JTLS-GO

Version Description Document

December 2024

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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 6.3.3.0)

[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 6.3.3.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 6.3.3.0 is a Maintenance release of the JTLS-GO 6.3 series that includes an updated repository of standard data, a demonstration scenario based in the western Pacific, as well as minor 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 Director of Development
Valkyrie Enterprises LLC
120 Del Rey Gardens Drive
Del Rey Oaks, California 93940
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 6.3.3.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO®) software suite. JTLS-GO 6.3.3.0 is a Maintenance delivery for the JTLS-GO 6.3 series of releases.

JTLS-GO 6.3.3.0 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 “wespac63”. There were no database format modifications between this Maintenance release and the original JTLS-GO 6.3.0.0 version. Appendix B of the *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 descriptions of the minor Engineering Change Proposals (ECPs) implemented for this release are provided in Chapter 2.0. Chapter 3.0 summarizes the Software Trouble Reports (STR) that have been corrected and are delivered with this version of JTLS-GO 6.3.

JTLS-GO 6.3.3.0 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 Air Services User Guide (*JTLS-GO Document 03, Version 6.3.0.0)
- *JTLS-GO Configuration Management Plan* (JTLS-GO Document 03, Version 6.3.0.0)
- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 6.3.1.0)
- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 6.3.0.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 6.3.0.0)
- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 6.3.2.0)

- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 6.3.2.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 6.3.0.0)
- *JTLS-GO Repository Description* (JTLS-GO Document 14, Version 6.3.0.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 6.3.0.0)
- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 6.3.0.0)
- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 6.3.1.0)
- *JTLS-GO DoD Architecture Framework* (JTLS-GO Document 22, Version 6.3.0.0)

1.2.3 Updated Documents

- *JTLS-GO Analyst Guide* (JTLS-GO Document 01, Version 6.3.3.0)
- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 6.3.3.0)
- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 6.3.3.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 6.3.3.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 6.3.3.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 6.3.3.0)

1.2.4 New Documents

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

1.2.5 Delivered Software Components

JTLS-GO 6.3.3.0 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)
- JTLS Object Distribution Authority (JODA)

  The current JODA build number is 214.
- 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)
- 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 6.3.3.0 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.

1.3 INTERFACE COMPATIBILITY

1.3.1 Support Software

JTLS-GO 6.3.3.0 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 6.3.3.0 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 6.3.3.0 is delivered with the Adoptium project Temurin Java Development Kit (JDK) 1.8 Update 432 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 database tools require a certified PostgreSQL 15.8 database server and the full PostgreSQL installation. PostgreSQL 15.8 that has been compiled under Linux 9.4 is bundled with the JTLS-GO 6.3 release tar files. It is not necessary to use the delivered solution, but it is the easiest method to meet the requirements of JTLS-GO 6.3.3.0. There are several alternative methods available for obtaining the PostgreSQL 15.8 software. Refer to Chapter 6 of the *JTLS-GO Installation Manual* for additional installation details.

JTLS-GO 6.3.1.0 was released with PostgreSQL 15.7. There is no currently known available method to upgrade PostgreSQL 15.7 to PostgreSQL 15.8.

- If your organization requires the use of the most current security release of PostgreSQL, download each of your scenarios held by PostgreSQL. Install PostgreSQL 15.8 by following the instructions in Chapter 6 of the *JTLS-GO Installation Manual* and reload your scenarios.
- If your organization is willing to skip this maintenance release, JTLS-GO will operate without error.

JTLS-GO 6.3.2.0 is delivered with a PostgreSQL 15.8 client. This is a security upgrade from the previous release of JTLS-GO 6.3.1.0. This version of the client software has been tested and works with both the previously-released PostgreSQL 15.7 server and the current PostgreSQL 15.8 server.

- 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.
- 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 6.3.3.0, 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 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. 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, Ms. Jessica Camacho (jessica.l.camacho.civ@mail.mil) to obtain the completed Cybersecurity paperwork.

1.3.3 JTLS-GO High Level Architecture Compliance

The JTLS-GO 6.3.3.0 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 has been accomplished. The reader should note that the JTLS-GO Development Team, to date, has not been able to test this federation. 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 6.3.3.0 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. VERSION 6.3.0.0 DATABASE CHANGES in the *JTLS-GO 6.3.0.0. Version Description Document* has a summary of all database changes. There were no database changes made between this maintenance release and the original JTLS-GO 6.3.0.0 release.

To upgrade your JTLS 6.2 scenario to JTLS-GO 6.3 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 6.3.0.0 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

You should not make any modifications to the Default Symbol Set delivered with JTLS-GO 6.3.3.0, 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

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 15.7 required to operate JTLS-GO 6.3.3.0.

2.0 ENGINEERING CHANGE PROPOSALS

This chapter summarizes the minor model capabilities added to JTLS-GO 6.3.3.0 as a result of implementing authorized Engineering Change Proposals (ECPs).

2.1 JTLS-2024-16875 Find Orders Frame Displays Orders Dynamically

Summary of Model Change Request

The current method for finding orders in the WHIP and TRIPP requires the user to search for the name of the desired order in a dialog, and then statically displaying the corresponding orders in a table based on the entire search input. If no orders are found, an error message is displayed and the user needs to repeat the search process again.

Design Summary

The order search frame was redesigned to display a table of all possible orders and a search bar in the same frame. The table will dynamically update and display corresponding orders based on each user keystroke input in the search field.

2.2 JTLS-2024-16939 Add Weather At Attack Location To Mission Report

Summary of Model Change Request

A user requested that the current weather at an air mission's attack location to be included in the mission report for attack missions.

Design Summary

The code was modified to include the current weather at the attack location in the mission report.

2.3 JTLS-2024-16998 Add Long Unit Name To SITREP

Summary of Model Change Request

The unit short name in the Situation Report (SITREP) is not sufficient to meet exercise requirements.

Design Summary

The routine that generates the SITREP was modified to include the UT LONG NAME, which can be as long as 40 characters. The message templates were modified to display the unit long and short names in the English and MTF formats.

2.4 JTLS-2024-17009 Setup For SDC Filters

Summary of Model Change Request

The Scenario Data Client (SDC) allows filters by Force Side and by JDSP (JTLS Data System Protocol) object.

The Force Side filters can be set within the Interface Configuration Program (ICP), but no user-friendly method exists to set filtering by JDSP object.

Design Summary

An extension tool for the ICP, much like the AAR Filter Interface (AARFI), has been created that allows Tech Control to view current SDC filters, modify as needed, and save them. This tool, called the SDC Filter Interface (SDCFI), can be accessed individually, or via the ICP's SDC entry for Advanced Setup.

The SDCFI first loads object names from the jds_protocol.xml file, turning all objects on, which is the default behavior of the SDC. It will then check for an existing $JGAME/<scenario>/config/jsdc.conf file, which the SDC reads on startup to determine user-defined filtering. If the file exists, the SDCFI will read the file, and any filters that are marked as "off" within the file will be switched "off" on the GUI. If the file does not exist, the SDCFI will look for a $JGAME/data_site/sdr/sdc-object.xml file, which serves as a site-specific scenario default. If this file exists, all object filters in the GUI will be switched to "off", and then only the objects listed in the sdc-objects.xml file will be turned "on".

Like the AARFI, the SDCFI requires the user to save their current settings on the extension tool. Saving on the ICP does not save the AARFI or the SDCFI filter settings.

The ICP was modified to allow the starting of the SDCFI from the SDR tab. The AARFI, which was previously named the AFI, has been modified to align with its new name.

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-2024-16874 AAR Script Ends Prematurely

The AAR script aar_air_other ends prematurely, with the error message "ERROR: relation 'aar_other_losses_air_run_1' does not exist".

The SQL statement in the script referenced the wrong view, and needed to access fields from another view that was not referenced. These errors have been corrected.

3.2 JTLS-2024-16876 Accessing Sensor OPM Page Error

When viewing the Online Player Manual (OPM) target category data, the link to the page for Sensor Types was invalid.

The Sensor Type data was saved to the subdirectory for Target subcategories, but the link was attempting to access that file in the Targets directory. The code was changed to access the proper directory.

3.3 JTLS-2024-16881 Remove Obsolete JTOI ICC 282

The JTOI (JTLS Transaction Operational Interface) build for ICC 2.8.2 is now obsolete, and should be removed from all build processes and delivered archives.

The JTOI build of ICC 2.8.2 has been removed from the build processes. The common code shared across JTOI ICC builds has been moved to jtoi_icc_common.

3.4 JTLS-2024-16882 OPM Unused Directories

The directory structure for the OPM included some unused directories.

The unused directories were removed from all places in the code.

3.5 JTLS-2024-16883 Inconsistent Use Of Memory Management Code

Throughout the JTLS-GO code, there were instances where memory management was done, but no feedback was given when errors occurred. Some common routines were not being used.

Code was modified to utilize the common memory management routines, which do provide feedback and error messages. Error messages were directed to the service log file or to the terminal where the program was running.

3.6 JTLS-2024-16884 SVP Error 256 Crashes SIP

If a database has an instance of Scenario Verification Program (SVP) Error 256, indicating that an Individual Intelligence Parameter's Visual Sensor is not the correct type, the Scenario Initialization Program (SIP) crashes.

The error was caused by the new manner in which JTLS-GO 6.3 holds a sensor's use capability. The code was corrected to properly access the new data structure.

3.7 JTLS-2024-16885 SIP Crash Generating OPM When HRU Owns Target

A typographical code error caused the SIP to crash when a checkpoint includes a High Resolution Unit (HRU) that owns a target.

The typographical error was corrected.

3.8 JTLS-2024-16886 Package Checker Script Missing RPM

The script used to verify server-installed packages for Linux is missing the RPM check necessary for linking Simscript executables.

Added "unixODBC-devel" to the list of RPMs that are checked for a development server.

3.9 JTLS-2024-16890 AAR Airbase Consumption Report Usage

The consumption of fuel and fired weapons by air missions should be recorded in the After Action Review (AAR) database, and that consumption data should be accessible by the Airbase Consumption Report. The report did not pick up the fuel usage and weapons fired by air missions originating from a particular air base.

The model was sending the fuel and weapons consumed by each air mission to the AAR database, but this information was not connected to the air mission's home base. This has been corrected.

In addition, the code made erroneous references to field names in the Airbase Consumption Report order that had to be corrected. These have also been corrected.

3.10 JTLS-2024-16891 BDA Battle Summary Inaccurate

In the past, each Combat System had a different name based on the Combat System Prototype (CSP) that owned the asset. This was changed, but the BDA Battle Summary Report still assumed this was true. The generated message format was changed, but the code supporting that message was not changed.

The information written out for the message was corrected. The CSP is no longer written to the message file. Several inefficacies were also found in the code and corrected.

3.11 JTLS-2024-16893 Updating Unit Transponders Does Not Work

When an HRU is created and takes an available Transponder from the parent unit, so it can continually report its location, the number of transponders that are left at the parent unit is not being updated appropriately in the JTLS Object Distribution Authority (JODA).

When updating a unit attribute, the JODA needs to be told the type of unit (ground, airbase, squadron, etc). This was not done because the variable used to tell the JODA the unit type was not appropriately set.

The proper attribute is now passed in the JODA message in the routine that was attempting to update the JODA

3.12 JTLS-2024-16895 Routine Unused Variable

There was an unused variable defined and set in the routine MANAGE LINK16 SOURCE. The variable held the faction of the object.

The faction variable was not needed and was removed from the routine.

3.13 JTLS-2024-16897 Local Variable For Fuel Priority Not Used

The read code for aircraft, combat system, and ship class fuel options each defined a local variable "fuel option priority", and then never used the unneeded variable.

The unneeded variable was removed from the code.

A few other routines with unused variables were also corrected.

3.14 JTLS-2024-16899 JTLS Menu Exercise Log Schema Initialization Fix

The Exercise Log Schema initialization function of the JTLS Menu failed to enforce a requirement that an existing scenario name be chosen.

The Exercise Log Schema initialization function now requires a user to choose an existing scenario name before proceeding.

3.15 JTLS-2024-16900 Hide Glassfish System Info

When a bad request to Glassfish is made, the system responds with an Error 404 "Not Found" message, saying that Glassfish 5.1 is being used. The knowledge that JTLS-GO is using Glassfish version 5.1 can be exploited by a hacker to gain access to the classified system.

The Glassfish server configuration was modified so that Error 404 messages will not display Glassfish server information.

3.16 JTLS-2024-16901 Exercise Log And DDSAS Mismatch

A recent change to the Database Configuration Program (DCP) display configuration process made the Exercise Log and Database Development System (DDS) Application Server (DDSAS) configurations run as separate background processes.

The Exercise Log depends on the existence of the Glassfish domain root. When the Exercise Log process thread runs faster than the DDSAS thread, the configuration fails.

A lock on the Exercise Log process was added to wait for the DDSAS to create the Glassfish domain root.

3.17 JTLS-2024-16902 Exercise Log Will Not Create If DDS Inactive

When only the Exercise Log is active, the DCP save process tried to validate the DDS database schema, which may not even exist, so the configuration process fails.

The DDS database validation was removed.

3.18 JTLS-2024-16903 OPM Apache Password Management Blocks Access

The OPM Apache password management prevented non-Controller WHIP users from accessing some subdirectories, ostensibly located under the Controller OPM, that should be accessible by non-Controller WHIPs.

An OPM "ALL-WHIP" user group was created and assigned to authenticate these common Controller subdirectories.

3.19 JTLS-2024-16904 SDR Initialization Checking For DDS Access

When Oracle was used as the database management system software in previous versions of JTLS-GO, the script to initialize the Scenario Data Repository (SDR) created a database link between the SDR database and the DDS database. In order to ensure the link will be successful, the script checks that the username and password for the DDS is valid.

This database link was never used by any of the configuration-managed software, and no longer exists under PostgresSQL. Being able to access the DDS database should not be required when performing a game setup, which is when the initialization of the SDR normally happens. But without access, the initialization of the SDR currently fails because the script checks for the existence of the DDS database.

The check for access to the DDS was removed from the sdr_action script.

3.20 JTLS-2024-16905 SVP Warnings Not Ignored

SVP Warnings 1160, 1615 and 1642 are not ignored when they are turned off.

The code to bypass the warnings was incorrect. The code was corrected.

3.21 JTLS-2024-16906 Routine Check Ground Fuel Incomplete

The routine CHECK.GROUND.FUEL is incomplete.

The routine CHECK.GROUND.FUEL was previously modified to report SVP Warnings related to the new multiple fuel category capabilities. The actual warnings were never completed to report them in the SVP.

Warnings 1254 and 1255 were updated to reflect new code and usage of multiple fuel categories:

- 1254 reports fuel categories where the basic load is not enough for 5 days normal consumption, as written in the code.
- 1255 reports when there is not enough fuel in the highest priority fuel category to support the larger of usage per KM, non-combat usage, or combat usage of a combat system.

3.22 JTLS-2024-16907 OPM Lanchester Assignment Table Frozen Column

The first column of the OPM Lanchester Assignment table needs to be frozen.

The first column of the Lanchester Assignment Table is the KILLER Posture and should always be in view. The code was corrected.

3.23 JTLS-2024-16908 Add Entity Count To Table Title

There is no way to know how many rows are in an Information Management Tool (IMT) table without scrolling through them.

The number of entities (in this case, rows) was added to the IMT table title.

3.24 JTLS-2024-16909 Targetable Weapon SAL Link Does Not Work

The link to the Surface-To-Air Lethality data on the Targetable Weapon page does not work.

The link for the Surface-To-Air Lethality was improperly formatted, and has been corrected.

3.25 JTLS-2024-16910 WHIP/TRIPP IMT Menus Inconsistent

The IMT menu in the TRIPP uses a separate menu definition file than the assigned WHIP menu definition. This required two files to update whenever a change to the IMT menu was made.

The IMT and Order menu definition portions of the WHIP menu files were moved out into their own separate files, and are instead referenced by the WHIP-assigned menu file. The change

centralizes each of the IMT and Order definitions to one file to improve consistency across the WHIP menus.

3.26 JTLS-2024-16911 IMT Screens Do Not Update

There have been reported instances of IMT tables going stale for unknown reasons and not updating their live data.

IMT tables subscribe to JTLS XML Serial Repository (JXSR) updates, and periodically request new data to render in its table. To prevent the IMT's data from going stale, a code change was made to re-subscribe to JXSR updates whenever an IMT screen is displaying its related data but is not actively requesting for updates.

3.27 JTLS-2024-16912 New WHIPs Order Group Editor Exception

When working with a new WHIP that has had no Orders or Report Orders previously submitted, the WHIP will not have existing associated Order or Report Order history files on the server until the first Order and Report Order are sent. When the Order Group Editor was opened before submitting any orders, there was an attempt to retrieve the non-existent Order history file, and the WHIP threw an exception.

The WHIP was changed to throw an exception if the creation of the Order or Report Order history file failed to create it on starting the Order Group Editor and Report Order Group Editor in new WHIPs.

3.28 JTLS-2024-16913 Exercise Log Export - Redact Classified Titles

The Exercise Log export function was not redacting the titles of log entries that were marked above the selected maximum classification.

The Exercise Log export function now redacts the titles of log entries above the selected maximum classification. The Exercise Log's dependencies were also updated.

3.29 JTLS-2024-16914 Mission Emitter Load Updates Not Sent To JODA

Numerous JODA errors are reported when an air mission's emitter load update was sent prior to the object existing in the JODA, and an initialization was sent when the object already existed. The IMT information shown for an air mission's emitters was not accurate.

When the mission's emitter load is assigned, the improper model entity was used to send the object to the JODA. The proper entity is now used to initialize the JODA, and the information is published correctly and is available on the IMT.

3.30 JTLS-2024-16915 Foreign Target Range Rings Not Available

In JTLS-GO 6.2, the WHIP showed Target range rings based on the database ranges provided. Because it is possible for a database builder to alter the range information for a specific Target, getting this data from the Target's subcategory may not have been accurate. For this reason, the WHIP was changed to always use the JODA range item information from the model for displaying range rings.

When this change was made, it became impossible for a Player WHIP to view range rings for foreign detected targets, because the JODA Range Items associated with an object are only seen by the Controller and the Force Side that owns the object.

Range Items were changed so that all Force Sides can see all range items. A Player WHIP will not be able to view range rings unless an object is detected, but once an object is detected all of its range information is now available to the detecting side.

There may be some circumstances under which this may be too much information, but the design decision for this STR was that it was more important to provide the range information that was missing than worry about providing too much information. An investigation of the circumstances under which too much information is provided is required and will be performed at a future date.

3.31 JTLS-2024-16916 Log Rollup Message Error For ADA Targets

If any of the Units included in a Logistics Rollup report have Air Defense Targets, the report has a formatting error.

The message definition file had a formatting error, which was corrected.

3.32 JTLS-2024-16917 NATO App6-B ID Codes Not Recognized

The NATO App6-B ID Codes symbology differs from the US 2525-B standard for SOF Naval units and SOF Civil Affair units. Because these do not conflict with any entries in the US 2525-B standard, they do not exist in the 2525-Symbols.xml file. This causes symbols using these ID Codes to throw an unrecognized ID Code error within the JSYMS program, and for the LC2IS Message Service to be unable to translate them into LC2IS symbol codes.

The NATO App6-B ID Codes differences were added to the 2525B-Symbols.xml file as valid entries. Because there is no conflict with already existing US 2525-B symbols, this should cause no issues.

3.33 JTLS-2024-16918 Submarines Detected By Radars

A submerged submarine was being detected by a radar.

The model correctly determines that a submerged submarine was not subject to detection by radar, but the logic did not properly skip over the radar after it came to that conclusion. The logic has been corrected.

3.34 JTLS-2024-16919 Alert Missions Show At Location 0N 0E

Missions told to go on alert at their home base show up at location 0N 0E.

There were two issues:

- The mission's own side perception data was not initialized to ground truth. The code was changed to ensure that the mission's perceived information for location and COP location were initialized to ground truth.

- The model assumed that if a mission was active, it was flying. This is not true for missions on alert. The model was changed. Foreign missions are considered active if they are flying, but a Side's own missions are considered active from the moment the mission enters the game to the moment at which the mission is COMPLETE, DESTROYED, or CANCELED. The mission may still exist in the game for reference purposes until the end of the Air Tasking Order (ATO) cycle, but is no longer considered active.

3.35 JTLS-2024-16920 Mission Not Given Pickup Supply Unit

A Mobility Mission was not given a unit from which to pick up supplies. When the mission reached the pickup location, and there was no feasible unit from which to pick up supplies, the model crashed.

The mission's load supply task was appropriately canceled, but when the task was being marked as complete, the model attempted to access the name of the unit from which the supplies were taken. There was no unit, so the model crashed.

The model now checks for this situation, and appropriately reports that no supplies were picked up due to a lack of a pickup unit.

3.36 JTLS-2024-16921 Bad Message File Single Aircraft Load Of NONE

If the user asked to show the aircraft load for a single type of load, and all alternatives were equal to NONE, the resulting message was formatted incorrectly.

The code to generate the Modify and Show Load Assignment data was simplified and the error condition is no longer feasible. The code was also placed in its own routine for easier code management.

At the same time, two other issues were corrected:

- The resulting massage from the order properly shows whether the information was based on a Modify Load Request or a Show Load Request.

- The model was unnecessarily updating the ATO Translator and Generator initialization files when the user simply asked to show the contents of the Load Assignment Array.

3.37 JTLS-2024-16922 Create Message Looking For Custom Message

There was a requirement to have a customized Logistics Rollup Report using tabs between tabular items instead of spaces. A new language locale "english-tab" was created to support this. A copy of the Logistics Rollup message definition file (MDF) was placed in the directory and modified for the requirements.

The create_message script was then run to generate the message formats for this new locale. This threw an error because it could not find the MDF under the $JGAME/data directory structure, because it was under the end user $JGAME/data_site directory structure.

The Message File Parser was checking if the locale contained the words "english", "mtf", or "taccis" to make the decision if the MDF was under the $JGAME/data directory, or the user-specific directory structure. The user-created locale "english-tab" does include the keyword "english". This step was switched from the locale containing "english", "mtf", or "taccis" to the locale equaling those keywords.

While testing this change, it was noted that a site-specific Kuwaiti Arabic MDF threw errors complaining that the message title did not match the entries in either the static_voc.xml or msg_shelf_life.xml files. This check is done to ensure system integrity, and the MDF in question was automatically translated by the i18nui tool. The i18nui tool should not translate the title because its only purpose is this integrity check. This issue was also resolved.

3.38 JTLS-2024-16923 Crash Detecting "Gone Ghost" Ship

A "Gone Ghost" in JTLS-GO is an object that is created when a detected object moves. For example, assume Side Blue detected a Red ship at Location X, Y. When the Red ship moves, it creates a "Gone Ghost" at Location X, Y. The next time Blue has a sensor covering Location X, Y it cannot see the Red ship because it is no longer there. Instead, Blue "detects" the Gone Ghost and reports "I expected to see a Red ship at this location, but it is not there."

The new JTLS-GO 6.3 reporting procedure needs a source for the detection of this Gone Ghost and it was not assigned. This caused the model to crash.

The assumption is that a Gone Ghost was a contact for an object on the surface. For this reason, a Source of "Radar" was assigned to the Gone Ghost detection.

This problem was found and reported as fixed in JTLS-GO 6.3.2.0. Somehow the proper routine fix was not included in the releases.

3.39 JTLS-2024-16924 SIP Crash Generating OPM

The SIP crashed while generating the OPM. The crash occurred when the SIP was printing the TPFDD data for a squadron.

The SIP was attempting to print the name of the Home Base for a specific squadron. In this case, the squadron had no home base.

The SIP code was modified to allow for the case when no home base was assigned to a squadron. In these cases, a blank name is printed in the column for the home base.

3.40 JTLS-2024-16927 AAR Documentation Improvements

Many attributes in the AAR object tables can change during the course of game play. The fact that they can change, and how to find those changes, should be part of the AAR online documentation for those tables.

The table column descriptions within the AAR documentation come from the attribute descriptions within the JTLS Data System (JDS) Protocol. The JDS Protocol had additional information added to the description of those attributes that are changeable, stating that they may change and the library to use to discover the current value based on a time.

3.41 JTLS-2024-16929 Create Runway Keyword Needs Upper Case

JTLS-GO orders are strings of keyword-value pairs. By convention, the keywords are always in upper case, but this is not a mandatory requirement.

In the Create Runway order, the keyword for the type of runway to be created did not follow the convention. This causes issues if a user is creating a read order file.

For consistency, the keyword was changed to upper case.

3.42 JTLS-2024-16933 Added Information to ato_constants.xml File

The ATO Translator (ATOT) obtains information concerning how to link ATO Mission types to JTLS-GO Mission Types. Several NATO Mission Types were encountered that did not exist in the ato_constants file.

The following updates were made to the ato_constants.xml file to seamlessly convert an Interactive Command Control (ICC) ATO to JTLS-GO missions:

- Mission Type "C3" was missing - it is now translated to a JTLS-GO "AWACS" mission.
- Mission Type "RECCE" was missing - it is now translated to a JTLS-GO "RECCE" mission.
- Mission Type "AAR" was missing - it is now translated to a JTLS-GO "REFUEL" mission.

Two other ato_constants.xml issues were uncovered:

- The three CAS mission types ("CAS", "XCAS", and "GCAS") had a Target Type of "UNK". This was changed to indicate it should go against a Target Type of "CBT". This allows the ATOT to specify the types of Target Type Groups these missions should plan on attacking.

- The mission type "MEDEV" (Medical Evacuation) was translated to a Transfer mission. It is better modeled as a Mobility mission, so that wounded personnel can be picked up from the various stops provided to the missions.

3.43 JTLS-2024-16935 NATO Report Infinite Loop

The model went into an infinite loop when a user submitted a request for a NATO Report with a time interval of zero, directing the model to generate the report only once.

The logic was changed to recognize an report interval of zero.

3.44 JTLS-2024-16937 Crash Acquiring Air Mission Order Attacking ACM

If an air mission is told to attack objects within an Air Control Mean (ACM) polygon, and the order is acquired by the enemy's Communications Intelligence (COMINT) capability, the model crashes attempted to create the message informing the intercepting Side of the acquired information.

Attacking an object in an ACM was not expected to be acquired by COMINT. The code was corrected to account for this possibility.

3.45 JTLS-2024-16938 Set Unit Parameter Message Truncates CS Names

A Set Unit Parameter order message that includes the Combat Systems owned by the unit has truncated Combat System names.

The message file used the old Combat System name length of 15 characters. It has been increased to 25 characters.

A search was accomplished to determine if any other messages had the same problem. Two other messages were found to have the same problem and were corrected.

While making these changes, it was found that there were two routines in JTLS-GO that did exactly the same thing, and both were accessing data that was unnecessary. This code was cleaned up at the same time.

3.46 JTLS-2024-16940 ATO Parser Bad Route Item

The ATO Parser may find an error in the ROUTE record of a mission. For example, the ROUTE may include a time for a route location, but no location. When parsing is complete, the Parser lists all the errors that were discovered, including the error with the bad route item. However, the Parser

will not produce any of the XML for the ATO until this particular error is fixed. ATO translation must wait until this error is fixed at the parsing phase.

The routine used by the Parser to scan and interpret the ROUTE record uses a variable to determine critical errors during the parse. This variable was changed so it will not be set to critical for this particular logic error. The Parser will continue to flag the error but allow it to be passed on via the XML to the ATOT for proper user adjustment.

3.47 JTLS-2024-16941 Unable To Modify Aircraft Load Comms Assets

Communications assets were added to Aircraft Loads in JTLS-GO 6.3. The order used to manage the contents of aircraft loads was never modified to allow the Controller to change the communications capabilities of loads.

The Manage Aircraft Load order was modified to allow changes to and show the communications capabilities in aircraft loads.

3.48 JTLS-2024-16942 ATO Parser Not Correlating AIRMOVE/ROUTE

The ATO Parser is skewing the landing task of an AIRMOVE record with the corresponding item in the ROUTE record, which are included within the records of a mission. The result is that the mission is directed to visit the landing location twice.

The ATO Parser reads all the records for each mission in the ATO. The items listed in the ROUTE record generally correspond with the other records. When an AIRMOVE record exists and properly references the corresponding ROUTE item, the Parser did not associate these location items. The Parser was not comparing the leg number and time slot whenever the AIRMOVE was not using an ICAO as the location. The ICC sometimes uses latitude/longitude locations in these records of the ATO, so the Parser was skipping the association of the location between these records.

The Parser will associate the AIRMOVE with the corresponding ROUTE item, whether or not latitude/longitude locations are used.

3.49 JTLS-2024-16943 Air Mission Ignores ACM Area Altitude

An air mission was given an air route which consisted of multiple ACM areas. The air route also specified an altitude for each ACM area. The mission failed to adhere to the specified altitude associated with each ACM area as it flew along the route. Instead, the mission flew at the original default altitude assigned to the pre-defined ACM areas.

When a Corridor, Line, Track, or Point ACM area is created using the Manage Air Control Mean Area order, "Default Altitude" is a required entry on the panel. The created ACM area can be included in an ingress/egress/transit route that is created by a subsequent Player air order. The route is specified in the Air Route utility that allows existing ACM areas to be incorporated in the route.

The Air Route utility included an optional altitude field for an existing ACM area which was not processed by the model. An ACM altitude entered in the Air Route utility did not override the original ACM Default Altitude value. Research showed that the ATO message also does not allow the altitude of an existing ACM area to be altered when part of a route (Note that ACM areas that are tasking locations, such as an orbit, can be altered by the ATO message and by the Player). For consistency with the ATO message, the ACM altitude field was removed from the Air Route utility.

Also, the routine that builds the mission task lists for the ingress/egress/transit route was changed to reference the Default Altitude in the ACM area, instead of the optional altitude field that was removed from the Air Route utility.

The ATOT was expecting to write an altitude value in the corresponding parameter of the utility for the ingress and egress route when the ACM option was used. Because the value was never provided by the calling routine, it was writing an empty value, which caused a problem displaying the ingress/egress route in the WHIP. The calling routine never would pass an altitude value because the allowed ATO records for a mission are not able to set an altitude for an ingress/egress. The altitude for an ingress/egress defined by the desired ACM cannot be changed. The ATOT was modified to simply provide the name of the ACM and entry/exit points when the ingress/egress utilities are generated.

3.50 JTLS-2024-16944 Bad OPM Page Links

When viewing the Target Category Data, some of the links to Jammer and Transportation Class target category OPM pages did not work.

The code to link the Jammer Type and Transportation Category pages had typos. These errors were corrected and the OPM pages are now linked correctly.

3.51 JTLS-2024-16945 Error Detaching Existing Unit

When a Player attempted to detach an existing unit from a parent unit, the detachment would sometimes fail when the parent unit was damaged. The error message reported that the parent unit had a shortage of either combat systems or supplies. Because it was an existing unit, it should always be detachable from the parent unit unless the resulting strength was below the wiped-out threshold.

Before finalizing the detachment, the code checked to see if the parent unit could provide the required combat systems and supplies. This check required that the full TOE of combat systems and the Basic Load of supplies be available at the parent unit. The check should have looked at the current amounts available at the parent and computed the detached unit strength based on partial quantities rather than the complete amounts.

The code was modified to use the proper amounts when evaluating the detachment.

3.52 JTLS-2024-16946 Combined Aircraft Loads Error

A Player attempted to change the aircraft load for a tanker mission using the Change Mission Parameter order before the mission entered Launch Capable posture. Although a permissible action, the model incorrectly combined the newly-assigned load and the aircraft's default load. In this case, zero fuel was loaded on the tanker.

The routine COMBINE AIRCRAFT LOADS did not include air mission fuel in the combined load. This was corrected.

In addition, the code was overly complicated to accommodate a blank default load and a blank new load. This situation could never occur because the new load was always present from the calling routines. Nevertheless, a logic check was added to the calling routine CHECK LOAD VALIDITY.

Custom code was used in the COMBINE AIRCRAFT LOADS routine instead of the existing COPY AIRCRAFT LOAD routine. COMBINE AIRCRAFT LOADS now uses COPY AIRCRAFT LOAD.

3.53 JTLS-2024-16948 WHIP/MDP Inconsistent Variable Arguments

Within the MDF format, a VARIABLE() function can be used to return pre-set values. For instance, VARIABLE(SIDE) would return the side or sides of a WHIP. The documentation states that this function recognizes the arguments "SIDE", "MPP", "RHOME", "RSIZE", and "ROUTING", but does not state the meaning of these values.

"MPP" is an obsolete concept.

"RHOME", "RSIZE", or "ROUTING" were used for email addressing, but JTLS-GO handles email differently now than when these were created.

The WHIP returns the WHIP name for "RHOME", "MPP", and "ROUTING", and the text "0" for "RSIZE".

The MDP does not recognize or handle arguments of "RHOME", "RSIZE", and "ROUTING", and will just return an empty string for them.

The VARIABLE() function has been modified to recognize the arguments "CLASSIFICATION", "SCENARIO", "SIDE", and "WORKSTATION". This change was made both within the WHIP's Message Browser code and within the MDP, as well within the Message File Parser.

The existing functions CLASSIFICATION() and SCENARIO() are now considered deprecated and should be replaced with VARIABLE(CLASSIFICATION) and VARIABLE(SCENARIO) respectively. They should be removed within the next major or minor release of JTLS-GO. All messages using the CLASSIFICATION() or SCENARIO() functions have been updated to use the VARIABLE() function. The *JTLS-GO Software Maintenance Manual* and the common.mdf file have been updated to reflect these changes.

3.54 JTLS-2024-16949 GIS Tool Import All Selected Terrain Features

A master database was created for a customer. The database was a copy of the GIS Tool's default database with additional Digital Terrain Elevation Data (DTED) provided by the customer.

An attempt was made to use the GIS Tool to import national boundaries, railways, rivers and roads from the new database to a GIS Tool project. Only the national boundaries were imported. The GIS Tool's console window showed that the railway data were read in, that networks were being merged, and that all networks were fully connected. At this point the import abruptly ended, and the railways, rivers and roads were not imported.

Rail and road nodes have an elevation attribute. At the time the import ended, an attempt was made to get the elevation for a rail node. The latitude and longitude of the node corresponded to an area covered by the customer-provided DTED. The attempt to get the elevation from the DTED failed and an exception was thrown, but was not caught, ending the import. There is now a check to catch this exception, allowing the GIS Tool to obtain the elevation from the default ETOPO1 elevation model.

3.55 JTLS-2024-16951 OPM Load Assignments Table Missing TTG Links

The Load Assignments Table does not have Target Type Groups (TTG) as a link.

The link to the TTG table was a minor oversight in the massive project to rewrite the OPMs. The code for the Load Assignment Table has been modified and tested to provide the link.

3.56 JTLS-2024-16952 OPM Aircraft Load Detail Missing Comm Site Count

The Aircraft Load OPM page summarizes the fuel, weapons, sensors, and jammers in the load, and then goes to the load contents page. The summary page does not list the number of Communication Assets in the load.

The code for the summary page has been modified to display the number of Communication Assets in the load. The load contents detail page was also checked for Communication Assets and was already correct.

3.57 JTLS-2024-16953 WHIP HRU/Target OPM Path Links Updated

The new OPM pages use different paths for individual HRUs and Targets, accessed from the WHIP's context-sensitive menus.

The OPM HRU and Target paths to the new links were updated.

3.58 JTLS-2024-16954 WHIP Login Verification Error Displayed Exception

A specific WHIP’s password file did not exist. When the user attempted to login, the WHIP login verification failed, and the message displayed an exception.

The code was improved to expect this exception and display an appropriate error message. The actual cause of the situation has not been determined and is being investigated.

3.59 JTLS-2024-16955 Aircraft Delay Missions Increasing Unit Strength

An aircraft squadron was well below full-strength after losing most of its aircraft in combat. A new mission order was submitted to the squadron. The mission went into Aircraft Delay posture, due to a lack of available aircraft. The Controller Change Unit order was submitted to bring the squadron back to full strength, increasing the number of available aircraft to match the TOE. The mission in Aircraft Delay did not launch, even though sufficient aircraft were available.

The routine that increases a unit's strength failed to check if any air missions were in delay status due to lack of available aircraft. The missing logic was added to invoke the CAN YOU LAUNCH routine, which checks the status of any delayed mission to determine whether it should launch. This new logic applies to airbases, FARPs, and naval vessels that possess aircraft squadrons, as well as separate squadrons, ground, and support units which can have aircraft.

3.60 JTLS-2024-16957 Multi-Send Panel Field Name Display Problem

The user interface for the Multiple Send (MSEND) button has a display problem for the field name to be modified.

The plain text value for the field label name is now used to display the modifying field in the MSEND panel, instead of the HTML label that is used for mandatory fields in an order.

3.61 JTLS-2024-16958 Mission Weapon Load Changes Incorrect

A Change Mission Parameter order was submitted to change the assigned load of an air mission that was in SCHEDULED posture. The messages received assumed that the new load only had new weapons, but the new load actually had new sensors.

The "Model Assigned" aircraft load assignment option of the Change Mission Parameter order sets the load for the mission to the default load for that aircraft and mission type. The message sent to the Player confirming the load change was incorrectly formatted. This was corrected.

3.62 JTLS-2024-16959 Remove Unused Variables

The JTLS-GO Development Team is continually tracking and monitoring code for unused variables and uncalled routines. Several variables were found in the preamble that were no longer used or needed by the currently-implemented algorithms.

The variables were removed. This maintenance effort extends the lifecycle of JTLS-GO.

3.63 JTLS-2024-16960 DDS Migration With New PostgreSQL HBA File

The DDS migration tool did not work with the tightened database server configuration in the new PostgreSQL host-based authentication (HBA) file.

The DCP had a bug where, when attempting to perform DDS migration, Glassfish improperly assumed that the destination database's "sslmode" value was always "false." This prevented connection between the source and destination databases, and has been fixed.

3.64 JTLS-2024-16961 Satellite Movement Infinite Loop

A moving satellite went into an infinite loop, repeatedly moving to the same location.

This could happen under two circumstances:

- If the satellite had a non-surface search sensor on board.
- If the satellite had a weapon on board that had a range of zero.

The code was changed to account for both situations:

- If a non-surface search sensor is on board, it is not considered.
- If a zero-range weapon is loaded, the range of the weapon does not impact the next movement location of the satellite.

  It is quite possible that the satellite will jump over the target it should hit. Satellites should have weapons with reasonable ranges so the satellite does not have to be directly overhead to hit the ground target.

3.65 JTLS-2024-16962 Need To Upgrade Apache For Cyber Approval

A customer could not load JTLS-GO 6.3 until it used the most current version of Apache (Version 2.4.62).

The Apache 2.4.62 was obtained, compiled, and installed on JTLS-GO.

3.66 JTLS-2024-16963 Crash Checkpoint For Fire Mission With ACM Target

The model crashed when taking a checkpoint while the game was holding a Fire Mission with an ACM target area.

The checkpoint write Fire Mission code was not properly handling Fire Missions that are targeting an ACM area or a Supply Convoy at the time of the checkpoint.

The read and write checkpoint code was corrected to properly handle these two cases.

3.67 JTLS-2024-16964 Mixed Case Field Names In Orders

An order had a field name written in lower-case letters, rather than the upper-case standard used by all other orders. Although this does not cause an issue, it does cause confusion and slows down debugging efforts. All field and group names should be written in upper-case letters.

The XML Schema Definition (XSD) files were modified to introduce two simple string types: alpha-numeric upper-case and alpha-number mixed-case. Both types also permit the underscore as a valid character.

The name attribute for fields and groups was modified from being a simple string to being an alpha-numeric upper-case string, and the OEC name field for fields and groups was modified to be an alpha-numeric mixed-case string. Two CEP orders violated the new XSD check and were modified. Numerous DDS orders violated the XSD check and had their order XML file and supporting code modified.

After these modifications, the Unit Vehicle Report failed verification for an unrelated problem. A field within the order retrieves data from the SUPPORT_UNIT table, but this table name was not a recognized entry for the field. The Order Verification Program had the keyword SUPPORT_UNIT added as a legal table for unit field fill requests.

The Aircraft Characteristics Report failed to generate due to the removal of the parameter ac_max_downlink_distance from the database. The retrieval and display of this obsolete field was removed from the report.

The checkorder script would always perform a verification of all files in the DDS and AAR directories when they were specified as the source, even if the command requested only a couple of files. This was also fixed.

3.68 JTLS-2024-16965 Split Mission Not Link 16 Assigned

If a Link 16 mission is split, the split-off mission is not properly assigned a Link 16 network.

The split-off mission is now assigned to the same network as the parent mission.

3.69 JTLS-2024-16974 Sending Message To Non-Existent Pending Task

The model crashed attempting to send a message concerning a Controller Kill Aircraft order. The message was designed to be sent to the WHIP responsible for the Air Mission that was killed.

The problem started when the user had a squadron change sides. The squadron had several active Air Missions. One of these Air Missions was on the ground waiting to take off at a future time as specified by the original order. The logic incorrectly removed all of the tasks from the Air Mission. This caused the mission to be stuck. It had no pending execution tasks (PETs) and therefore nothing could make it take off and fly back home.

The Player got frustrated and killed the Air Mission using the Controller Kill Aircraft order. When the logic attempted to tell the Player that they lost some aircraft, the model used a routine called SEND TO PET FUNCTIONS, which accessed the variable representing the PET without checking if it existed, causing a crash.

This STR does not fix the original problem; instead, it adds code to check if the PET exists before using it within the SEND TO PET FUNCTION Routine. The original problem will be fixed in a future STR.

3.70 JTLS-2024-16976 Add Speed To Air Mission IMT

The speed of an Air Mission should be available on the IMT.

Speed was added to the Air Mission IMT screen.

In addition, there was a need to filter the Air Mission IMT side by region. Because the location of the Air Mission was not a column on the IMT screen, this could not be done. The current location of the Air Mission was also added.

Both new columns were hidden, and the user can display them if desired, but the IMT screen is initially no larger than the old IMT display.

While investigating, a spelling error was noted in the Foreign_Air_Mission_Intel.xml file. It was corrected at the same time.

3.71 JTLS-2024-16977 LC2MS Major Equipment Strength Incorrect

A file generated by the LC2IS Message Service was listing the percent strength of a unit's major equipment as 16%, but was listing the unit's major equipment status as 1, which would indicate a strength greater than 90%.

It was discovered that the code was using the personnel strength to determine the major equipment strength status, rather than using the major equipment strength. The code was modified to use the proper percent strength when converting to the strength status.

3.72 JTLS-2024-16978 OPM Errors

The OPM links to the *JTLS-GO Data Requirements Manual* were pointing to the wrong page, and the graphics symbols were printed at a size which was too small to be useful.

Code changes were made to correct these errors.

3.73 JTLS-2024-16979 Crash Starting ELS After CEP Had Advanced

The Entity Level Server (ELS) crashed when it had been started long after the CEP had begun execution. The CEP had been running long enough to have newly-created HRUs and Convoys in the game.

The crash occurred because the ELS was attempting to cancel an update event for the new objects, but no such event existed. The new HRUs and convoys had never been updated, because they were new to the ELS at the time when the execution started for the ELS. As such, no update event existed, and canceling the non-existent events caused the ELS to crash.

Code was added to not attempt a cancellation unless the event actually existed.

3.74 JTLS-2024-16980 ATO Constants Missing NEC CCIS Mission Type

The ato_constants file contains pre-defined codes and values for mission types used in the ATO. This file is used to decode the mission types found in the typical ATO messages that are processed by the ATO Translator. The "FCF" code for a mission used by NEC CCIS was not included in this file.

Added the "FCF" mission type code to the ato_constants file and associated it with the Reconnaisance mission type in JTLS-GO.

3.75 JTLS-2024-16981 ATOT Writes Bad Mobility Mission Stop Location

The ATOT may receive a mission from the ATO which contains a stop location for a mobility mission. The stop location ICAO or ACM may be unknown. In this case the ATOT writes bad data as the stop location.

The ATOT was modified to fill the location properly when the location name is known, but will leave the location empty when it is unknown. The operator will need to properly translate the named location when it is unknown, so the ATOT can fill the location properly.

3.76 JTLS-2024-16983 Alter Terrain Correcting Dual Capable Flag

A grid's Dual Capable Flag indicates that the terrain is both land and water. Any grid that is not labeled as Ocean or Small Island needs to have its Dual Capable Flag set to "Yes" when the grid has a depth specified and is categorized as any of the various land terrain types. This was not always happening.

The routine that accomplishes the Alter Terrain logic was reworked and the logic error corrected. The routine checks to ensure that the Dual Capable Flag is properly set, based on the final values assigned to the grid, prior to completing.

3.77 JTLS-2024-16985 ATO Parser Does Not Process Altitude

The NEC CCIS ATO Parser reads records for each mission in the ATO. For missions that use a "PTRCPLOT" record, the Parser was not discovering and processing the altitude held in Field 3 of this record. The resulting ACM did not get an associated altitude when it was saved for processing by the ATOT.

This Parser was not processing "PTRCPLOT" records completely. Field 3 of this record holds the specified altitude. The Parser was changed to read and process this field for the altitude values and store them with the enclosed ACM.

3.78 JTLS-2024-16989 JDSP Bad Memory Values In Error Message

A user received an error while using their JODA client, which is linked with the JTLS-GO 6.3 JDSP libraries. The error reported that the memory was exhausted, but gave erroneous values for the memory addresses.

The error was valid for exhausted memory, but the print command was using an improper data type for writing the memory addresses used for debugging. The print command was modified to properly format the memory values when this error occurs and is written.

3.79 JTLS-2024-16990 SUP OPM Column Vague

There is a column "Targets Using This SUP" on the Ship Unit Prototype (SUP) OPM page. The entries in this column are all "none."

The "Targets Using This SUP" column has been relabeled "Category Ship Targets Using This SUP". Because it is rarely used, it has been moved to the end of the row for each SUP. This data item represents a ship hull, such as might be found in a graveyard of old ships, or a single ship that has been destroyed or decommissioned.

The "Owned Targets" column was relabeled "SUP Owned Targets". This represents the special equipment that is automatically added to the SUP when it is created in the game (such as radars, large guns, SSM launchers, etc).

3.80 JTLS-2024-16991 Port Amphibious Operation Does Not Work

A Port amphibious operation did not take place.

The Port Operation should have worked like a Beach Operation, using the port throughput numbers rather than the Over Beach numbers. There is no code in the routine PORT AMPHIBIOUS OPERATION - the routine simply canceled the task.

The routine START AMPHIBIOUS OPERATION calls the routine PORT AMPHIBIOUS OPERATION when a port operation is specified. It was determined that this call could be replaced by calling the existing routine START BEACH OPERATION. A new PORT FLAG argument was added to START

BEACH OPERATION to distinguish between whether a port landing/pick-up was taking place (PORT FLAG is YES), or this was an actual beach operation (PORT FLAG is NO). Port or Over Beach throughput numbers would then be applied depending on the value of PORT FLAG.

Because of this new argument, the routine START SHIP BEACH WAVE and the event AMPHIB OPERATION DONE had to be modified to account for the PORT FLAG. The routine PORT AMPHIBIOUS OPERATION has been removed.

3.81 JTLS-2024-16992 CAT Terrain Non-Visible Values Missing

The CAT.TT.PERCENT.NONVISIBLE parameter value is not in the OPM. This parameter is the percentage of Combat Systems using the specified Combat Arms Type (CAT) that are not visible when located in the specified Terrain Type.

The CAT.TT.PERCENT.NONVISIBLE values should a child table of the CAT data. When selected, the OPM should link to the table and highlight the records for that CAT and all Terrain Types. The code has been modified to accomplish this.

3.82 JTLS-2024-16993 Magic Air Ops Reference Limit

The Change Mission Parameter order and the Manage Air Mission Tasks order both accommodate a 25-character Reference field length. The Magic Air Ops order is limited to only a 12-character Reference field, which prevents pasting of full mission names. For consistency and ease of use, all three orders' Reference fields should be the same length.

The Reference field was increased from 12 to 25 characters in the Magic Air Ops order.

The Help text in all three orders was updated to reflect 25 characters instead of 12.

3.83 JTLS-2024-16995 Mixed Perceived/Common Attributes Logic Error

The CEP generated a logic error when sending Air Mission updates to the JODA. When the mission type was changed, the CEP sent updates for the type and the symbol assigned to the mission.

The mission type was a common attribute, and the symbol was a perceived attribute. The updates for these two attributes were separated into individual updates, which prevented the logic error from occurring.

3.84 JTLS-2024-17000 Munitions Report Incomplete Information

The data contained in munition reports did not have accurate counts for the weapon expenditures by units. All of the counts were reported as zeroes, regardless of weapon type and method of usage.

The code used to tabulate the weapon counts was incomplete. When munitions were used by units, the structure used to hold the expenditures was never updated with the amounts used. Modifications were done to account for the weapon usage and to provide these data in the Munitions Report.

3.85 JTLS-2024-17001 Cannot Save AAR Reports To HTML

The user has a choice to save an AAR report as either raw XML or HTML. The raw XML option works. The HTML option does not work. The resulting HTML only contains the table and cell borders of the report but no data.

The method used to generate the HTML file did not retrieve the data values to allow them to be printed in the report. A different method was substituted and the report now generates properly in HTML.

3.86 JTLS-2024-17003 Compute Supply Amount Missing Code Used

When a unit was moved, either through airlift or amphibious operation, the weight of supplies was computed. In some cases the needed supplies did not match the current amount on hand. The code to make this calculation was incomplete.

Additional code was implemented to add up the amount of needed supplies to support the Combat Systems in the unit. This code was only used when the unit had an unlimited amount of supplies that were used for specific Combat Systems.

3.87 JTLS-2024-17004 OPM Bad TPFDD Unit Table

There are a number of errors or missing data in the OPM TPFDD Page:

- Not printing out the alternate base for Airbases - only for Naval Units and FARPs.
- Using an index value as a row identifier.
- Missing runway information for Airbases.

The code was modified to properly print out the alternate landing base for Airbases.

The row index value was changed to use the Unit Short Name for the TPFDD unit.

A new page was created for listing the Airbases and the runways they use. The TPFDD page now has a link to the new Airbase/Runway page. Also links were created from the runway names in the Airbase/Runway page to the corresponding individual runway target page. The "Runway" link in the Airbase-specific page was also modified to link to the new Airbase/Runway Page.

Several other coding changes were made to improve efficiency and speed.

3.88 JTLS-2024-17005 HRUs Starting Next Task Did Not Vacate Shelters

When an HRU began to execute its next task, it did not leave its protective shelter as expected.

The routine which handled the execution of HRU tasks contained a flag to signal that the HRU should vacate any protective shelters, but this flag was not being used. The code was modified to instruct the HRU to leave the shelter if it was appropriate to do so.

3.89 JTLS-2024-17006 ATOT Not Using ACMs For Attack Missions

The ATOT is not using ACMs when building the attack missions that call for ACMs in the ATO. When the translation is performed, any attack mission that is suppose to attack an ACM is instead created as an attack of a location point.

The ATOT uses a routine to build the utility order for tasking of a mission. This routine was not considering ACM types of targeting. The default type was simply to use an attack location when Unit, Target and JDPI were not used.

This routine was modified to consider ACMs in writing the utility for attack objects for the OAS order.

3.90 JTLS-2024-17007 Mobility Mission Actions Missing From MISREP

An Interim Mission Report (MISREP) was requested for a Mobility air mission. The report contained an incomplete reference to a transport action and omitted all tasks completed by the mission.

The Interim MISREP is designed primarily for non-Mobility missions. The Air Movement Report is designed specifically to monitor currently executing Mobility missions and details the mission's lift-related tasks. However, it is not unreasonable to expect the MISREP to contain entries for each completed load/offload task in a Mobility mission.

Investigation revealed several related issues that required correction:

- Completion of Unit pickup and delivery (i.e. airlift/airdrop option) was missing from the final MISREP as well as the Interim MISREP.
- Completion of supply loading & offloading (i.e. air transport option) was missing from the Interim MISREP.
- Completion of HRU movement (i.e. insert/extract) was missing from the Interim MISREP.

All missing mission activities were added to the MISREPs in chronological order to include the units/HRUs or supplies involved at each load/offload event.

3.91 JTLS-2024-17010 Correct Multiple Range Item Updates

The model is sending out multiple unnecessary range item updates.

One attribute of a Range Item is whether the Range Item has changed, and an update is needed. If an update is needed, the update is sent to the JODA. The problem was the internal attribute was not being reset to "No" and in future looks at the range Item, the model believed an update had been made.

The code now correctly resets the internal variable to indicate no update is required.

3.92 JTLS-2024-17011 OPM Unit Hierarchy Improvements

The OPM interactive unit hierarchy required updates to address existing bugs and improve user experience and usefulness.

The OPM unit hierarchy has been improved in the following ways:

- Increased the size of unit symbol icons;
- Added a political country flag icon to each unit;
- Added a search function to the chart;
- Added the ability to navigate to a unit in the unit hierarchy chart from that unit's OPM page;
- Fixed a bug where navigating to a unit's OPM page from the unit hierarchy did not work;
- Fixed a bug that caused a black background when making the chart full-screen.

The following unit types now write additional data:

- Ground Combat Units and Support Units now write out command level and strength.
- Airbase and FARPs now write out ICAO codes
- Squadrons now write out aircraft type and home base.

The OPM was also modified to:

- Increase the size of unit symbols;
- Add conditional queries to data tables;
- Add styling to reduce white space and make page headers easier to read.

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 6.3.3.0.

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 6.2.

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 exist 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
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
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 6.3.0.0 DATABASE CHANGES

Refer to the JTLS-GO 6.3.0.0 Version Description Document (VDD) for the list of database changes between the JTLS-GO 6.2 series and the JTLS-GO 6.3 series.

APPENDIX C. VERSION 6.3.2.0 REPOSITORY CHANGES

No significant changes were made to the JTLS-GO 6.3.2.0 repository.