

JTLS-GO

Version Description Document

August 2021



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

JOINT THEATER LEVEL SIMULATION - GLOBAL OPERATIONS
(JTLS-GO 6.0.4.0)

[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 air, land, naval, and Non-Governmental Organization (NGO) environments.

This *JTLS-GO Version Description Document (VDD)* describes the new features of the Version 6.0.4.0 delivery of the configuration-managed JTLS-GO software suite.

JTLS-GO 6.0.4.0 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, as well as some minor model functionality improvements implemented as Engineering Change Proposals (ECPs). These ECPs are summarized in Chapter 2. Code modifications that represent corrections to known Software Trouble Reports (STRs) are described in Chapter 3. Remaining and outstanding STRs are described in Chapter 4.

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

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

Copyright 2021 - ROLANDS & ASSOCIATES Corporation - All Rights Reserved

[Blank Page]

TABLE of CONTENTS

ABSTRACT	iii
1.0 INTRODUCTION	1-1
1.1 SCOPE	1-1
1.2 INVENTORY OF MATERIALS	1-1
1.2.1 Obsolete/Outdated Documents	1-1
1.2.2 Unchanged Documents	1-1
1.2.3 Updated Documents	1-2
1.2.4 Delivered Software Components	1-2
1.2.5 Released Databases	1-4
1.3 INTERFACE COMPATIBILITY	1-4
1.3.1 Support Software	1-4
1.3.2 JTLS-GO Cybersecurity Compliance	1-8
1.3.3 JTLS-GO High Level Architecture Compliance	1-8
1.4 DATABASE MODIFICATIONS	1-9
1.4.1 JTLS-GO Using Legacy Default Symbol Set	1-9
1.4.2 JTLS-GO Using New Default Symbol Set	1-9
1.4.3 Standard Repository Changes	1-10
1.5 INSTALLATION	1-10
2.0 ENGINEERING CHANGE PROPOSALS	2-1
2.1 JTLS-2021-15278 VIRTUAL BATTLE SPACE ADDED AS EXTERNAL FEDERATE	2-1
3.0 SOFTWARE TROUBLE REPORTS	3-1
3.1 JTLS-2021-15228 NAVAL UNITS WITH ICAO MISSING VCP CHECK	3-1
3.2 JTLS-2021-15229 VCP WRITING CSP_CAT TABLE RECORDS MISSING DATA	3-1
3.3 JTLS-2021-15230 IIP RECORDS FAIL TO LOAD IN DATABASE	3-1
3.4 JTLS-2021-15231 NATIONAL ASSET TBM DETECTION CRASH	3-2
3.5 JTLS-2021-15232 FIRE MISSILE UNASSIGNED WEAPON ISSUE	3-2
3.6 JTLS-2021-15233 CROSS DECKING SPECIFIC SHIP ASSETS CRASH	3-2
3.7 JTLS-2021-15235 SVP ERROR 815 UNCLEAR TEXT	3-2
3.8 JTLS-2021-15236 TPFDD UNIT OUT WHEN IN PORT ERROR	3-3
3.9 JTLS-2021-15237 DDS PROTOTYPE OWNED TARGET RENAMING ERROR	3-3
3.10 JTLS-2021-15238 SMALL BOAT DAMAGE REPORTED AS SUPPLY DAMAGE	3-3
3.11 JTLS-2021-15239 DDS TABLE DID NOT TAKE ENTERED VALUE	3-3
3.12 JTLS-2021-15240 INCORRECT WET/DRY STOCKAGE OBJECTIVE	3-4
3.13 JTLS-2021-15241 LINK16_REPORT_TYPE OUT OF BOUNDS	3-4
3.14 JTLS-2021-15242 HELP/DRM FILE NAME INCORRECT	3-4
3.15 JTLS-2021-15243 HRU LOG REPORT REPEATING SUPPLY CATEGORY	3-5
3.16 JTLS-2021-15244 LC2IS REPORTABLE EQUIPMENT	3-5
3.17 JTLS-2021-15245 LOGFAS SUPPLY CAT ONLY OWNED BY NAVAL UNIT CRASH	3-5
3.18 JTLS-2021-15246 ARMED RECCE TRYING TO FIRE ON SHIP WITH NO PRIORITY ..	3-6

- 3.19 JTLS-2021-15247 SHIP MOVING TO SAME POINT CRASHES SYSTEM 3-6
- 3.20 JTLS-2021-15248 JOI MODULE SERVICE LABEL NOT UPDATING 3-6
- 3.21 JTLS-2021-15249 ATOT NO SEPARATE COMAO CHIT ORDER GROUP 3-6
- 3.22 JTLS-2021-15250 ATOT NOT REREADING INITIALIZATION FILE 3-7
- 3.23 JTLS-2021-15251 ATOT INCORRECT FIELD FOR AIR MISSION PACKAGES 3-7
- 3.24 JTLS-2021-15252 OWN SIDE DETECTIONS NOT REPORTED VIA LINK16 3-7
- 3.25 JTLS-2021-15253 DATA SYNCHRONIZATION CLARIFICATION 3-7
- 3.26 JTLS-2021-15254 XMS CRASHES PROCESSING RESTART DOWNLOAD 3-8
- 3.27 JTLS-2021-15256 COMPLETE FAILED CHECKPOINT 3-8
- 3.28 JTLS-2021-15257 SYNCHRONIZE BUTTON NEEDS TOOLTIP 3-8
- 3.29 JTLS-2021-15258 AAR QUERY TO RETRIEVE MISSION LAUNCHES FAILS 3-9
- 3.30 JTLS-2021-15260 OPM DROP-DOWN NOT CONVERTING TIME ZONE 3-9
- 3.31 JTLS-2021-15261 CRASH SWITCHING PUMPING STATION PIPELINE 3-9
- 3.32 JTLS-2021-15262 TRIPP ADDED AFTER GAME START REFUSED CONNECTION .. 3-10
- 3.33 JTLS-2021-15263 MISSING CHECKPOINT IN AAR VIEWS 3-10
- 3.34 JTLS-2021-15264 MENU FILES REFERENCE NONEXISTENT UTILITY 3-11
- 3.35 JTLS-2021-15266 CRASH READING COUNTRY CODES 3-11
- 3.36 JTLS-2021-15267 ATO PARSER ADJUSTS AIRMOVE END TIME INCORRECTLY 3-11
- 3.37 JTLS-2021-15268 ELS CRASH DELETING TARGET 3-11
- 3.38 JTLS-2021-15269 MISSING WEB SERVICES JAVADOC DOCUMENTATION 3-11
- 3.39 JTLS-2021-15271 ATOT NAME AND USE GRAPHIC SLIDES 3-12
- 3.40 JTLS-2021-15273 DUPLICATE MAP OBJECT SELECTION EVENT 3-12
- 3.41 JTLS-2021-15274 AAR DATA DELETED ON CHECKPOINT 0000 RESTART 3-12
- 3.42 JTLS-2021-15275 SVP WARNING 1233 CORRECTION DOES NOT WORK 3-12
- 3.43 JTLS-2021-15276 HRU ATTACK CONVOY ISSUES 3-13
- 3.44 JTLS-2021-15277 ATOT LINK FOREIGN AIRBASES FOR SHARED ICAOS 3-13
- 3.45 JTLS-2021-15279 ADDING BRANCH TO SEALANES CRASH 3-14
- 3.46 JTLS-2021-15280 SIP CRASHES FOR OUT OF MEMORY 3-14
- 3.47 JTLS-2021-15281 LOCK LACKS CEP MACHINE NAME 3-14
- 3.48 JTLS-2021-15282 WARNING 1462 CORRECTIONS NONFUNCTIONAL 3-14
- 3.49 JTLS-2021-15283 MDP "NOT AN ACCEPT SOCKET" ERROR ON START UP 3-15
- 3.50 JTLS-2021-15284 MDP MESSAGES FROM ACTIVE TCP CONNECTIONS 3-15

- 4.0 REMAINING ERRORS 4-1
 - 4.1 DDSC - TMU LINE MODE CHANGES MULTIPLE GRIDS 4-1
 - 4.2 DDSC - MULTIPLE TYPES IN DDS HISTORY TABLE 4-1
 - 4.3 WHIP - PIPELINE NOT SHOWN ON IMT 4-1
 - 4.4 DDSC/WHIP/JOBE - CADRG MAP ZOOM 4-1

- APPENDIX A. ABBREVIATIONS AND ACRONYMSA-1

- APPENDIX B. VERSION 6.0.0.0 DATABASE CHANGES B-1

- APPENDIX C VERSION 6.0.0.0 REPOSITORY CHANGES C-1
 - C.1 NEW ADA CLASSES: C-1

C.2 NEW AIRCRAFT CLASS:	C-1
C.3 NEW AKL:	C-1
C.4 NEW COMBAT SYSTEMS:	C-1
C.5 NEW GROUND UNITS:	C-1
C.6 NEW HRU'S:	C-1
C.7 NEW JAMMERS:	C-2
C.8 UPDATED PH TABLES:	C-2
C.9 NEW PH TABLES:	C-2
C.10 UPDATED PK TABLES:	C-3
C.11 NEW PK TABLES:	C-3
C.12 NEW SALS:	C-3
C.13 NEW SENSORS:	C-3
C.14 NEW SKL:	C-3
C.15 NEW SMALL BOATS:	C-3
C.16 NEW SSMS:	C-3
C.17 ROGUE.NSM(2).LNCHR.VEH	C-3
C.18 UPDATED SUPPLY CATEGORIES VALUES:	C-4
C.19 NEW SUPPLY CATEGORIES:	C-4
C.20 RENAMED SUPPLY CATEGORIES:	C-4
C.21 NEW TUPS:	C-4
C.22 NEW TWS:	C-4
C.23 NEW UOMS:	C-5

1.0 INTRODUCTION

1.1 SCOPE

This *JTLS-GO Version Description Document (VDD)* describes Version 6.0.4.0 of the configuration managed Joint Theater Level Simulation - Global Operations (JTLS-GO[®]) software suite. JTLS-GO 6.0.4.0 is a Maintenance delivery for the JTLS-GO 6.0 series of releases.

JTLS-GO 6.0.4.0 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.

One Engineering Change Proposal (ECP) is delivered with this maintenance release and is described in [Chapter 2.0](#). 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 6.0.2.0)
- *JTLS-GO Air Services User Guide* (JTLS-GO Document 03, Version 6.0.0.0)
- *JTLS-GO Data Requirements Manual* (JTLS-GO Document 05, Version 6.0.1.0)
- *JTLS-GO DDS User Guide* (JTLS-GO Document 06, Version 6.0.3.0)
- *JTLS-GO Director Guide* (JTLS-GO Document 07, Version 6.0.0.0)
- *JTLS-GO Executive Overview* (JTLS-GO Document 08, Version 6.0.2.0)
- *JTLS-GO WHIP Training Manual* (JTLS-GO Document 10, Version 6.0.0.0)
- *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 Federation User Guide* (JTLS-GO Document 20, Version 6.0.0.0)
- *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 DDS Training Manual* (JTLS-GO Document 23, Version 6.0.3.0)
- *JTLS-GO Configuration Management Plan* (JTLS-GO Document 24, Version 6.0.0.0)

1.2.3 Updated Documents

- *JTLS-GO Controller Guide* (JTLS-GO Document 04, Version 6.0.4.0)
- *JTLS-GO Installation Manual* (JTLS-GO Document 09, Version 6.0.2.0)
- *JTLS-GO Player Guide* (JTLS-GO Document 12, Version 6.0.4.0)
- *JTLS-GO Version Description Document* (JTLS-GO Document 17, Version 6.0.4.0)
- *JTLS-GO Entity Level Server User Guide* (JTLS-GO Document 19, Version 6.0.0.0)

1.2.4 Delivered Software Components

JTLS-GO 6.0.4.0 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 170.

- 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 (JOBED)
- JTLS Geographic Information System (GIS) Terrain Building Program
- JTLS Master Integrated Database (MIDB) Tool

Instructions for installing JTLS-GO 6.0.4.0 are provided in the *JTLS-GO Installation Manual*. Installing a previous version of JTLS prior to installing JTLS-GO 6.0.4.0 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 6.0.4.0, 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 6.0.4.0 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 6.0.4.0 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 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 future 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 6.0.4.0 is delivered using the AdoptOpenJDK 1.8 Update 302.

Note: AdoptOpenJDK Update 302 explicitly disables the deprecated Transport Layer Security (TLS) protocol versions 1.0/1.1. With this TLS protocol disabled, the Glassfish CLI commands to the Glassfish secure admin services do not work.

Altering Glassfish to solve this problem would require complete recertification of the JTLS-GO Glassfish server. Instead, the JTLS-GO Development Team has decided to re-enable this deprecated TLS protocol so the system follows the current certification standard. The older TLSv1.0/1.1 is only used for accessing Glassfish secure admin services, which is always password protected. The TLS 1.2 cyphers are used for all other http/https: access requirements to Glassfish.

This Glassfish issue will be fixed properly in JTLS-GO 6.1 so the new Glassfish Server can be re-certified along with all of the other JTLS-GO 6.1 software.

- 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 6.0.4.0. 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-2.0> OHLA 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 prebuilt 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 6.0.4.0 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 6.0.4.0 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.

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 6.0.4.0, 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.

STR JTLS-2021-15237 required changes to some post-update database triggers. In order to update your database scheme to use this trigger correction, perform the following procedure:

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

The following Engineering Change Proposals (ECPs) were added to this Maintenance Release of JTLS-GO.

2.1 JTLS-2021-15278 Virtual Battle Space Added As External Federate

Summary of Model Change Request

A link to the external simulation Virtual Battle Space (VBS) is under development. Forming this link requires the capability to use VBS as the owning federate for units in the game.

Design Summary

VBS was added as an allowed federate, involved only minor changes to the Combat Events Program (CEP), Scenario Initialization Program (SIP), Entity Level Server (ELS), and JTLS-GO Object Distribution Authority (JODA). The full implementation of the link to VBS will be delivered early in 2022.

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-2021-15228 Naval Units With ICAO Missing VCP Check

Several naval units were not loaded when the Version Conversion Program (VCP) was run on an error-free JTLS-GO 5.1 database.

The VCP checks to ensure that a naval unit that has a specified ICAO will have CARRIER, HELO CAPABLE, AMPHIB HELO CAPABLE, or SUPPLY HELO CAPABLE as its assigned Ship Unit Prototype. Several naval_unit records violated this check and were not loaded.

The VCP now checks whether each naval unit has an ICAO specified. If the unit has an ICAO, the unit's SUP TYPE is checked. If the unit's SUP TYPE is neither CARRIER, HELO CAPABLE, AMPHIB HELO CAPABLE, nor SUPPLY HELO CAPABLE, the naval unit's ICAO is set to "" (empty string), so that the VCP will output it as NONE.

3.2 JTLS-2021-15229 VCP Writing CSP_CAT Table Records Missing Data

When running the VCP on an error-free JTLS-GO 5.1 database, records reading "<CSPName>/<CATName>/0/0/0/0/0/0/" were written to the csp_cat file. This caused the DDS to crash.

There were missing records in the JTLS-GO 5.1 csp_cat file, and the VCP created records with all zeros in the JTLS-GO 6.0 csp_cat file to account for the missing records.

The VCP now initializes the CSP CAT MEAN TIME BETWEEN REPAIR to 0.25 over all the COMBAT SYSTEM PROTOTYPE/COMBAT ARMS TYPE compound entities before reading the 5.1 csp_cat file. If there are any missing 5.1 records, the CSP CAT MEAN TIME BETWEEN REPAIR is never zero. This will help the user identify which 6.0 records come from missing 5.1 records. The user can then decide what real data should go into the record.

3.3 JTLS-2021-15230 IIP Records Fail To Load In Database

The JTLS-GO 5.1 database had a value of 0.0416 for the parameter IIP TAC INTEL VARIABILITY. The VCP was writing out the value as 0.041600000050 instead of 0.041600000000. PostgreSQL was rejecting the record during scenario load because it violated the check constraint of "less than or equal to" 0.0416.

The upper limit in the related check constraint was modified to be "less than" 0.014667 to successfully load the Intel Information Prototype (IIP) records in the scenario after conversion.

3.4 JTLS-2021-15231 National Asset TBM Detection Crash

The Controller turned on the National Collection Assets capability for a radar, but did not create a radar sensor owned by the Force Side commander. When the national asset capability picked up a theater ballistic missile, the model crashed because it could not determine if the detection was or was not a Link 16 asset.

The model will now allow the detection in this circumstance, and assume the detection could not be Link 16 capable.

3.5 JTLS-2021-15232 Fire Missile Unassigned Weapon Issue

When a user submits a Fire Missile order using the model select weapon option, the weapon that was selected was not properly assigned to the fire mission, causing the mission to be canceled.

Once the model selects the weapon, it is now properly saved on the order and can be accessed when preparing the fire mission.

3.6 JTLS-2021-15233 Cross Decking Specific Ship Assets Crash

A crash occurred when a ship was instructed to cross-deck a unit's assets from a specific ship to two other vessels. This crash occurred because the unit did not actually have any assets on the specified ship.

The order verification code for cross decking was changed to properly determine and check for the crash condition. Now under all circumstances, only the unit assets on board the "giving" ship are relocated to the new cross-deck ships.

When investigating the crash, a local variable was found to be conditionally set within a loop and was not being properly reinitialized between iterations of the loop. This was also fixed.

3.7 JTLS-2021-15235 SVP Error 815 Unclear Text

It was unclear whether the text of SVP Error 815 referred to Target long name or short name.

Error 815 identifies when a Target to be created at a future date has a CCF Number or Long Name that matches another Target in the database.

In order to clearly identify which name was causing the problem, Error 814 was added. Errors 814 and 815 now specifically state whether the Target Long Name or the Target CCF Number is the problem.

Further testing found that as each future Target was checked, it was removed from the Target set, preventing users from checking the names of a future Target against other future Targets. This was also corrected.

3.8 JTLS-2021-15236 TPFDD Unit Out When In Port Error

The CEP crashed when a unit that was TPFDDed out of the game and back in.

When the unit was originally TPFDDed out of the game, the code failed to remove the unit from the IN.PORT set. When the unit was TPFDDed back in, the code attempted to place it into the proper unit port set. This caused the crashed. The problem was corrected by removing units from the IN.PORT set when TPFDDed out.

3.9 JTLS-2021-15237 DDS Prototype Owned Target Renaming Error

When the DDS user was trying to rename the base name of a Prototype Owned Target record in the TUP_POT or SUP_POT table, they were receiving a "Column Reference TG name is ambiguous" error.

This problem is caused by differences between the way Oracle managed post-trigger code and PostgreSQL processes these triggers. The post-update trigger functions for the TUP_POT and SUP_POT database tables were altered to properly work under PostgreSQL. Due to this change, users should download and then reload their scenarios to have the new trigger functions in their database schemas.

3.10 JTLS-2021-15238 Small Boat Damage Reported As Supply Damage

A CEP routine that records Air Mission weapon damage and sends the damage items to the ICC was incorrectly assuming damage to a unit was either Hull-related, Combat System-related, or Supply Category-related, and defaulting to the last item if the damage was not the first two. However, the damage could be Small Boat-related, and this case needed to be added.

The code was modified to also expect Small Boat-related damage to units and to handle any other unexpected cases with a logic error.

3.11 JTLS-2021-15239 DDS Table Did Not Take Entered Value

When several rows in the TUP Supply Table screen are highlighted, and an attempt is made to set the On Hand value to 1.0, the DDS did not update the field properly and always set the highlighted On Hand value to zero.

The On Hand column of this table represents a supply amount that always has a DDS-specified Unit of Measure (UOM). The process of changing a value in a Supply Category value cell is a two step process:

1. Call a method to return the user's specified UOM for the cell.

2. Pass the newly-entered value to the cell editor. This process converts the entered value, assumed to be in the cell's UOM, to the database-held value and updates the database with the computed value.

When highlighting several rows of the table, each of which may have a different user-specified UOM, the method used to return the user's specified UOM was always NULL. The method cannot determine the UOM because several rows are highlighted.

When the process moved on to the second step, a null UOM was passed to the cell editor. When the cell editor received a NULL UOM, it always converted the entered value to 0.0.

The logic of this procedure was changed. The conversion process now simply returns the entered value when the passed in UOM is NULL. When highlighting several rows, and setting the value of a specific cell in each row, the entered value will be assumed to be in the default UOM.

3.12 JTLS-2021-15240 Incorrect Wet/Dry Stockage Objective

The SVP reported the wet and dry stockage objectives as being larger than they actually are.

The SVP attempts to compute values that can be used for automatic corrections. The TUP storage capacity is normally based on the total TUP stockage objective, and in some rare cases, the basic load values may be higher than the stockage objectives. The code used the greater than or equal (\geq) operator and computed the needed storage capacity as the total basic load * 1.05 (.05 was used for spare capacity), when it should only have used the greater than ($>$) operator. The code was changed to correct the operator.

3.13 JTLS-2021-15241 Link16_Report_Type out of bounds

A user requested the Set Aircraft Parameter Order report all database values for a selected Aircraft Class. The selected Aircraft Class had an AC.LINK16.CAPABLE value of 3. The user was able to open the message, but the WHIP threw an exception stating that the Link16_Report_Type was out of bounds. The message was not able to translate the Link16_Report_Type.

When the data was being read in on game start/restart, the CEP was assigning a value of 3 for the AC.LINK16.CAPABLE field for all aircraft classes using "SEND_AIR_INFO". In the static_voc file, messages for the AC.LINK16.CAPABLE variable in the CEP are translated using the Link16_Report_Type. The Link16_Report_Type only accepts values 0, 1, or 2, so a value of 3 was out of bounds. In the JDSP, the LINK16_REPORT_TYPE values can also only be 0, 1, or 2.

The routines which perform the translations during the read and writes in the CEP have been updated to only use 0, 1, or 2 for the AC.LINK16.CAPABLE field.

3.14 JTLS-2021-15242 Help/DRM File Name Incorrect

The column name "attrition_factor" in the DDS tables referenced a DRM file name that did not exist.

The DDS screen definition file was corrected to reference the correct DRM file name (FLP.CAT.LC.WC.ATTRITION.FACTOR.pdf).

3.15 JTLS-2021-15243 HRU Log Report Repeating Supply Category

An HRU Logistics Report is broken into multiple sections, starting with the status of combat systems/equipment, followed by fuel supplies, followed by ammunition supplies, and finally all other supplies held by the HRU. When writing out the last section, the supply category name is always the same for all entries and does not represent the actual supply category.

This was a failure to initialize the supply category index that was being used. This initialization has now been added.

3.16 JTLS-2021-15244 LC2IS Reportable Equipment

When reporting equipment, combat systems that were not personnel and did not have a Reportable Item Code (RIC) were being skipped. This statement should have been combat systems that were personnel or did not have a RIC were skipped. The mix up in the logic was causing personnel systems to be printed under equipment when they should not have been.

Additionally, subordinates were not listed alphabetically on the Online Player Manual (OPM) web browser interface. The WHIP Command Hierarchy provides an alphabetical listing of subordinates. Not alphabetizing entries makes the OPM difficult to use.

The logic statement was modified to reflect the proper conditional for skipping the combat system.

Subordinates are now sorted alphabetically when they are retrieved to display on the web browser interface.

3.17 JTLS-2021-15245 LOGFAS Supply Cat Only Owned By Naval Unit Crash

If a database used LOGFAS profiles, and that database contained a Supply Category only used by a Naval Unit, the SIP would crash when writing out the LOGFAS initialization file.

The issue occurred because the SIP first loops through all legal units to create the data structures needed to track the LOGFAS initialization data, and then loops through all legal units to actually write out the LOGFAS initialization data. The first loop properly skipped over Naval Units without a RIC. The second loop did not skip over these units. When attempting to write out the Supply Category Information for a Naval Unit without a RIC, the model crashed because the internal data structures were not created.

The solution was to also skip over Naval Units without a RIC when looping through the second loop.

3.18 JTLS-2021-15246 Armed Recce Trying To Fire On Ship With No Priority

The model crashed when an Armed Recce mission attempted to fire on a ship that was not within its assigned priority list.

The model should never have attempted to fire. The ship held some equipment and supplies that were on the priority list, but when it was time to fire, only the ship's hull was considered. The hull was not on the priority list. The routine that determined whether the ship was on the priority list should not have checked supplies and equipment for ships. That check is only applicable to ground units.

The routine in which this check is done was called ARE NEEDED SUPPLIES AVAIABLE. The supplies should not have been checked. The routine was fixed and renamed so there is no further confusion. The new name for the routine is DOES UNIT MATCH PRIORITY.

A logic error was added for safety at the exact crash location in case there is another way to make it through the logic inappropriately.

3.19 JTLS-2021-15247 Ship Moving To Same Point Crashes System

The model crashed because it was attempting to determine if a unit would cross a barrier during its next move. The problem was that the next move was not a move, but a move to the ship's current location.

The logic was changed to catch this situation. The ship could not possibly cross a barrier or boundary if its next move is to the ship's current location.

3.20 JTLS-2021-15248 JOI Module Service Label Not Updating

Users are able to switch between active JOI services from the JOI module. This applies to both OTHGold and Link16 services and modules. The JOI module displayed the name of the JOI service that it initially connected to, but failed to update when swapping to a different service.

The OTHGold and Link16 JOI modules have been updated to update the service name labels when swapping between services.

3.21 JTLS-2021-15249 ATOT No Separate COMAO Chit Order Group

Whenever an ATO contains refuel missions and refuel chits, the generated orders that correspond to the chits should be placed in a separate order group identified by the time or COMAO of the associated missions. This separate chit group is being created for translations having the non-COMAO option, but is not being created for the COMAO option.

The ATO Translator (ATOT) was modified to create a separate group for chit orders, similar to the group used to hold chits when the non-COMAO option is selected. Having a group of chit orders

allows the user to sequence the sending of orders to the simulation. In this way, refuel missions and missions receiving fuel are known by the simulation before the chit that reference them.

3.22 JTLS-2021-15250 ATOT Not Rereading Initialization File

When the ATOT starts up, it reads the initialization data file provided by the CEP. However, if the CEP rewrites the initialization data file, the ATOT does not detect the newer file and reread the data file. In this case, any translations would require the ATOT to be restarted in order to discover the new data.

The ATOT was modified to check the modification time of the initialization data file before any attempt to use the data obtained from this data file. When the time is newer, the ATOT will reread the file and replace the older data with the data from the newer file. All subsequent translations will use newer data, such as mission weapon loads that may have been added after the ATOT was first started.

3.23 JTLS-2021-15251 ATOT Incorrect Field For Air Mission Packages

The Manage Attack Package order requires an Initial Routing field, which the ATOT is not writing into the order for creating the package. This occurs whenever the ATO includes mission packages.

The ATOT was modified to include this field within the Manage Attack Package order.

3.24 JTLS-2021-15252 Own Side Detections Not Reported Via Link16

An AWACS was not sending Link16 Track messages for own-side aircraft that were within range of the sensors and were showing as being detected on the WHIP. Sufficient track blocks existed for track assignment and reporting.

The squadron that launched the aircraft was listed as being in a Link16 network but did not have any track blocks assigned. This assignment to the Link16 network was passed to the air mission when it launched. The Link16 detection logic processed the air mission, saw that it was associated with a Link16 network and therefore assumed it was a self-reporter. Since the squadron was not assigned JU blocks, the mission had none and therefore had no mechanism to self-report.

The check for self-reporting was modified to verify the mission is associated with a network and that the mission has a JU number assigned. Once the mission did not pass this new self-reporting check it was properly detected and reported by the AWACS.

3.25 JTLS-2021-15253 Data Synchronization Clarification

After completing data synchronization, the user is given the message "Synchronize complete. Reload table." This looks like the user is being required to perform an action, but no actions are required by the user. This is confusing.

The wording of this message was changed to "Synchronize complete. Reloading table."

3.26 JTLS-2021-15254 XMS Crashes Processing Restart Download

The XMS may crash when it remains running and the CEP is restarted, which causes a fresh download of data.

When the XMS receives a new set of data during a download from the JODA, it must purge the old data. If the XMS has no previous data to purge, there is no problem. However, when the new downloaded data is replacing the previous data, the XMS must purge all the units in its hierarchy structure. While iterating through this structure, the unit entries were being deleted, which caused simultaneous modification and access to the hierarchy structure.

The XMS routine used for purging all units was modified to prevent recursive accesses within the structure.

3.27 JTLS-2021-15256 Complete Failed Checkpoint

The CEP encountered a crash situation during the taking of a checkpoint. The Tech Controller then chose to run the "Complete a Failed Checkpoint" procedure. This procedure is responsible for creating the next checkpoint directory, adding an entry to the scenario .tim and .run files, and moving the current order file to the checkpoint. This creates the shell of a checkpoint that can be used during the pushing of orders on a restart. These processes are also the first thing that the CEP accomplishes when it starts a checkpoint.

This procedure should only be run when there is a hard crash situation, a power outage, or some other situation where a checkpoint could not be started, not when the game crashes during a checkpoint. The Tech Controller should be prevented from making this mistake.

One of the first things that the CEP does when taking a checkpoint is to move the current order file, the .ciO file, into the new checkpoint directory. One of the last things that is done during a checkpoint is to create a new current order file. If this file is missing, then the CEP assumes that a checkpoint was started, and that the three essential activities of the checkpoint were accomplished (creating the directory, updating the .run and .tim files, and moving the .ciO file), and should prevent the procedure from being run.

The "Complete a Failed Checkpoint" script has been modified to include this logic.

3.28 JTLS-2021-15257 Synchronize Button Needs Tooltip

The Synchronize Data table in the Data Repository Tool has a button that does not display a tooltip when the cursor hovers over it.

A tooltip was added to the synchronize data button.

3.29 JTLS-2021-15258 AAR Query To Retrieve Mission Launches Fails

The user requested data for missions that had launched within a specified time period using the `aar_missions` script. This query failed because the launch time was not part of the primary select statement, but was being used by the primary select statement to attempt to filter. This caused an SQL error and the failure of the query.

The script is designed to return missions that have had their initial launch sequence, with the option of limiting output to a time period. Because an air mission can have multiple launch events (every time an OAS goes on an intercept, or when a mission refuels on the ground), we want to only include the first launch event for the mission. This means the query must find if the first launch event falls within the time period, rather than finding the first launch event within the time period.

To attempt to enforce this distinction, we inserted a nested query to only retrieve the first launch event (based on time). This time retrieval only existed within the nested query, but the time constraint filters were applied to the master query, resulting in the failure.

The query was rewritten, still as a nested query, so that the inner query will retrieve all first launches for all air missions. The outer query, which has access to all the retrieval data, including the time, is then used to filter by the launch time.

Additionally, all times within the AAR database are stored as character strings, rather than timestamps. The time filtering was actually comparing ASCII strings rather than actual times. This issue was addressed by converting all times into timestamps for the purposes of comparison and filtering.

3.30 JTLS-2021-15260 OPM Drop-Down Not Converting Time Zone

The time zone was not being updated for any OPM pages that use the drop-down menu for converting time zones.

The script "`make_js_arrays`" was not updated to parse the `<scenario_name>.glo` file to get the scenario start time. The script has been updated.

3.31 JTLS-2021-15261 Crash Switching Pumping Station Pipeline

If a Pumping Station Target is originally assigned to operational Pipeline A, and the Controller switches the Pumping Station to be assigned to Pipeline B, a crash will occur. Code exists to accomplish this task, but the code attempts to remove the wrong pointer from the operating unit of Pipeline A.

The code was modified to remove the proper target from the operating unit of Pipeline A.

3.32 JTLS-2021-15262 TRIPP Added After Game Start Refused Connection

If a new TRIPP-enabled WHIP is added after game start, the SYNAPSE will allow a request for this TRIPP to login, but the Replay JXSR refuses to connect to it. The new TRIPP is therefore unable to replay any of the saved game data.

The Logging JODA creates the log files of JODA packet data as they are received from the model. The stream of packets contain all the MIP data for each Player WHIP (TRIPP). The MIP packets available at the start of the game are received within the first complete download of data from the model and written at the very start of the log file.

When one of the newer TRIPPs makes a request to connect to a Replay JXSR, the Replay JXSR start reading the log of packets at the start of the log file (time zero for the game). If a MIP for this newer TRIPP does not exist in the set of MIPs at time zero, then the request to connect is rejected regardless of whether the MIP packet for this TRIPP exists later in the log file.

The fix was to allow valid TRIPPs to connect even before they were created in the packet stream by the Logging JODA. The SYNAPSE only allows valid TRIPPs to login. This prevents a TRIPP by an invalid name from connecting. The Replay JXSR was modified to be able to distinguish between different TRIPPs during logins and internal management for side perception(s).

3.33 JTLS-2021-15263 Missing Checkpoint in AAR Views

All records that are stored in the AAR tables have a checkpoint associated with them. This is always the next checkpoint, so records inserted after checkpoint 0003 will have a checkpoint of 0004 associated with them. The concept of the next checkpoint allows the AARC to remove records if the game is hard aborted without checkpoint 0004 ever being taken.

The purpose of the checkpoints is so that the AARC can create views on each table for every run. Thus, the view `aar_object_lookup_run_1` will limit the retrievals of the `aar_object_lookup` table to only those records that are in the checkpoints associated with run #1. This allows data across multiple runs to be gathered by the AARC and for the reports to limit their retrievals to a specific run. The AARC will read the scenario run file and create these views when it starts and whenever it gets a checkpoint notification.

The start checkpoint notification was being used as a trigger, but the CEP had not always updated the scenario run file when the AARC performed its action, resulting in the new checkpoint being left off the views for the current run.

The code was modified to use the checkpoint complete notification instead of the start checkpoint notification as the trigger to read the scenario run file and create all the views for each run.

3.34 JTLS-2021-15264 Menu Files Reference Nonexistent Utility

There were some configuration-managed menu files that referenced the old Generate Traffic HRU probability list utility (HPL). This utility no longer exists.

The utility was renamed to the Generate Traffic HUP List (GTH). The menu files were changed to reference the new name of GTH, rather than the old name of HPL.

3.35 JTLS-2021-15266 Crash Reading Country Codes

A site-specific Country Code overwrite file was created in the \$JGAME/data_site directory that was a full copy of the configuration-managed version, with some additions. Because the site-specific file is considered an overwrite and addition of the existing file, this discarded every entry from the configuration-managed versions. During the discard process, the entries were not properly removed from a hash table, but all data about the entries was deleted. This later resulted in a crash when the hash entries to these deleted items were accessed.

The discarding of an overwritten country code was fixed to properly remove it from the hash tables and perform the deletion.

3.36 JTLS-2021-15267 ATO Parser Adjusts AIRMOVE End Time Incorrectly

The ATO Parser used the wrong month for the ending date-time-group for all AIRMOVE records when an ATO period time shift is desired.

The Parser read the start time after the time shift had been applied, but read the end time before the shift had been applied. This caused an incorrect month to be computed. This has been corrected to properly shift both times.

3.37 JTLS-2021-15268 ELS Crash Deleting Target

The ELS crashed, deleting a killed multiple-element target. Portions of the target elements were supposed to be transferred up the chain of command. They were, but these elements were not removed from the target prior to deletion.

The problem was solved. When transferring a target element, the element is now also removed from the target's element set.

3.38 JTLS-2021-15269 Missing Web Services Javadoc Documentation

The Software Maintenance Manual describes the availability of the Application Programming Interface (API) specification for the JTLS-GO Web Services. The HTML-formatted documentation was missing from the delivered JTLS-GO documentation suite.

The javadoc generating process is now incorporated in the JTLS-GO nightly build so that the documents will always be up to date.

3.39 JTLS-2021-15271 ATOT Name And Use Graphic Slides

During translation, the ATOT will create slides for individually-named ACO airspaces or slides for Use groups of the ACO airspaces, but will not create both at the same time.

The ATOT has a routine for discovering which slide options have been selected by the user through the interface GUI. The logic in this routine was not properly checking both options. This logic has been modified to check for both options and write the slides for the desired options.

3.40 JTLS-2021-15273 Duplicate Map Object Selection Event

When selecting a position on the map while preparing an order, the WHIP received a duplicate order query when the user selected on a unit from the map.

The duplicate problem was corrected by stopping logic to request another query.

3.41 JTLS-2021-15274 AAR Data Deleted On Checkpoint 0000 Restart

A number of runs of the CEP were conducted and AAR data was gathered during the course of these runs. The CEP was then restarted from checkpoint 0000 without a push, and the message the CEP sent to the AARC was that a game start was in progress. This caused the AARC to assuming the CEP was throwing out all the old checkpoints and that a fresh start was happening. Therefore, the AARC truncated all the database tables, resulting in the loss of all the collected data.

The Start/Restart Program recognizes the difference between a Start and a Restart from Checkpoint 0000. However, to the CEP, both actions count as a game start and that is the signal sent to the AARC.

The code within the CEP to determine the signal to send to the AARC was improved so it does not look at just whether it thinks its a game start or a restart, but to also check what the next checkpoint will be. If the next checkpoint is checkpoint 0001 then it is a true start. Otherwise it constitutes a restart in AARC terms and that message will be sent.

3.42 JTLS-2021-15275 SVP Warning 1233 Correction Does Not Work

The automatic correction for Warning 1233 for aviation fuel does not work.

The automatic correction for Warning 1233 for updating the aviation fuel had a hard-coded supply category value of POL. This has been long outdated.

The supporting routine was modified to provide the correct supply category as defined in the user database. The change itself only recalled the prototype supply record if it existed. The software was further modified to also allow the user to insert a new record. This required the use of another warning to properly implement correction options. Because the warnings were similar

and should be kept together, Warning 1233 was renumbered to 1270 and new Warning 1271 was created.

Another problem was found which threw out supply records internal to the program that were all zero. This confuse the software which would try to insert a new record when one actually existed in the database. This problem was also corrected and new Warning 1233 was created.

3.43 JTLS-2021-15276 HRU Attack Convoy Issues

An HRU was ordered to Patrol a small polygon along a road, with Seek Combat set to Yes and Combat Object List set to Truck_Rail_TTG. When a truck convoy entered the patrol polygon, the HRU engaged the convoy, but did not inflict any losses on the convoy.

The model permitted the HRU to detect and attack the convoy, but a logic flaw caused the Direct Damage Supply Run routine to immediately exit, because the attacked object priority type was a Combat System instead of a Transportation Class. The flaw was corrected in the Evaluate Highres Fire routine by setting the subject priority type to Transportation Class if the attacked object is a convoy. The priority type is passed down the calling chain to the Direct Damage Supply Run routine, allowing the damage adjudication process to proceed.

While testing this correction, the message 4510 "Spot Report Convoy Under Attack" was found to not correctly handle an attack on a convoy by an HRU. The message template did not accommodate an HRU attack. This problem was corrected by adding a new submessage #10 for the HRU attack situation and modifying the underlying Quick Attacked Convoy Report subroutine to use the new submessage.

Further testing uncovered a model crash that occurred when all vehicles in the convoy were destroyed by the HRU attack. The model failed to recognize that the entire convoy was destroyed and attempted to execute the return fire logic (convoys cannot return fire anyway and would not do so even if not destroyed). The model crashed trying to reference the now-destroyed convoy. The problem was corrected by checking the Dead Flag (set at the end of the Assess Damage routine) before beginning the return fire process in the HRU Assess Combat event. If the dead object was a convoy, the return fire logic is now skipped, while destroyed HRUs and Units are still allowed to return fire one last time.

3.44 JTLS-2021-15277 ATOT Link Foreign Airbases For Shared ICAOs

The ATOT automatically forms links between ICAOs specified in the ATO message and the associated airbases defined in the scenario data. When two or more airbases on different Sides have the same ICAO (as would occur with a coalition ATO), the ATOT may select the foreign Side as the airbase when linking with the ICAO. When this happens, the ATO translation will not verify.

The ATOT was modified to only include the link side when computing the automatic linking.

3.45 JTLS-2021-15279 Adding Branch To Sealanes Crash

When adding a branch to the sealane network, the model crashed.

When the Manage Movement Network code was cleaned up, the name of the Sealane network was changed to SEALANE, but the real name of the network is SEALANES. The code could not find the sealane network and crashed. The spelling of the network name was corrected.

3.46 JTLS-2021-15280 SIP Crashes For Out Of Memory

The SIP crashed when the system ran out of memory.

A memory leak in the CREATE.SPHERICAL.POLYGONS routine caused the problem. The routine used temporary arrays that were not being released when the routine was exited. The SIP routine code was corrected.

3.47 JTLS-2021-15281 Lock Lacks CEP Machine Name

The ./game/<scenario_name>/<scenario_name>.lck file does not contain the name where the CEP is currently executing.

When a game is started, the code creates a file in the ./game/<scenario_name> directory with the name <scenario_name>.lck. That file contains the name of the machine that the CEP was started on. If another individual should attempt to start a game for the same scenario, the code will see the .lck file exists, will deny the start/restart of the scenario, and will notify the user whose machine the CEP is currently running on.

Originally, the code used a called the function OS.GETHOSTNAME.F. That function has not worked in the latest releases of Simscript. To fix the problem, the function call was changed to OS.GET.ENVVAR.F using the parameter "HOST". This returns the name of the host the CEP is running on, which is then written to the .lck file.

3.48 JTLS-2021-15282 Warning 1462 Corrections Nonfunctional

The Warning 1462 corrective action to open the combat system table does not work.

The parameter that is passed to indicate which specific combat system should be displayed was passing an incorrect data item. This was corrected.

Two new corrective options were added to include displaying the DOWNED PILOT HUP and to automatically update number of combat systems needed. The wording was also modified.

Warning 1445 was also reviewed and also found to have errors. The corrective actions were expanded and the wording was also changed.

3.49 JTLS-2021-15283 MDP "Not An Accept Socket" Error On Start Up

The MDP will sometimes report an error, reading "socket does not reference an accept socket", after start up, when trying to send the first messages over any TCP connection. The error report is reoccurring and fills up the error log.

In the MDP Connection Manager class, the default state for a disconnected socket ID is supposed to be -1, but during startup, the socket takes a value of 0 prior to any connection attempts. The socket ID is eventually set to -1 after running through a message cycle. The MDP connection manager class was modified to set the socket to -1 when it gets initialized during startup.

3.50 JTLS-2021-15284 MDP Messages From Active TCP Connections

Whenever the MDP has more than one TCP connection configured and active, it sends all the messages for each connection to only one of the active TCP connections. They are all received, but by only one of the receiving hosts.

The MDP Connection Manager class is used for connecting, sending, and closing a single TCP connection. For each delivery profile that included an active TCP connection, only the first profile would be configured and saved to this Connection Manager. When messages were sent over each connection, the one Connection Manager was being used.

This code was expanded so the MDP will maintain a list of Connection Manager objects. Each of the active TCP connections is therefore managed by a separate Connection Manager.

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 represents a major release of new functionality, remaining outstanding errors 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
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 6.0.0.0 DATABASE CHANGES

Refer to Appendix B in the JTLS-GO Version 6.0.0.0 VDD. No database format changes were needed to this Maintenance release.

APPENDIX C VERSION 6.0.0.0 REPOSITORY CHANGES

The following changes were made to the JTLS-GO 6.0 repository.

C.1 New ADA Classes:

- GBI.FACILITY
- MADIS.MK1.STINGER(SP)VEH

C.2 New Aircraft Class:

- 6GEN.FIGHTERS

C.3 New AKL:

- KH101.HE_AKL

C.4 New Combat Systems:

- BMK130_AMPHIB
- EWTV.M1235.M2-APC.HA.OW
- MADIS.MK2.M134-APC.LA.TW
- OVERLORD.LUSV_AMPHIB
- SEA.HAWK.MSUV_AMPHIB
- T72B.125MM-TANK_IT
- VITYAZ-DT10P_VEH-LA
- VITYAZ-DT30_VEH-LA

C.5 New Ground Units:

- FT.GREELY.GBI_US
- USMC.NMESIS.BTRY_US

C.6 New HRU's:

- OVERLORD.LUSV
- OVERLORD_LUSV

- SEA.HAWK_MSUV
- SEA.HAWK_MSUV

C.7 New Jammers:

- 1L267.MOSKVA.RDR.JAM
- 1L269.KRASUKHA-2.RDR.JAM
- AVIACONVERS.GPS.JAM
- EW1600.GPS.JAM
- KRASUKHA-4.COM.JAM
- MODI.II.COM.JAMMER
- R330ZH.ZHITEL.COM.JAM
- RB109A.BYLINA.COM.JAM
- RB341V.LEER-3.COM.JAM
- RB531B.INFAUNA.COM.JAM
- RB636AM2.SVET-KU.COM.JAM
- REPELLENT.UAV.COM.JAM
- TORN.COM.JAM
- TORN-MDM.COM.JAM
- VLQ12.CREW.DUKE.COM.JAM

C.8 Updated PH Tables:

- TLAM-C
- TLAM-D
- TLAM-E

C.9 New PH Tables:

- KH101_PH

- SIAW_PH

C.10 Updated PK Tables:

- TLAM-C
- TLAM-D
- TLAM-E

C.11 New PK Tables:

- KH101_PK
- PRESEVET_PK
- SIAW_PK

C.12 New SALs:

- GBI_SAL
- NULKA_SAL

C.13 New Sensors:

- RPS42_LAA

C.14 New SKL:

- KH101_SKL
- SIAW

C.15 New Small Boats:

- BMK130
- OVERLORD_LUSV
- SEA.HAWK_MSUV

C.16 New SSMs:

C.17 ROGUE.NSM(2).LNCHR.VEH

New SUPs:

- LAW_US

C.18 Updated Supply Categories Values:

- SUP SAN.ANTONIOL_US For CL.V.SS-SR.LASER-XN-1

C.19 New Supply Categories:

- CL.V.TORP.HAAWC.MK54
- CL.V.AA-AIM260
- CL.V.AS-AGM158D
- CL.V.AS-AGM88G
- CL.V.AS-HACM
- CL.V.AS-SIAW
- CL.V.AS-TV-LR-KH101
- CL.V.SA-LR-GBI
- CL.V.SA-SR-NULKA.DECOY
- CL.V.SS-ATACM1A.UN
- CL.V.SS-LR-NGLAW

C.20 Renamed Supply Categories:

- CL.V.SS-SR.LASER-XN-1

C.21 New TUPs:

- GBI.FACILITY_US
- MC.LAAD.(MADIS).BN.HQ_US
- MC.LAAD.(MADIS).BTRY_US
- MC.ROGUE.BTRY_US

C.22 New TWs:

- AGM158D.JASSM-XR

- AGM88G.HARM
- AIM260
- ATACMS-BLK1A.UN
- GBI
- HAAWC.MK54.AL324TP
- HACM
- KH101
- NGLAW
- NULKA.CHAFF.RKT
- SIAW
- STINGER-(SP).MADIS.MK1

C.23 New UOMs:

- AGM158D.JASSM-XR
- AGM88G.HARM
- AIM260
- ATACMS-BLK1A.UN
- GBI
- HAAWC.MK54.AL324TP
- HACM
- KH101
- NGLAW
- NULKA
- SIAW