recent change in the logic, this is no longer true. The mission now intercepts at its maximum speed, unless ordered to travel at a different speed. As soon as the mission started to intercept, its fuel consumption rate increased, and the model determined the mission would need more fuel.

The interceptor feasibility check now checks to see if the mission has enough fuel at its expected interception fuel consumption rate.

3.19 JTLS-2026-17554 Crash Rescheduling Pre-Launch Event

The model crashed attempting to find a delayed mission event.

The exact cause of the crash could not be determined, so a patch was incorporated into the model to ensure the model will not crash.

3.20 JTLS-2026-17557 ATO Parser Incorrectly Sums Aircraft

The new C2Core system provides multiple Mission Aircraft ("MSNACFT") records for some missions. The ATO Parser for this ATO format is tabulating all the aircraft in the provided MSNACFT records of the mission and uses the total for the count of aircraft in the JTLS-GO mission. It is also using the call sign from the first record as the call sign for the mission in JTLS-GO. This is not correct.

The Parser should be considering each MSNACFT record and using the count of aircraft and call sign from the MSNACFT record having the lowest call sign postfix number.

The Parser was modified to find the MSNACFT record having the lowest call sign number and only use the count of aircraft for the mission having that record. It will also use the call sign specified in that record as the call sign for the associated mission in JTLS-GO.

3.21 JTLS-2026-17559 ATOT Mission Available Time Scenario Start Time

When a mission is suppose to start its first tasking at a time that occurs before the start time of the scenario, the ATO Translator (ATOT) is not including any time value as the mission available time in the mission's order.

The ATOT uses a routine to determine the mission duration from all tasks included for the mission. This routine also computes the mission available time for the mission.

When the start time for the first task falls before the start of the scenario start, the ATOT did not compute the mission available time. This available time was then left empty in the order. Although this situation is unique to the very first ATO for an exercise, it can happen if a user, for example, shifts the ATO period of an older ATO using the ATO Parser.

The routine for computing the mission available time was modified to allow the early task time, even when it may fall before the scenario start time. The simulation will handle these early times by adjusting them to the scenario start time.

3.22 JTLS-2026-17561 ELS Event Timing Problems

The ELS experienced problems with the timing of events. In some cases, when the ELS was restarted from a checkpoint, the program would successfully download objects on the EODA but would never advance in time. On other occasions, the ELS would run for a period of time and then crash with a reason that it attempted to decrease the simulation time.

When the ELS was restarted, it would schedule the existing events, but it would never set the time for the next ELS event. This prevented the ELS from advancing in time because it never reached the portion of the code which incremented the game time. In the case of the crash associated with a decrease in time, the code was inadvertently incrementing the game time by an amount which exceeded the time of the next event. A code correction was made to prevent either of these problems from happening.

3.23 JTLS-2026-17563 ELS Crash Computing HRU Radius

The ELS crashed while attempting to compute the radius of an HRU.

The code was attempting to compute the radius, using the type of entity when it was expecting an object type. The code was modified to use the correct object type in the called routine.

3.24 JTLS-2026-17565 ATO Parser Incorrect Takeoff Task Start Time

The C2Core service provides "AIRMOVE" records for each of the stops for Mobility missions. However, the first record listed is typically missing a start time in the record (the stop time is always provided). The C2Core service is representing the home takeoff location in the first AIRMOVE record, so a start time for this location would not be necessary. In fact, a stop location for the home takeoff location is also not necessary.

The ATO Parser discovers the stop time associated with this first AIRMOVE record, but is unable to extract the associated start time. The Parser stores an unassigned time for this tasking and the ATOT later converts this time to 01 Jan 1970 (start of epoch).

The ATO Parser was modified to exclude the stop location held by the AIRMOVE record as the home takeoff location. This stop is not necessary.

3.25 JTLS-2026-17567 ELS Crash Deleting Active Engagement

The ELS crashed while attempting to delete an active engagement. These structures are used to hold combat engagements which are bundled by the CEP according to their engagement identifier.

The ELS stored active engagements in hash buckets to make their access more efficient. When an active engagement was deleted, the object was not properly removed from the hash bucket. This caused some corruption in the memory and led to an ELS crash. The code was modified to remove the engagement from its associated hash bucket.

3.26 JTLS-2026-17569 JSXR Responses Empty For Georegions Crossing IDL

The JXSR uses special filtering to discover simulation objects contained within a polygon area. This is used primarily for IMT requests made by the WHIP. When an object falls within the polygon, the unit is listed in the reply back to the requesting WHIP. However, when any portion of that polygon crosses the International Date Line (IDL), the reply is empty, even though objects exist within the polygon.

The JXSR stores the latitude/longitude vertex values for one or more polygons in a georegion. When any object is checked to see if it is within the georegion, the JXSR employs a formula to determine whether the coordinates of the object falls within any of the polygons of the filter. The formula fails when vertices of a polygon are on opposite sides of the IDL.

The georegion filter was changed to only store positive longitudes, in effect transforming the range of longitudes for vertices from (-180,+180) to (0.0,+360). Then, temporary coordinates for each object in the comparison are transformed in this same way and applied to the formula. This coordinate transformation allows the formula to properly function.

3.27 JTLS-2026-17570 ATOG Crash Writing MTF Data

The ATO Generator crashed after creating an ATO and writing out the MTF version of the ATO. The code was accessing an array for mission roles using the mission type instead of the mission role.

The code was modified to access the proper mission attribute when accessing the mission role array.

3.28 JTLS-2026-17573 ATO Parser Missing ACM Use Field

When two or more ACO messages are combined by the ATO Parser, the Parser leaves out the "Use" field for each of the ACMs contained in the ACOs that come after the first.

The ATO Parser reads the ACO files in order and combines the ACMs in a set of ACMs obtained from the first ACO. The routine that adds each ACM was not copying the "Use" field of subsequent ACOs whenever this was done.

The ATO Parser was modified to include the "Use" field of each ACM for each ACO following the first.

3.29 JTLS-2026-17577 SVP Warning 1133 Incorrect Wording

There are six different versions of Warning 1133, all of which are similar to one another. Due to a copy and paste error, all six warnings indicated the problem was with a Surface Search sensor, when many sensors did not have any surface search capability.

The code's copy and paste error was corrected. Warning 1133 now reports the correct sensor use type that has a problem.

3.30 JTLS-2026-17579 Prevent MUSE-Owned Air Mission Ownership Transfer

When the link to the MUSE simulation was disconnected, the ELS attempted to regain ownership of the air missions being flown by MUSE. Ownership of the missions should not change, because they are reacquired when MUSE reconnects.

The ELS was incorrectly attempting to gain ownership of the MUSE missions. Code was changed to prevent this from happening in the future.

3.31 JTLS-2026-17581 ELS Attempting To Move Targets To Offset Locations

The ELS was attempting to move an owned Target to a location which was offset from the owning Unit’s location. This was occurring when the Unit was not moving.

The ELS was incorrectly trying to reposition an owned Target when the owning Unit was stationary. Code changes were made to stop this from happening.

3.32 JTLS-2026-17583 Issues ELS Publishing Updates To EODA

The ELS was pushing object updates to the EODA during times when the object updates were not yet complete. This caused some objects to be updated before the data should be changed.

The ELS code was modified to update each type of object in a separate routine. This allowed the code to process each type separately.

3.33 JTLS-2026-17585 ELS Not Launching External Missions

A Controller changed the aircraft type at a Ground Unit, but the ELS did not reflect this change and was not launching missions from that Unit.

The ELS was only updating the aircraft type for squadrons and not for all other unit types. The code was modified to update aircraft type for all unit types.

The ELS was only launching missions from squadrons. The same was also true for external missions being created in the CEP. This has been corrected.

3.34 JTLS-2026-17586 ELS Events Unnecessary Execution At Startup

The ELS restarted very slowly after the CEP had been running for an extended period of time.

When the ELS was restarted, it was scheduling and executing events which had occurred in the past without regard to the current game time.

Modifications were made to prevent the scheduling and execution of past events. This allowed the ELS to restart more efficiently.

3.35 JTLS-2026-17588 WIA/Remains Removed From Medical Units

When a convoy has delivered supplies, it looks around to determine if there are any Remains or Wounded in Action (WIA) that need to be evacuated. The design indicates it should not try and evacuate personnel that are located at a medical-capable unit which represents a trauma center. The check to exclude such units was not implemented correctly.

The check to determine if a unit was a medical-capable unit was not correct. The code was corrected to match the manner in which a medical-capable unit is specified.

3.36 JTLS-2026-17591 Radar Vocabulary List Does Not Include SAR

The SIP and the CEP generate a vocabulary list for the permanent entities, such as Sensor Types. There is a vocabulary list called Radar_ST, that should include all of the sensors that are considered radars. The list does not include Synthetic Aperture Radars (SAR). For the purposes of this list, SAR Sensors should be considered radars.

The SIP and CEP routines used to create these data vocabulary lists were corrected to include SAR sensors in the Radar_ST vocabulary list.

3.37 JTLS-2026-17593 ATOT Chit Orders For Residual Missions

When the ATO contains missions that have their Residual Mission Indicator set to "R", the ATOT will not generate orders for these missions when the option to Process Residual missions is deselected in the ATOT module. However, in this case the ATOT is still incorrectly generating the refuel chits for these missions, and they will not pass the group checking.

The ATOT was not checking either the residual mission indicator for each mission or the desire for Processing Residual Missions while building the chit orders. This caused the ATOT to withhold the residual mission orders but still produce the refuel chit orders for the missions.

The ATOT was modified to include these two checks during the chit processing.

3.38 JTLS-2026-17595 Crash Processing Jammed Unit SITREP

The model crashed when a unit is being jammed and periodically wants to send a Situation Report reminding the Player that it is being jammed. The automatically-generated order was missing an assignment statement indicating that it was a single unit report.

The missing assignment statement was added to the automatic model-generated report.

3.39 JTLS-2026-17597 ICP Exception Clicking On Tree Root Node

When clicking on the scenario name/root tree node in the ICP, the ICP will throw a Java exception in the terminal. Functionality is not affected and the ICP behaves normally.

The ICP will no longer throw an exception when clicking on the root node/scenario name.

3.40 JTLS-2026-17599 Support Unit Has Not Arrived In Theater Crash

In the SVP, there is a warning that indicates that a unit's assigned support unit arrives later than the unit that may need supplies. This is a warning because it should not cause a crash. It does cause a crash.

The SVP-reported issue is considered a warning because support units may arrive before the requisitioning unit within the initialization database, but the Controller could easily TPFDD the unit out of theater and cause the same situation. Therefore, the problem was solved by ensuring that a unit's logistics requisition order does not get delivered to a support unit that is no longer alive or no longer in theater.

3.41 JTLS-2026-17601 Order Group Editor External Update Names

The WHIP Order Group Editor is showing all saved External Update orders as "EU_NONAME", even though we can retrieve the saved orders and see them in the actual order panel.

The External Update order did not have a <queue> element fully defined. Defining the element corrected the problem.

3.42 JTLS-2026-17603 Crash Reading In Checkpoint Weapon Usage Data

A checkpoint crashed being read into the Shadow Game. The problem was pinpointed to the artillery weapon usage data for a specific unit. Instead of a number, the number of rounds used was output as four stars (****).

A Simscript integer write statement specifies the number of digits that should be printed to the file. If the data parameter is larger than the number of digits allowed by the write statement, the compiler outputs a series of stars (****).

The user had been giving an artillery unit a large number of Fire Artillery orders. The rounds fired was greater than 10,000 rounds over a 48-hour period. The Simscript write statement only allowed for four digits; therefore, the model output the four stars (****).

The code was changed to write up to ten digits, which allows for any normal game usage.

3.43 JTLS-2026-17605 ELS Crash Restarting From Checkpoint

The ELS crashed while it was restarting from a checkpoint. The program was trying to destroy an entity-level convoy object because the associated aggregate convoy no longer existed in the game. This situation occurred because the CEP had advanced in time from the time when the ELS checkpoint was saved. This happened when the game time was advanced for several hours while the ELS was not running.

The ELS checkpoint data contained a convoy which no longer existed in the CEP. The code was attempting to clean up the associated entity-level data. An error in the code resulted in a situation where the aggregate convoy was deleted, but the entity level objects still existed in the ELS. The code to delete the entity-level objects was prematurely exiting before they were cleanly removed. Changes were made to correct this problem.

3.44 JTLS-2026-17607 Update PostgreSQL Database Client

Minor PostgreSQL updates are essential for improving system stability, ensuring security, and fixing bugs. PostgreSQL 15.16 has been released and needs to be delivered as part of the next maintenance release.

The PostgreSQL Client, which is distributed with JTLS-GO, has been updated to version 15.16. Also, a custom-created PostgreSQL 15.16 database server .tar file has been included with the JTLS-GO release. Detailed security information for PostgreSQL version 15.16 can be found at:

https://www.postgresql.org/support/security/15/

3.45 JTLS-2026-17609 Cannot Change Port Flag

The Controller has the ability to alter the flag that indicates that a unit represents a naval port, but it does not work.

The code to have the Controller change this flag was accessing the incorrect tab of the order; therefore, the code did not see the request to alter the flag. This has been corrected.

3.46 JTLS-2026-17613 Crash Creating Unit After Game Time 99 Days

The model crashed when the user attempted to create a unit at a game time greater than 99 days into the simulation. The model had a hard-coded unit Time Phased Force Deployment Data (TPFDD) time of 99 days. This caused the model to attempt to schedule the Unit Arrive event at 99 days, which was a decrease in simulation time.

The model wants to schedule the unit arrival a long time in the future to give the Controller a chance to establish the remainder of the unit arrival parameters. The code was changed to have the unit arrive not at day 99, but 99 days in the future.

3.47 JTLS-2026-17615 IMT Help Menus Display "null"

The WHIP's IMT help menus, inside the Help -> "This Screen" option, would display the string "null" at the beginning of the text description. This occurs across all IMT tables.

The "null" string was removed, so that it no longer appears in the help menu's text.

3.48 JTLS-2026-17626 Order Redirect Cruise Missile Crash

A player submitted an order to redirect a cruise missile. The order checked the box indicating a cancellation of the current orbit. No new route for the missile was specified. The order included only a new attack location.

The crash occurred in the code where the model was attempting to re-use the receiver number for the missile's route. The receiver number specified the unique identifier for the route on the JODA. Because there was no existing route, the model crashed when trying to get the old receiver number by dereferencing a null pointer. Code was modified to give the route a new receiver number if there was no existing route

3.49 JTLS-2026-17628 WHIP Exceptions On Emitter Objects

The WHIP was found to be periodically throwing exceptions on the Emitter objects in the game. Although it had no negative effect on the objects, Emitters are not currently reading the TARGET_SUBCATEGORY attribute.

The TARGET_SUBCATORY attribute was added to the Emitter object so that value does not throw a read exception.

3.50 JTLS-2026-17630 Resting Unit Rejects Sequential Move Order

A Move order with a sequential Execution Time option that specified a sequence number 1 was submitted to a ground unit currently executing a Resting task (sequence number 0). The order was not accepted and a non-compliance player message was generated. The sequential Move order should have been accepted and then executed after the Resting task was complete.

The existing logic properly rejected an attempt to order a resting unit to move. However, the logic failed to make an exception for a Move order with a specified sequence number. Note that the sequence number is restricted in the Move order panel to a value greater than zero (the currently executing task). The logic was corrected to accept a Move order with a Sequential Execution Time option.

While investigating this issue, the logic was found to reject Move orders with a Start or Arrival time that was later than the end time of the resting task. This oversight was also corrected.

3.51 JTLS-2026-17633 Frame Title Spacing

The frame titles for the DDS, DCP, and ICP were missing some spaces after the tag lines. For example, the DDS frame title bar displayed "Scenario:scenario_name" when it should be "Scenario: scenario_name".

The spacing issues were fixed in the DDS, ICP, and DCP title bars.

3.52 JTLS-2026-17635 ICP Panels Do Not Scroll To Position

Inside the main scroll panel in the WHIP ICP tab, clicking on a WHIP entry will also update the left scroll panel to locate and automatically scroll to the corresponding WHIP entry. This does not work in vice versa (upon clicking on a WHIP entry in the left scroll panel, the main scroll panel will locate and highlight the corresponding WHIP entry, but it will not scroll to it).

The main panel in the ICP was fixed, so that it scrolls to the selected position from the left scroll/ tree panel.

3.53 JTLS-2026-17638 DDSC Supply Categories UOM Preferences

The preference settings for the Supply Categories UOM were always the default UOMs listed for either Dry Weight or Wet Weight, depending on the Supply Category's Shipment Type, and the default UOMs for Dry or Wet Weights are determined by the first UOM entry in the UOM table whose multiplier is 1.0 and additive is 0.0. However, in the Supply Categories table, many Class V supply categories had specific Preferred UOMs set for those supply categories, but the preference setting in the DDS Client did not pick up those preferred UOMs by default.

The DDSC code was modified so that it now picks up the preferred UOMs set for the supply categories. For a supply category, if it has a preferred UOM set for it, the preference setting by default will use the indicated preferred UOM instead of the default UOM of the relevant shipment type. Also, if the user changes the preferred UOM for a supply category, the preference setting of that supply category will update accordingly.

3.54 JTLS-2026-17639 ICP OTH-GOLD Country Code Default Setting

The JOI panel in the ICP defaults the outgoing country code for OTH-GOLD to ISO-3166-1-TRIGRAPH, when it should be set to OTH-GOLD as standard format.

Set the OTH-GOLD outgoing country code to OTH-GOLD as the default setting.

3.55 JTLS-2026-17643 Reject Order Status Not Set When Unit In Formation

Every order that is rejected should have a rejection status. When an order is rejected because a user did not indicate the ship had permission to leave a formation, it did not set any rejection status in PLAYER.ORDER entity.

In all such cases, the rejection status is now set.

3.56 JTLS-2026-17644 SVP Java Panel Does Not Update Correctly

The SVP Java window (started from the Javamenu) does not update the checkboxes upon switching the scenario after the initial startup.

The logic has been fixed to where the checkboxes update properly upon switching to multiple scenarios.

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.

The following list of issues is known and have not been fixed in time to make it into this release of JTLS-GO 6.3.9.0.

4.1 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.

4.2 MHE Targets Loading Air Mission Can Cause a Crash

MHE targets should be avoided for loading and unloading air missions. It is suggested that the database be set to “Do Not Use” for Air Missions.

4.3 JTLS-GO Strategic Lift Missions Are Not Working Properly

Strategic Lift Missions, used to move TPFDD assets into the Theater and report the results to a real-world TPFDD processing system, has not been updated to work within JTLS-GO 6.3.

4.4 Tactical Ground Formation Attacks Do Not Work

The ability to send a Tactical Ground Formation on an Attack mission has been temporarily disabled due to reliability issues.

4.5 ATOT Spreadsheet Lacks Detailed Field Checking

The ATOT Spreadsheet Parser has been found to have numerous issues within the Spreadsheet format that are not caught and cause the spreadsheet parser to crash. Fixing the uncovered issues are being worked and should be fixed prior to the next maintenance release of the JTLS-GO 6.3 series.

4.6 Moving Combat System Supplies Can Reduce Unit Strength To Zero

If a user does a mandatory transfer of Combat System supplies from one unit to another, the providing unit can be emptied out and exists without any Combat Systems or personnel. This situation needs to be thoroughly and properly handled.

4.7 Upgrade Procedures For Maintenance Release of PostgreSQL

The Development Team continues to look for viable options to upgrade from one maintenance release of PostgreSQL to a newer maintenance release.

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
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
SDR	Scenario Data Repository
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 6.3.9.0 DATABASE CHANGES

Due to changes made in support of STR JTLS-2025-17024, the JTLS-GO 6.3.4.0 database format was changed to add a new column to the text_symbol_lu table, related to unit symbols.

This change requires that users **unload** their scenarios prior to installation of JTLS-GO 6.3.4.0, and then **load** them following installation, if users are upgrading to JTLS-GO 6.3.9.0 from JTLS-GO 6.3.3.0 or earlier.

APPENDIX C. VERSION Version 6.3.9.0 REPOSITORY CHANGES

No significant changes have been made to the structure of the JTLS-GO 6.3.9.0 repository.