JTLS-GO

Version Description Document

September 2024

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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 6.3.2.0)

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

JTLS-GO 6.3.2.0 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 6.3.2.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO®) software suite. JTLS-GO 6.3.2.0 is a Maintenance delivery for the JTLS-GO 6.3 series of releases.

JTLS-GO 6.3.2.0 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 6.3.2.0 executes on the Red Hat Enterprise Linux Version 9.4 and Oracle Linux 9.4 64-bit operating systems. The Web-Hosted Interface Program (WHIP®) user workstation interface can be executed on any 64-bit operating system from any Java-compatible Web browser.

1.2 INVENTORY OF MATERIALS

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

1.2.1 Obsolete/Outdated Documents

No documents have been deleted or become outdated as a result of this release.

1.2.2 Unchanged Documents

- *JTLS-GO Analyst Guide* (JTLS-GO Document 01, Version 6.3.0.0)
- *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 Data Requirements Manual* (JTLS-GO Document 05, Version 6.3.0.0)
- *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 Player Guide* (JTLS-GO Document 12, Version 6.3.0.0)
- *JTLS-GO Standard Database Description* (JTLS-GO Document 14, Version 6.3.0.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 6.3.0.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, 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 DDS User Guide* (JTLS-GO Document 06, Version 6.3.2.0)
- *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 Version Description Document* (JTLS-GO Document 17, Version 6.3.2.0)

1.2.4 New Documents

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

1.2.5 Delivered Software Components

JTLS-GO 6.3.2.0 may be delivered either on a CD or as a set of compressed TAR files to be downloaded. Either method includes the complete suite of software executable code and command procedures. The following software components are included with this release:

- Combat Events Program (CEP)
- Scenario Initialization Program (SIP)
- Interface Configuration Program (ICP)
- Reformat Spreadsheet Program (RSP)
- JTLS Symbols Application (JSYMS)

- Database Development System (DDS)

    Database Configuration Program (DCP)

    DDS Client User Interface (DDSC)

- ATO Translator Service (ATOT)
- ATO Generator Service (ATOG)
- ATO Retrieval Program (ATORET)
- JTLS Convert Location Program (JCONVERT)
- Count Critical Order Program (CCO)
- JTLS HLA Interface Program (JHIP)
- After Action Review Client (AARC)
- Scenario Data Client (SDC)
- Order Entry Client (OEC)
- Order Verification Tool (OVT)
- JTLS Object Distribution Authority (JODA)

    The current JODA build number is 214.

- Web Services Manager (WSM)
- Web-Hosted Interface Program (WHIP) and its component programs:

    Apache Server (APACHE)

    JTLS XML Serial Repository (JXSR)

    Order Management Authority (OMA)

    Synchronized Authentication and Preferences Service (SYNAPSE)

    XML Message Service (XMS)

    Total Recall Interactive Playback Program (TRIPP)

- Entity Level Server (ELS)
- JTLS Operational Interface (JOI) for both OTH-Gold and Link-16 generation

- Tactical Electronic Intelligence (TACELINT) Message Service
- Keyhole Markup Language (KML) Operational Interface (KOI)
- JTLS Transaction Interface Program (JTOI)
- JTLS Interface Network Navigator (JINN)
- JTLS Order of Battle Editor (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 6.3.2.0 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 6.3.2.0 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 6.3.2.0 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 6.3.2.0 is delivered with the Adoptium project Temurin Java Development Kit (JDK) 1.8 Update 422 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 6.3.2.0. 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 6.3.2.0, 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 6.3.2.0 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 6.3.2.0 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 6.3.2.0, 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 6.3.2.0.

2.0 ENGINEERING CHANGE PROPOSALS

This chapter summarizes the minor model capabilities added to JTLS-GO 6.3.2.0 as a result of implementing authorized Engineering Change Proposals (ECPs).

2.1 JTLS-2024-16817 Side Border Colors For Super WHIP Components

Summary of Model Change Request

Changing the Force Side perspective in a Super WHIP does not update the component border color in the following WHIP components: Map, Command Hierarchy, Logistics Hierarchy, IMT, and ATO Generator, Translator, and Viewer).

Design Summary

All relevant Super WHIP components now dynamically update their border color to the color of the corresponding Force Side whose perspective is selected.

2.2 JTLS-2024-16853 Reformat WHIP Password Reset Dialog

Summary of Model Change Request

The password characters requirement of the current WHIP password reset dialog is confusing to the user.

Design Summary

The dialog was reformatted with bullet points to communicate to the user that passwords must match one of each of the following criteria:

- 8 – 30 characters
- At least one uppercase letter
- At least one lowercase letter
- At least one digit
- At least one symbol
- Passwords must match

As the user dynamically types in the password field, each requirement will highlight green and the bullet point will change to a checkmark. All requirements must be checked for the system to accept the password (unless 'NONE' is accepted).

2.3 JTLS-2024-16854 LC2IS Message Service Modifications

Summary of Model Change Request

Two minor modifications were requested for the LC2IS Message Service, related to the new database C4I Name for units:

- The force description “surnom” (French for “nickname”) tag should always use the unit short name in the SIF output.
- When a unit's short name and C4I name are different, both should be displayed on the web browser interface.

Design Summary

The modifications improvements have been implemented.

2.4 JTLS-2024-16858 Add Satellites To Range Ring Filter Panel

Summary of Model Change Request

Satellite range rings can only be displayed on the Map from their respective context-sensitive menus; users must click on each satellite individually on the Map to display their range ring.

In addition, the map.colors data file contains numerous unused range ring filter default colors.

Design Summary

A Satellite option was added to the Range Ring filter tab, so users can display multiple satellite range rings at once. Users are able to display and filter satellites by type and Force Side. Satellite range rings can currently be displayed and filtered for Targetable Weapons, sensors, and jammers.

The map.colors file was cleaned up to remove any unused range ring filter colors, and added a new default color (yellow) for satellite range rings.

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-2024-16813 Online Player Manual Corrections

As a result of JTLS-GO user input several issues were identified in the new Online Player Manual (OPM) format. Many of these issues included the slow display of some large data tables that required a reorganization of the OPM files

This STR is similar to STR JTLS-2024-16764 released as part of JTLS-GO 6.3.1.0. There are too many problems to list individually. As JTLS-GO 6.3 continues to go through testing to support its first major exercise in October 2024, issues are coming in daily to improve the performance and rearrange the new OPM to better meet the needs of JTLS-GO users.

3.2 JTLS-2024-16814 WHIP/DDS Map - Minor Fixes

Several issues existed in the Web Hosted Interface Program (WHIP) and the Database Development System Client (DDSC) map component:

- Unit names clutter the WHIP Map on startup.
- The STARTUP_VIEW file contains the default map view values on startup. This file should never be modified by the user.
- Users can create and save Map view/location/filter files without specifying a name.

Each of these issues were corrected under this single STR.

- Unit names are now filtered to “Off” by default on startup.
- Users are now prohibited from manually renaming, overwriting, or deleting the STARTUP_VIEW file. An error message in the WHIP or DDSC appears if the user tries to modify the file.
- Creating or renaming view/location/filter files with empty names is now prohibited. An error message appears if the user tries to do so.

3.3 JTLS-2024-16816 HRU Pickup/Delivery Supplies

Several issues existed in the quickly developed logic for pick up and drop off of supplies by HRUs:

- Picking up unavailable supplies from a unit caused the Combat Events Program (CEP) to crash.

- Delivering unavailable supplies from an HRU to a unit caused the CEP to crash.
- The calculated load/offload time was wrong.
- Supplies were not actually being delivered from the HRU to the unit during a Drop-off event.

Each of these issues were corrected.

- The crash caused by picking up unavailable supplies was caused by an incorrect variable in the logic, and was fixed.
- The crash caused by dropping off unavailable supplies was caused by code that should not have been executing, given the situation. The code is now properly bypassed.
- The incorrect calculated load/offload times were caused by an additive operator being used when a multiplicative operator should have been used. The operator was corrected.
- Supplies were not being added to the "Due In" column, which marked them as being delivered, in the proper routine. This caused an empty "Due In" situation, which was corrected.

3.4 JTLS-2024-16818 Programs Failed To Connect As JODA Peer

The Entity Level Server (ELS) and JTLS Satellite Service (JSAT) failed to connect to the JTLS Object Distribution Authority (JODA) when compiled using Linux 9.4. They both used a connection to the JODA for the purpose of sending orders to the CEP.

For these programs, the JODA connection failed when initializing the Order Verification Tool (OVT) library. This failure was due to a floating point exception.

The OVT library uses a standard library which treated the value of "Not-A-Number" (NAN) differently in Linux 9.4 than was done in earlier versions of Linux. As a result, the OVT library would not initialize until code changes were made to handle the floating point environment. Once the proper environment was established, the floating point exception was averted and the programs connected to the JODA as expected.

3.5 JTLS-2024-16819 JHIP Crashes Reconnecting To JODA

The JTLS-GO High-Level Architecture Program (JHIP) uses the OVT library for validating orders prior to sending them to the CEP, via the JODA. The OVT library is linked with the libxml2 libraries on the system, which are known to have a bug in determining an internal constant for the NAN value. During testing to determine whether this libxml2 bug is present in the JHIP, the JHIP was manually disconnected from the JODA, resulting in a JHIP crash.

The JHIP crash was caused by multiple initializations of the OVT library. Each time the JHIP reconnected to the JODA during a single run, the JHIP reinitializes the OVT library, even though. library only needs to be initialized once for each run of the JHIP.

The JHIP was modified to initialize the OVT library only once, at startup.

3.6 JTLS-2024-16820 HRU Ambush Convoy Issues

An HRU was tasked to ambush a rail convoy, and failed to destroy any of the railcars when the train arrived within range of the HRU's weapons.

A flaw in the logic prevented the HRU from carrying out the ambush task when a convoy, of any type, first arrived or was seen by the HRU. These two situations were missing from the logic that set the ambush flag to “Yes”, thereby skipping the ambush process. The missing convoy situations were added to the logic to start the ambush and assess any direct damage.

Secondary blast damage was not possible in the existing logic when the convoy was moving. The logic was incorrectly using the location of the convoy’s receiving unit to determine how far the weapon impact point lay from the convoy location. This error made the HRU’s weapons out of range for secondary blast damage assessment.

The code was corrected to use the location of the moving convoy when determining whether each weapon was in range. If the convoy was loading or unloading, the code correctly used the location of the donating or receiving unit to place the convoy assets relative to the unit and calculate the distance between the weapon impact point and each asset.

3.7 JTLS-2024-16822 HRU Combat System Range Rings Not Available

HRUs only have range rings for their tactical intelligence collection range. HRUs possess Combat Systems, and the new range ring capability does not show the HRU’s Combat System range rings.

The Combat System range rings were added and now can be displayed on the WHIP.

3.8 JTLS-2024-16823 SDR Database Initialization Terminal Closes

When a Scenario Database Repository (SDR) database is initialized from the Javamenu, an command terminal is launched in which the sdr_action script is executed. This script performs the actual SDR initialization. Before initialization is started, a series of error checks is performed. If an error is found, a message is written out, and the user is prompted to enter a carriage return to continue and close the terminal. This allows the user the time to read the message and take corrective action.

In one case, the sdr_action script found an error and printed a message, but did not prompt the user to enter a carriage return. Instead, the terminal closed before the user could read the message.

The sdr_action script contains a function that pauses the script when an error is found, until the user presses a carriage return. One of the error checks determines whether any existing process is currently accessing the SDR database, but this error did not call the pause function.

The AAR Glassfish was running when the user initiated the sdr_action script. The script properly determined that the AAR Glassfish was connected to the SDR database, but the script did not properly pause resulting in an immediate termination of the terminal.

The pause function is now called in all error situations detected by the sdr_action script.

3.9 JTLS-2024-16824 Unit Hierarchy Report Context-Sensitive Menu

A feature of messages within the Message Browser is the ability to bring up the context-sensitive menu for any unit name that appears in the message text. This feature did not work in a "Command Unit" Report for any on the listed subordinate units.

The problem was that the message as formatted so the indentation symbol (-->) for each hierarchy level was next to the unit's name. This prevented the Message Browser from recognizing the unit's name. The format of the message was changed to put a space between the indentation symbol and the unit's name, solving the problem.

The change was only put in the English version of the message. The MTF version of the message was not changed due to the rules of MTF message format.

3.10 JTLS-2024-16825 Downed Pilot Detects Aircraft That Shot It Down

An air-to-air engagement resulted in an aircraft being shot down, and a downed pilot HRU being created. In preparation to report its current status, the downed pilot attempted to detect not only the enemy aircraft that shot it down, but the remainder of the air mission to which it had once belonged. The downed pilot did not have enough sensor capability to accomplish the detection and the model crashed.

The code was corrected to check for the situation and not attempt to make the infeasible detection.

3.11 JTLS-2024-16826 Leave Formation Message Typo

There was a misspelled word in the message generated by the Leave Formation order.

The misspelled word was fixed in the message definition file.

3.12 JTLS-2024-16828 All Load Assignments Incorrect Change

The Set Load Assignment order allowed the Controller to change the load assignments for all target categories. This option also changed the load assignments for the extended target type

groups, such as the AWACS Load and Tanker load. It should have only changed the assignments for the main target type groups used by offensive missions.

When the option to change all "Target Category" assigned loads, the code was modified to change only the offensive mission target type groups, and not the extended target types. The extended target type loads can still be altered using the Order's specific load assignment option.

3.13 JTLS-2024-16832 Refuel Mission Quick Order Mission Select

The Refuel Mission quick order did not allow the user to select air missions from different Force Sides, even when their side relationship was Friendly.

Both Manage Air Refueling Chits order and the Refuel Mission Quick order were missing the criterion to allow missions from a friendly Force Side to be selected. Each order was corrected.

3.14 JTLS-2024-16834 Dashboard Service Fixes

Two issues were discovered with the Dashboard service:

- The Dashboard Apache module was writing errors to requesting clients in XML, but successful responses are written in JSON.
- The Dashboard service was not handling a connection refusal from the JODA, which ignores errors that should not be.

The Dashboard Apache module was updated to use JSON for both errors and standard responses.

The Dashboard service now handles connection refusal events from the JODA.

The Dashboard client error page format was enhanced.

3.15 JTLS-2024-16835 Missing Report Tables When Creating New Database

When the user selects the "Create a New Database" option from the Database Development System Menu, the Glassfish-related report tables and related sequences were not created.

The related SQL script was modified to create the Glassfish-related report tables and related sequences.

3.16 JTLS-2024-16837 TRIPP IMT Menu Not Consistent With WHIP

The Total Recall Interactive Post-Processor (TRIPP) Information Management Tool (IMT) menu did not have the Satellite Assets IMT table available. The TRIPP used a different definition of the IMT menus than the Player and Controller WHIPs.

The IMT menu definition was centralized so the TRIPP and WHIP use the same IMT definition file.

3.17 JTLS-2024-16838 Reporting Impact When Firing At Location Crash

If the Entity Level Server (ELS) is running and a user fires at a location, the model crashes.

This problem was fixed in JTLS-GO 6.2, but was never moved into JTLS-GO 6.3. The code from JTLS-GO 6.2 was moved into the JTLS-GO 6.3 code base.

3.18 JTLS-2024-16840 Replay JXSR Wrong End Time During CEP Restart

When the user stops the CEP without taking a checkpoint, and restarts the game from the last checkpoint of the previous run, the TRIPP's time slide-bar should show the end time for the last checkpoint of the selected run. However, the slide-bar still showed the end time from the last execution of the CEP, incorrectly implying the selected run would continue to a later, unavailable time.

Whenever the CEP is restarted from the last checkpoint of any existing run, the CEP rewrites the scenario run file. However, none of the contents of the file have changed because the CEP will continue writing checkpoint numbers onto the selected run in the file. When the CEP was restarted in this way, the Replay JXSR assumes that this was a simple checkpoint event, because the list of runs had not been changed in the run file for the scenario.

The Replay JXSR was modified to consider this special case. If nothing has changed, the Replay JXSR understands the CEP was restarted. Otherwise, the JXSR assumes the CEP simply took a checkpoint. The Replay JXSR will properly conduct a reset in case of a CEP restarting in this way.

3.19 JTLS-2024-16841 MDP GUI WHIP Names Fail To Populate

The Message Delivery Program (MDP) GUI delivery panel displays a list of WHIP names for filtering. This list was failing to populate because the database file was moved.

The MDP GUI Delivery panel now successfully populates the list of WHIPs.

3.20 JTLS-2024-16842 Message/Report Browsers Static Table Heights

The table heights in the Message and AAR Report Browsers are set to a fixed size, even if the divider location is moved.

The table height was updated to change dynamically as the user changes the divider location.

3.21 JTLS-2024-16844 Crash OPM Link 16 Missile Type

There are two Link 16 database parameters for Air Defense Classes. The OPM was attempting to write out the Link 16 Missile Type for an Air Defense Site when the OPM generation procedure crashed.

The OPM generation code was improperly accessing the Air Defense Class Link 16 Site Type, when attempting to print the Link 16 Missile Type. The proper variable is now accessed

3.22 JTLS-2024-16845 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 that it expected to see a Red ship at this location, but the Red ship is not there.

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

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

3.23 JTLS-2024-16846 NULL Allowed Variables OPM Crash

Some variables are allowed to be NULL within a JTLS-GO database, such as the default icon database parameters, the Common Operational Picture (COP) Force Side, and the Downed Pilot HUP. The new OPM code was not checking if these data parameters were filled or NULL. If NULL, the OPM generation procedure crashed.

The OPM generation code now checks if the database parameter is set or not. If not set, an appropriate entry is now made on the required OPM page.

3.24 JTLS-2024-16847 Exercise Log - Admin Classification Link Fix

A link to the "Classification" edit page in the admin section of the Exercise Log incorrectly navigated to the "Status" edit page.

The "Classification" link now navigates to the correct edit page.

3.25 JTLS-2024-16848 Ballistic Missile Altitude Issues

JTLS-GO was linked to a real-world C4I system that could display the flight profile for ballistic and hypersonic missiles. Algorithm improvements were needed for more realistic flight profiles.

The algorithm changes were made.

During testing, it was also discovered that the new hypersonic variables were not transferred to a new Targetable Weapon that is created during game play. This transfer is now accomplished for newly-created Targetable Weapons.

3.26 JTLS-2024-16849 Civilian Traffic Order Crash

The Civilian Traffic order crashed the model.

The problem was caused by a change in the Air Route definition made to support other changes in JTLS-GO 6.3.

Normally Air Routes are specified as Utility Directives, submitted when air orders are submitted. The Civilian Air Mission order automatically generates an Air Route for every Civilian Air Mission. This is done by the model to ensure that the Civilian Air Mission does not immediately transfer to its destination squadron. This allows the exercise to cancel missions and have them return to their starting location.

The model code to generate this automatic route was not changed to match the new Air Route format requirements. The problem was fixed.

3.27 JTLS-2024-16850 Detached Squadrons Missing Link 16 Network

When a Link 16 aircraft returns to a different base from which it was launched, a new detachment squadron is created. The assigned Link 16 mission number and track block are correctly given to the newly detached squadron. The problem is that the squadron is not placed on the Link 16 network assigned to the aircraft, so the Link 16 Mission block never gets used.

The Link 16 network assigned to the mission's home squadron is now given to the mission's newly detached squadron.

3.28 JTLS-2024-16852 OVT Library Crash When Releasing Memory

When the HIP disconnects and reconnects to the JODA, it causes a fresh re-read of all the order files. This first involves releasing the memory associated with any previous reads of the order files. When this step happened, a crash would occur attempting to access an invalid order.

The code to release memory had never been updated from when the OVT library would read the orders.def file, generated from the XML order files, and most of it had been commented out. The crash happened when attempting to determine if an order existed by looking whether the name had been set. This was the old way to determine if an order existed.

The OVT Library now reads the XML order files directly, and whether an order exists or not is checked by whether the pointer is non-zero. This change was made along with updating the code to properly release the memory structures that are now used.

3.29 JTLS-2024-16855 DDS Edit Ground Visual Sensor On IIP Table

Double-clicking the Ground Visual Sensor and Air Visual Sensor fields of the Intel Information Prototype (IIP) table in the DDSC has no result, so the two fields cannot be edited.

Both of these fields had their editing capabilities inadvertently removed from the IIP screen definition file, because they had pointed to an obsolete field in the Sensor Type table to determine whether the sensors were for ground or air.

The editing capabilities were restored.

- For the IIP GROUND VISUAL SENSOR, The user is now only allowed to select a sensor that has an ST SURFACE RANGE value greater than 0.0.
- For the IIP AIR VISUAL SENSOR, the user is now only allowed to select a sensor that has an ST AIR RANGE value greater than 0.0..

If the same visual sensor has both ranges specified, the same sensor can be used for as the default surface visual sensor and default air visual sensor.

3.30 JTLS-2024-16856 DCP/ICP Customize Alternative Map Shape File

The DCP/ICP Customize Alternative Map Shape File dialog is confusing. It implied that the user could add multiple shape files through the dialog, but only the first shape file specification is used, and subsequent files are ignored.

The dialog was modified so that the user can only add one alternative shape file for a given scenario.

3.31 JTLS-2024-16857 ICP Generates OPM Password Modifying WHIP

The ICP generated the Apache password used to access the OPM when the ICP was used to modify a WHIP's Password even if the ICP was not saved.

The ICP code was changed so that the OPM password is no longer generated when the ICP is used to modify a WHIP's password. The OPM password is now only generated when the ICP is saved.

3.32 JTLS-2024-16859 AAR Data Convoy Status Change Incorrect

Whenever a convoy changes its status (such as from "moving" to "offloading"), an event is sent by the CEP to the AARC to record the new status. This is showing up within the AAR tables as "ITOT.F(NEW.STATUS)" instead of being the actual enumeration for the new status.

The value sent to the AARC is within quotes in the code, instead of being only the command to switch the status enumeration to a text and send it. The incorrect quotes were removed from the code.

3.33 JTLS-2024-16861 Link 16 Crash In Process Special Message

The Link 16 Message Service forwards Engagement Status messages from the CEP to the Link-16 Common Operational Picture (COP). The Link 16 Message Service was crashing during the process if the reporting object was an Air Mission.

The Link 16 Message Service now correctly handles the processing of a special engagement message from the CEP in which the reporting object is an Air Mission.

3.34 JTLS-2024-16863 LOGFAS Force IDs Change In LOGFAS XML File

The LOGFAS Force Code IDs are not constant throughout the LOGFAS XML Initialization file.

The logic error causing this issue was identified and fixed. The routine that creates the LOGFAS Code IDs was being called twice in the XML file generation code. The Force ID assignment routine now checks to determine if the unit or HRU has been assigned a Force Code ID. If so, the routine does not generate a new Force Code ID.

3.35 JTLS-2024-16865 Minefield Damage Not In SDC Tables

When a unit encounters an unknown minefield, it sustains initial damage before switching to a mine clearing mode. This damage can be viewed in the IMT Combat System screens, but is not reflected in the unit's strength within the SDC tables.

Upon suffering damage, the CEP was not updating the unit's strength and sending the update to the JODA, so the SDC was never informed of the change. The CEP was modified to update the unit strength after the assessment of the minefield damage.

3.36 JTLS-2024-16867 SDC Indicates Translation Errors

The SDC generated a log item for a Range Class of 37 as "unrecognized".

The static_voc.xml file, which defines all the possible range ring values, did not include the target range ring types. These types were added to the static_voc.xml file.

3.37 JTLS-2024-16868 Link 16 WHIP Subordinate JMessages In Filters Label

The Link 16 WHIP Module allows users to filter by JMessage type. Five JMessages that are filterable by the service have subordinate messages that only make sense if they are sent along with the parent message. However, this was not communicated to the user.

The Link 16 WHIP Module filter table now displays any subordinate messages supported by a parent message. If the parent message is filtered off, then the subordinate message is implicitly filtered as well.

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 6.3.2.0.

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 The 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
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.