JTLS-GO

Version Description Document

May 2022

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

**JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 6.0.8.0)**

[Blank Page]

ABSTRACT

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

This *JTLS-GO Version Description Document (VDD)* describes the new features of the Version 6.0.8.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 6.0.8.0 is a Maintenance release of the JTLS-GO 6.0 series that includes an updated repository of standard data, a demonstration scenario based in the western Pacific. There were no functional improvements implemented as Engineering Change Proposals (ECPs) within this release, For consistency with other release VDDs, an empty Chapter 2 is included in this document. Code modifications that represent corrections to known Software Trouble Reports (STRs) are described in Chapter 3. Remaining and outstanding STRs are described in Chapter 4.

This publication is updated and revised as required for each Major or Maintenance version release of the JTLS-GO model. Corrections, additions, or recommendations for improvement must reference specific sections, pages, and paragraphs with appropriate justification and be forwarded to:

> JTLS-GO Development Team Leader
> ROLANDS & ASSOCIATES Corporation
> 120 Del Rey Gardens Drive
> Del Rey Oaks, California 93940 USA
> jtlsgo@rolands.com

[Blank Page]

TABLE of CONTENTS

1.0 INTRODUCTION

1.1 SCOPE

This *JTLS-GO Version Description Document* (VDD) describes Version 6.0.8.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO®) software suite. JTLS-GO 6.0.8.0 is a Maintenance delivery for the JTLS-GO 6.0 series of releases.

JTLS-GO 6.0.8.0 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 "wespac60". Database modifications that were accomplished to upgrade the previous JTLS-GO database format to the JTLS-GO 6.0 series data format were summarized in the VDD for JTLS-GO 6.0.0.0. No data format changes were made between JTLS-GO 6.0.0.0 and this Maintenance release.

There are no Engineering Change Proposals (ECPs) implemented for this release, but for consistency between this VDD and other provided VDDs, an empty Chapter 2.0 is included. All software corrections are summarized in Chapter 3.0.

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 6.0.6.0)

- *JTLS-GO Air Services User Guide* (JTLS-GO Document 03, Version 6.0.0.0)

- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 6.0.4.0)

- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 6.0.1.0)

- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 6.0.0.0)

- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 6.0.6.0)

- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 6.0.4.0)

- *JTLS-GO Standard Database Description* (JTLS-GO Document 14, Version 6.0.0.0)

- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 6.0.2.0)

- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 6.0.3.0)

- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 6.0.4.0)

- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 6.0.6.0)

- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 6.0.0.0)

- *JTLS-GO DoD Architecture Framework* (JTLS-GO Document 22, Version 6.0.2.0)

- *JTLS-GO Configuration Management Plan* (JTLS-GO Document 24, Version 6.0.0.0)

1.2.3 Updated Documents

- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 6.0.8.0)

- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 6.0.8.0)

- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 6.0.8.0)

- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 6.0.8.0)

- *JTLS-GO DDS Training Manual* (JTLS-GO Document 23, Version 6.0.8.0)

1.2.4 Delivered Software Components

JTLS-GO 6.0.8.0 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 138.

- 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

- 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

Instructions for installing JTLS-GO 6.0.8.0 are provided in the *JTLS-GO Installation Manual*. Installing a previous version of JTLS-GO prior to installing JTLS-GO 6.0.8.0 is not necessary. No other upgrade beyond installation of the compressed TAR files (or CD) is required. The software provided with this delivery is a complete release that includes all files and code required to execute JTLS-GO.

1.2.5 Released Databases

This release includes the following sample unclassified databases:

- The scenario that serves as a repository of engineering level data is called "repository60". 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. With JTLS-GO 6.0.8.0, it is possible to access and copy records from the repository60 database into your own developed scenarios.

- The scenario "wespac60", which is suitable for training and demonstrations.

1.3 INTERFACE COMPATIBILITY

1.3.1 Support Software

JTLS-GO 6.0.8.0 requires the following versions of support software, including operating systems, compilers, scripting utilities, database tools, transfer protocols, and display managers.

- Operating system for the model: Red Hat Linux Enterprise Edition Version 7.9 (ES), 64-bit architecture.

  JTLS-GO 6.0 has been tested with the following versions of Linux 7:

    RedHat Linux 7.9 - this operating system license must be purchased.

    Oracle Linux 7.9 - This operating system is free to download, use, and distribute, and is provided in a variety of installation and deployment methods. It has been approved by DISA for use by U.S. Government Agencies.

    CentOS 7.9 - a free version of Linux 7 that has not been approved by DISA for use by U.S. Government agencies.

- There are no restrictions on the operating system for client workstations, except that the operating system must have a Java-enabled web browser. JTLS-GO 6.0.8.0 has been tested on the following operating systems:

   Red Hat Linux Enterprise Edition Version 7.9.

   CentOS Linux Version 7.9.

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

Although JTLS-GO 6.0 is only approved for use on a RedHat Linux 7.9 system, several users have already upgraded their JTLS-GO Linux servers to RedHat 8. There have been no reported problems running the JTLS-GO 6.0 series on RedHat 8 Linux servers, as long as the servers are not Federal Information Processing Standards (FIPS) enabled. Each user organization needs to decide whether to move to RedHat 8 with a FIPS-disabled environment. To summarize:

- JTLS-GO 6.0 series on RedHat 7.9 can be run in a FIPS-enabled or disabled mode.

- JTLS-GO 6.0 series on RedHat 8.0 must be run in a FIPS-disabled mode.

- JTLS-GO 6.1 series must be run on RedHat 8 and can be run in a FIPS enabled or disabled mode.

- The JTLS-GO 6.0 series no longer relies on the Open Java Development Kit (OpenJDK™) to be installed at the system level using the Red Hat Package Manager (RPM). Instead, the JTLS-GO series is delivered with the AdoptOpenJDK package, which is equivalent to the current version of OpenJDK. Using AdoptOpenJDK provides two benefits:

   a. Only the JTLS-GO account on the system servers access this version of Java. An installation site can use the JTLS-GO servers for programs other than JTLS-GO without impacting the version of Java used by other programs.

   b. Security releases of AdoptOpenJDK software are produced on the same schedule as the Oracle OpenJDK security release procedure. An organization can expect to receive a bug release version of JTLS-GO within two-weeks of a new Java 1,8 security release. As long as a user organization installs all of the JTLS-GO bug releases, JTLS-GO can guarantee that the latest Java security release is being used on the servers. JTLS-GO no longer depends on system administration for implementing proper Java security update procedures.

   JTLS-GO 6.0.8.0 is delivered using the AdoptOpenJDK 1.8 Update 332.

- JTLS-GO uses IcedTea to provide the Java Web Start capability to start the JTLS-GO web-enabled applications. JTLS-GO supports IcedTea version 1.8.4. The user has two possible avenues to access the supported version of IcedTea when starting a web-enabled application on the JTLS-GO Linux servers. The IcedTea 1.8.4 RPM can be installed on the JTLS-GO Linux servers, or the user can set their Browser to use the proper version of IcedTea, located in the bin_support/Linux64 directory for "jnlp" applications,

  Client machines should be setup to also use the supported version of IcedTea,

- JTLS-GO database tools require a certified PostgreSQL 11.8 database server and the full PostgreSQL installation. A containerized solution, that fulfills this specification, is provided as part of the JTLS-GO download. It is not necessary to use the delivered containerized solution, but it is the easiest method to meet the requirements of JTLS-GO 6.0.8.0. There are several alternative methods available for obtaining the PostgreSQL 11.8 software. Refer to Chapter 6 of the *JTLS-GO Installation Manual* for additional installation details.

- Windows software, X11R5 server, Motif 1.2 Library, Motif Window Manager: These items are included as part of the supported versions of Red Hat Linux ES.

- TCP/IP is required for inter-process communication between the JODA data server and all user interface programs. The version of TCP/IP included with the supported versions of Red Hat Linux ES is sufficient.

- The Perl script language is used by the JTLS-GO system and game setup scripts. The version of Perl included with the supported versions of Red Hat Linux ES is sufficient. The Perl program is typically located in the /usr/bin directory. If Perl is installed in a another location, a link should be created from the /usr/bin directory to this program.

- SIMSCRIPT III Version 3-5.2.0 (SIMSCRIPT to C) translator/compiler: SIMSCRIPT is required for recompiling JTLS-GO code. It is not necessary to have a SIMSCRIPT compiler to execute JTLS-GO, because all JTLS-GO software executables are statically linked with the SIMSCRIPT libraries. The compiler is needed only if you are a U.S. Government organization that can obtain source code and plan to re-compile JTLS-GO SIMSCRIPT code. To obtain a SIMSCRIPT compiler, contact CACI Inc.

- ANSI C Compiler: It is not necessary to use a C compiler to execute JTLS-GO. This compiler is used only by U.S. Government organizations that can obtain source code and intend to re-compile any of the JTLS-GO component programs. The C Compiler version delivered with the supported versions of Red Hat Linux ES is sufficient.

- C++ Compiler: It is not necessary to use a C++ compiler to execute JTLS-GO. This compiler is used only by U.S. Government organizations that can obtain source code and intend to re-compile any of the JTLS-GO HLA component programs. The C++ Compiler version delivered with the supported versions of Red Hat Linux ES is sufficient.

- The JTLS-GO DDS (Database Development System) application uses these open source libraries:

     JFreeChart, licensed under LGPL (GNU LESSER GENERAL PUBLIC LICENSE) by Object Refinery Limited, http://www.object-refinery.com

     JCommon, licensed under LGPL2.1 (GNU LESSER GENERAL PUBLIC LICENSE version 2.1 or later) by Object Refinery Limited, http://www.object-refinery.com

     Commons-math3-3.0.jar, licensed under Apache Software Foundation (Apache License, Version 2.0) http://www.apache.org/licenses/LICENSE-2.0HLA 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 the COP, C2PC, or other C4I systems may be able to install and use Google Earth and configure the KOI to provide a capability that resembles C4I for observing perception Force Side data.

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

- JTLS-GO 6.0 implements SSH Tunneling between Apache and the services and among the services. Rigorous testing should be done prior to use in any exercise, and particular attention should be paid to network performance under load.

1.3.2 JTLS-GO Cybersecurity Compliance

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

One of the DoD requirements is that the software must implement a methodology that ensures that the end user keep the software up-to-date and all security patches are properly installed. In previous versions of JTLS-GO, Java 8, as delivered by Oracle, fulfilled this mandate by implementing an expiration date for its software. After a certain date, all Java related programs stopped working, whether connected to an open network or not. All JTLS-GO releases were closely linked to the Java expiration date.

OpenJDK™ has not implemented an expiration date. In order to fulfill this DoD Cybersecurity requirement, JTLS-GO has moved to AdoptOpenJDK, a full OpenJDK Java environment with

licensing alternatives allowing an application to deliver the software. The following procedure has been established and approved by the JS/J7 Cybersecurity branch:

- Within days of an Oracle Java security release, AdoptOpenJDK produces an equivalent version using infrastructure, build and test scripts to produce pre-built binaries of the OpenJDK class libraries. All AdoptOpenJDK binaries and scripts are open source licensed and available for free.

- Within two-weeks of the AdoptOpenJDK release, JTLS-GO provides a Maintenance release version (JTLS-GO 6.0.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 Maintenance released versions when distributed.

Contact the U.S. Government Program Manager, Mr. James E. Blank (james.e.blank.civ@mail.mil) to obtain the completed Cybersecurity paperwork.

1.3.3 JTLS-GO High Level Architecture Compliance

The JTLS-GO 6.0.8.0 release is fully High Level Architecture (HLA) compliant, and includes all the programs required to run JTLS-GO in an HLA mode. JTLS-GO currently belongs to one federation known as GlobalSim. GlobalSim is a comprehensive constructive simulation solution for joint training and wargaming that helps commanders and all levels of staff prepare for a range of operational scenarios.

The solution combines JTLS-GO with CAE's GESI constructive tactical entity-level simulation system. CAE's GESI constructive simulation system is designed to run complex and comprehensive exercises from the company level up to division level. The GESI system is used to represent a virtual battlefield, including weapons, vehicles, aircrafts, ground forces and more.

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

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

The HLA RTI (Run Time Infrastructure) executive program (rtiexec) recommended for use with this release is Pitch pRTI Evolved 4.4.2.0. However, this program is not included in the JTLS-GO 6.0.8.0 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 JTLS-GO 6.0.0.0 and the previous JTLS-GO 5.1 series database structure. There are no data format changes as a result of this Maintenance release.

To upgrade your JTLS 5.1 scenario to JTLS-GO 6.0 compatibility, see instructions listed in the *JTLS-GO DDS User Guide*, Chapter 3.1.

> STR JTLS-2022-15608 Grid Lower-Left Location Not Exact On Restart was released with JTLS-GO 6.0.6.0 in March 2022.
>
> This STR required a change to the ASCII checkpoint format. Any saved checkpoints from versions of the JTLS-GO 6.0 series earlier than JTLS-GO 6.0.6.0 will not work with JTLS-GO 6.0.8.0.
>
> STR JTLS-2022-15660 Convoy Didn't Load All Specified Supplies required an ASCII checkpoint format change. Any saved checkpoints made with JTLS-GO 6.0.7.0 will not work with JTLS-GO 6.0.8.0.

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 6.0.0.0 formatted data from your PostgreSQL database server into the JTLS-GO 6.0.0.0 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 6.0.8.0, but end-user organizations are free to use the Default Symbol Set in their scenarios and alter the scenario symbol set to meet specific organizational needs. Some new symbols have been created to meet end-user requirements. No previously existing symbols were deleted nor were any of the preexisting symbol names changed.

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

1.4.3 Standard Repository Changes

R&A has continued to improve and expand the unclassified data repository, "repository60." 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.

> The JTLS-GO 6.0.4.0 release, included a procedure needed to correct STR JTLS-2021-15237 which required changes to some post-update database triggers. If the following procedure was not accomplish with the release of JTLS-GO 6.0.4.0, it should be performed after the installation of this Maintenance release. Please:
>
> 1. Unload your scenario using the JTLS-GO Menu, Options 1 -> 1 -> 5.
>
> 2. Load your scenario using the JTLS-GO Menu, Options 1 -> 1 -> 4.

1.5 INSTALLATION

The *JTLS-GO Installation Manual,* a Portable Document Format (pdf) file available for direct download, is part of this JTLS-GO delivery, It provides detailed instructions for installing the new version of JTLS-GO and the installation of PostgreSQL 11.8 required to operate JTLS-GO 6.0 series of software.

2.0 ENGINEERING CHANGE PROPOSALS

No Engineering Change Proposals (ECPs) were added to this Maintenance Release of JTLS-GO.

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.

Chapter 4.0 describes STRs that remain outstanding from previous versions. These issues are being addressed and solutions will be included in future Maintenance releases in the JTLS-GO 6.0 series.

3.1 JTLS-2022-15646 Sensor/Jammer Coverage Not Magic Moved

When a unit was magic moved to a new location, the model did not relocate the effective area of its sensors and jammers.

If the unit was magic moved to a new location, the associated grid coverage for sensors and jammers was not moved. This meant the unit did not make detections or influence jamming of nearby objects at the new location. Code was added to correct this issue.

3.2 JTLS-2022-15652 Upgrade Third-Party Libraries

Vulnerabilities were identified in three third-party libraries. The newest version of the following libraries need to be used.

1. **jai-imageio-core-1.3.1.jar - The buffer copy does not check size of input.**

2. **libxml2.dll - Use after free trial period allows remote attackers to execute arbitrary code by controlling the arguments to a replaceChild call.**

3. **xtream-1.4.18.jar - A remote attacker may be able to allocate 100% CPU time on the target system, resulting in a denial of service, by manipulating the processed input stream.**

The libraries have been upgraded; jai-imageio-core to version 1.4.0, libxml2.dll library to version 2.09, and xtream to version 1.4.19.

3.3 JTLS-2022-15656 ELS Updates Missing Side Attribute

On the EODA, some update packets did not include the attribute for the side of the objects.

The side attribute was not included on the packets because it was assumed to be unchanged. This caused some issues for a direct connection with an external model. Changes were made to always publish the side information for all updates to object data on the EODA.

3.4 JTLS-2022-15659 Insufficient Capacity In Truck Convoy

A Directed Resupply order was submitted to generate a truck convoy. The convoy created by the model did not have enough trucks of sufficient capacity to carry the specified supplies, even though there were plenty of trucks available at the support unit. As a result, the convoy delivered only a partial load of supplies.

The model correctly determined the required trucks to transport all the specified supplies when assessing the feasibility of the Player order, but the process that actually assigned the trucks to the convoy lacked that information. Specifically, the process knew only the total number of cargo, tanker, and HET trucks to assign, but not their specific transportation classes (i.e. 2.5 ton vs 5.0 ton truck, etc).

Because the information was incomplete, the logic simply grabbed the first cargo trucks available up to the number required, without regard to their capacities. The result was the wrong truck assets were assigned to the convoy and could not carry all the specified supplies. This problem was corrected by adding logic to access the number of trucks required by transportation class as originally determined by the model.

While testing these changes, a canceled convoy returned to its origination (support unit) and disbanded. The LOGSITREP correctly showed the trucks were no longer in the Unavailable column, but the Combat System IMT showed the trucks still in the Unavailable column. The model had correctly moved the trucks from the Unavailable to Available. A missing JDS update was added to correct the display problem.

3.5 JTLS-2022-15660 Convoy Not Loaded With All Specified Supplies

A Directed Resupply order was submitted for a Rail convoy. After the Load task was complete, some of the supply categories specified in the order were not loaded on the convoy even though enough rail cars with sufficient capacity were part of the convoy (confirmed in the Supply Run Status Report). As a result, the convoy delivered only what it carried at the destination.

The model correctly selected the rail cars necessary to transport all the supply categories specified in the order. However, the process that actually loaded the supplies failed to differentiate between "wet" and "dry" supply categories, which caused some categories to be omitted when a transportation class could carry both. The logic treated both types identically and wrongly mixed quantity measures (i.e. gallons vs tons) when assigning supplies based on transportation class capacities. The logic was modified to load wet and dry supplies separately to correct the problem.

While testing this solution, it was discovered that sometimes the Offload task failed to unload all the supplies carried by the convoy at the delivery location. The problem arose when a transportation class happened to have a capacity to carry both wet and dry supplies (such as a 64-ton flat car) and a particular sequence of transportation classes and supplies were processed during the Offload task. The Elementary Loading Facility (ELF) was mis-allocated to unload wet

supplies instead of unloading dry supplies from the dual-capable transportation class. This logic error caused all the wet supplies to be unloaded and none of the dry supplies. The error was corrected by forcing all the dry supplies to be unloaded before the wet supplies so that the ELF did not take wet supplies from a transportation class that had both a wet and dry capacity.

Also while testing, during the convoy offload process, the amount left on the convoy for a particular supply category could be extremely small (after repeated subtractions). The supply category remained in the requested offload set when ordinarily it would have been deleted when approaching zero. As a result, the logic failed to terminate the Offload Task, leaving the convoy in limbo until the Manage Convoy Tasks order was sent to cancel the Offload Task. This rare occurrence was corrected by changing the conditional test to use almost zero (0.0000001) instead of exactly zero.

Similarly, a convoy might not have carried enough to fulfill the expected amount to be offloaded which triggered a player message. The logic was modified to use the almost zero value instead of exactly zero so that the message is not generated when the convoy is short a very small amount.

Note that these load/unload problems were intermittent and independent of the convoy type (truck, rail, or barge). None of them may have been experienced in your execution scenarios. The problems depended on the supply types requested, the transportation classes used, and the sequence of processing.

Important note: As a result of these changes, the checkpoint read/write files had to be modified to include a new inventory item attribute. Therefore, all existing checkpoint files for the JTLS-GO 6.0 series are no longer viable.

Code was added to display the name of the transported unit in the Supply Run IMT.

3.6 JTLS-2022-15662 POT Bridge Target Number Of Elements

The SVP should not limit POT Bridge Targets to one.

The SVP currently limits the number of elements for a POT target of type BRIDGE to one. There should be no limit. A unit may have more than one Pontoon Road Bridge that it carries around, emplaces, and picks up as needed.

The check in the SVP code limiting POT target Bridges to one element has been removed.

During testing, a problem with a unit being unable to retrieve a formerly Unit-owned emplaced POT bridge was identified. During design discussions it was determined that any unit from any side that is Bridge Ops capable should be able to retrieve a mobile bridge.

The bridge retrieval logic to match this real-world capability was also implemented as part of this STR.

3.7 JTLS-2022-15664 ELS Crash Reading Terrain Data During Restart

The ELS crashed in the process of reading the terrain data during an operational assessment test. The ELS was expecting a different format to the data than was saved by the CEP.

The terrain data ASCII checkpoint format was recently changed. This change was documented as STR JTLS-2022-15637 delivered with JTLS-GO 6.0.6.0. The ELS could not be restarted from a checkpoint which was saved in the improved data format. The ELS required code changes similar to those in the CEP to read the same data files.

3.8 JTLS-2022-15670 AAR Air Missions With Unknown Home Squadron

One of the data items that is inserted into the AAR database when an Air Mission is created is its home squadron. For some Air Missions, this is being inserted into the database as an unknown reference. This appears to be linked to either a checkpoint restart or a restart of the AARC.

The AARC keeps a full list of all objects that it knows about and ensures those objects that pass its filter settings are inserted into the database. For each of these objects, the AARC tracks whether it is in the database and whether it has passed filter checking. The filter tracking is used to maintain a consistent and closed database with references to objects that will not be in the database being zeroed out. For example, if a target passes filters and will be inserted into the database, but its associated unit does not, the reference to the associated unit will be zeroed out.

In order to accomplish this, when the AARC received a new JODA download, it dumps all known objects and starts to rebuild this list by retrieving everything from the database. During this retrieval objects were being marked as in the database, but were not being marked as having passed filters leading subsequent references to them to be zeroed out. The problem was fixed by marking objects retrieved from the database as having passed filters, which they would have done at the time of insertion.

3.9 JTLS-2022-15672 GIS Tool - New Terrain Layers Not Listed

The user created terrain layers and then selected the Calculate Grid Parameters option. The Calculate Grid Parameters dialog appeared, and the new layers were listed and correctly color-coded red. The user selected all of the new layers and had the grid parameters calculated. The user next proceeded to create additional terrain layers and then selected the Calculate Grid Parameters option again. The new layers were not listed by the Calculate Grid Parameters dialog.

The GIS Tool was correctly updating its internal table of terrain layers as the new layers were created. However, the Calculate Grid Parameters dialog was not properly updating its list of terrain layers displayed to the user. The dialog was supposed to update its list by reading the layers stored in the Tool's internal table every time the user selected the Calculate Grid Parameters option, but the dialog was not doing so. The dialog now correctly updates its list so that the user can see the new layers.

3.10 JTLS-2022-15674 Crash Sending HRU ROE Order For "No Side"

The model crashed when a user entered an HRU Task Order and specified an ROE change for a Side called "No Side". This does not exist in JTLS-GO.

The problem was that the order panel provided an incorrect drop-down list. The list should have included all Sides other than the HRU's Side and the Unknown Side. The order panel was corrected to include the correct drop-down list referred to as ROE_Force_Side, rather than the Other_No_Side drop-down list.

3.11 JTLS-2022-15677 Infinite Loop Gathering Grid

A naval ship was located exactly at the lower left corner of a grid. When computing whether there was any terrain masking from a foreign sensor attempting to detect the ship, the model went into an infinite loop attempting to gather the grids between the sensor and the ship.

The logic did not properly handle this situation. Code was added to check for the situation and to stop as soon as the grid location of the ship was entered. The same logic improvement was placed in the Scenario Initialization Program (SIP). The gather routine is needed for several of the SVP checks.

3.12 JTLS-2022-15678 Missile Hits Dead Ship

Some databases are built so there is no time required for a ship to sink. As soon as the model computes that a ship should sink, it sinks and is removed from the game. If, for this type of database, the user submits two or more fire missile orders, it is very possible that a follow-on missile will arrive after a ship is removed from the game. The model does not crash, but some confusing messages are generated.

The damage assessment logic checks to determine if the missile is about to hit a ship that is no longer in the game. Depending on the type of missile and the objects in the vicinity, the missile may end up hitting a different ship. This situation is as intended and designed.

3.13 JTLS-2022-15681 Runway Warnings For Future Arriving Airbases

A user was getting numerous warnings indicating that an airbase was not assigned a runway. Some of these errors were accurate, but some were not, because the unit was not arriving in the game for several days, if at all. A warning should not be generated if an airbase that arrives after game start has no database-assigned runway.

The user request was implemented.

3.14 JTLS-2022-15683 Incorrect Warning HUP Crew Needed

The SIP was incorrectly reporting a warning, indicating that there were not enough personnel in the HUP to support the Combat System crew requirements.

There was a logic error in the check. If a HUP has three Armored Personnel Carriers (APCs), it was computing the crew requirements for four APCs. For every crewed combat system that the HUP owned, the check was adding in an extra combat system. The logic error was corrected.

3.15 JTLS-2022-15685 PKs Not Always Desired When PH Exists

A user pointed out that they did not want Probability of Kill (PK) data for some weapons, but did want Probability of Hit (PH) data. Reporting this undesired situation causes them to miss real problems.

Leaflets should have PH but no PK data. Similarly for Chemical Weapons, weapons should be able to hit something, but once the chemical agent is released, the chemical immediate and longer term damage data takes over.

Both situations were reasonable. The code was changed to not report the lack of PK data for leaflet weapons and chemical weapons when PH exists.

3.16 JTLS-2022-15686 Incorrect Air Drop Task End Time

The model was incorrectly computing the time required for an Air Drop of supplies or a unit task to complete. Airdrops are instantaneous, and attempting to compute an end task time was unnecessary and caused a logic error.

Airdrop task completion is now reported as occurring immediately. The logic error is no longer being produced.

3.17 JTLS-2022-15690 Warning 1223 Improvement

Warning 1223 needs an option to display the entire TUP/SUP supply table.

An option to display the entire TUP/SUP supply table was added. This allows the user to review all supplies before making corrections to the supply category in question.

3.18 JTLS-2022-15691 Warning 1254 Improvement

Warning 1254 needs an option to display the entire SUP supply table.

An option to display the entire SUP supply table was added. This allows the user to review all supplies before making corrections to the supply category in question.

4.0 REMAINING ERRORS

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

As JTLS-GO 6.0.6.0 represents a major release of new functionality, all outstanding errors have been reviewed. If the error could not be reproduced, it was considered obsolete and no longer relevant to JTLS-GO. These errors have been removed from consideration for correction at this time. This chapter contains the remaining errors that could be reproduced. As development team assets become available, these remaining known issues will be corrected.

In future maintenance releases, newly uncovered outstanding errors related to JTLS-GO will be listed in this chapter, along with information regarding the extent of the error, as well as suggestions to avoid or minimize the effects of the problem.

4.1 DDSC – TMU Line Mode Changes Multiple Grids

When using the line mode in the TMU, more grids than the ones the line passes through are changed. This can also cause a warning about trying to change multiple layers to appear.

4.2 DDSC – Multiple Types In DDS History Table

If records for more than one table type are selected in the DDS History table, "History Details" will display details for only one type.

4.3 WHIP - Pipeline Not Shown On IMT

A pipeline being operated by a non-detected unit is not shown in the pipeline IMT.

4.4 DDSC/WHIP/JOBE - CADRG Map Zoom

When using the CADRG map projection, if the width of the map is less than the height, the zoom tool does not work correctly.

APPENDIX A. ABBREVIATIONS AND ACRONYMS

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

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

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

DPICM	Dual Purpose Improved Conventional Munitions
DS	Direct Support
DSA	Directed Search Area
DTG	Date Time Group
EC	Electronic Combat
ECM	Electronic Counter Measure
ECP	Engineering Change Proposal
EEI	Essential Elements of Information
ELINT	Electronic Intelligence
ELS	Entity Level Server
EODA	Entity Level JTLS Object Data Authority
ETA	Estimated Time of Arrival
FARP	Forward Arming and Refueling Point
FLP	Fire Lethality Prototype
FLOT	Forward Location of Troops
FOL	Forward Operating Location
FWL	Frederick W. Lanchester (originated a differential equation model of attrition)
GAL	Gallon
GCCS	Global Command and Control System
GRTE	Ground Route
GS	General Support
GSR	General Support Reinforcing
GUI	Graphical User Interface
HARM	High-speed Anti-radiation Missile
HE	High Explosive
HELO	Helicopter
HMMWV	High Mobility Multipurpose Wheeled Vehicle
HQ	Headquarters
HRU	High Resolution Unit
HTML	Hypertext Markup Language
HTT	High resolution unit Target Type
HUP	High resolution Unit Prototype
ICM	Improved Conventional Munitions
ICP	Interface Configuration Program

ICPLogin	Interface Login Program
ID	Identifier
IFF	Identification Friend or Foe
IIP	Intelligence Information Prototype
IMT	Information Management Tool
INFO	Information
INTEL	Intelligence
JCATS	Joint Conflict And Tactical Simulation
JDA	Japan Defense Agency
JDPI	Joint Desired Point of Impact (formerly DMPI: Desired Mean Point of Impact)
JDS	JTLS Data System
JDSP	JTLS Data System Protocol
JEDI	JODA Entity Data Identifier
JMCIS	Joint Maritime Combat Information System
JMEM	Joint Munitions Effectiveness Manuals
JODA	JTLS Object Distribution Authority
JOI	JTLS Operational Interface
JPL	Jet Propulsion Laboratory
JRSG	Joint Rapid Scenario Generation (formerly JIDPS: Joint Integrated Database Preparation System)
JSDF	Japanese Self-Defense Force
JTLS	Joint Theater Level Simulation
JTLS-GO	Joint Theater Level Simulation - Global Operations
JTOI	JTLS Transaction Operational Interface
JXSR	JTLS XML Serial Repository
KIA	Killed In Action
KM	Kilometer
KNOTS	Nautical miles per hour
LA	Lethal Area
LAN	Local Area Network
LAT	Latitude
LB	Login Build (JTLS order type)
LDAP	Lightweight Directory Access Protocol
LDT	Lanchester coefficient Development Tool
LOG	Logistics

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

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

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

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

APPENDIX B. VERSION 6.1.0.0 DATABASE CHANGES

No database format changes were made to this security release of the JTLS-GO 6.0 database. All previous JTLS-GO 6.0 databases will work with this version of the software.

APPENDIX C. VERSION 6.0.6.0 REPOSITORY CHANGES

No changes were made to the JTLS-GO 6.0.7.0 data repository.