JTLS-GO

Version Description Document

February 2021

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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 5.1.11.0)

[Blank Page]

ABSTRACT

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

This *JTLS-GO Version Description Document (VDD)* describes the new features of the Version 5.1.11.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 5.1.11.0 is a Maintenance release of the JTLS-GO 5.1 series that includes an updated wespac51 demonstration database as well as repository data held in the repository51 database. There are no Engineering Change Proposals (ECPs) included with this release. 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
jtlsdev@rolands.com

[Blank Page]

TABLE OF CONTENTS

1.0 INTRODUCTION

1.1 SCOPE

This *JTLS-GO Version Description Document* (VDD) describes Version 5.1.11.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO®) software suite. JTLS-GO 5.1.11.0 is the last Maintenance release planned for the JTLS-GO 5.1 series.

JTLS-GO 5.1.11.0 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 “wespac51”. No database format modifications have been made for this release, but a static data error for allowable OTH-Gold Ship Types was corrected in Version 5.1.1.0. This does require a user to execute a correction procedure for all of their Version 5.1.0.0 scenarios loaded in Oracle prior to the release of Version 5.1.1.0. Information on this procedure can be found on Page 1-11.

No Engineering Change Proposals (ECPs) were implemented for this release, but for consistency an empty Chapter 2.0 stating this fact is included. Explanations of all Software Trouble Reports (STRs) corrected in this release are provided in Chapter 3.0. Outstanding STRs are provided in Chapter 4.0. Changes made to the JTLS-GO 5.1 engineering data repository are usually provided in APPENDIX C, but updates for the JTLS-GO 5.1 engineering data repository are no longer supported.

JTLS-GO 5.1.11.0 executes on the Red Hat Enterprise Linux Version 7.6 64-bit operating system. The Web-Hosted Interface Program (WHIP®) user workstation interface can be executed on any operating system from any Java-compatible Web browser.

1.2 INVENTORY OF MATERIALS

This section lists documents and software that are relevant to JTLS-GO. All JTLS-GO documents included in this delivery are provided in Portable Document Format (PDF) 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 5.1.3.0)
- *JTLS-GO Configuration Management Plan (*JTLS-GO Document 03, Version 5.1.2.0)
- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 5.1.9.0)

- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 5.1.7.0)
- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 5.1.2.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 5.1.3.0)
- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 5.1.6.0)
- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 5.1.6.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 5.1.3.0)
- *JTLS-GO Repository Description* (JTLS-GO Document 14, Version 5.1.2.0)
- *JTLS-GO Software Maintenance Manual* (JTLS-GO Document 15, Version 5.1.2.0)
- *JTLS-GO Technical Coordinator Guide* (JTLS-GO Document 16, Version 5.1.5.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 5.1.2.0)
- *JTLS-GO Federation User Guide* (JTLS-GO Document 20, Version 5.1.9.0)
- *JTLS-GO C4I Interface Manual* (JTLS-GO Document 21, Version 5.1.9.0)
- *JTLS-GO Air Services User Guide (*JTLS-GO Document 24, Version 5.1.2.0)

1.2.3 Updated Documents

- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 5.1.11.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 5.1.11.0)

1.2.4 New Documents

No new documents are delivered with JTLS-GO 5.1.11.0.

1.2.5 Delivered Software Components

JTLS-GO 5.1.11.0 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)
- Modernized Integrated Database (MIDBTool)
- JTLS Object Distribution Authority (JODA) Build 137
- 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)

When operating the TRIPP capability in current JTLS-GO releases, users are not prevented from logging into an actively running TRIPP making a connection to the same Replay JXSR. A TRIPP, as documented, requires its own Replay JXSR to control and perform the replay of the recorded simulation events.

This situation as a consequence can have more than one user concurrently control the playback of the game, as the shared Replay JXSR will honor each of the user's playback requests and will then change what each connected user sees on their TRIPP instance. This issue has been addressed and fixed within the JTLS-GO 6.0 release by only permitting one login per TRIPP user instance.

Organizations should develop their own procedures to ensure that only one user logins to a given TRIPP at a time.

- 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

Instructions for installing JTLS-GO 5.1.11.0 are provided in the *JTLS-GO Installation Manual*. Installing a previous version of JTLS-GO prior to installing JTLS-GO 5.1.11.0 is not necessary. No other upgrade beyond installation of the compressed TAR files (or CD) is required. The software provided with this delivery is a complete release that includes all files and code required to execute JTLS-GO.

1.2.6 Released Databases

This release includes the following sample unclassified databases:

- The scenario "wespac51", which is based on the Western Pacific theater of operations and is suitable for training and demonstrations.

- The scenario “repository51” serves as a repository of engineering level data. Although not useful as a scenario, it does follow all of the database requirements for a scenario, and should be loaded into your Oracle scenario table-space. With JTLS-GO 5.1.11.0, it is possible to access and copy records from the repository51 database into your own developed scenarios.

1.3 INTERFACE COMPATIBILITY

1.3.1 Support Software

JTLS-GO 5.1.11.0 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 7.6 (ES), 64-bit architecture.

  Security Enabled (SE) Linux must not be enabled on systems used to execute JTLS-GO or its components. Tests reveal that SE Linux-enabled systems cause frequent and random JXSR crashes and errors. Use of SE Linux to support JTLS-GO is currently not authorized.

- JTLS-GO 5.1 has been tested with the following versions of Linux 7:

  Red Hat Linux 7.6 - This operating system license must be purchased, but it has been approved by the Defense Information Systems Agency (DISA) for use by U.S. Government Agencies.

  Oracle Linux 7.6 - 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 Linux 7.6 - 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 used for client workstations, except that the operating system must have a Java-enabled web browser. JTLS-GO 5.1.11.0 has been tested on the following operating systems:

  Red Hat Linux Enterprise Edition Version 7.6.

  CentOS Linux Version 7.6.

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

- The JTLS-GO 5.1 series no longer uses Oracle Java, and has moved to the latest version of OpenJDK 8, which is OpenJDK 8 Version 282. For the JTLS-GO 5.1 series, the Java Runtime Environment (JRE) is not provided within the JTLS-GO delivered software TAR files. Each user organization must obtain the latest version of the OpenJDK Red Hat Package Manager (RPM) and install the RPM on the servers used by JTLS-GO.

- JTLS-GO uses IcedTea to provide the OpenJDK web start capability that implements the web-enabled JTLS-GO functionality. The current version of JTLS-GO supports IcedTea version 1.8.3.

  IcedTea version 1.7.1 does not support use of unsecured HTTP to access web-enabled JTLS-GO components for the DDS Client application. Users must configure their Glassfish instance to use HTTPS, by enabling the "Use SSL" column within the DCP when running with IcedTea version 1.7.1. The use of SSL is the current default setting in the DCP.

  Red Hat Linux version 7.7 continues to distribute with IcedTea version 1.7.1. There are available RPM packages for a later version of IcedTea, so users who wish to use unsecure HTTP with JTLS-GO must explicitly install IcedTea 1.8.3 on the JTLS-GO servers and client workstations.

- JTLS-GO database tools require use of a certified Oracle database server and the full Oracle Client installation for runtime requirements. Additional installation details can be found in Section 1.5.2 of this chapter.

- 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 service 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 II.5 (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 by a U.S. Government Agency that can obtain source code and plans 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 needed only by a U.S. Government Agency that can obtain source code and plans 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 needed only by U.S. Government Agency that can obtain source code and plans 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 LICEN.SE 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 5.1 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 Cyber Security Information Assurance Compliance

Because of recent incidents of intrusions into software systems, the United States Department of Defense (DoD) has implemented a strong and strictly enforced Cyber Security Information Assurance (CSIA) program. JTLS-GO, as software that executes on DoD systems, must comply to the mandates of the CSIA program, as well as requirements of all of the third party software used by JTLS-GO, such as Oracle and Java.

One of the DoD CSIA requirements is that the software must implement a methodology that ensures that the end user keeps the software up-to-date and properly installs all security patches. The primary purpose of this release is to provide a version of JTLS-GO compiled with the latest security release of OpenJDK, OpenJDK 8 Version 282.

To meet these CSIA requirements, each user organization should ensure that this version of OpenJDK is loaded on the JTLS-GO servers and any client machines used to connect to JTLS-GO.

JTLS-GO has completed the CSIA program mandates and the JTLS-GO 5.1 series of releases has been granted an Authority To Operate (ATO) on DoD systems. Contact the U.S. Government Program Manager, Mr. Don Weter (donald.e.weter.civ@mail.mil), for additional information.

1.3.3 JTLS-GO High Level Architecture Compliance

The JTLS-GO 5.1.11.0 release is fully High Level Architecture (HLA) compliant, and includes all the programs required to run JTLS-GO in an HLA mode. JTLS-GO uses the Federation Object Model (FOM), located in the $JGAME/data/hla directory. Federation testing of JTLS-GO with CAE's GESI wargaming system has been accomplished. Future plans include expanding the capabilities of the JTLS-GO-GESI federation, called "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 as a federation. CAE's GESI constructive simulation system is designed to run complex and comprehensive exercises from the company level up to division level. The CAE GESI system is used to represent a virtual battlefield, including weapons, vehicles, aircraft, and ground forces.

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.

All JTLS-GO software needed to run GlobalSim is included in this delivery.

The HLA RTI (Run Time Infrastructure) executive program recommended for use with this release is Pitch pRTI Evolved 4.4.2.0. However, this program is not included in the JTLS-GO 5.1.11.0 delivery. Users may obtain a full installation package of the RTI executive program from Pitch Corporation (www.pitchtechnologies.com). 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 5.1 series and the previous JTLS-GO 5.0 series database structure. Refer to Appendix B in the *JTLS-GO 5.1.0.0 Version Description Document*, included with this release, for data structure changes made for the JTLS-

GO 5.1 series. To upgrade your JTLS-GO 5.0 scenario to JTLS-GO 5.1 compatibility, see instructions listed in Chapter 3.1 of the *JTLS-GO DDS User Guide*.

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 5.1.0.0 formatted data from your Oracle database server into the JTLS-GO 5.1.0.0 scenario American Standard Code for Information Interchange (ASCII) text files, you must execute the JSYMS program using the procedure outlined in Appendix B.11 of the *JTLS-GO DDS User Guide*. This procedure will reorganize the structure of the .gs and .scf symbols-related files.

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 5.1.11.0, 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.

1.4.3 Standard Repository Changes

The JTLS-GO 5.1 series of JTLS-GO is the first series in which R&A is delivering an unclassified data repository called “repository51”. In future Major releases of JTLS-GO, APPENDIX C will provide a summary of the data structure changes made to the data repository. No data structure changes have been made is this Maintenance release; therefore, APPENDIX C is empty.

1.5 INSTALLATION

1.5.1 Installation Instructions

The *JTLS-GO Installation Manual,* a PDF file available for direct download, is part of this JTLS-GO delivery. It provides detailed instructions for installing a new version of JTLS-GO.

1.5.2 Oracle Compatibility and Installation

A full Oracle Client (not Instant Client) installation that matches your database server version is currently a requirement for running some JTLS-GO applications. The Oracle Instant Client is not sufficient for JTLS-GO applications because certain Oracle utilities, such as sqlldr, imp, exp, and tnsping, are missing. If you have applied a patchset to your database server, the same patchset should be applied to the Oracle Client installation. A 64-bit Oracle Client installation must be used.

The JTLS-GO scenario/database modification process also expects Oracle 11.2.0.1 or higher full Oracle Client installation. Some sites NFS mount their database server as Oracle Client; other sites prefer a full installation of the Oracle Client to a different directory that mounts to JTLS-GO (a simple NFS mount will suffice). Your system administrator can choose the appropriate installation.

Assigning the full Oracle Client installation location (or mount point) as the ORACLE_HOME in the JTLS-GO .cshrc file allows connecting to an Oracle database server (11.2.0.1 or higher - including 11gR2 XE) running on any Oracle-certified database server platform.

Oracle offers free Express Editions (XE) of the Oracle relational database management system. Compared to the 11gR2 XE version, the newer 18c XE has a larger footprint and a much more complex database architecture. For test environments and scenario building purposes, or for collecting AAR data for a short period of time, the installation and setup of the 11gR2 XE version is much simpler.

The DDS application utilizes the Oracle GlassFish J2EE server, which, like the JTLS-GO WHIP Apache server, is delivered with JTLS-GO and requires no separate installation.

Refer to Chapter 6 of the *JTLS-GO Installation Manual* for additional details pertaining to the Oracle installation.

1.5.3 Special Installation Instructions

This section describes special instructions that should be followed because of errors corrected in this version and previous bug releases for the JTLS-GO 5.1 series.

1.5.3.1 Generation Of Combat System Summary Files

STR JTLS-2019-14518 Move Combat System Summary Files, delivered as part of JTLS-GO 5.1.4.0, solved a problem overwriting the summary Combat System Character Separated Value (.csv) files during Batch Runs. This STR was solved by moving the location of the .csv files from the game/<scenario_name>/location directory to a sub-directory under each checkpoint.

For every active game scenario, one of the following procedure options must be executed to establish the directory structure needed by the STR solution. Users who have already executed this procedure for their scenarios after installing JTLS-GO 5.1.4.0, do not need to execute one of the selected options again:

- Option 1: Rerun the Setup Procedure for each active scenario.
- Option 2: Hand-create a cbtsys_summary sub-directory in the game/<scenario_name> directory. This can be accomplished using the following steps from a command terminal for each existing game that has already been set up and prepared for execution:

  a. Enter the command: game - this puts the terminal in the $JTLSHOME/game directory.

  b. Enter the command: cd <scenario_name> (for example, cd wespac51) - this puts the terminal in the game/<scenario_name> directory.

  c. Enter the command: mkdir cbtsys_summary. This creates the necessary new directory.

1.5.3.2 OTH-Gold Ship Types

JTLS-GO 5.1.0.0 was delivered with an old list of OTH-Gold ship types. This problem was corrected in JTLS-GO 5.1.1.0. To properly implement this solution, users must execute the following additional procedures for each of their JTLS-GO Version 5.1.0.0 scenarios loaded in Oracle.

Users who have already executed this procedure for their scenarios after installing JTLS-GO 5.1.1.0, do not need to execute the procedure again.

New OTH-Gold ship types were added to JTLS-GO, due to STR JTLS-2019-14238.

1. Execute the following command:

   cd $JTLSHOME/script/dds/version5.1/scripts/

2. Execute the following command:

   sqlplus yourScenario/OraclePassword @update_oth_gold_types.sql

3. Verify the related execution listing file under the $JDATA/scenario/ directory for errors.

4. Unload your scenario using the JTLS-GO Menu, Options 1 -> 1 -> 4

5. Verify the .srw ascii file for your scenario.

To fix the problem renaming Force Sides, due to STR JTLS-2019-14267:

6. Reload the database.

1.5.3.3 Aircraft Type Foreign Key Definitions

STR JTLS-2019-14541 Referenced Aircraft Class Can Be Deleted, delivered as part of JTLS-GO 5.1.5.0, solved a problem that allowed the deletion of aircraft types that were referenced by units. The STR was solved by removing a rule in the database that allowed the user to set the aircraft type-related foreign keys of the unit tables to NULL.

After loading JTLS-GO 5.1.5.0, the user must unload and then reload their JTLS-GO 5.1 scenarios to have the new foreign key definitions in their database schemas.

Users that have accomplished this procedure after installing JTLS-GO 5.1.5.0, do not need to re-execute this procedure after installing this version of JTLS-GO.

2.0 ENGINEERING CHANGE PROPOSALS

No model capabilities were added to JTLS-GO 5.1.11.0 as a result of implementing authorized Engineering Change Proposals (ECPs).

3.0 SOFTWARE TROUBLE REPORTS

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

3.1 JTLS-2020-15037 Tech Tool Crash During

The Tech Tool crashed during initialization. The problem was related to shared order groups.

When the Tech Tool initializes, it creates the tree directories that are displayed in the Order Group Copy panel. A particular node in the tree that the Tech Tool was expecting went missing and caused a null pointer exception. A check is now made to restore a missing node.

While investigating the original problem, another problem was uncovered, where the Tech Tool was sometimes retrieving the wrong home directory for the remote host's directory tree, as displayed in the Order Group Copy panel. The problem was resolved by adding a correct retrieval method to the code.

3.2 JTLS-2020-15053 Crash Splitting Formation

An argument mismatch existed when splitting a formation and calling the routine to find a good station location for each ship. The routine to find the station location was expecting the latitude and longitude of the formation, but received the desired range and bearing of the ship from the formation center. In the instance of the crash, this resulted in a latitude of 163 being sent to calculate a new location resulting in the crash when performing spherical math.

The code was modified to pass the correct arguments.

3.3 JTLS-2020-15056 HRU With Same Movement Lat/Lon Crash

An HRU was performing a very short move through a minefield. When the CEP was checking if the HRU had crossed any barriers, a calculation was done between the movement vector of the HRU and the barriers in the grid. The small distance moved caused an arithmetic underflow in the spherical math. The starting and ending locations of the move were the same when examined in the debugger, which does not show the full 64-bit real number.

The code was protected if the locations had been exactly the same, but not for almost exactly the same. Therefore, the check for sameness was switched to a function call that determines if the locations are within 1 meter, and treats them as the same if they are.

Additionally, when investigating the crash it was initially assumed the error was in a barrier with identical points, which did exist. This barrier had been created by a Controller order with a duplicate point. This should not be allowed, and to avoid it in the future, the utility file that is used to construct the points was modified to require unique locations.

3.4 JTLS-2020-15058 Negative IIR Reports Not Generated

Whenever a Non-Theater DSA Negative Collection Report was generated by the model, the resultant XML message file was improperly formatted/corrupted. The report would appear on the Message Browser with the title "Bad Message File" and was not viewable.

The XML corruption was due to data items not being properly identified as data items within the CEP code. The code was modified to add the data item designators around the data.

3.5 JTLS-2020-15083 Emitter IMT Screen Help Misleading

The IMT screen help was inaccurate and misleading.

The help was updated for the entire IMT screen, including the individual column help.

3.6 JTLS-2020-15085 SDC Shortens Supply Category Names

If the name of a supply category is longer than 15 characters, the SDC will cut the name length down to 15 characters prior to adding the supply category to the associated table in the database repository.

Back when the names for game objects were standardized to 25 character lengths, several changes were made to ensure the new generic name length was properly restricted for data entry, storage and display. This included length changes to the definitions controlling the DDS, IMT and messages and various other display utilities. One change that was not made was the size variable for generic names in the JDS Protocol. This variable is used for automatically configuring database table generation and limiting name lengths within the AAR programs such as the AARC, SDC and OEC.

This variable was changed from 15 to 25 characters in the JDS Protocol configuration file. From the Protocol configuration the new 25 character length is now available to the AAR programs for limiting name lengths. After generating the code for each of the AAR programs the names are now properly limited to 25 characters.

3.7 JTLS-2020-15088 Temporary Array Not Released From CEP Memory

A CEP routine was found to reserve storage in memory for a temporary data array, but the reserved memory was not freed when the routine was finished.

A missing RELEASE statement was added to the routine to correct the problem.

3.8 JTLS-2020-15092 OPM TTG Target Definition Not Alphabetical

Checking the OPM to determine if a specific target category/subcategory is part of a Target Type Group (TTG) is nearly impossible for the larger target subcategories, such as SUPs, because the

data are listed in priority order, not alphabetical order. This data should be arranged alphabetically.

The SIP did not actually impose an ordering on the arrangement of the data. When dealing with checkpoint zero, downloaded from a DDS database, the data was arranged alphabetically and thus appeared in the OPM alphabetically. However, the CEP stores the data in priority order, and this ordering was reflected in a checkpoint. Generating the OPMs from a checkpoint would pick up this ordering and list the items by priority.

To solve the problem the SIP was modified to impose its own ordering on the data, by target subcategory which is alphabetical for each target category.

3.9 JTLS-2020-15097 Loop When Mission Told To Rearm

The model went into an infinite loop when a Mobility air mission was told to rearm after it dropped off a unit, but without specifying the optional resupply unit. When the mission finished offloading the unit, it went to rearm but there were no legal units in the area.

At this point, the mission did not know what to do and generated a message telling the player about the problem. After reporting the issue, the bad task was not canceled, so the mission repeatedly executed the same task over and over, each time coming to the conclusion there was no one available from which to rearm and generating a problem report for the user.

The rearm task under this circumstance was canceled to resolve the problem.

3.10 JTLS-2020-15101 CCP Attributes Report Error

An error was found in the Command Control Prototype (CCP) message that was generated from the Controller Set CCP Parameter order. In the report section that display the Density Modifiers for Terrain, Posture, and Base Value, the message template incorrectly referenced Combat Systems instead of Combat Arms Types. The Density Modifiers applied to Combat Arms Types, not Combat Systems in the CEP. The error was confined only to the generated message and was not present elsewhere in model processing.

The message template was changed to reference Combat Arms Types for the Density Modifiers.

3.11 JTLS-2020-15110 Manage ULN Show Going Out As Broadcast

The manage ULN show option was generating a message and sending it out as a broadcast message. The generated message is designed as a directed message.

The code was changed to make the message a directed message.

3.12 JTLS-2020-15113 CEP Out Of Memory Computing Optimal Route

A tester entered a value of 56 for the Ground End Overlap Factor and 20 for the Ground Half-Width Factor. The result for a 300 KM move was that a convoy was searching an area of 33,900 KM by 12,000 KM for an optimal route. The model ran out of memory attempting this computation.

The order panel should have limited the various end-overlap factors and half-width factors to a maximum of 2.0. The maximum was changed and the help was improved so the user better understands the meaning of the database parameter. It was noted that the database builder could also enter a large value. A temporary SVP check was created to make sure the parameter is not too large to cause model issues.

3.13 JTLS-2020-15116 Pickup From Ground Unit Transport Instructions

A Directed Resupply convoy order was created, and the convoy was told to pick up supplies from a ground unit and drop off the supplies to a different ground unit. The transport instructions list could be created, but the order was rejected because the pickup unit was a ground unit.

Within JTLS-GO, there is a rule that the model will never automatically take supplies from a Ground Combat unit. This rule should not apply to an order in which a Player wants to specifically pick up supplies from a ground unit. The order constraint that was causing the problem was removed from the utility panel.

The model also needed to be changed to allow the convoy to pickup from a ground unit.

When incorporating the order changes into the system, it was noted that the script to perform the task was not finding the stylesheet to generate OEC code and scripts. This file was moved to a new directory, but the generation script was not updated for this new location. The generation script has now been updated to look in the proper place for the OEC stylesheet.

3.14 JTLS-2020-15117 Lat/Long Pickup/Dropoff Incorrect Task Order

If the user indicates that a convoy should pick up supplies from a unit and at the same time drop off supplies at the same unit, the model ensures that the drop off occurs before the pickup so the convoy has room for the newly planned supplies to be loaded.

If the user creates an order task list that picks up and drops off supplies and no unit is specified, the model incorrectly switched the pickup and dropoff tasks.

If two sequential tasks have no unit specified, then the location of the two tasks are now checked. If the location is identical, then the drop off continues to occur first. If the two locations are not the same, the task check is not made and the ordering submitted by the user is implemented.

3.15 JTLS-2020-15118 Inconsistent SSA Target Creation

The supplies contained in a Supply Storage Area target were not the same if the target arrived at game start, if it arrived after game start, or was created by a Player during game play.

The code was changed to show consistent behavior, no matter whether the target arrives at game start or after game start.

3.16 JTLS-2020-15123 Move Checkpoint To Shadow Game Scripts

The JTLS Menu contains an option to move a checkpoint from an exercise game to a shadow game when the two games are on the same file system, such as an NFS system. However, NATO and a number of NATO countries that use JTLS-GO set up their shadow games on an independent file system, so the delivered JTLS-GO procedure does not meet their needs.

Usually, when supporting an exercise for these organizations, the Technical Controller ends up writing scripts to perform the procedure, which end up being deleted at the end of the exercise, which is a poor utilization of engineering time.

The two scripts that were written for one exercise were saved and placed into the $JTLSHOME/ script directory as templates for users. These scripts require minor modifications for each site and exercise. Directions on the use of the scripts has been added to Section 12.3 of the *JTLS-GO Technical Coordinator Guide*.

3.17 JTLS-2020-15125 Mission Task Sequence Number Change Error

An air mission was created and scheduled to launch at a future time. The mission included an Orbit in its task list. The Player attempted to force the mission to launch earlier than its scheduled time by using the Manage Air Mission Tasks order to change the Orbit task sequence number to occur before its Begin (launch) task. The CEP incorrectly accepted the change request, which led to a crash.

The CEP was modified to not allow a mission task to be moved before the Resource or Begin (launch) tasks. If a Player attempts to change the task sequence number to a value less than the Resource or Begin task sequence number, the request is denied and a message is sent to the Player.

Note: The correct method to change the launch time of a scheduled mission is to alter the Start Task Time of the mission's first primary task. The model will then appropriately adjust the mission's Resource Task and Begin Task time automatically.

3.18 JTLS-2020-15127 Center World View Near Equator

When selecting the World View option in the WHIP/DDS map, the map was not centered to the equator.

WHIP/DDS code was modified to correctly center the map on the equator when selecting the World View option.

3.19 JTLS-2021-15130 Order Field Inconsistent Data Pass HRU Control

The order to pass control of an HRU used an inconsistent method to fill an order field. This order is used to give control of an object to an external model, such as GESI.

The order panel was filled with a list of HRUs which had a side relationship of friendly or neutral. This incorrectly allowed a Player to pass control of an HRU which was not on the same side as the WHIP. The order definition was changed to only allow the user to pass control of HRUs which were on the same side as the WHIP. The Controller retained the option to pass control of all HRU in the game.

3.20 JTLS-2021-15137 Air Movement Report Crash

An Air Transport order was submitted that directed a mission to launch from its home base, pick up supplies at a different base, and then return home to unload. While the mission was loading supplies at the pickup base, an Air Movement Report was requested just as the LOAD_SUPPLIES task completed. The CEP crashed while attempting to generate the Air Movement Report.

When the Air Movement Report started to generate, the model had just finished the LOAD_SUPPLIES task and had only the MSN_COMPLETE task remaining in the queue. The report logic attempted to access the successor to the current task. Because the current task was the MSN_COMPLETE task, there was no successor task. The model crashed attempting to determine if the non-existent successor task was a REFUEL or REARM task in order to calculate the take off time.

The problem was corrected by adding a check to ensure the successor task exists before attempting to access it.

3.21 JTLS-2021-15144 English IIR Message Title Confusing

When the exercise audience views the English version of the IIR message, the message say Intelligence Information Report. This confuses the intelligence cell and the correct terminology "Imagery Interpretation Report" should be used.

All IIR English-version messages were changed to use the appropriate terminology.

3.22 JTLS-2021-15146 Single Path OAS Mission Crash

This problem was first seen and solved as part of STR JTLS-2020-14936 in September 2020. During this test situation, an extremely short single path orbit OAS mission with auto-assign of Yes was created in a target-rich environment. The model crashed because the mission had no route.

When the mission fired its weapons and postponed its current Orbit Task, its single path orbit route was not updated properly, the mission ran out of route points before it fired all of its weapons. Properly updating its planned route when its move task was postponed to fire its weapons solved the problem.

3.23 JTLS-2021-15148 Push Order Convoy Missing Supplies

A Push order was submitted to automatically send supplies from a support unit to a ground unit every hour. The Show Push report verified the hourly automatic push shipments were in effect, but no supplies were listed. The new convoy was displayed on the Supply Run IMT, but the Query Supply Run Status order showed no supplies to be loaded and offloaded. After the Load task was finished, no supplies were taken from the support unit. When the empty convoy arrived at the receiving unit, a Player message stated no supplies were available to offload.

The supply list was located in the wrong section of the order panel. The underlying Simscript routine was unable to find the supply list and, as a result, no supplies were found to ship and an empty convoy was sent. The Push order panel was changed to hold the supply list in the same section of the order panel as other convoy-related orders using the same routine.

3.24 JTLS-2021-15156 JDLM Cannot Own Federate Unit

Units with JDLM as their Owning Federate caused an issue when using the map selection button for orders that could possibly select the incorrectly labeled units. The map select button in this situation was non-operative.

JDLM is no longer a valid Federate for JTLS-GO, but the database builder could assign units to a JDLM federate. To remove the JDLM federate from the DDS would require a database format change which cannot be fixed until a new database format is implemented in JTLS-GO 6.1. To alleviate this issue, SVP Error 456 has been added to identify units with JDLM as the Owning Federate. The user must change the unit's Owning Federate before using the scenario database.

3.25 JTLS-2021-15159 BDA Report SSM Format Error

The BDA Report did not display the number of combat system losses caused by Surface-to-Surface (SSM) missiles. The combat system names that suffered losses were displayed, but not the actual counts. Other combat system losses from artillery and air strikes were properly displayed in the report.

A typographical error in the message template prevented the loss counts caused by SSM attacks from being displayed in the BDA Report. The format error was corrected.

3.26 JTLS-2021-15161 Cancel Unit Offload Supplies Task Message Error

A Cache Supplies order was submitted to a ground unit. The Offload Supplies task appeared on the Unit Tasking Order IMT. After the task started executing, a Manage Land Unit Tasks order was

submitted to cancel the task. The task was successfully canceled, leaving a new Supply Cache target on the WHIP holding the partially delivered supplies displayed in the Target Supplies IMT.

However, the Player received a "Bad Message" translation error in the Message Browser, instead of the expected confirmation that the Offload Supplies task had been deleted.

In this case, the Supply Cache target type (subcategory) did not exist anywhere in the game.

When the unit began to execute the Offload Supplies task, the model created the target and updated the WHIP map filters so that the target could be displayed on the map. The Player message had already been started when the logic that wrote the dynamic map filters altered the output destination from the original message output file. When the logic later attempted to finish writing the Player message, the output did not go to the right output location. The message processing logic was unable to properly process the incomplete message and generated the error.

The problem was fixed by saving the original message output file (unit #) in the routine that writes the dynamic filters and restoring that number at the end.

3.27 JTLS-2021-15163 Convoy Creates Supply Cache Crash

If a convoy is sent to a location to drop off supplies and there are no suitable units in the area, the convoy creates a Supply Cache. If the convoy did not have the supplies on board to fulfill the player's request to drop specific supplies off, the model crashed trying to inform the Player not all supplies were placed in the supply cache.

The rejection message assumed that the supplies were destined for a specific unit. That was not the case. The supplies, if they had existed, would have been placed in a supply cache. The model was corrected by properly removing the assumption that the supplies would always go to a unit.

4.0 REMAINING ERRORS

Every effort has been made to correct known model errors. All reproducible errors that resulted in Combat Events Program (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 5.1.0.0 represents a major release of new functionality, remaining outstanding errors from the JTLS-GO 4.1 series and earlier have been considered to be obsolete and no longer relevant to JTLS-GO and have been removed from consideration for correction at this time. In future Maintenance releases, outstanding errors related to JTLS-GO will be listed in this chapter, with information provided regarding the extent of the error, as well as suggestions to avoid or minimize the effects of the problem.

4.1 DDSC – TMU Line Mode Changes Multiple Grids

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

4.2 DDSC – Multiple Types In DDS History Table

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

4.3 WHIP - Pipeline Not Shown On IMT

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

4.4 DDSC/WHIP/JOBE - CADRG Map Zoom

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

APPENDIX A ABBREVIATIONS AND ACRONYMS

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

AAA	Anti-Aircraft Artillery
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
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
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
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	Japan Self-Defense Forces
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
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-GO 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
XML	Extensible Markup Language
XMS	XML Message Service

APPENDIX B Version 5.1.11.0 DATABASE CHANGES

No database structure changes were made for JTLS-GO 5.1.11.0.

APPENDIX C Version 5.1.11.0 REPOSITORY CHANGES

No changes were made to the JTLS-GO 5.1 repository.